UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Facultat d'Informàtica de Barcelona

Computer interfacing (CI)

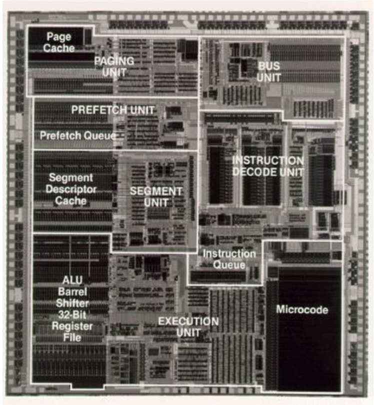
1. Introduction

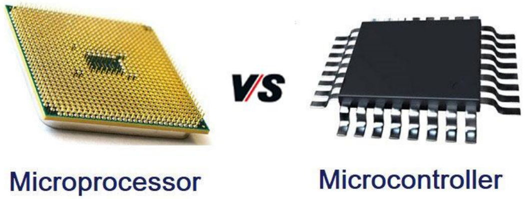
1.1 Expected knowledge

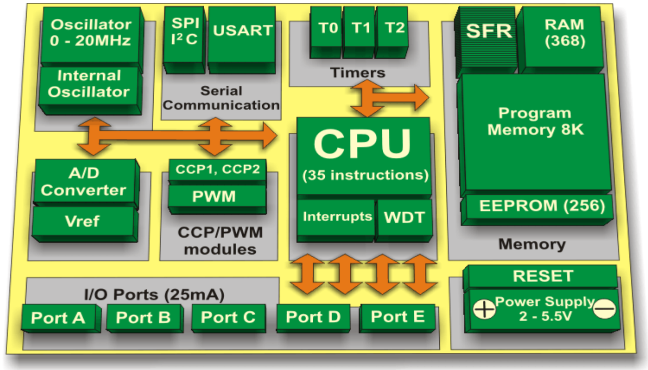
Among others:

- Use and behavior of electronic components like:
  - *Passive devices*: R, C, L
  - *Semiconductors*: diodes, LED, BJT and MOSFET transistors.
- DC circuit analysis. Ohm's law. Voltages (V), currents (A) and power (W) calculation.
- C Language programming.
- Understanding of technical documentation in english.

1.2 Basic ideas (1): Microprocessor vs microcontroller

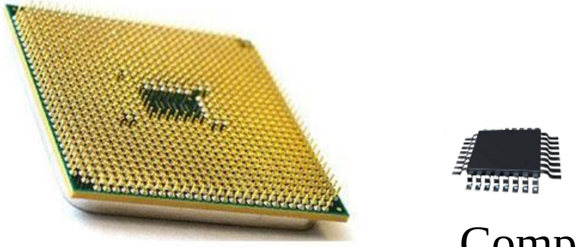






A **microprocessor** is a CPU in a single chip. Resources like main memory or peripherals are available through an external device (in a PC the mainboard).

A **microcontroller** is a small computer on a single chip, containing a processor core, memory, and programmable input/output peripherals.



Comparison of relative sizes.

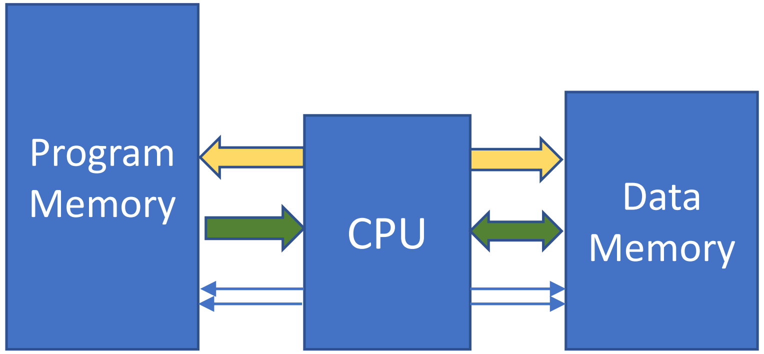
1.2 Basic ideas (2): RISC vs CISC architectures.

RISC	CISC
Simple instruction set	Complex instruction set
Regular and fixed instruction format	Irregular instruction format
Wired instructions, take only one or a few clock cycles to execute.	Complex microcoded instructions, take many clock cycles to execute.
Simple address modes	Complex address modes
Separated data and program memory	Combined data and program memory
Most operations are register to register	Most operations can be register to memory
Take shorter time to design and debug	Take longer time to design and debug
Provide large number of CPU registers	Provide smaller number of CPU registers
Simple operations, longer programs EFFICIENCY?	Complex operations, shorter programs EFFICIENCY?

1.2 Basic ideas (3): Harvard vs Von Neumann Archs.

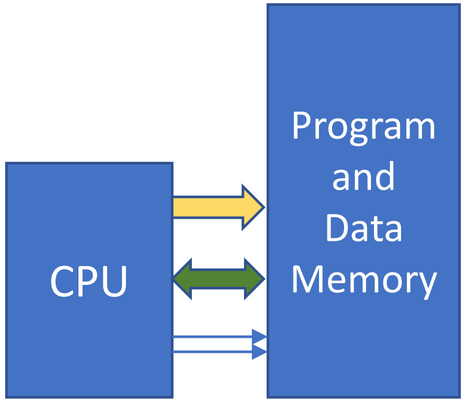
Harvard Architecture

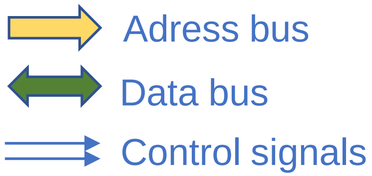
Is a computer architecture with separate storage and signal pathways for instructions and data.



Von Neumann Architecture

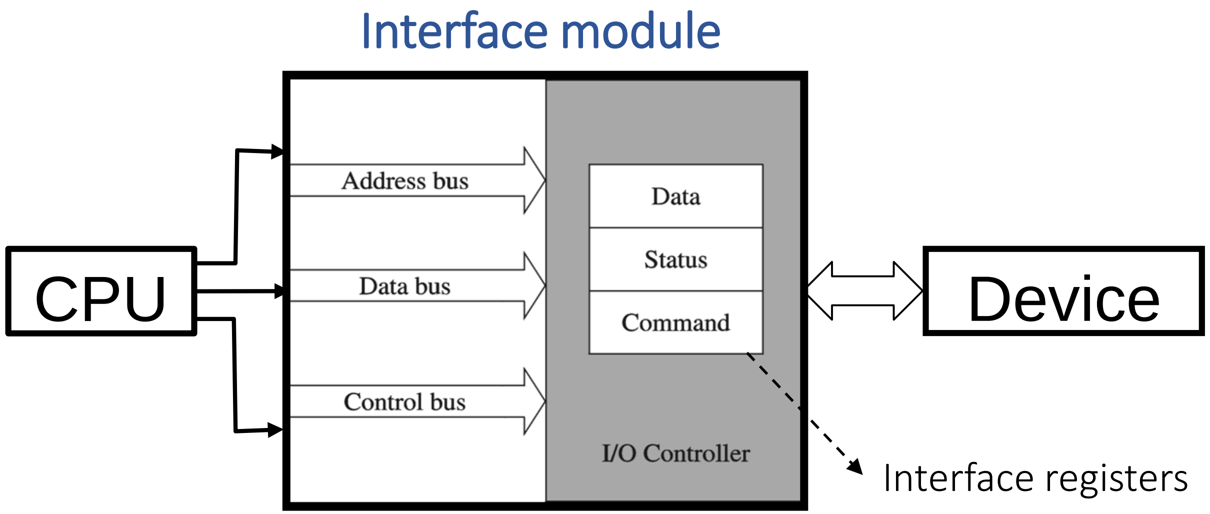
Instructions and data are stored in the same memory, so instructions are fetched over the same data path used to fetch data.





1.2 Basic ideas (4): I/O Registers and Memory Space.

All devices (keyboards, audio, screens…) are managed through **interfaces**, when there is an Operating System, drivers hide this interfaces to the users.



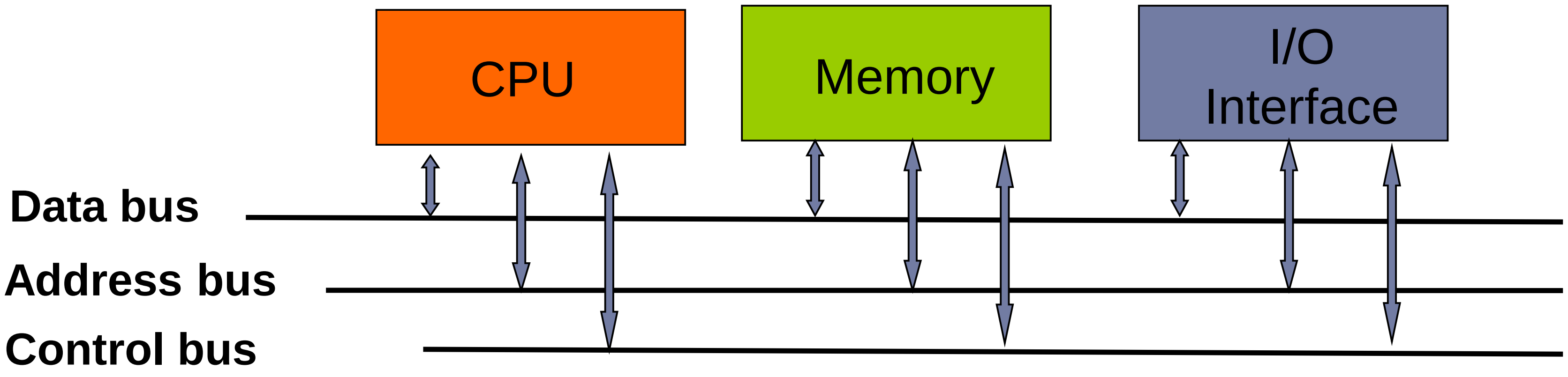
Digital interface registers are also called **PORTS**

System/microcontroller interfaces are programmed with **registers** (command, data…)

Where are Data, Status and Command registers located?

1.2 Basic ideas (4): I/O Registers and Memory Space.

If the same address bus is used for both memory and I/O, **how should the hardware be designed to differentiate between memory and I/O reads and writes?**



1.2 Basic ideas (4): I/O Registers and Memory Space.

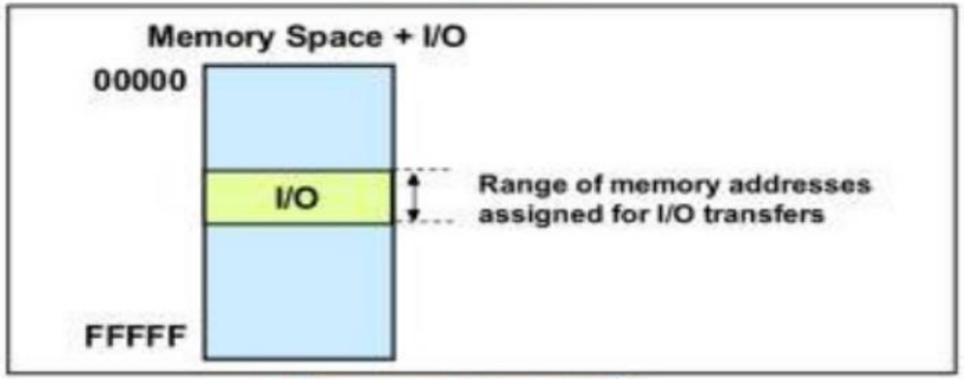
Solution 1: Memory Mapped I/O (MOTOROLA)

Any instruction that reads or writes memory can read/write I/O Port.

Address specifies which module (input, output, RAM, ROM), will communicate with the processor.

Example:

```
LDAA $200A   ;load value from address $200A
STAA $0024   ;store value to Port address $0024
```



Memory-Mapped I/O

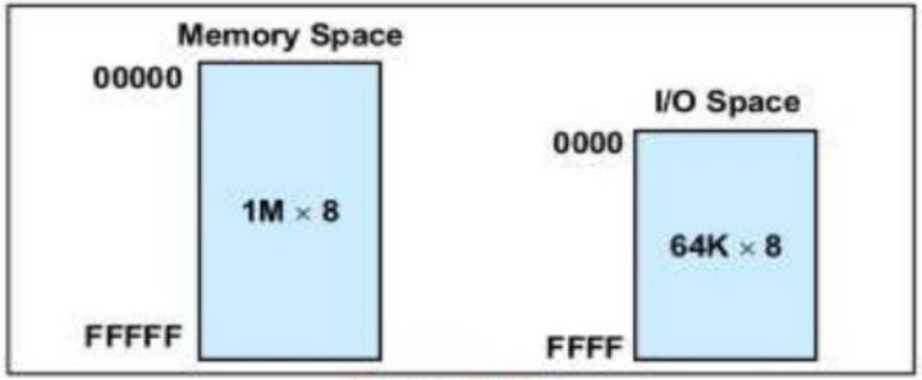
Solution 2: Isolated I/O (INTEL)

The control bus signals that activate the I/O are separate from those that activate the memory device.

These systems have a separate address space.Separate instructions are used to access I/O and Memory.

Example:

```
IN AL,$10      ;read from port $10
MOV $200A,AL   ;store value to RAM address $200A
```



Isolated I/O

1.2 Basic ideas (5): Memory technologies

Main classification

Volatile: loses data when power is turned off.

Typical volatile: Random-Access Memory (**RAM**). It’s a Read/Write memory.

Non-volatile: keeps stored data when power is turned off.

Typical non-volatile: Read-Only Memory (**ROM**). Can only be read but not written by the processor.

1.2 Basic ideas (5): Memory technologies

Types of RAM:

Dynamic random-access memory (DRAM): need periodic refresh.

- Capacitor technology.
- Big capacities (8GB, 32GB).
- Use as PC's main memory

Static random-access memory (SRAM): no periodic refresh is required

- More power consuming. Transistor technology.
- Fast and simple.
- Smaller capacities (KB—MB)
- Use as microcontroller's memory, PC registers, PC Caches.

1.2 Basic ideas (5): Memory technologies

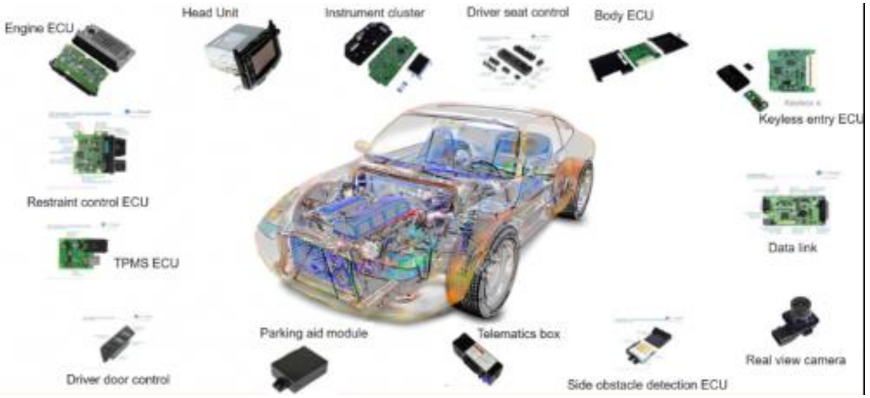
Types of ROM:

Mask-programmed read-only memory (MROM): programmed when being manufactured.
Programmable read-only memory (PROM): can be programmed by the end user (OTP).

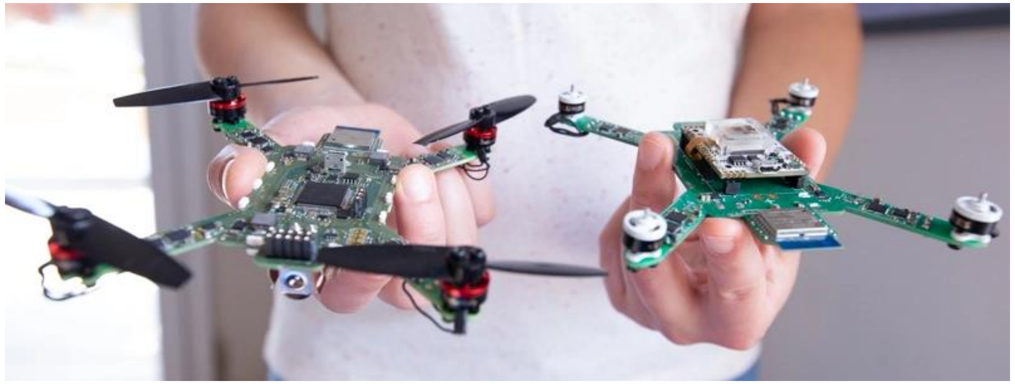
Electrically erasable programmable ROM (EEPROM)
- electrically programmable many times
- electrically erasable many times
- can be erased one location, one row, or whole chip in one operation

Flash memory (e.g. pendrive, SSD disks)
- electrically programmable many times
- electrically erasable many times
- can only be erased in bulk (either a block or the whole chip)

1.3 Use of microcontrollers



Car microcontrollers



Drone microcontrollers

IoT (Internet of things)

The global microcontroller market size was estimated at USD 20.61 billion in 2022 and is expected to reach USD 22.73 billion in 2023.

1.3 Use of microcontrollers: Setup for CI

MICROCHIP

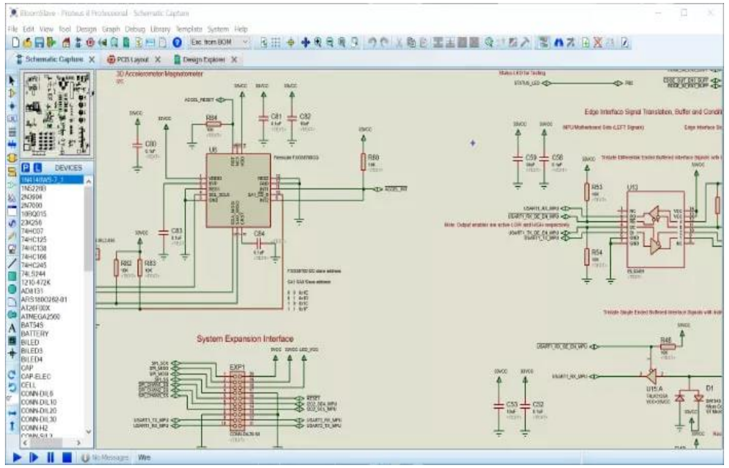
Sellings of 8,3 billion dollars in 2023.
22500 employees.
Leader in the microcontroller industrial sector.

PIC 18F45K22, microcontroller device

MIKROE
Time-saving embedded tools

Mikroe: EasyPIC 7, development board.

Proteus



Proteus: Hardware & Code simulation.

1.3 Use of microcontrollers: Setup for CI

MICROCHIP

PIC 18F45K22,
microcontroller device

- RISC architecture.
- Harvard Architecture.
- Memory mapped I/O registers.
- FLASH memory to store programs.
- EEPROM memory for user data.
- SRAM memory for registers and data RAM.

Entregable Nº 1

Q1 Un microcontrolador amb l’entrada/sortida mapejada en la memòria RAM serà més aviat RISC o CISC?

Q2 Un xip microcontrolador que pretén ser de baix consum tindrà molta, poca RAM o no té res a veure?

Q3 Si en la ROM es guarda el programa, en la RAM es guarden els valors de les variables, llavors on es guarden els valors de les comandes, consignes i paràmetres?