

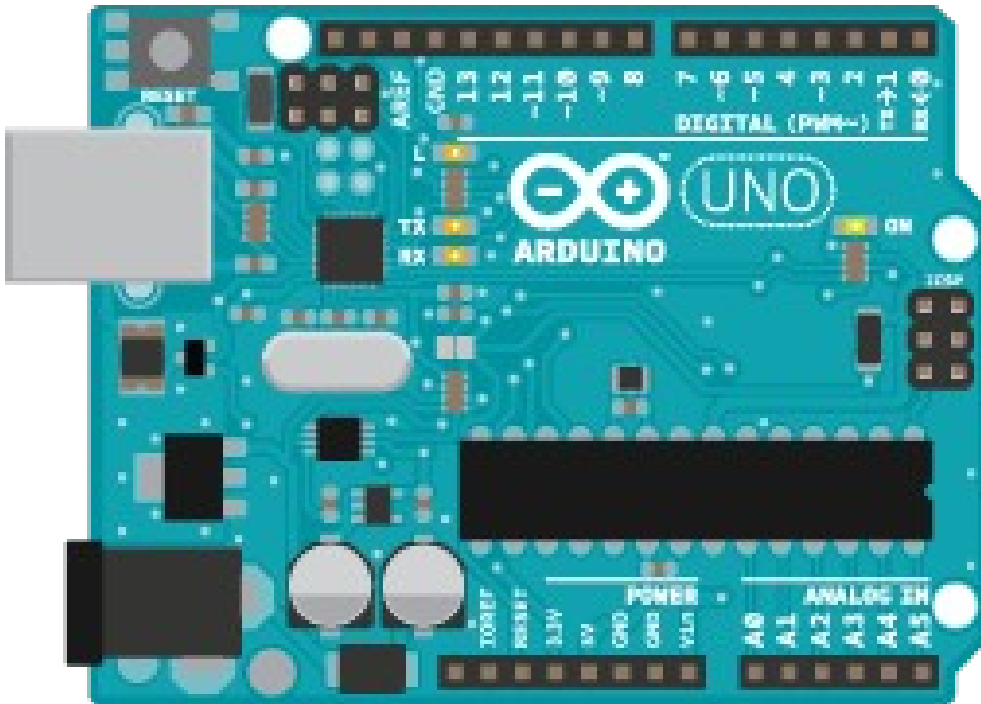
HardWare

- **Microcontrollore:** circuito integrato programmato per eseguire molteplici funzioni
- Istruzioni da eseguire passo per passo: rappresentate da sequenze di numeri binari, contenute nella memoria di programma; l'insieme delle istruzioni viene chiamato **firmware**
- Comunica attraverso **porte di Input-Output:** collegano direttamente i piedini fisici del dispositivo al bus dati e quindi può leggerne lo stato oppure impostarlo ad uno specifico livello logico

Schede elettroniche

- A **basso costo**: includono il microcontroller e tutti i componenti elettronici necessari per l'alimentazione elettrica, la distribuzione dei segnali e la comunicazione con altri dispositivi ed il computer
- Includono porte Seriali, USB, moduli WiFi, processori ARM o architetture custom
- **Linux-controlled board**: al firmware si può sostituire un Sistema Operativo basato su kernel GNU/Linux.; sono disponibili con processori ARM o x86 e frequentemente dispongono di interfacce di rete Ethernet o WiFi
- I linguaggi di programmazione più utilizzati sono C / C++, Python, con ambienti di sviluppo sia proprietari e commerciali che OpenSource e multi-piattaforma.

Cosa è Arduino



Il progetto Arduino è nato nel 2005 presso il ***Interaction Design Institute*** di Ivrea

Fornire agli studenti un'alternativa semplice ed economica alle tradizionali piattaforme di prototipazione elettronica utilizzate per i progetti durante i corsi.

Interagire con il mondo

- I **sensori** sono componenti elettronici in grado di percepire qualcosa dall'ambiente che li circonda e trasformarlo in un segnale elettrico
- Gli **attuatori** trasformano i segnali elettrici che ricevono in fenomeni percepibili nell'ambiente dagli utenti
- Sensori e attuatori **comunicano** con il processore, che interpreta gli input dei sensori e determina il comportamento degli attuatori in base alle istruzioni con cui è stato programmato.

A chi si rivolge

- Chi è incuriosito dall'elettronica e vuole sperimentare in sicurezza e semplicità
- Partendo dalla spiegazione dei principi della programmazione e della prototipazione con Arduino
- Verranno illustrati i concetti base e descritte alcune applicazioni della scheda seguendo una serie di esempi pratici.

Di che cosa abbiamo bisogno

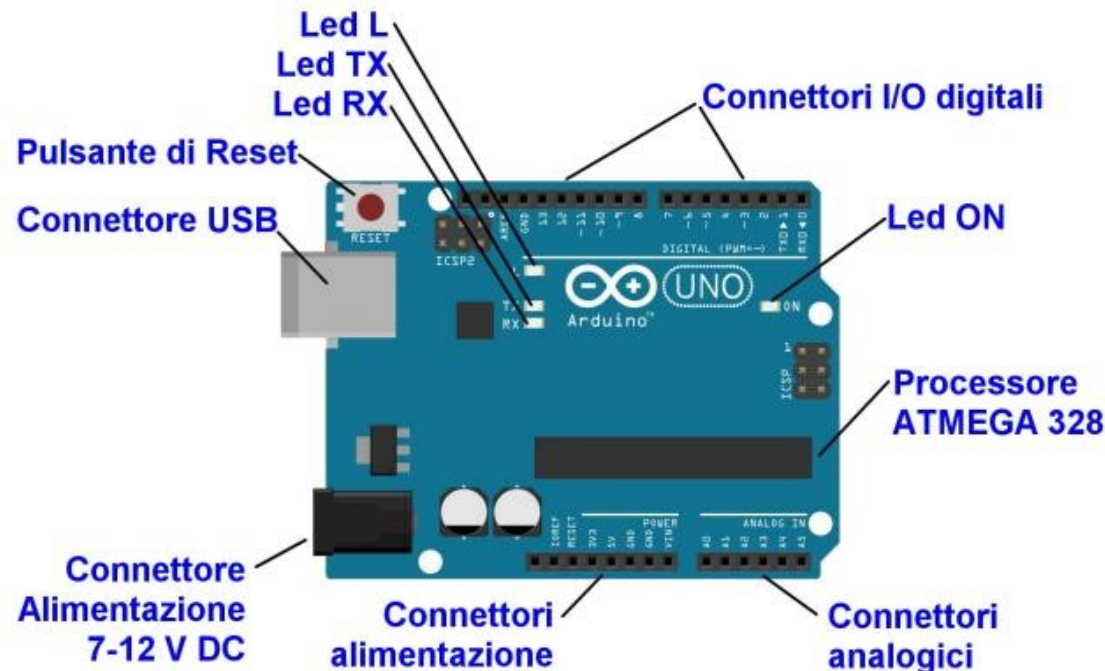
- Scheda Arduino Uno.
- Cavo USB A-B.
- Alimentatore da 9-12V a 1A per collegare la scheda alla rete domestica a 220V.
- Breadboard.
- Cavetti (jumper) di varia lunghezza da utilizzare sulla breadboard.
- LED da 5 mm.
- Resistori (diversi valori).
- Pulsanti adatti a essere posizionati sulla breadboard.
- Potenzziometro da 10k Ohm.
- Fotoresistenza VT90N2.
- Sensore piezoelettrico.
- Transistor BC547.
- Relè con bobina da 5V.
- Diodo 1N4007..
- Sensore di umidità e temperatura DHT11 (o DHT22).
- Sensore di pressione e temperatura BMP180 (o BMP085).

Il primo approccio ad Arduino

- Per prima cosa è indispensabile scaricare dal sito ufficiale del progetto: **<http://arduino.cc>** il software di sviluppo: seleziona dal menu la voce **Download**
- IDE Arduino è distribuito per Windows, OS X e Linux.
- Installare il software e far sì che il sistema operativo riconosca l'hardware.
- Utilizzare la connessione USB: il software Arduino riconoscerà l'hardware e ne permetterà la programmazione.

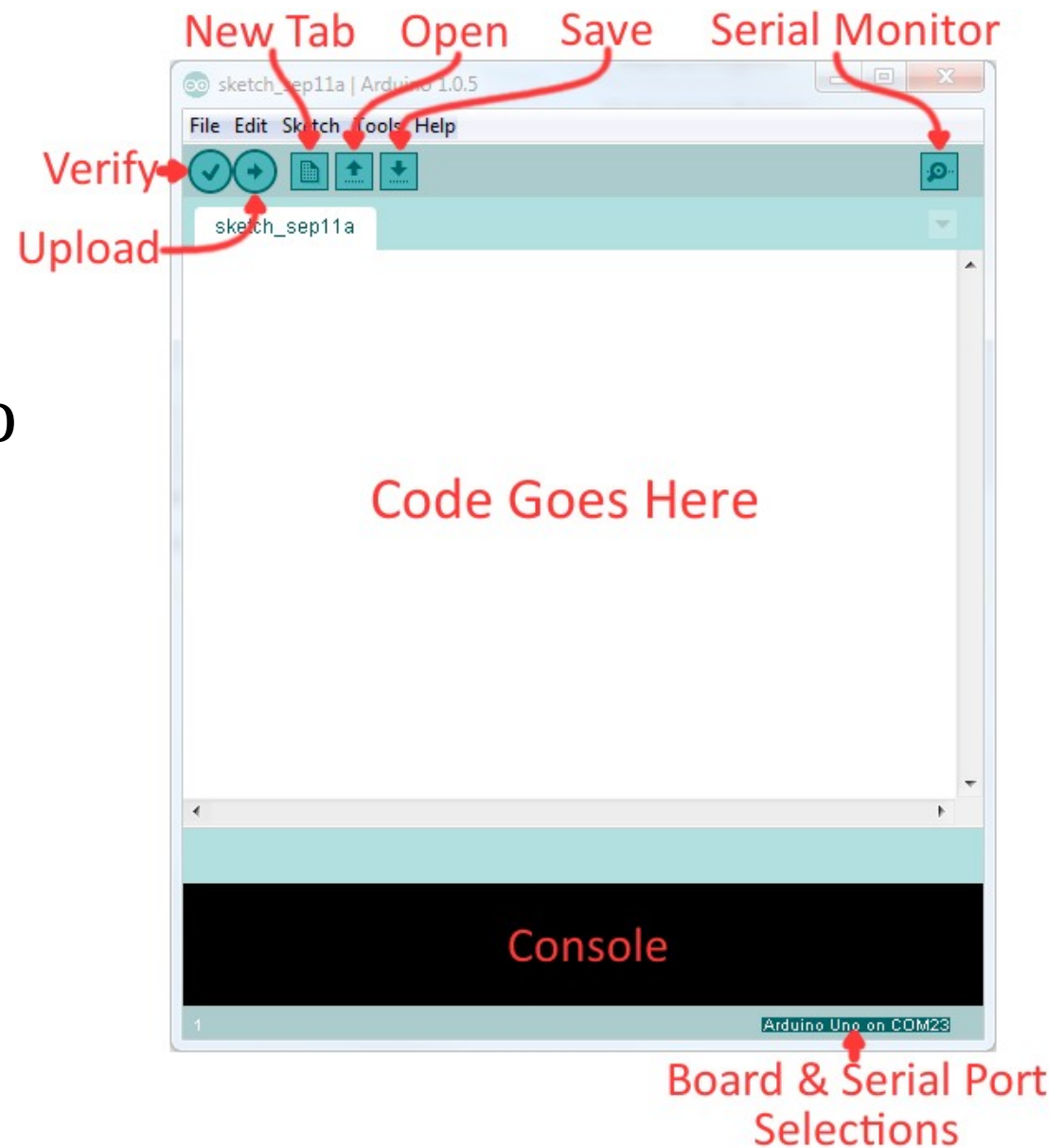
Componenti saldati sulla scheda

- Il tasto di reset.
- Il connettore USB.
- Il jack di alimentazione alternativa.
- I pin di input/output digitale
- I pin dedicati all'alimentazione: voltaggi predefiniti e pin GND
- LED: contrassegnato da L collegato al pin 13; segnalano le comunicazioni seriali in ingresso (RX) e in uscita (TX).
- Il led ON si illumina quando la scheda riceve alimentazione.
- Il “cuore” della scheda: il processore centrale.
- I pin di input analogico.

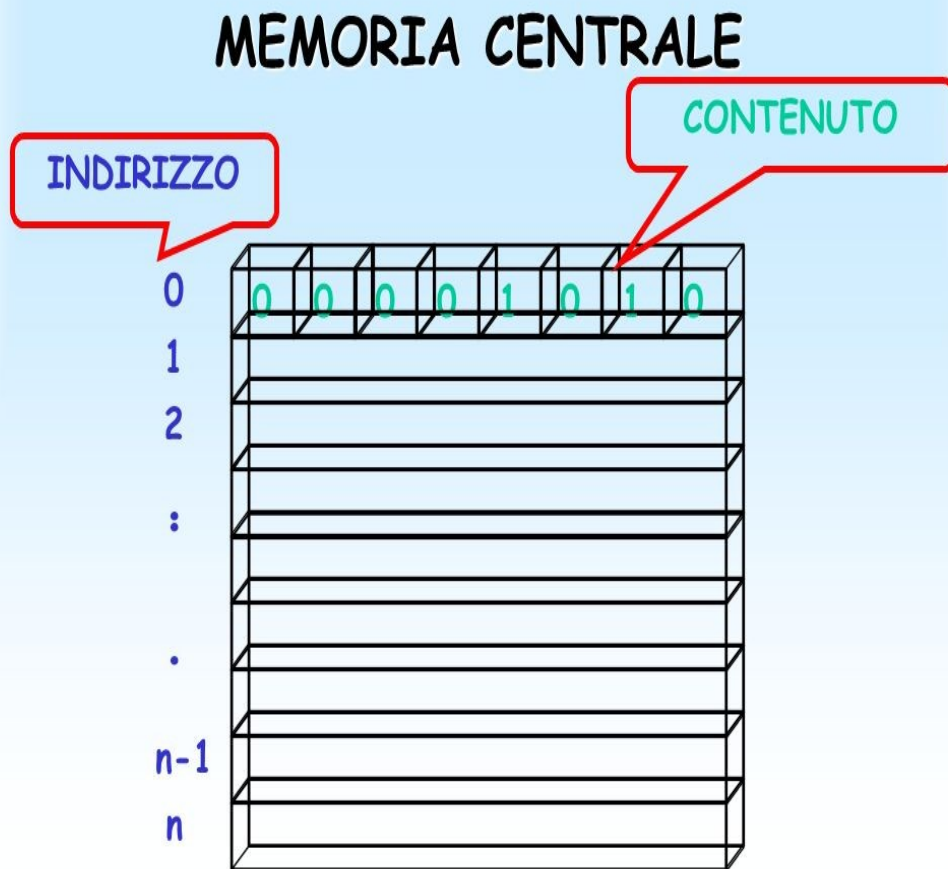


IDE Arduino

- **IDE**: acronimo di **I**ntegrated **D**evelopment **E**nvironment (ambiente di sviluppo integrato)



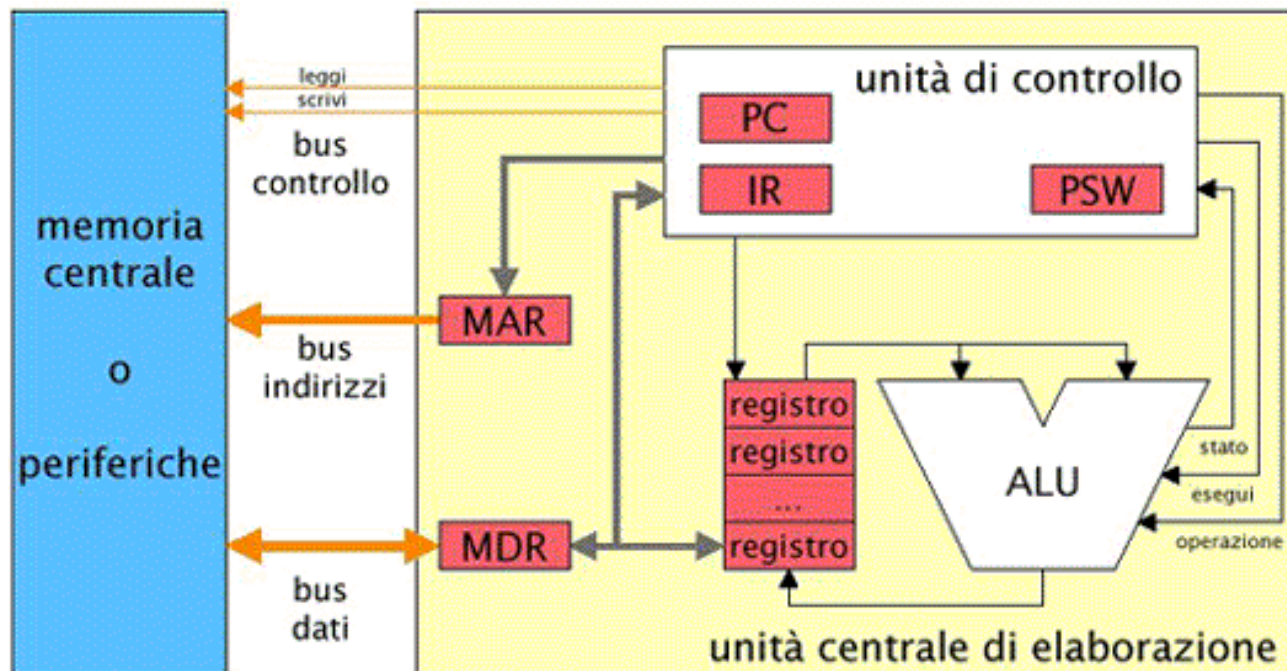
Processore e Memoria



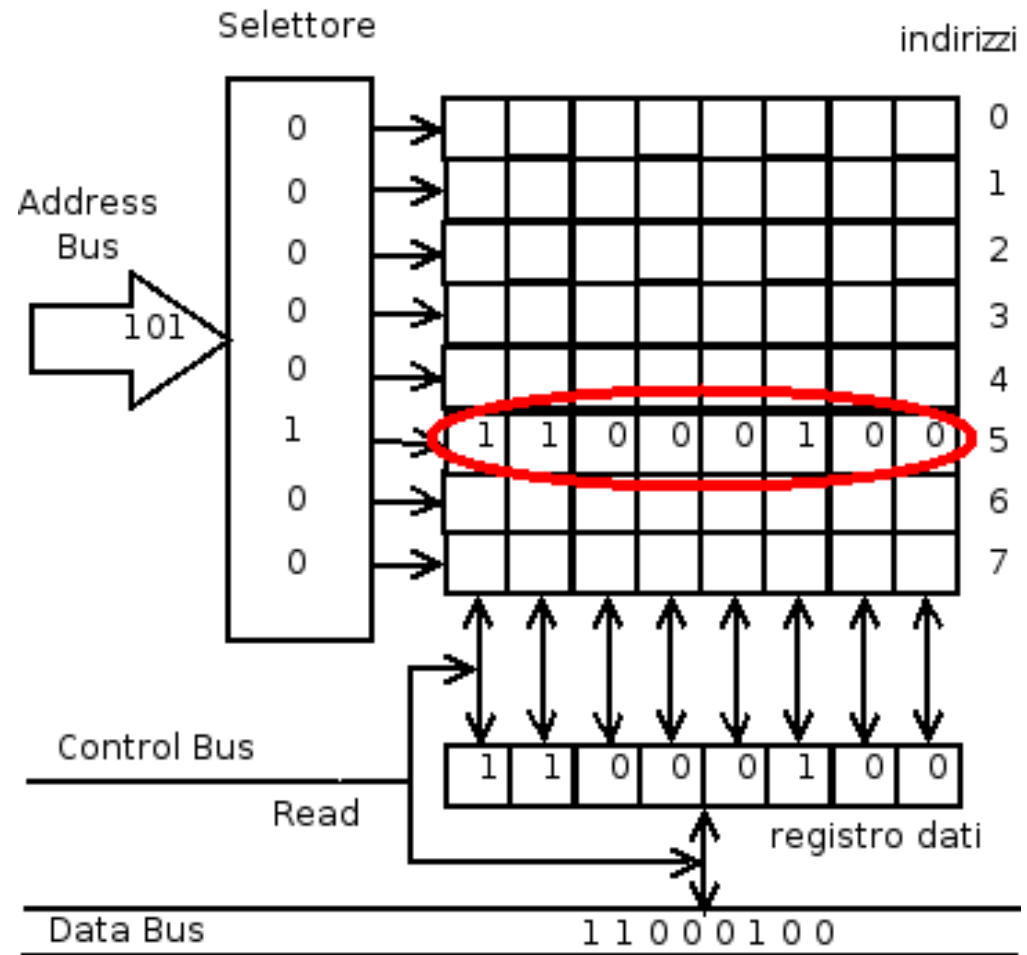
- Il processore
 - elabora I dati
 - contiene svariati registri
 - comunica con la memoria
- La memoria
 - conserva le informazioni elaborate
 - conserva i programmi da eseguire

Come opera il Processore

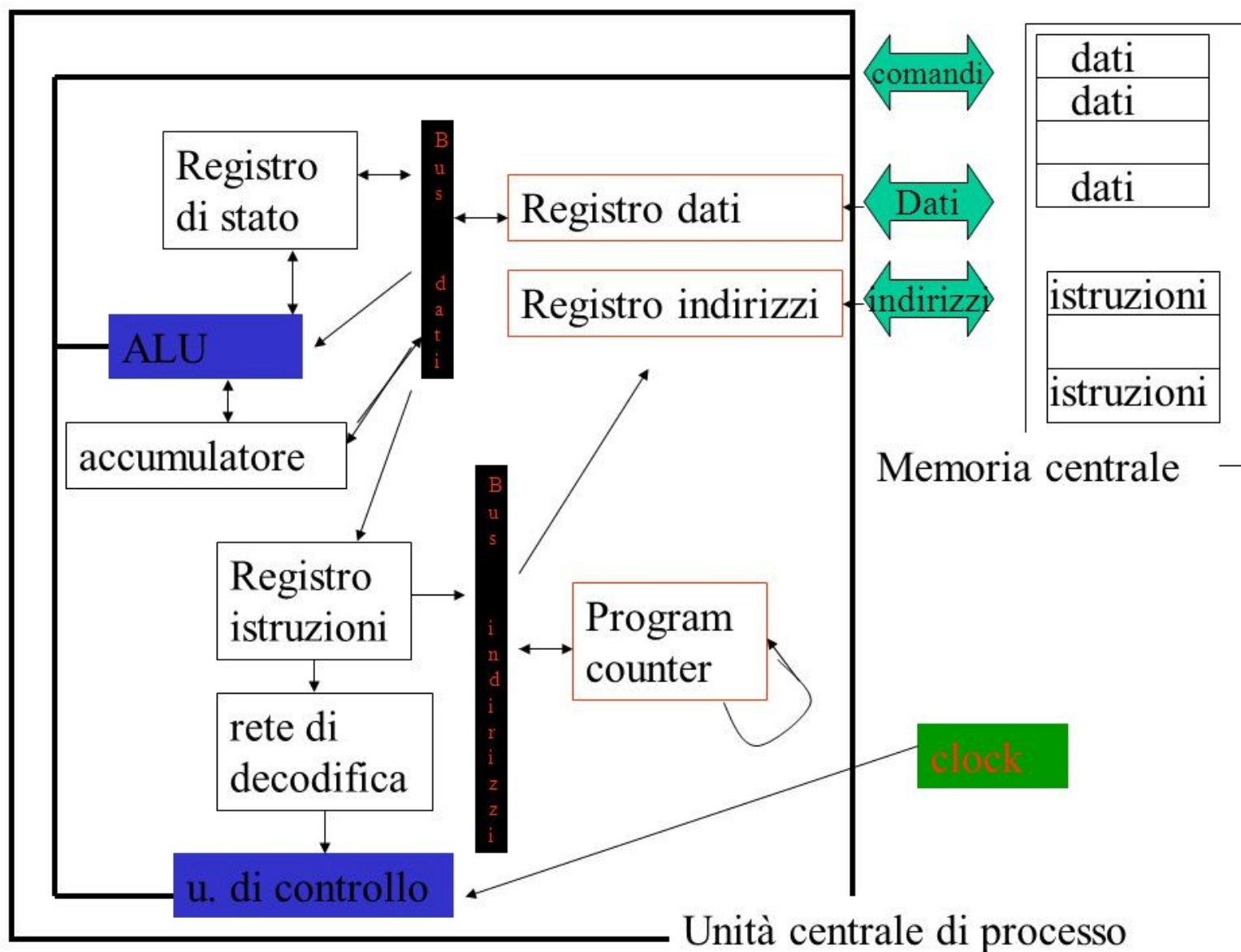
- Il processore
 - preleva i dati da elaborare dalla memoria
 - esegue la elaborazione
 - spedisce I risultati nella memoria



La memoria principale



Memoria e istruzioni



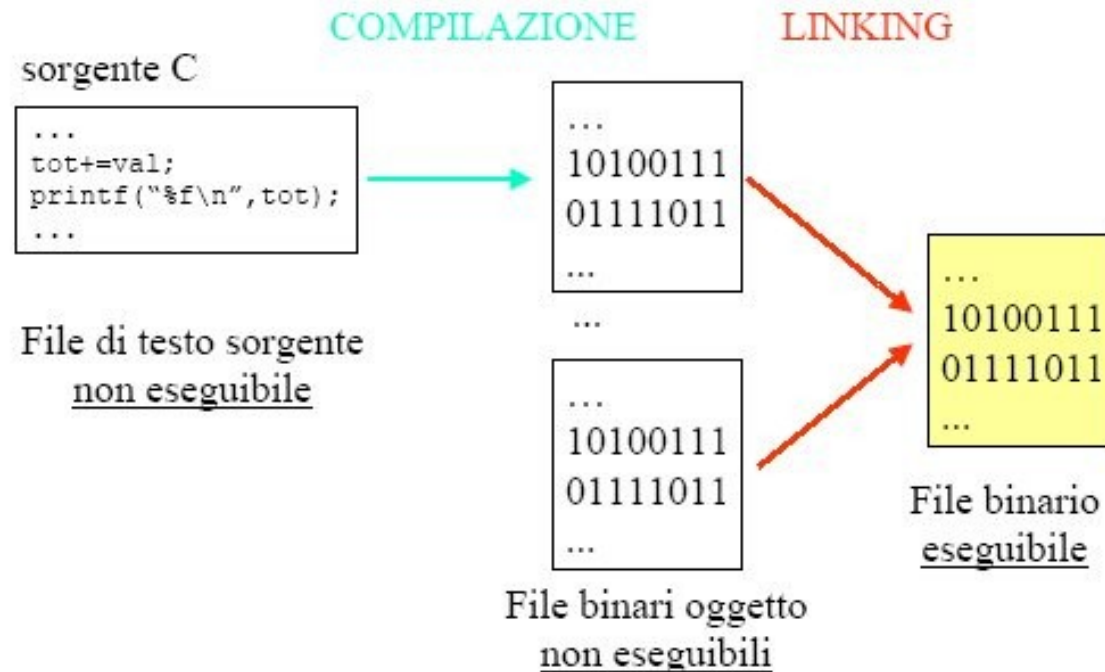
Linguaggi di programmazione

Il linguaggio C++

```
#include <stdio.h>
int main(void)
{
    printf("Hello, World!\n");
    return 0;
}
```

C/C++

Compilatore



- Traduce un programma sorgente in codice macchina