

# Corso di Architettura degli Elaboratori e Laboratorio (M-Z)

## Introduzione al corso

*Nino Cauli*



UNIVERSITÀ  
degli STUDI  
di CATANIA

Dipartimento di Matematica e Informatica

- Comprendere il funzionamento fisico del calcolatore
- Conoscere le parti che compongono la struttura di un calcolatore
- Conoscere i fondamenti teorici del funzionamento di un calcolatore
- Saper programmare un calcolatore utilizzando linguaggi assembly
- Conoscere i formati di rappresentazione delle informazioni

- **Rappresentazione dell'informazione:** Sistemi di numerazione, Conversione tra sistemi di numerazione, Numerazione binaria, Complemento a 2, Virgola fissa, Virgola mobile.
- **Circuiti Logici:** Algebra di Boole, Porte logiche, Circuiti combinatori, Analisi e sintesi di funzioni logiche, Circuiti sequenziali.
- **Componenti di un calcolatore:** CPU, ALU, Circuiti di Input/Output, Memoria, Circuiti di Controllo.
- **Instruction Set Architecture:** Architetture RISC e CISC, Istruzioni assembly, Modi di indirizzamento, Gestione degli interrupt.
- **Dispositivi di Input/Output:** BUS sincrono e asincrono, Porte Input/Output, USB, PCI.
- **Dispositivi di memoria:** Memoria principale, Memoria cache, Memoria di massa, Gerarchia delle memorie.

- **Libro di testo:**

Hamacher, Z. Vranesic, S. Zaky & N. Manjikian. **Introduzione all'architettura dei calcolatori**. Terza edizione, McGraw-Hill Education (Italy), 2013

- **Sito docente:**

Qui verranno pubblicate le slides aggiornate delle lezioni.

<https://nigno17.github.io/>

- **Sito Archelab:**

<http://www.dmi.unict.it/archelab/>

- **Sito Moodle:**

<http://moodle.dmi.unict.it/moodle/>



- **Prova scritta:**

Domande a risposta multipla e aperta. Scrittura di un programma assembly.

- **Prova orale:**

Domande sulla teoria.

- **Progetto (opzionale):**

Progetto legato agli argomenti del corso.

Personale o gruppi di 2/3 persone. Da 0 a 8 punti aggiuntivi.

## Nino Cauli

**Ufficio:** DMI, I blocco, II piano, stanza 325

**Email:** *nino.cauli@unict.it*

**Telefono:** 0957383007

**Sito web:** *<https://nigno17.github.io/>*

**Orario di ricevimento:** Lunedì e mercoledì dalle 15:00 alle 17:00

## Cos'è un calcolatore elettronico?

- È una macchina **ELETTRONICA** e **PROGRAMMABILE** che è in grado di eseguire **SEQUENZIALMENTE ISTRUZIONI** complesse in modo **AUTOMATICO**.
- È in grado di ricevere **INPUT** da un utente, generare **OUTPUT** di risposta e possiede una **MEMORIA** per immagazzinare dati ed istruzioni.



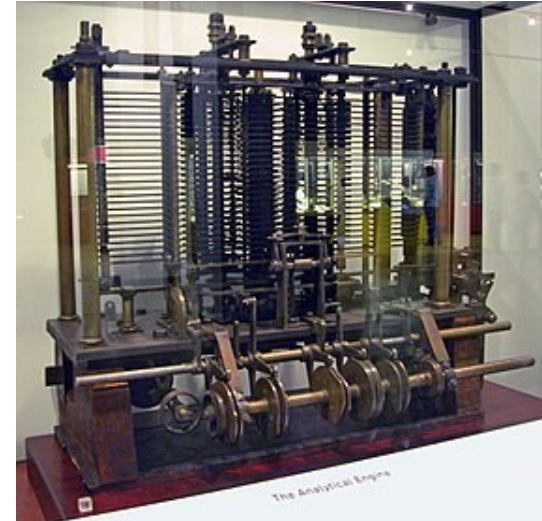
# I primi calcolatori meccanici (1600-1700)

- Dispositivi automatici per facilitare calcoli astronomici già presenti nell'antichità classica
- 1600-1700: primi dispositivi **MECCANICI** di epoca “moderna” per automatizzare calcoli matematici.
- 1642: **PASCALINA** (Blaise Pascal): somma e sottrazione di numeri.

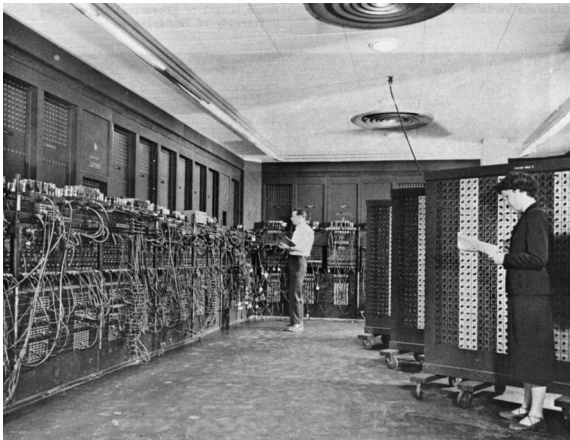




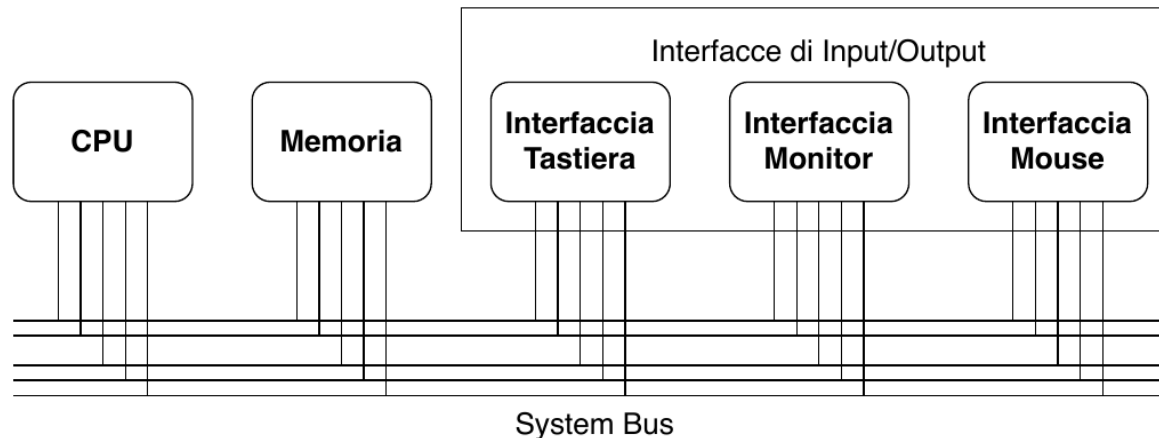
- 1801: **TELAIO JACQUARD** (Joseph-Marie Jacquard): Telaio meccanico **PROGRAMMABILE** tramite **SCHEDE PERFORATE**.
- 1837: **MACCHINA ANALITICA** (Charles Babbage): Progetto di calcolatore **GENERAL-PURPOSE**. Introduzione della **ALU**. Programmi e dati tramite **SCHEDE PERFORATE**.



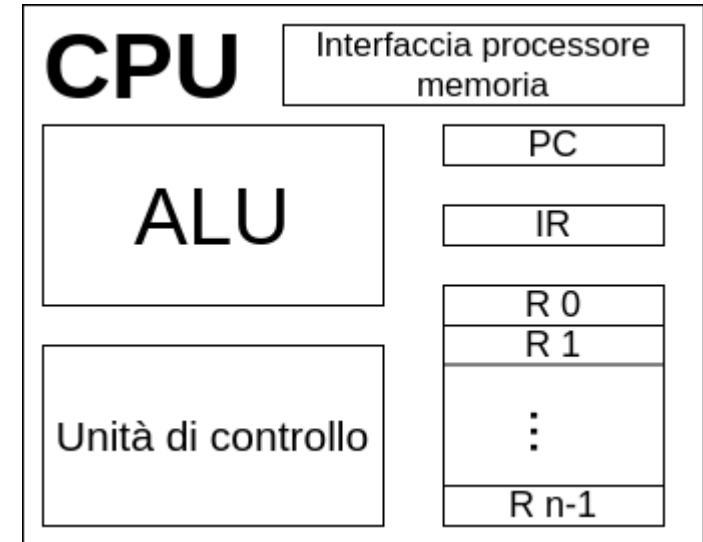
- Anni 40: Primi calcolatori **ELETTRONICI**. **COLOSSUS** (T. Flowers), **ENIAC** (Eckert & Mauchly). Valvole termoioniche.
- 1951: **EDVAC**. Architettura di **Von Neumann**. Dati e programmi nello stesso spazio di memoria.
- Fine anni 50: Introduzione dei **TRANSISTORI** e dei **CIRCUITI INTEGRATI**.



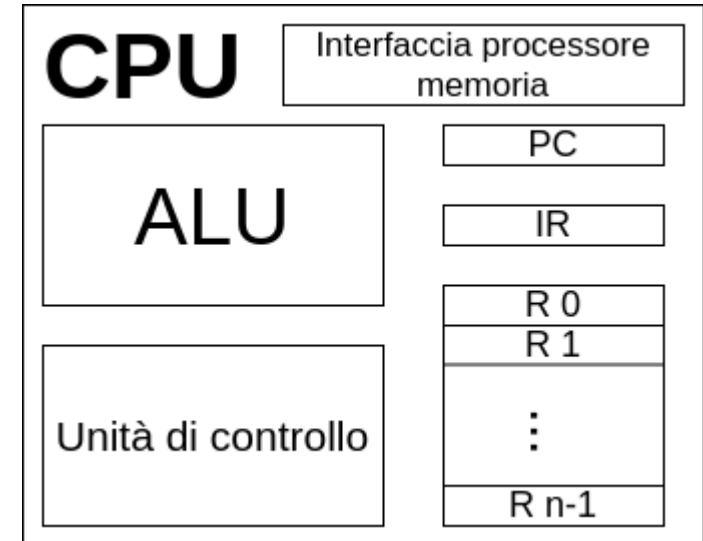
- **CPU**: esegue istruzioni elementari
- **MEMORIA**: contiene il programma (sequenza di istruzioni elementari) che la CPU deve eseguire e i dati necessari
- **INTERFACCE DI INPUT/OUTPUT**: circuiti elettronici che permettono di connettere la CPU al mondo esterno
- **BUS DI SISTEMA**: insieme di collegamenti elettrici che interconnettono i vari componenti di un calcolatore



- È un **CIRCUITO ELETTRONICO INTEGRATO** (chip) con il ruolo di **CERVELLO** del calcolatore
- Capace di caricare ed eseguire le **ISTRUZIONI ELEMENTARI** necessarie per eseguire i **PROGRAMMI**
- Esempi di istruzioni elementari: operazioni aritmetiche, operazioni logiche, confronti, salti incondizionati e condizionati.



- L'**UNITÀ ARITMETICA-LOGICA (ALU)** esegue le operazioni aritmetiche e logiche necessarie ad eseguire le istruzioni
- L'**UNITÀ DI CONTROLLO** coordina il funzionamento (carica le istruzioni e dati, gestisce i salti)
- **REGISTRI**: blocchi di memoria interni alla CPU contenenti l'istruzione in esecuzione, indirizzi e dati
- **INTERFACCIA PROCESSORE MEMORIA** gestisce il trasferimento di dei dati tra memoria e CPU



- Le unità memoria sono usate per immagazzinare informazione necessaria per eseguire i programmi
- Sono circuiti elettronici in grado di preservare l'informazione che può essere costituita da:
  - **ISTRUZIONI**, eseguite dalla CPU
  - **DATI**, utilizzati dalle istruzioni eseguite
- La memoria si può dividere in **MEMORIA CENTRALE** e **MEMORIA DI MASSA**

- La memoria centrale è **VELOCE**, con **CAPACITÀ LIMITATA** e **VOLATILE**
- La tecnologia usata si chiama **MEMORIA AD ACCESSO CASUALE (RAM)**
- Organizzata su **LIVELLI** (bassi + veloci e – capienti, alti – veloci e + capienti)
- **CACHE**: livello più basso (molto veloce, integrata nel processore)



- La memoria secondaria è **LENTA**, con **CAPACITÀ ELEVATA** e **NON VOLATILE**
- Viene usata per immagazzinare **GROSSE QUANTITÀ** di dati in modo **PERMANENTE** o per **LUNGI PERIODI**
- Varie tecnologie disponibili: **DISCHI MAGNETICI**, **DISCHI OTTICI** (CD e DVD), **MEMORIE FLASH**, etc.





- Un calcolatore ha necessita di comunicare con il mondo esterno
- Le interfacce di I/O sono tutti i circuiti elettronici che permettono alla CPU di interagire con l'utente:
  - Monitor
  - Tastiera
  - Mouse
  - Stampante
  - Connessioni di rete
  - ...



- E' un insieme di collegamenti elettrici che **INTERCONNETTE** tutti i componenti di un calcolatore: CPU, memoria, interfacce di I/O
- Ogni collegamento ha un ruolo ben preciso e “trasporta” una tipologia di informazione ben precisa
- Il numero di collegamenti del bus dipende da alcune caratteristiche specifiche di CPU e memoria

- Il calcolatore elettronico esegue **SEQUENZIALMENTE** una serie di **ISTRUZIONI**
- Le istruzioni definiscono delle operazioni da eseguire e sono raggruppate in **PROGRAMMI**
- Spesso le operazioni devono essere eseguite su dei **DATI**
- L'utente può interagire con il calcolatore tramite le **INTERFACCE DI I/O (PERIFERICHE)**

- Calcolatori sono costituiti da circuiti a **DUE STATI STABILI**
- Perciò le istruzioni e i dati sono rappresentati da **SEQUENZE** di **CIFRE BINARIE (bit)**
- Per convenzione una sequenza di 8 bit è detta **Byte**
- Il calcolatore lavora su gruppi di bit detti **PAROLE** di lunghezza da 8 a 64 bit (sempre potenze di 2)



# Come si programma?

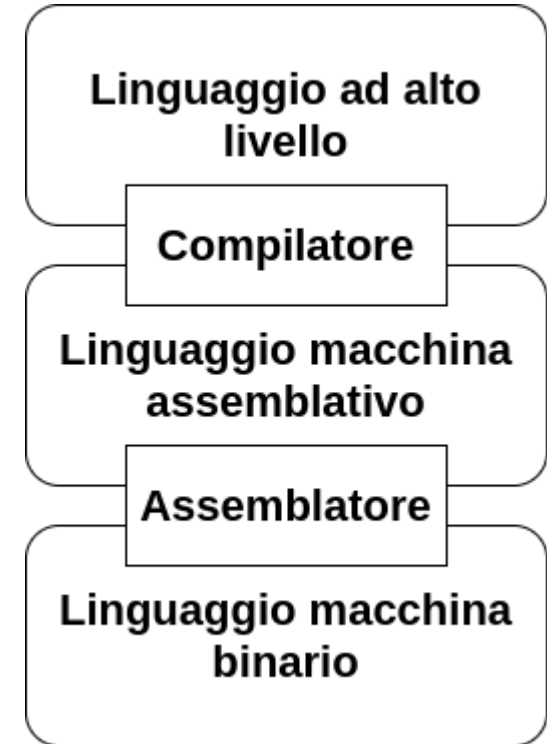
```
000111010101110011010100110101001010010001001011101010  
01110101000011010101010101111000110100100101010101000101  
101010101010011101011010100101010100101011
```



Load	R2, LOC
Add	R4, R2, R3
Store	R4, LOC

# Come si programma?

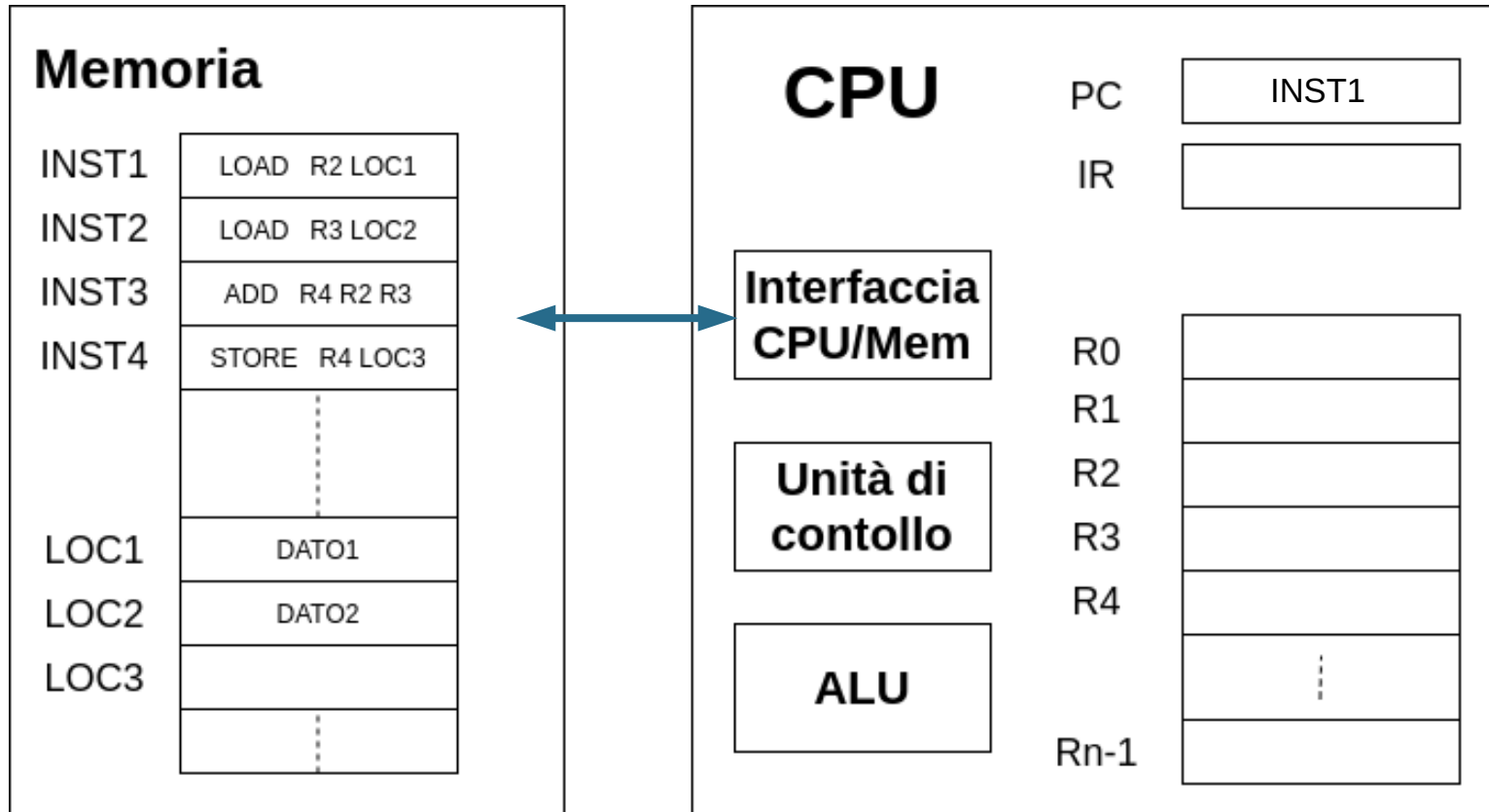
- Il programmatore scrive i programmi in **LINGUAGGIO ASSEMBLATIVO (ASSEMBLY)**
- Il programma assembly viene tradotto in sequenze binarie dall'**ASSEMBLATORE**
- Linguaggi ad alto livello (C, C++, etc.) ancora più espressivi
- Il **COMPILATORE** traduce il codice ad alto livello in codice assembly



## Passi operativi elementari per ciascuna istruzione:

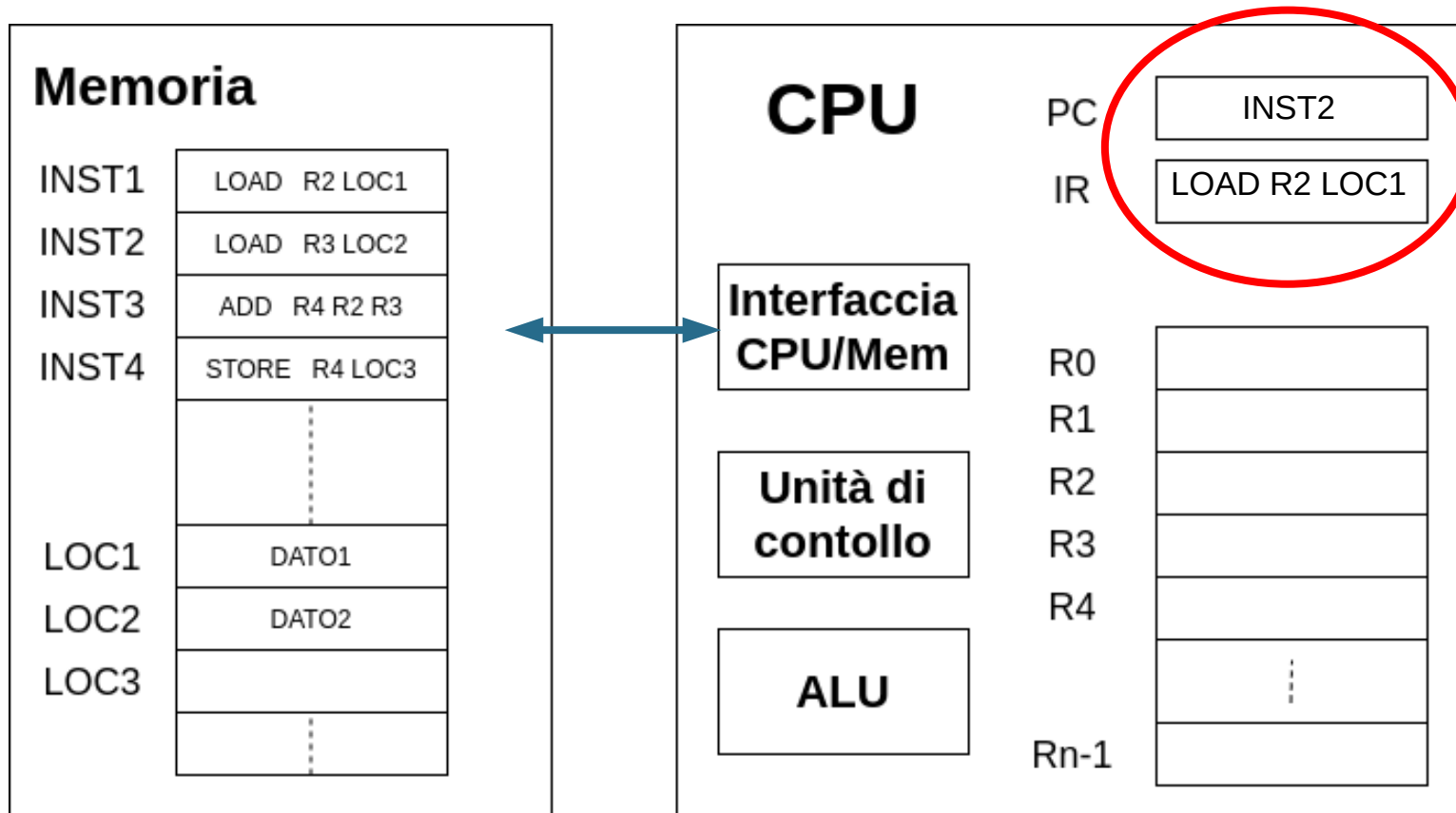
- **PRELIEVO**: prelievo della prossima istruzione dalla memoria (scrivere la prossima istruzione nel registro di istruzione IR)
- **DECODIFICA**: decodifica dell'istruzione (quale operazione bisogna eseguire? Dove si trovano i dati da usare?)
- **ESECUZIONE**: esecuzione dell'istruzione (leggere o scrivere un dato in memoria, eseguire operazioni matematiche e logiche sui registri)

## Stato Iniziale

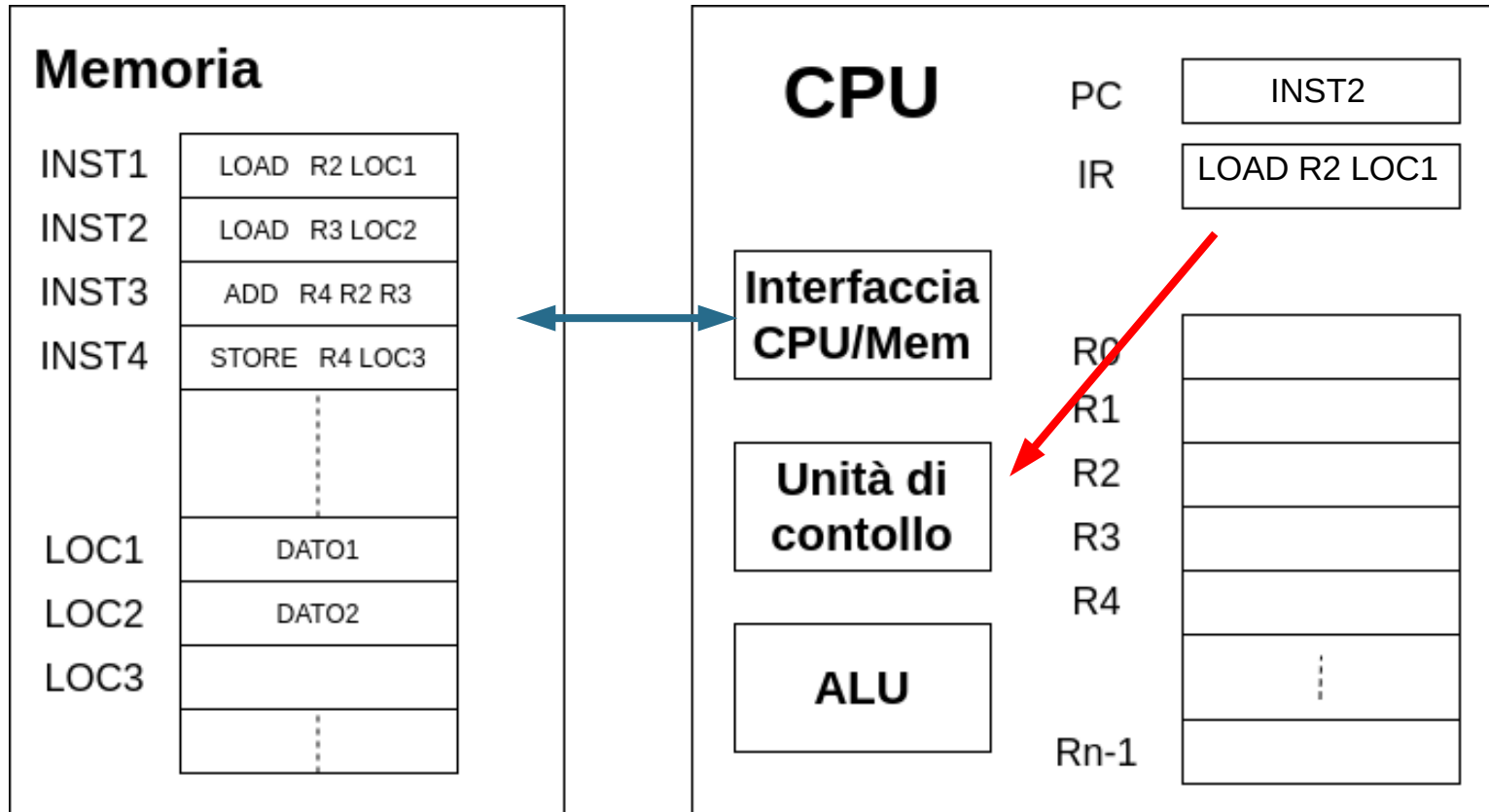




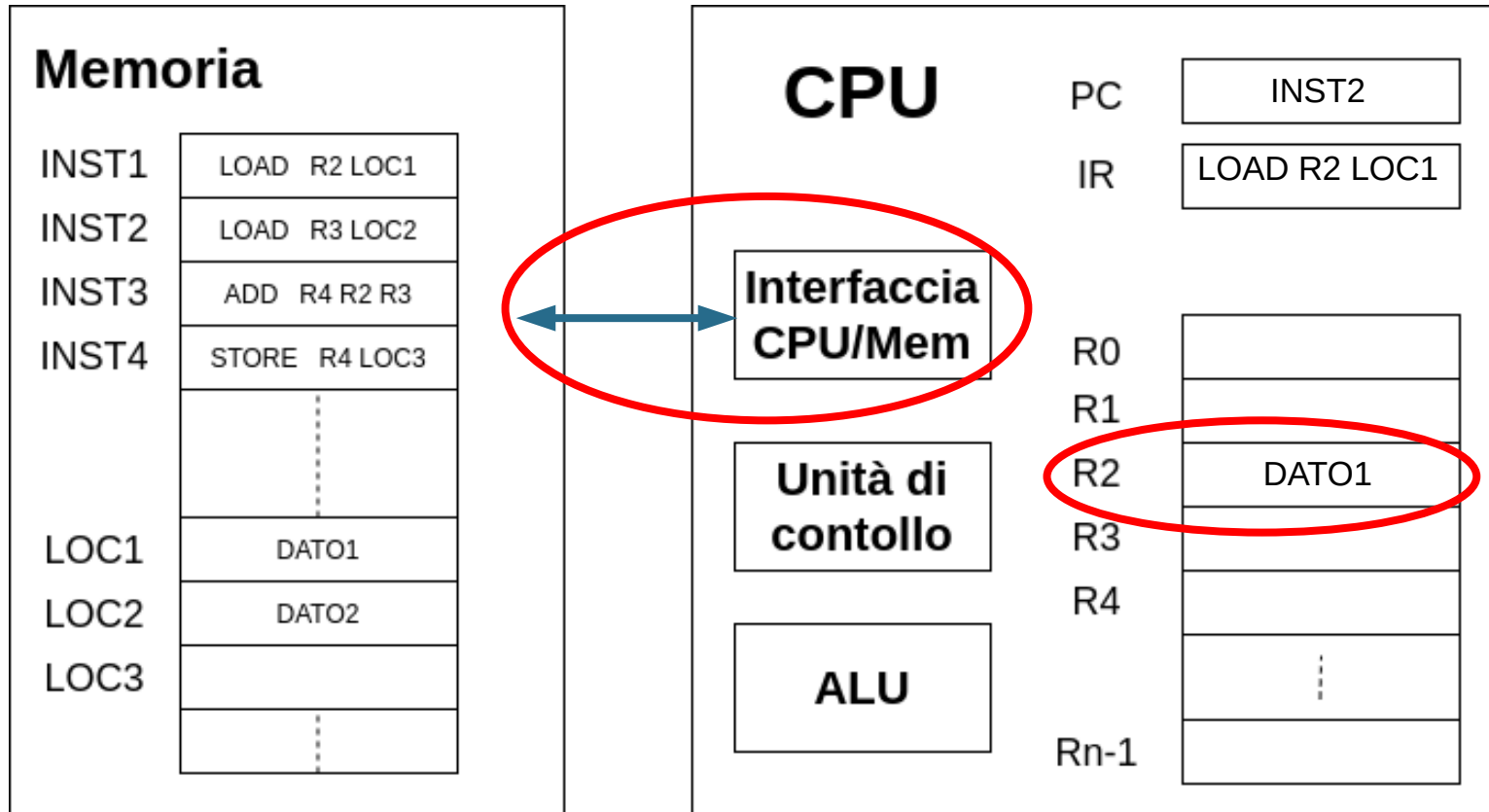
## Prelievo



## Decodifica



## Esecuzione



## Come avvisare il processore inviare dati da/a periferiche di input/output?

Si usano le **INTERRUZIONI!** (segnali di avviso)

Funzionamento:

- segnalazione della richiesta
- sospensione (se è il caso) dell'esecuzione del programma corrente, e salvataggio dell'informazione necessaria alla sua successiva ripresa
- esecuzione della relativa **ROUTINE DI SERVIZIO DELL'INTERRUZIONE**
- al termine di questa, ripristino dell'informazione salvata e ripresa dell'esecuzione interrotta