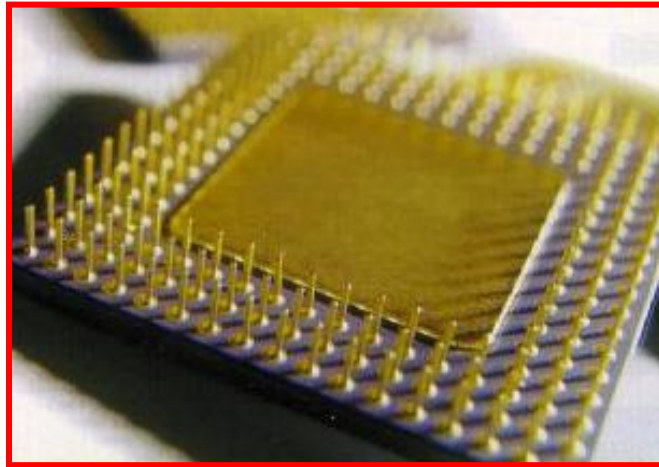
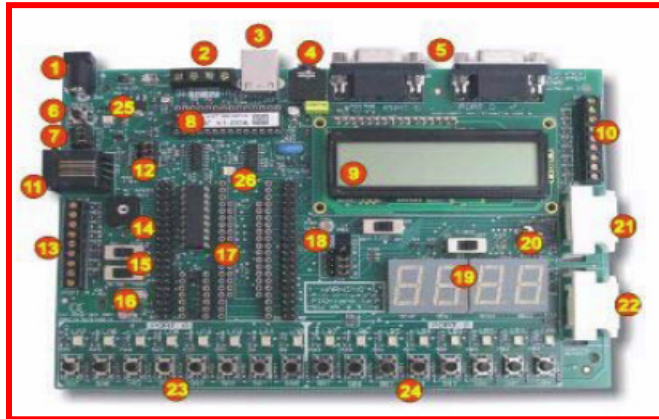




# Università degli studi di Cassino

**Corso di Laurea in  
Ingegneria della Produzione Industriale**



## **Corso di Informatica Applicata**

### **Lezione 3**

Ing. Saverio De Vito  
e-mail: [saverio.devito@portici.enea.it](mailto:saverio.devito@portici.enea.it)  
Tel.: +39 081 7723364



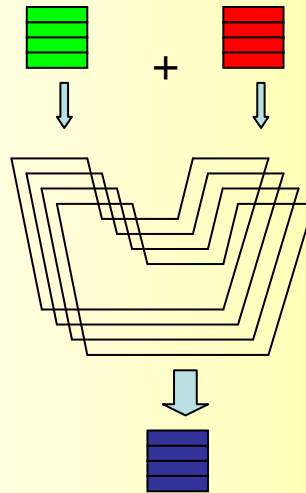
# Breve Riepilogo - Il processore

- Abbiamo inquadrato il processore nell' ambito di architetture von Neumann-like;
  - Architettura di Von Neumann
  - Ciclo di Von Neumann
- Il processore come esecutore continuo di –istruzioni-
  - L' istruzione è l' operazione atomica eseguibile dalla macchina processore
  - Il set di istruzioni processore (ISA) definisce la macchina virtuale “standard”
  - Dualismo CISC-RISC, chi vince
  - In realtà, le istruzioni a livello macchina “standard” sono, in molti casi, microprogrammate ossia eseguite da una macchina virtuale sottostante detta Microprogramma



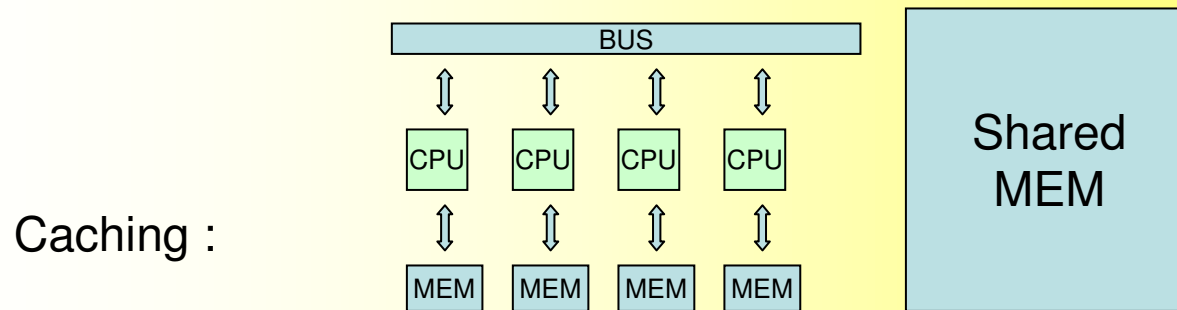
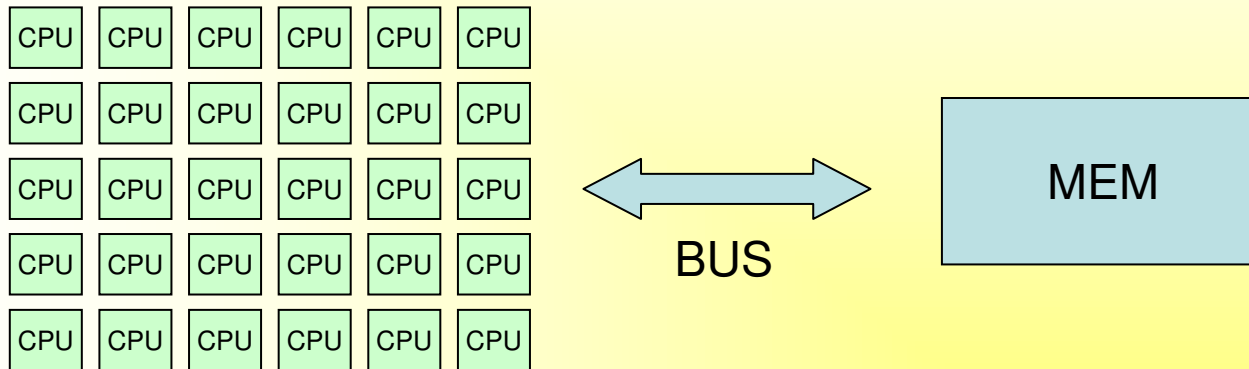
# Esecuzione Parallela

- Architetture SIMD [Single Instruction Multiple Data (flows)]
  - Es.: Architetture con ALU vettoriali



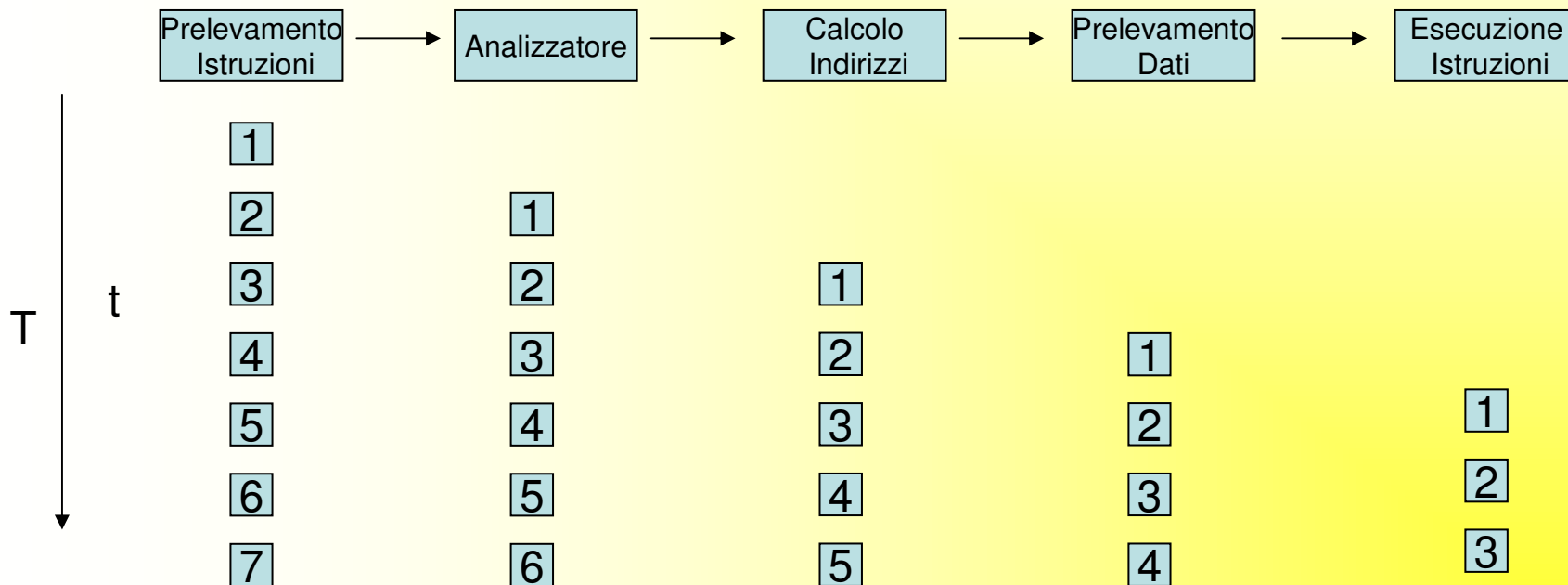
# Esecuzione Parallela

- Architetture MIMD
  - Es.: Architetture multiprocessore
    - Conflitti di accesso in memoria



# Pipelines

- Questa strategia viene adottata per ottenere una sorta di parallelismo in macchine di tipo SIMD:
  - L' esecuzione della singola istruzione viene assegnata a parti specializzate del processore che possono essere alimentate (nella maggior parte dei casi) in maniera autonoma



# Istruzioni Macchina

- Le istruzioni macchina sono tipicamente costituite da stringhe binarie che codificano sostanzialmente:
  - La tipologia di istruzione
  - Gli operandi (Registri interni, Registri di Memoria, etc.)
  - La modalità di indirizzamento (semantica degli operandi)
- Le istruzioni della macchina possono essere definite mediante un linguaggio “Assembly”, molto vicino al livello macchina standard ma più semplice da trattare per un operatore umano
- La vicinanza alla macchina standard fanno sì che il linguaggio assembly risulti fortemente dipendente dalla macchina standard pur garantendo un certo carattere di generalità
- Durante questa unità didattica cercheremo di estrarre le caratteristiche generali di un linguaggio Assembly



# Linguaggio Assembly

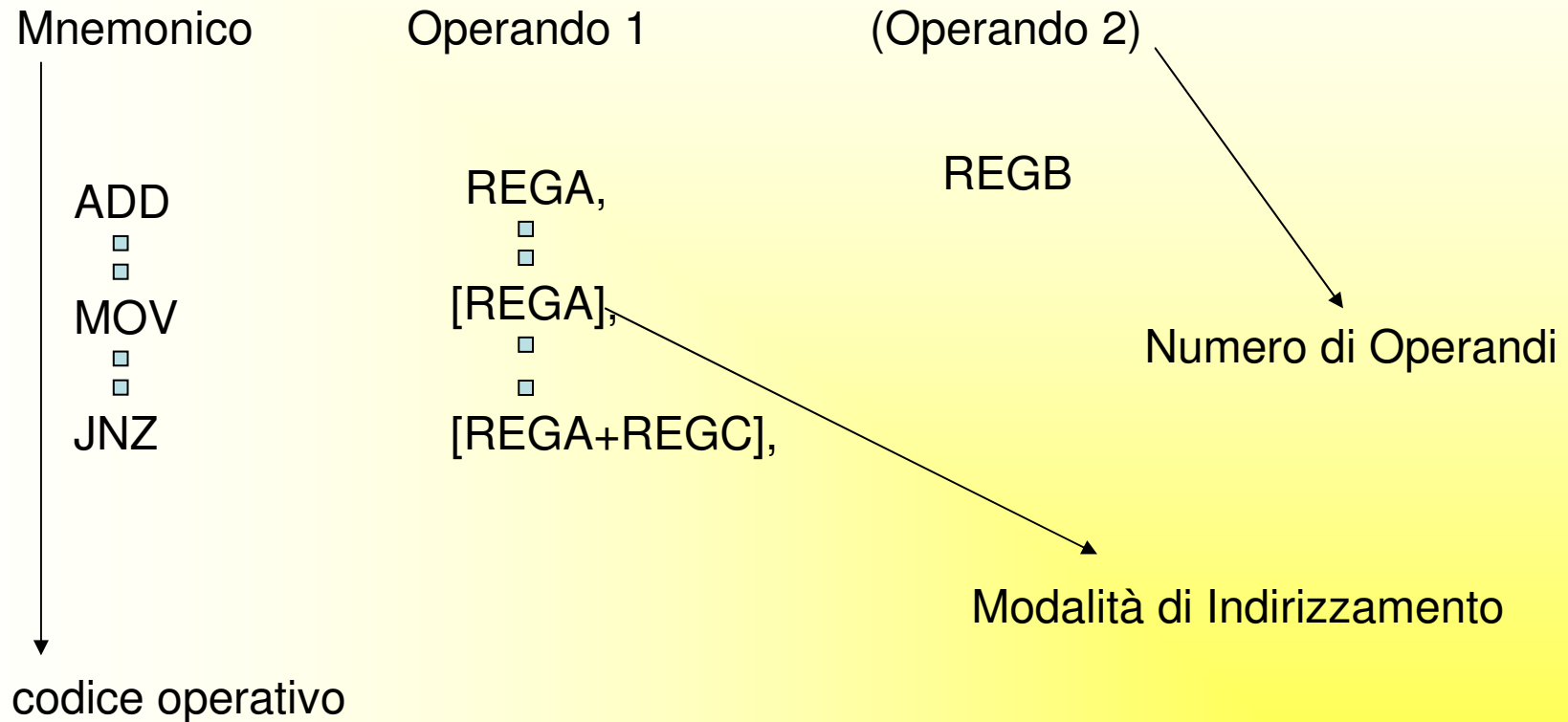
- Il set di istruzioni Assembly, unito al
  - formato del programma,
  - alle modalità di specifica dette direttive,

Definisce il linguaggio Assembly

Il linguaggio assembly può essere associato però a macchine virtuali leggermente differenti dalla macchina “standard”



# Formato Istruzione Assembly



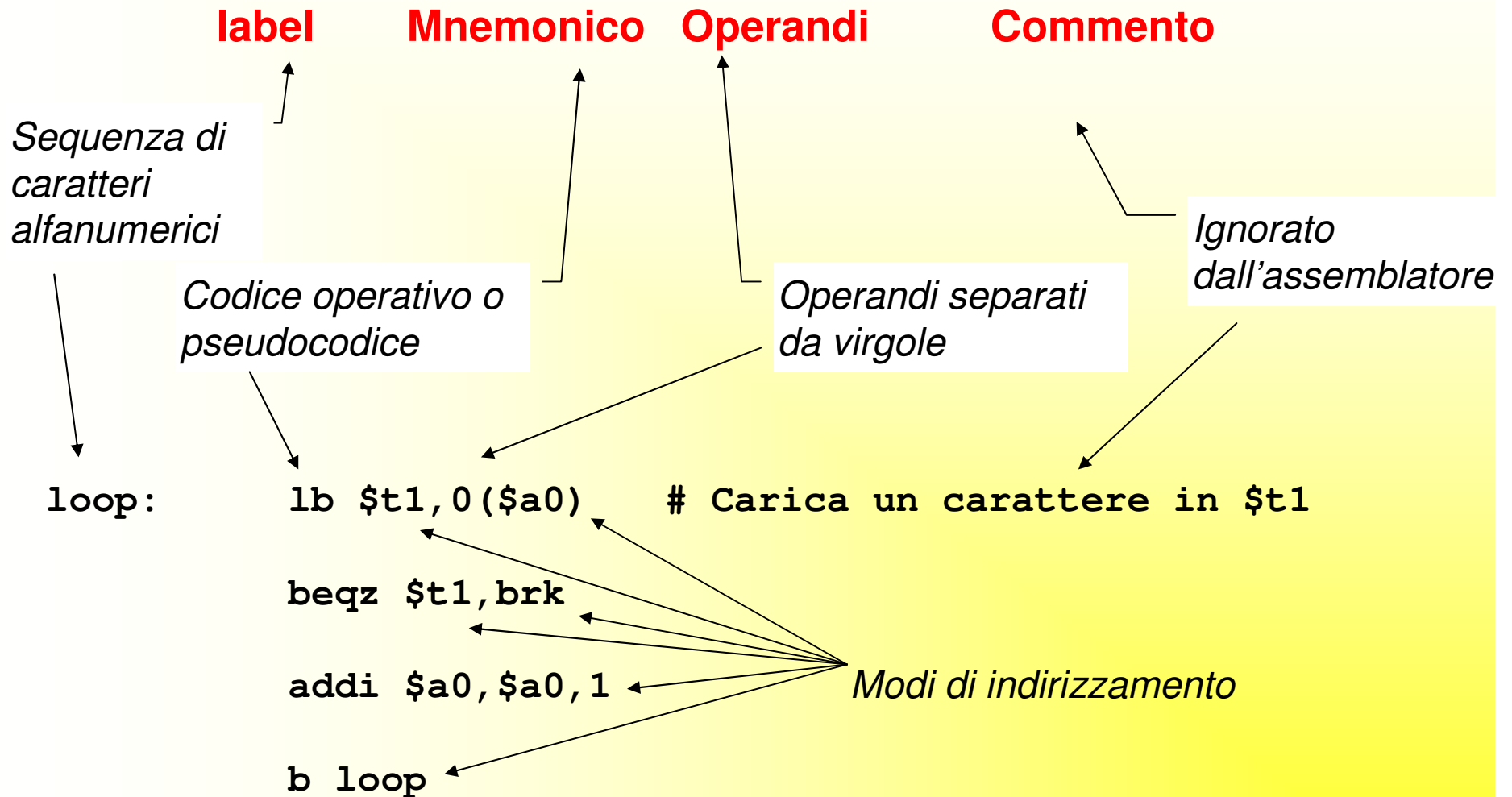


# Programma Assembly

- Un programma assembly è tipicamente formato da sequenze di istruzioni di formato simile a quello precedentemente esposto.
- Le istruzioni possono essere etichettate per delimitare dei riferimenti lungo il Codice
- Le stesse istruzioni possono essere seguite da commenti che vengono ignorate in fase di traduzione verso il linguaggio della macchina standard
- Il codice vero e proprio viene tipicamente preceduto da direttive che specificano il processo di traduzione implementato dall'Assemblatore



# Esempio



# Modalità di indirizzamento

- Gli operandi di una istruzione assembly descrivono in maniera sintetica gli oggetti della computazione effettuata dalla macchina virtuale associata.
- L'operando, eventualmente un numero o un codice mnemonico relativo ad un registro, può di volta in volta assumere il valore semantico di un numero su cui effettuare la computazione, un registro contenente il valore, una locazione di memoria contenente il valore, un registro contenente l'indirizzo di una locazione di memoria ecc.
- La semantica dell'operando viene attribuita attraverso particolari codici (parentesi quadre, \$ etc.)



# Indirizzamento Immediato

- La semantica più semplice per i codici operando è data dall' utilizzo dell' indirizzamento immediato.
- In questo caso il codice operando (numerico) è esso stesso il valore su cui si effettuerà la computazione
- Nell' istruzione macchina corrispondente sarà incluso il valore dell' operando. Non sarà necessario un ulteriore accesso alla memoria, una volta prelevata l' istruzione dalla memoria il processore avrà a disposizione tutte le informazioni necessarie all' esecuzione dell' istruzione
- Es.: `ADD AX, 1` (dove `AX` è un registro “accumulatore”)



# Indirizzamento Diretto

- Leggermente più complesso è il caso dell'indirizzamento diretto dove il codice operando rappresenta l'indirizzo del registro di memoria che contiene il valore necessario all'esecuzione dell'istruzione
- Ipotizzando di avere & come specificatore di indirizzamento diretto avremo:
- **ADD AX, &A12E**



# Indirizzamento tramite registri

- Concettualmente simile all' indirizzamento diretto:

Il valore dell' operando è contenuto nel registro specificato:

**ADD AX, BX** (BX è un secondo registro dell' architettura ipotetica cui ci riferiamo)



# Indirizzamento Indiretto

- In questa modalità l' indirizzo specifica quale parola contiene l' indirizzo del valore cercato e non il valore stesso come avveniva nell' indirizzamento diretto.
- `ADD AX, [BX]`



# Modalità di indirizzamento (2)

- Le modalità di indirizzamento presentate possono essere definite come principali.
- Esistono altri schemi di indirizzamento particolari che non tratteremo, ed altri come l'indirizzamento tramite indice che analizzeremo più avanti.
- Gli schemi di indirizzamento supportati sono fortemente dipendenti dalla macchina. Ogni processore possiede un proprio insieme di schemi di indirizzamento che differisce da quello di altre architetture.





# Istruzione Macchina (2)

- L'istruzione macchina sarà quindi una stringa di bit di formato generale:

COD. Operativo

Campo Indirizzamento 1

Campo Indirizzamento 2 (?)

- Le dimensioni in bit dei campi suddetti possono variare:
  - Ad es. il campo di indirizzamento nel caso l'operando sia un registro (indirizzamento diretto tramite registri) conterrà un numero di bit atto a codificare il numero di registri della macchina in esame.
- Il formato specifico delle istruzioni macchina è variabile con la tipologia di istruzione ed lo schema di indirizzamento

