



UNIVERSITA' DEGLI STUDI DI PARMA

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

# Fondamenti di Informatica B

---

Lezione n. 12

Alberto Broggi – Gianni Conte

A.A. 2005-2006

# Fondamenti di Informatica B

---

## Lezione n.12

- STACK, SOTTOPROGRAMMI, INTERRUZIONI
- GESTIONE DELLE INTERRUZIONI
- SET DI ISTRUZIONI

**In questa lezione approfondiremo le caratteristiche interne delle architetture con attenzione alle chiamate a sottoprogramma.**

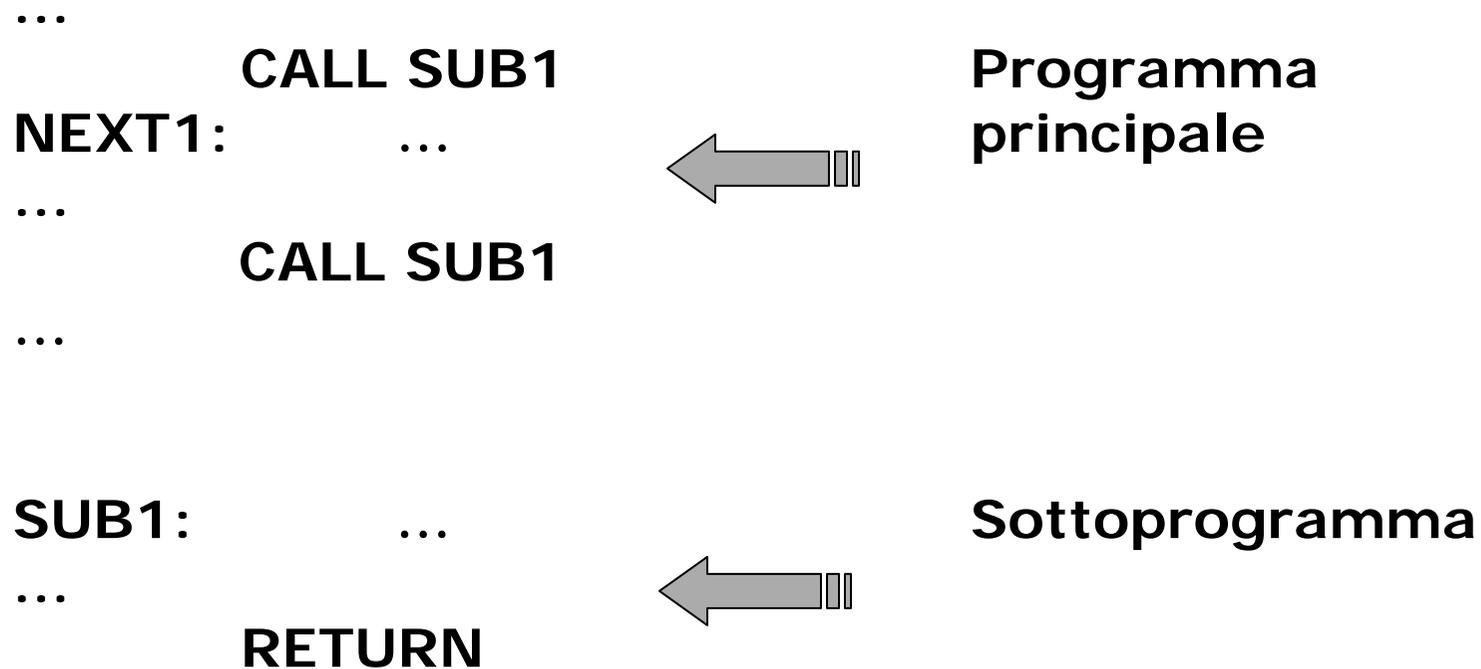
**Illustreremo le ragioni e i meccanismi legati alla presenza delle interruzioni.**

**Descriveremo poi le caratteristiche principali del set di istruzione di una architettura di una CPU di tipo generale.**

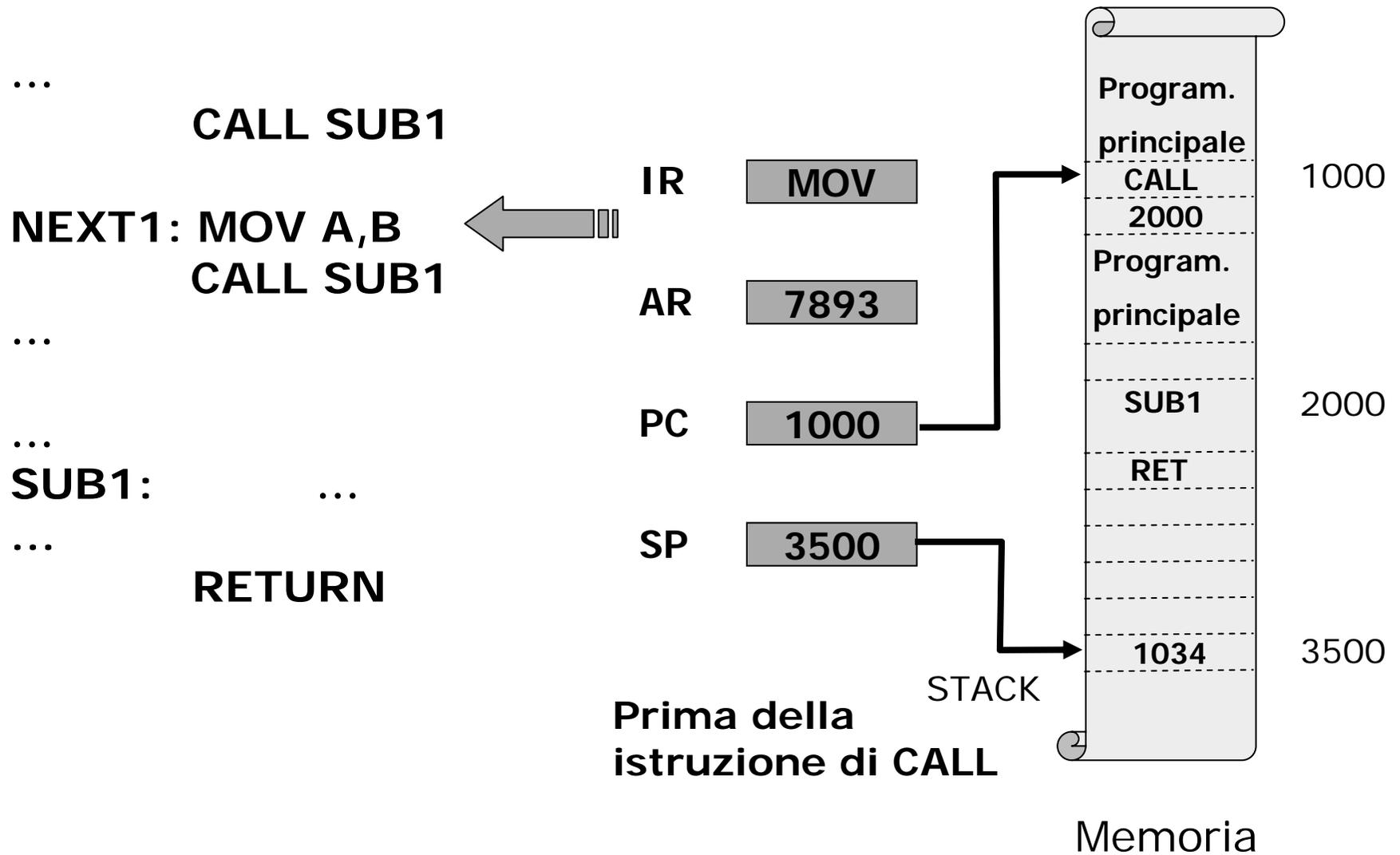
# Sottoprogrammi

---

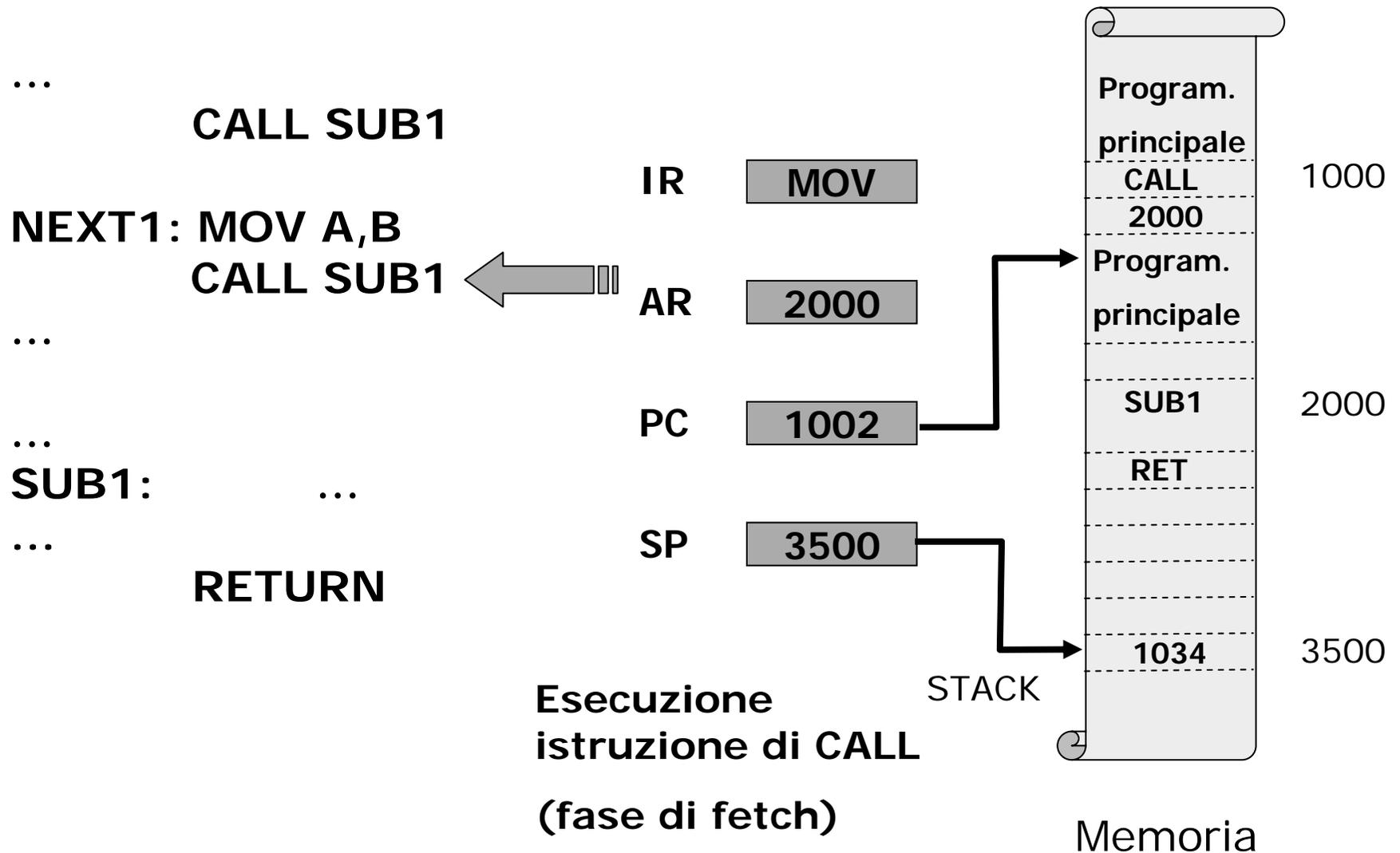
Un sottoprogramma è una sequenza di istruzioni che viene usata più volte nel corso di un programma



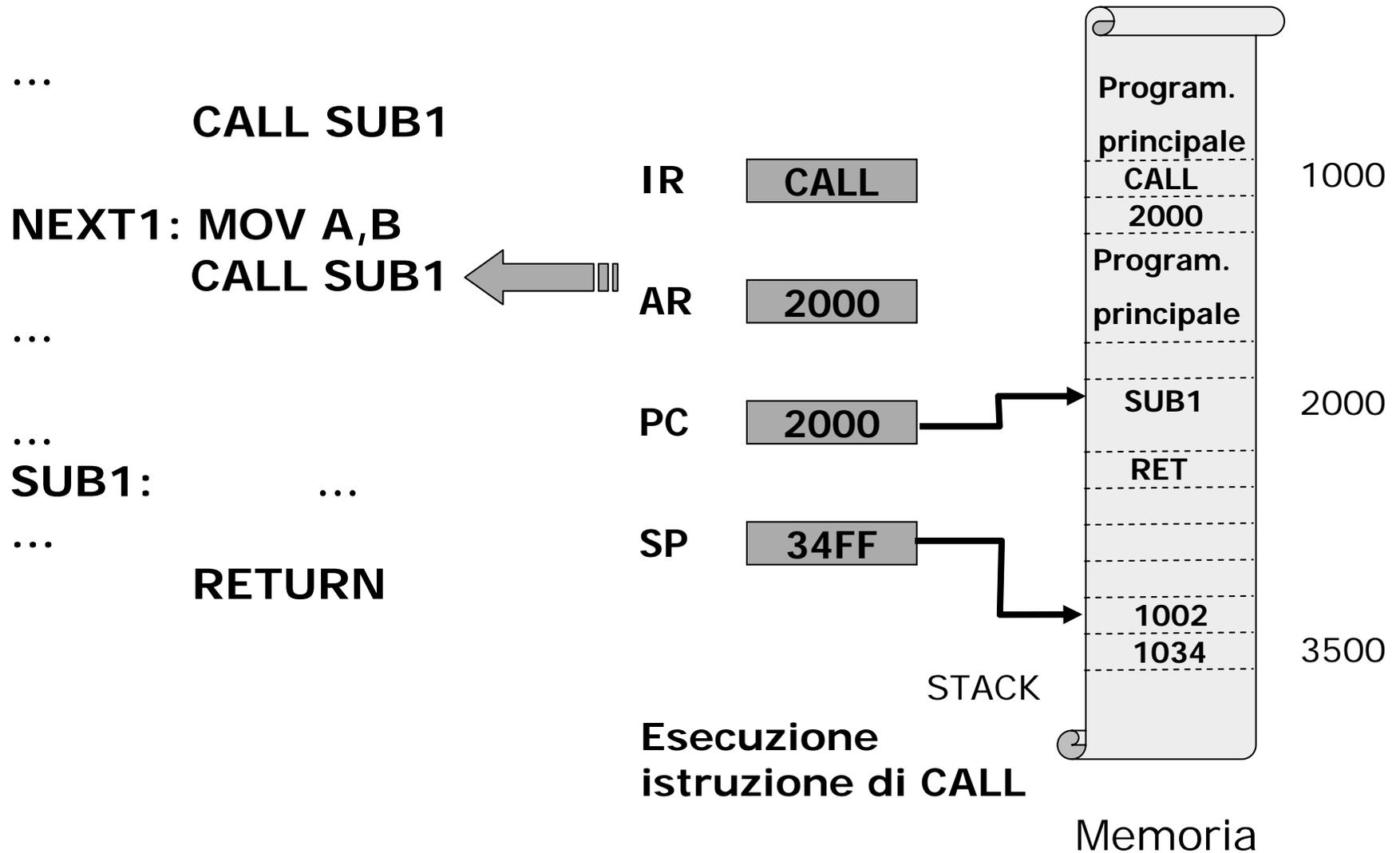
# Sottoprogrammi



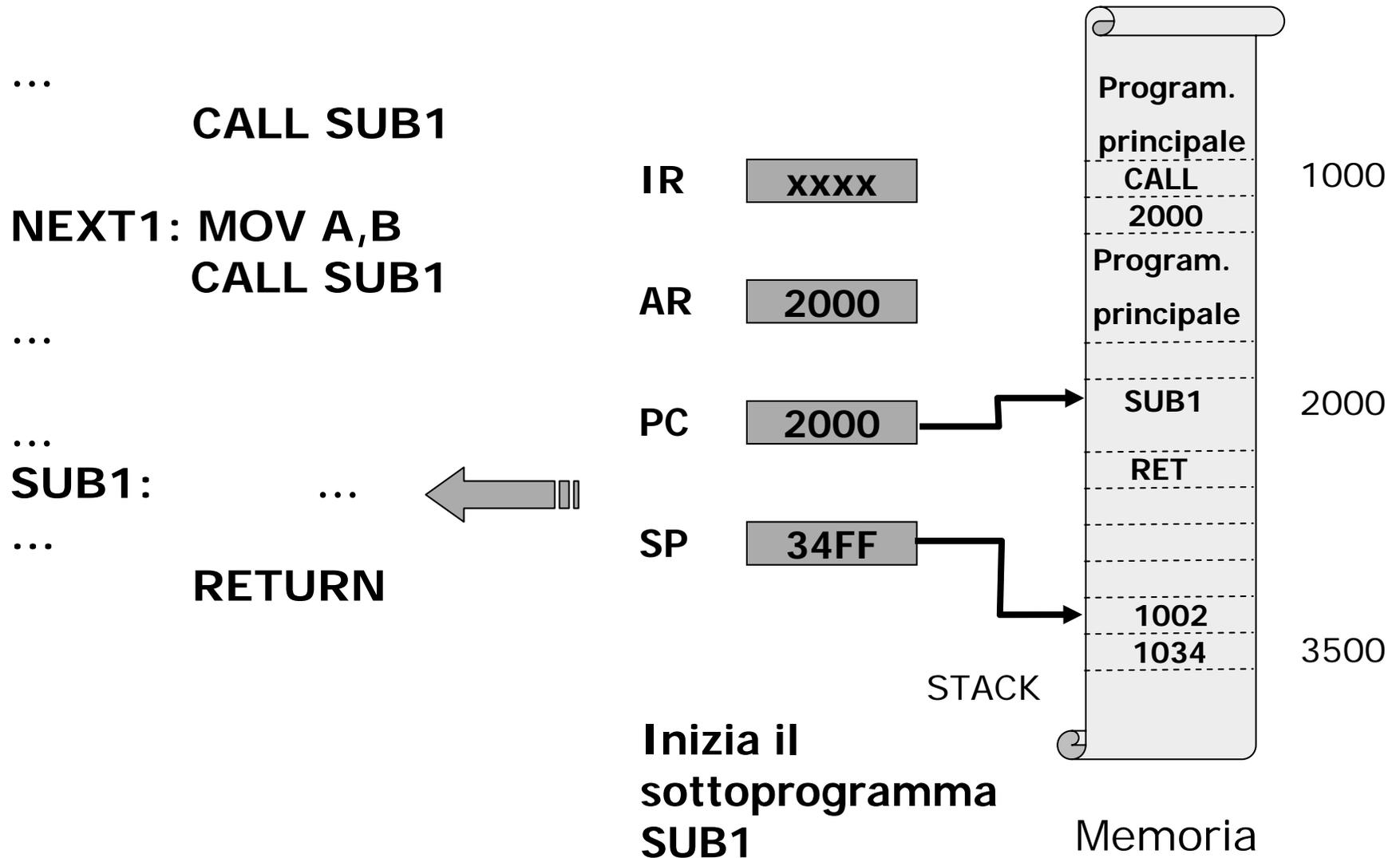
# Sottoprogrammi



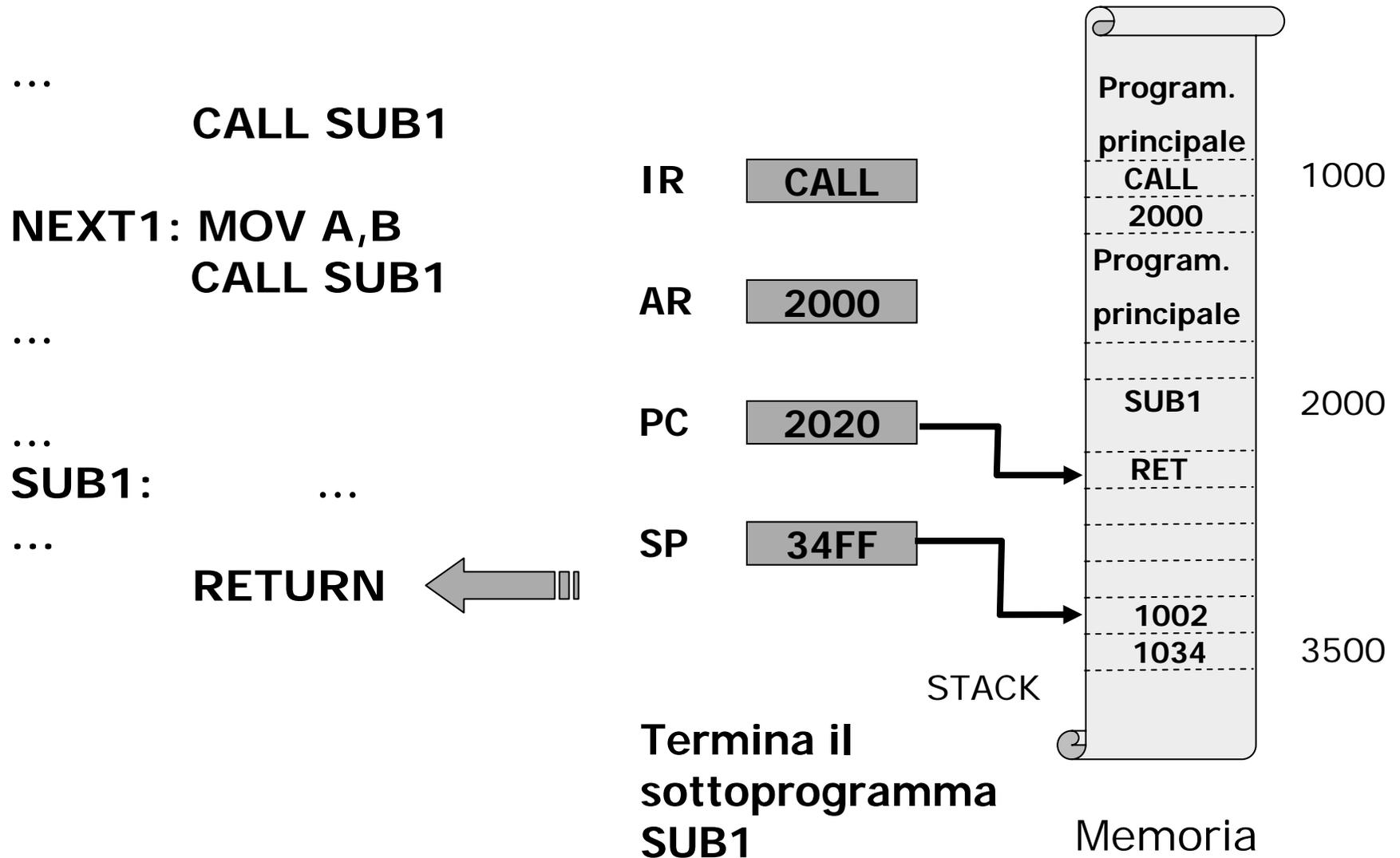
# Sottoprogrammi



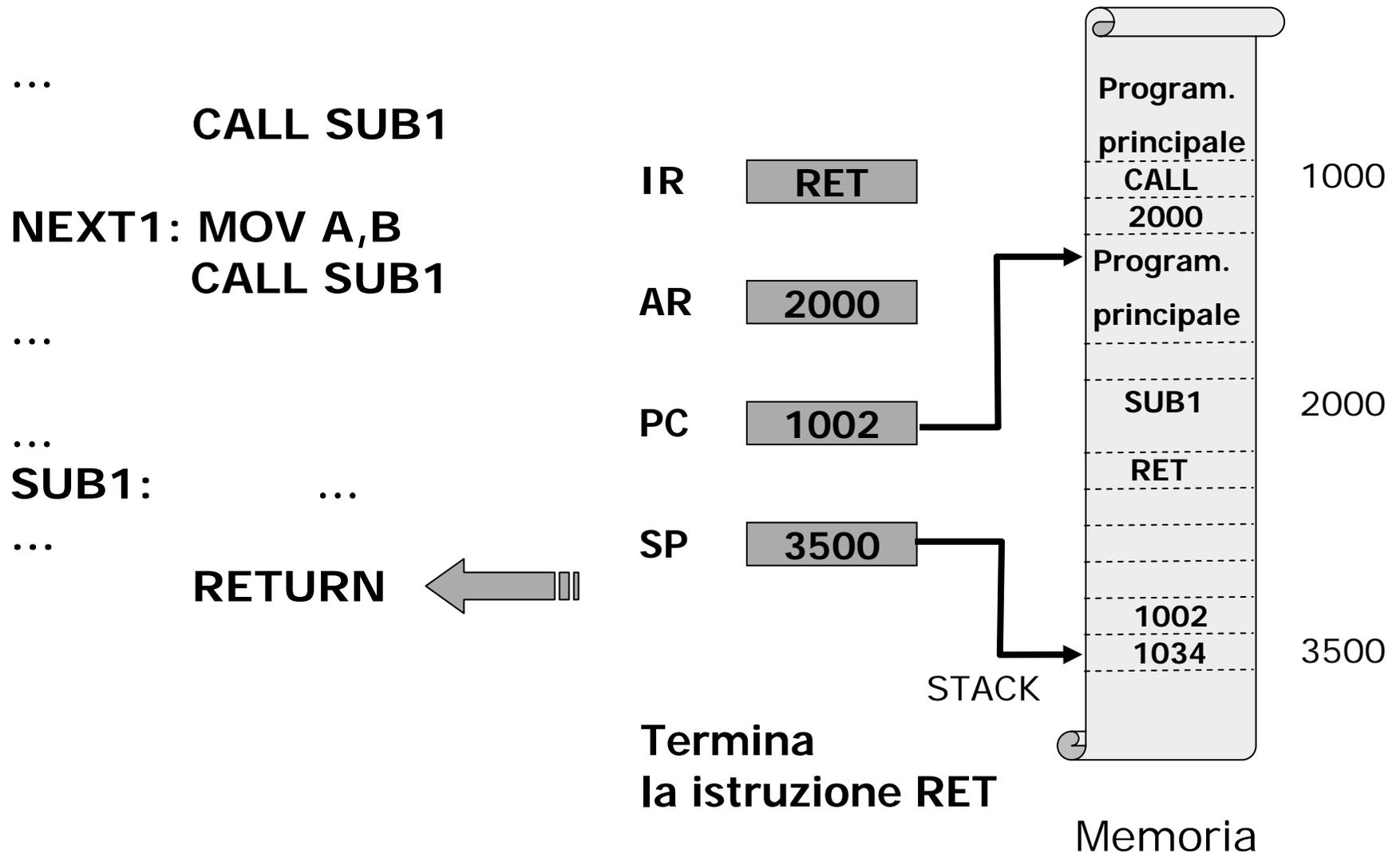
# Sottoprogrammi



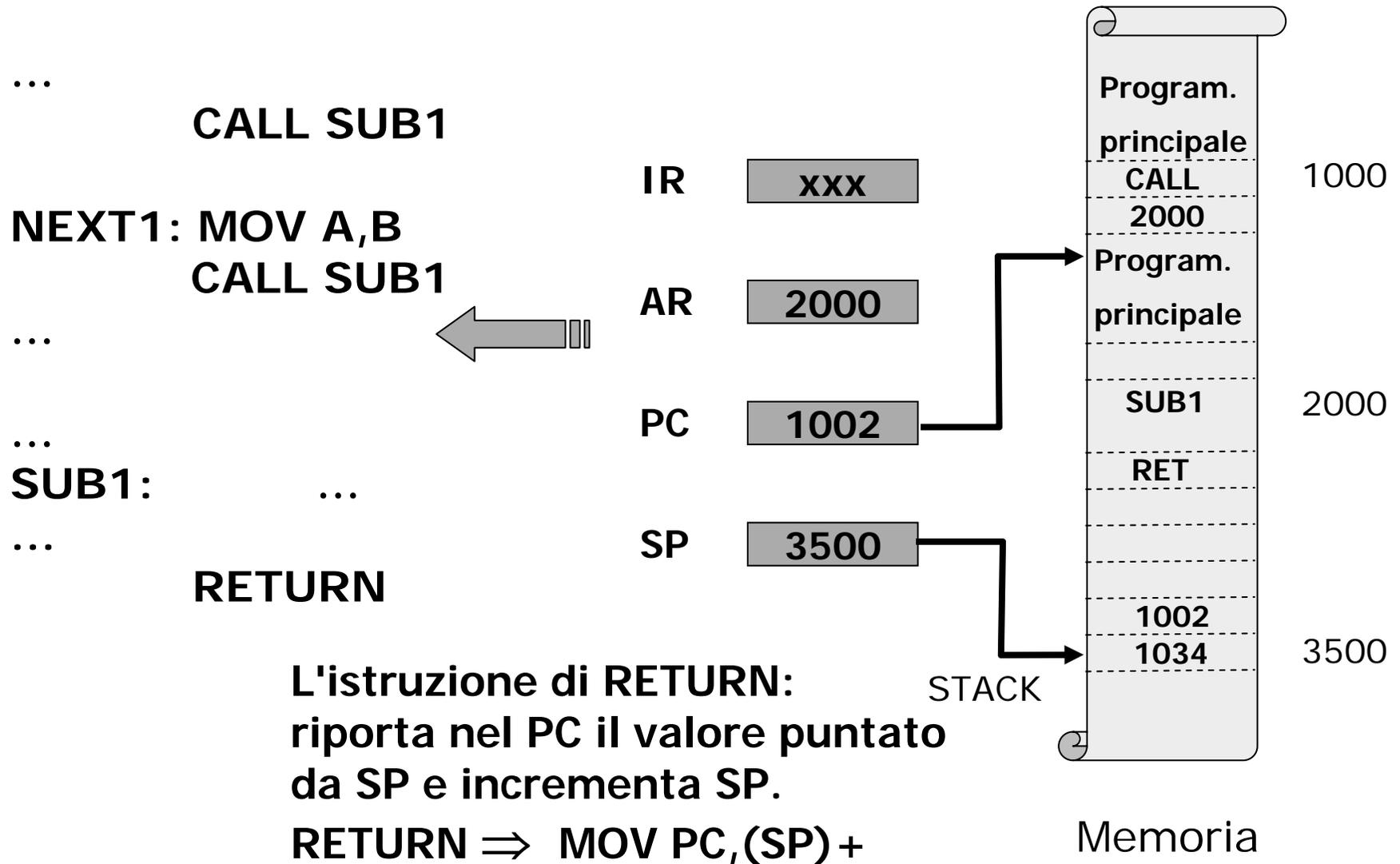
# Sottoprogrammi



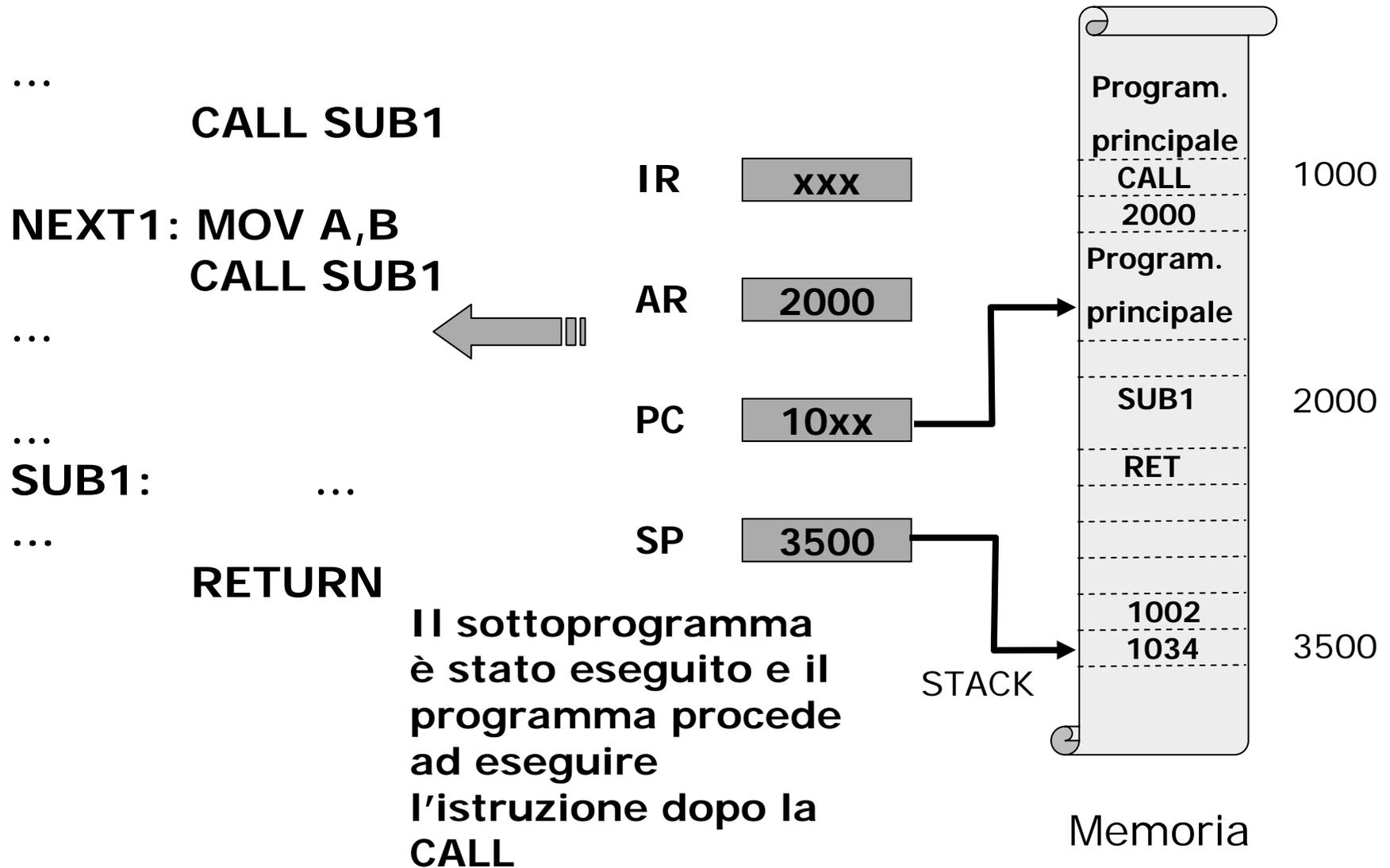
# Sottoprogrammi



# Sottoprogrammi



# Sottoprogrammi

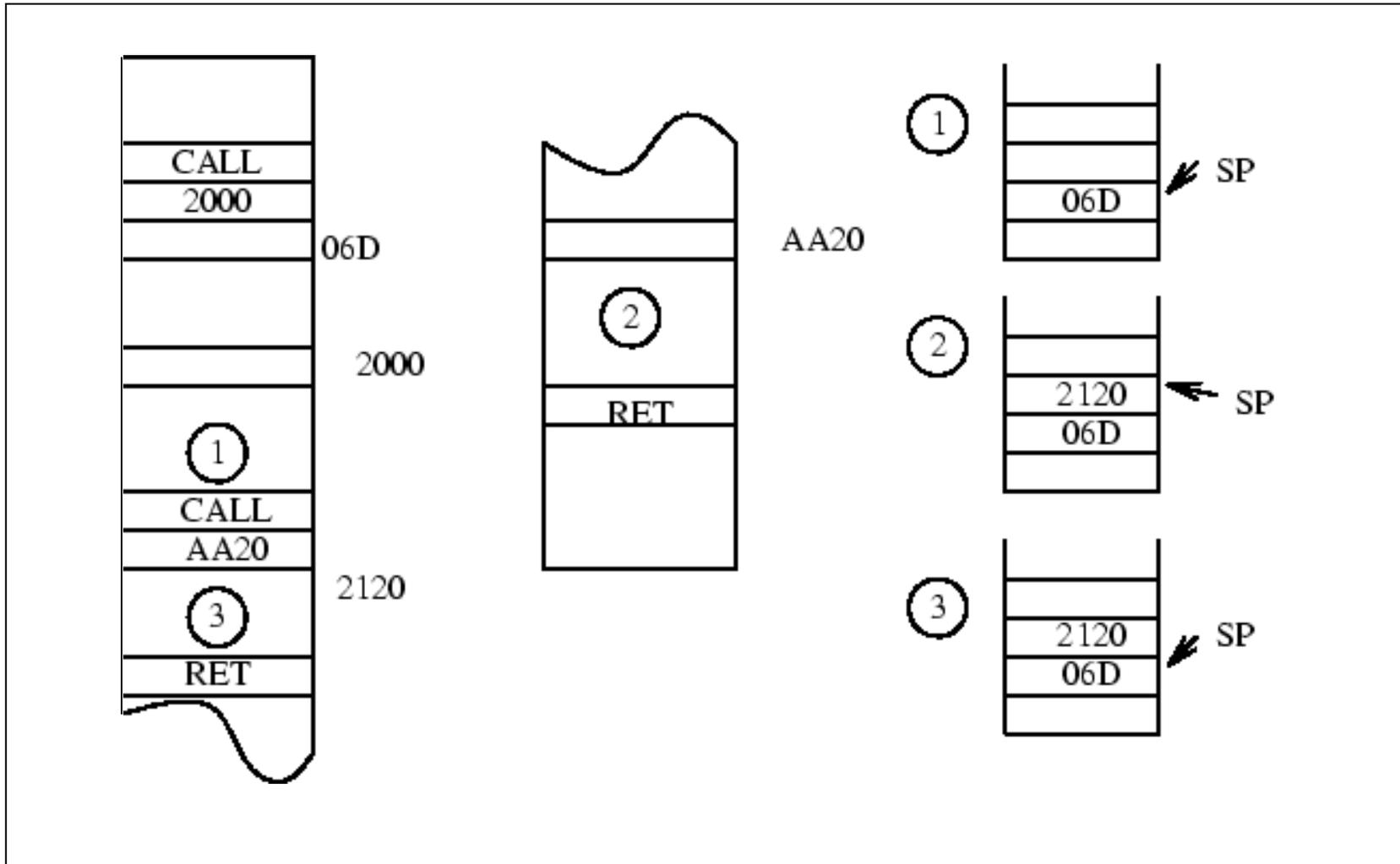


# Annidamenti

---

- Questa tecnica permette l'annidamento (o *nesting*) dei sottoprogrammi
  - Annidamento: chiamata a sottoprogramma all'interno di un sottoprogramma
- Lo Stack si trova in memoria principale
- Il dimensionamento dello Stack riveste particolare importanza

# Annidamenti



# Interrupt

---

- Evento infrequente ed eccezionale
- Generato internamente o esternamente
- Causa il trasferimento del controllo dal programma corrente a un programma specifico di servizio dell'evento

# Interrupt

---

- L'interruzione è utilizzata per la gestione degli apparati di I/O. Per ottenere risposte rapide nella gestione di dischi, tastiere, modem,...
- Questi elementi devono richiamare l'attenzione della CPU molto rapidamente e il trasferimento dei dati da e verso gli questi organi deve avvenire nel modo più efficiente possibile
- Un ulteriore aspetto è la necessità di assegnare urgenze o priorità diverse alle varie richieste di intervento in caso di contemporaneità

# Gestione delle Interruzioni

---

- La presenza di un interrupt è segnalata alla CPU da una linea proveniente dall'esterno (o da un segnale interno)
- Il segnale viene memorizzato ed è testato dalla CPU alla fine di ogni ciclo di istruzione
- La CPU risponde trasferendo il controllo a un altro programma (Interrupt Service Routine)
- L'evento che causa l'interruzione è asincrono rispetto all'esecuzione del programma
- Trasparenza delle interruzioni

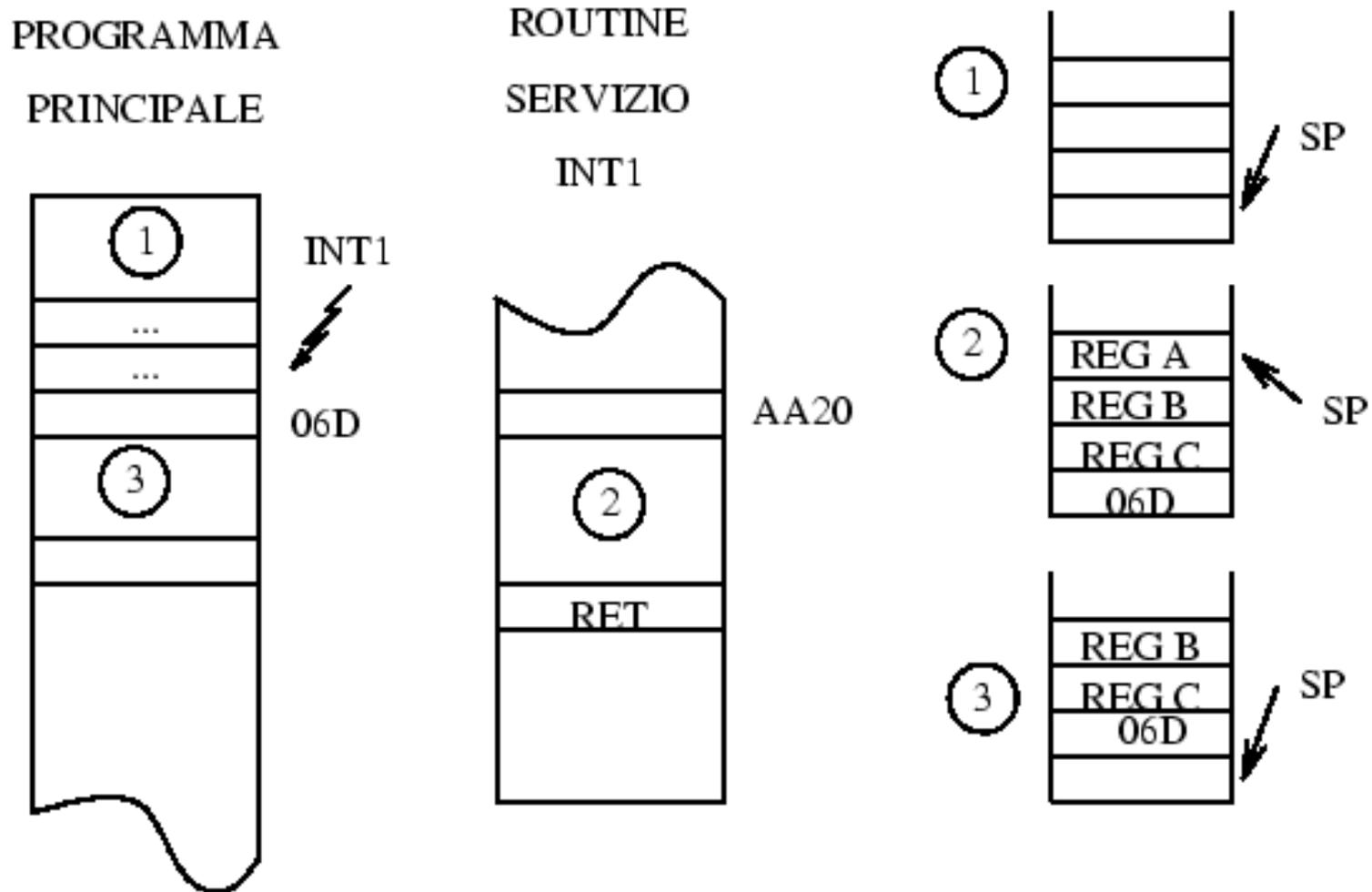
# Gestione delle Interruzioni

---

Azioni della CPU in risposta alla richiesta di interruzione (continua):

- Lo stato complessivo della CPU viene memorizzato a cura del programma di gestione
- Nel PC viene immagazzinato l'indirizzo del sottoprogramma di gestione
- L'esecuzione del sottoprogramma continua fino all'istruzione di RETURN che riporta il controllo al programma interrotto

# Gestione delle Interruzioni



# Set di Istruzioni

---

- Caratteristiche:
  - Completezza
  - Efficienza
  - Regolarità
  - Compatibilità

# Set di Istruzioni

---

## COMPLETEZZA:

- Deve essere possibile valutare qualunque funzione che sia calcolabile con una disponibilità ragionevole di memoria

## EFFICIENZA:

- Istruzioni usate frequentemente devono essere eseguite rapidamente
- Il controllo sull'efficienza deve essere misurato sul codice generato da compilatori
- Fornire primitive, non soluzioni

# Set di Istruzioni

---

## REGOLARITA':

- Le istruzioni devono comportarsi in modo omogeneo rispetto ai modi di indirizzamento
- Quando ogni istruzione consente tutti i modi di indirizzamento possibili su tutti gli operandi si ha ortogonalità

# Set di Istruzioni

---

## COMPATIBILITA'

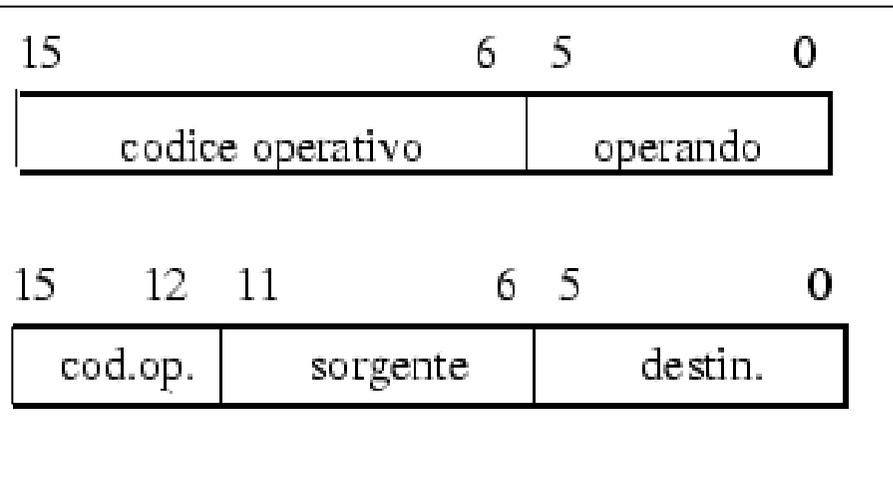
- Compatibilità sorgente (sono compatibili i codici mnemonici del linguaggio assembly) e compatibilità binaria (sono compatibili i codici macchina)
- Il codice macchina deve essere eseguibile su processori precedenti della stessa famiglia
  - 8080 >> 8086 >> 80186 >> ... >> Pentium >>  
...
  - PDP11 >> VAX

# Set di Istruzioni

---

## ESEMPIO PDP11

Istruzioni a un operando:



Istruzioni a due operandi:

## MODI DI INDIRIZZAMENTO

Operando (sorgente o destinazione):

3 bit modo:

1o bit per ind. indiretto

2o-3o bit (registro, autoinc., autodec., con indice)

3 bit per selezionare 1 su 8 registri

# Tipi di Istruzioni

---

## **TRASFERIMENTO DATI**

MOVE	Trasferisce dati da sorgente a destinazione
LOAD	Trasferisce dati dalla memoria
STORE	Trasferisce dati nella memoria
EXCHANGE	Scambia dati
SET/RESET	Valori su I/O
PUSH/POP	Gestione stack

## **OPERAZIONI ARITMETICHE**

ADD/SUB	Somma / differenza
MULT/DIV	Moltiplicazione / divisione
ABS	Valore assoluto
NEG	Cambio segno
INC/DEC	Incrementa / decrementa

# Tipi di Istruzioni

---

## OPERAZIONI ARITMETICHE

	Scalari		Vettoriale	
	Virg.fissa	Virg.mobile	Virg.fissa	Virg.mobile
	+, -	X, :	+, -, X, :	+, -, X:
Intel 8085	x			
Motor. 68020	x	x		
IBM 360	x	x	x	x
Cray-1	x	x	x	x
Pentium	x	x	x	(x)

Le attuali CPU VLSI svolgono attualmente le stesse operazioni che svolgevano i supercalcolatori di una decina di anni fa

# Tipi di Istruzioni

---

## OPERAZIONI LOGICHE

AND/OR/NOT/EX-OR

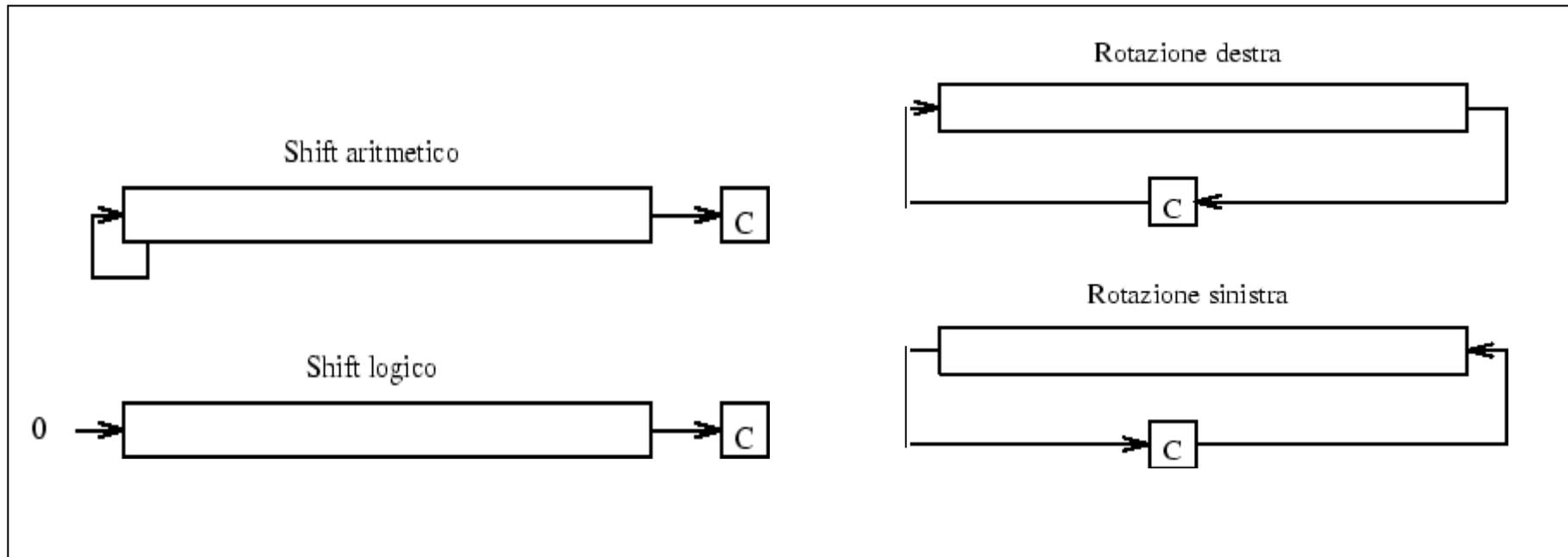
CONVERT

Calcola le funzioni logiche

Cambia formato ai dati

## SHIFT

## ROTATE



# Tipi di Istruzioni

---

## **CONTROLLO PROGRAMMA**

JUMP	Salto incondizionato
JUMP COND	Salto condizionato
CALL (COND)	Salto a sottoprogramma
RET (COND)	Uscita da sottoprogramma

## **CONTROLLO CPU**

HALT	Blocco operazioni
WAIT / HOLD	Blocco operazioni con ripresa condizionata o su interrupt
NOP	Non svolge operazioni

## **INGRESSO E USCITA**

INPUT (READ)	Trasferimento dati da I/O a memoria o registro
OUTPUT (WRITE)	Trasferimento dati verso porta di I/O.