

Modi di indirizzamento del processore MC68000 (parte seconda)

Prof. Roberto Canonico



Università degli Studi di Napoli Federico II
Facoltà di Ingegneria

Riepilogo modi fondamentali

- **Register direct addressing** - È usato per variabili che possono essere mantenute in registri di memoria
 - **Literal (immediate) addressing** - È usato per costanti che non cambiano
 - **Direct (absolute) addressing** - È usato per variabili che risiedono in memoria
-

LEA: Load Effective Address

Operazione: $[An] \leftarrow \langle ea \rangle$
Sintassi: LEA $\langle ea \rangle, An$
Esempio: LEA table, A3
Attributi: Size = longword

Descrizione:

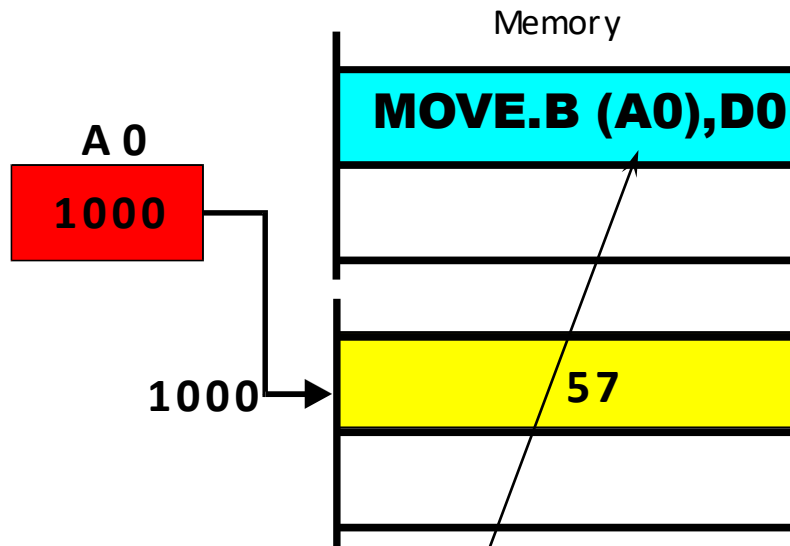
Calcola l'indirizzo effettivo ($\langle ea \rangle$) del primo operando, generalmente espresso in forma simbolica, e lo pone nel registro indirizzo specificato dal suo secondo operando. Non influenza i flag di stato:

X N Z V C
- - - - -

Address Register Indirect Addressing

- L'istruzione specifica uno dei registri indirizzo
 - Il registro indirizzo specificato contiene l'indirizzo effettivo dell'operando
 - Il processore accede all'operando puntato dal registro indirizzo
 - Esempio:
 - `MOVE.B (A0),D0`
-

Address Register Indirect: funzionamento

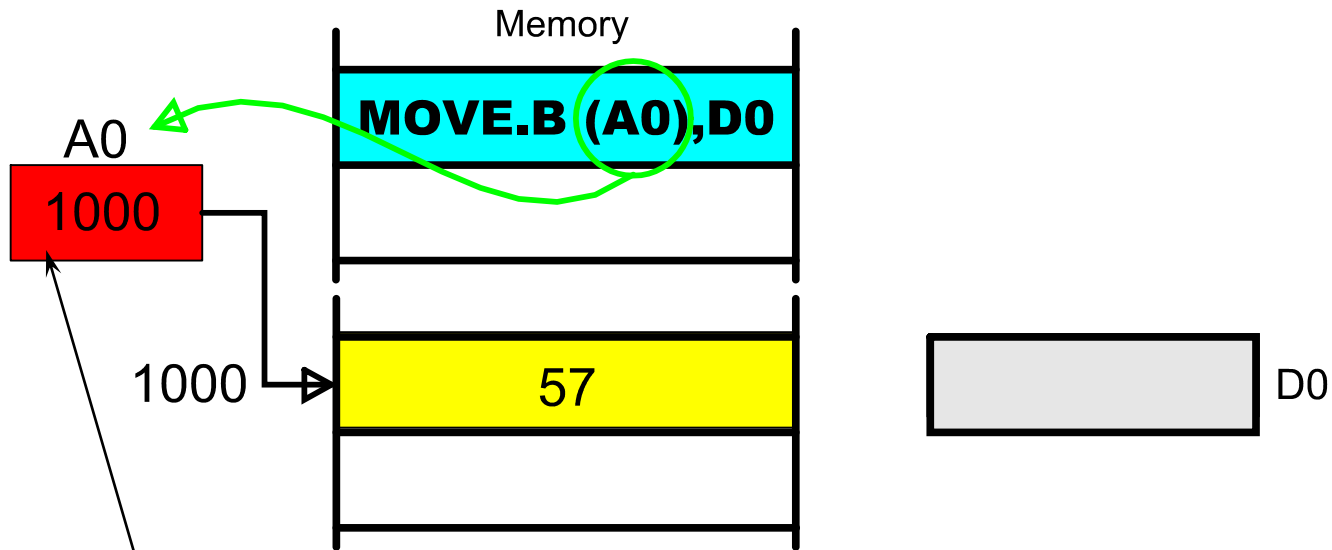


Questa istruzione significa: carica D0 con il contenuto della locazione puntata dal registro indirizzo A0



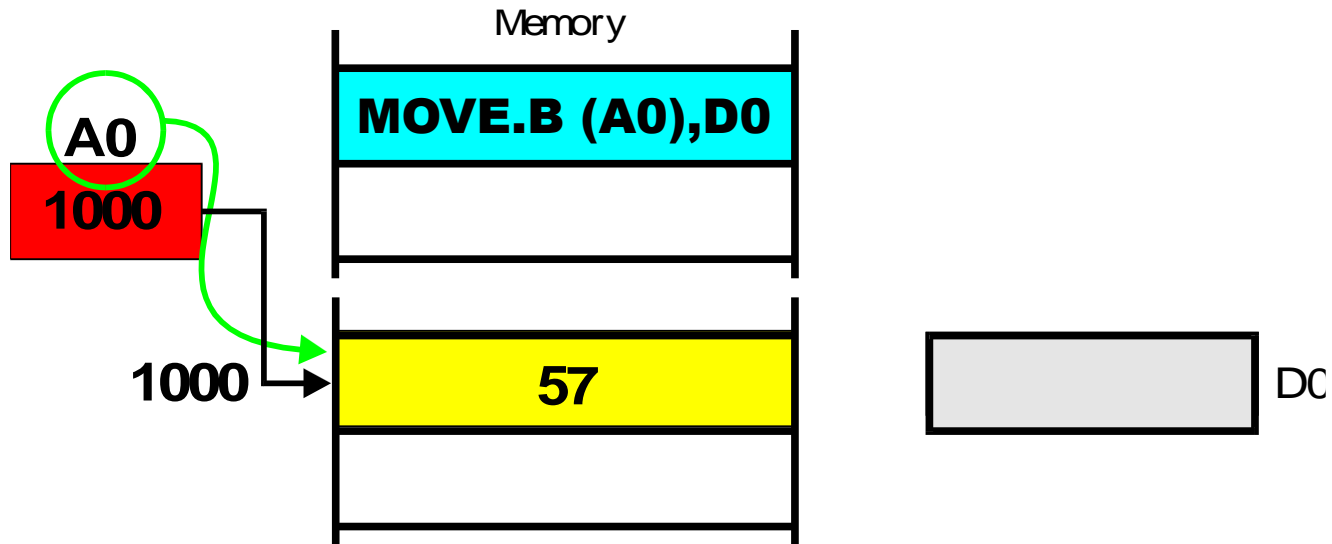
L'istruzione specifica l'operando sorgente come (A0)

Address Register Indirect: funzionamento



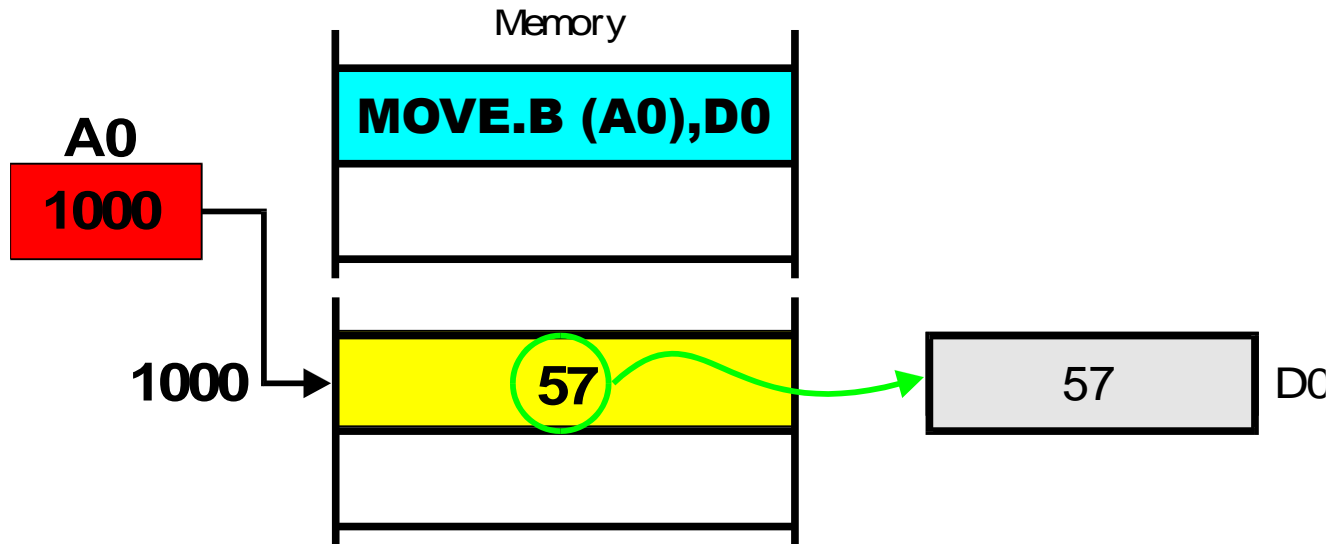
Il registro indirizzo nell'istruzione
specifica un registro indirizzo che
contiene l'indirizzo dell'operando

Address Register Indirect: funzionamento



Il registro indirizzo è usato per accedere all'operando in memoria

Address Register Indirect: funzionamento



Alla fine, il contenuto della
locazione puntata da A0 viene
copiato nel registro dati

Auto-post-increment

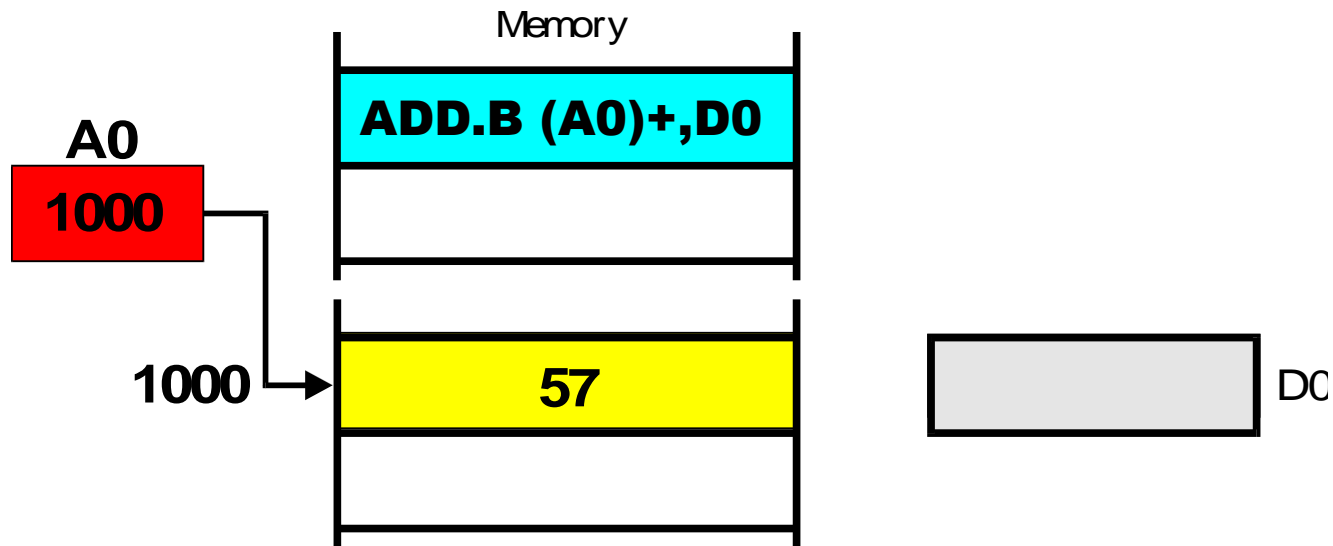
- L'istruzione specifica uno dei registri indirizzo, usando la modalità Address Register Indirect.
 - Se il modo di indirizzamento è specificato come (An)+, il contenuto del registro indirizzo è incrementato di una quantità pari alla dimensione dell'operando *dopo l'uso* ("post-incremento")
 - Esempio:
 - `MOVE.W (A0)+, D0` Usa A0 per la MOVE e poi gli aggiunge 2 (2 poiché l'accesso è di tipo .W = 2 byte). Di fatto, l'istruzione esegue un pop in D0 dallo stack puntato da A0
-

Auto-pre-decrement

- L'istruzione specifica uno dei registri indirizzo
 - Se il modo di indirizzamento è specificato come $-(An)$, il contenuto del registro indirizzo è decrementato di una quantità pari alla dimensione dell'operando *prima dell'uso* ("pre-decremento")
 - Esempio:
 - `MOVE.W D0,-(A0)`

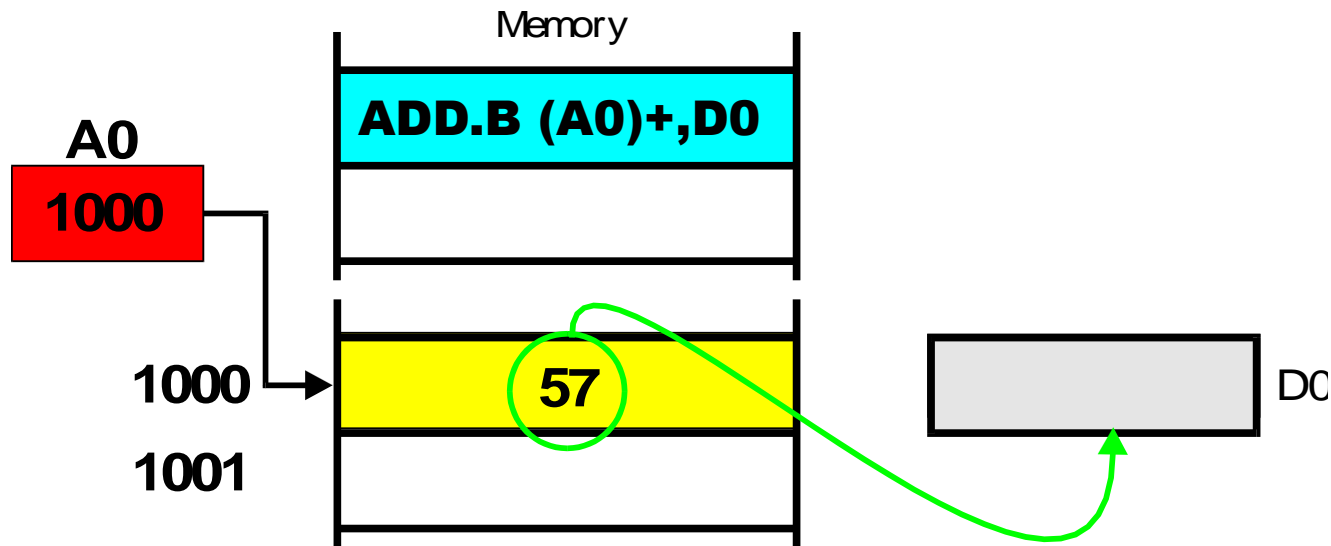
Sottrae 2 ad A0 e poi lo usa per la MOVE (2 poiché l'accesso è di tipo `.W` = 2 byte). Di fatto, l'istruzione esegue un push di D0 sullo stack puntato da A0.
-

Auto-post-increment: funzionamento



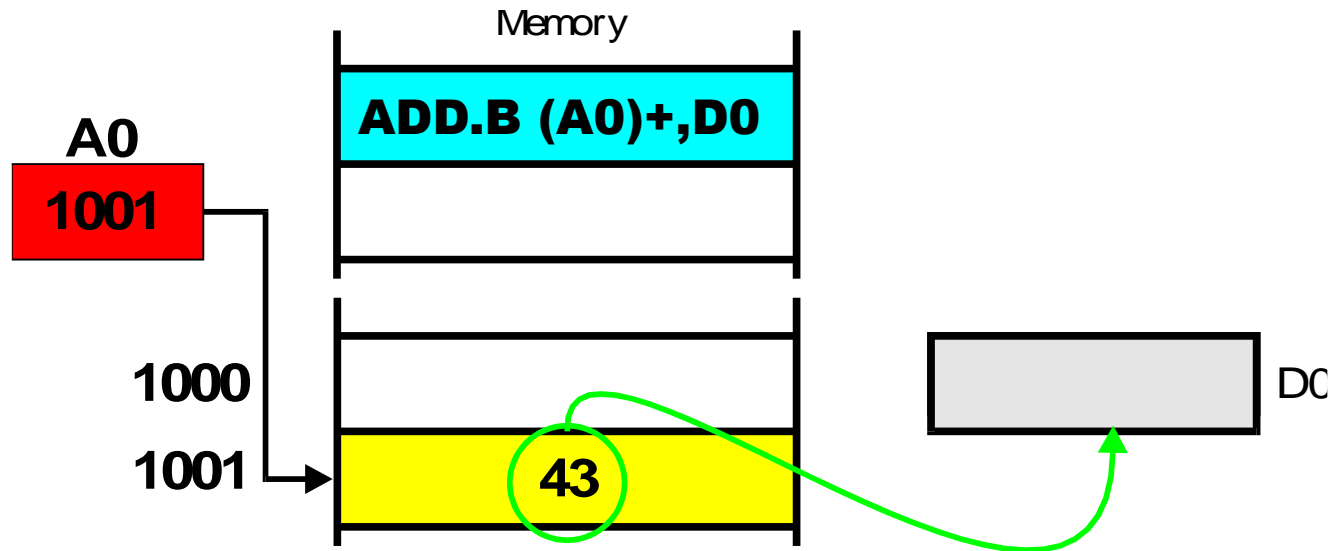
Il registro indirizzo contiene 1000
ovvero “punta” alla locazione 1000

Auto-post-increment: funzionamento



Il registro A0 viene usato per accedere alla locazione di memoria 1000 e il contenuto di questa locazione (57) viene sommato a D0

Auto-post-increment: funzionamento



Dopo che l'istruzione è stata eseguita,
il contenuto di A0 viene incrementato, per
puntare alla locazione successiva
