10. Modi di Indirizzamento e Formato delle Istruzioni

Modi di Indirizzamento

I modi di indirizzamento definiscono come viene specificato l'indirizzo degli operandi nelle istruzioni.

Indirizzamento Immediato

L'operando è contenuto direttamente nell'istruzione, nel campo indirizzo. Non richiede accessi in memoria per recuperare l'operando, ma il valore è limitato dalla dimensione del campo indirizzo.

Indirizzamento Diretto

Il campo indirizzo contiene l'indirizzo effettivo dell'operando in memoria. Richiede un singolo accesso in memoria. Lo spazio di indirizzamento è limitato dalla dimensione del campo.

Indirizzamento Indiretto

Il campo indirizzo contiene l'indirizzo di una cella di memoria che a sua volta contiene l'indirizzo dell'operando. Permette di indirizzare 2^N entità con parole di lunghezza N, ma richiede due accessi in memoria.

Indirizzamento Registro

L'operando si trova in un registro specificato nel campo indirizzo. Poiché i registri sono pochi, servono pochi bit per identificarli, risultando in istruzioni più corte e fetch più veloci (nessun accesso in memoria).

Indirizzamento Registro Indiretto

L'operando è in una cella di memoria il cui indirizzo è contenuto in un registro. Offre grande spazio di indirizzamento (2^n) con un accesso in memoria in meno rispetto all'indirizzamento indiretto puro.

Indirizzamento con Spiazzamento

Combina indirizzamento diretto e registro indiretto. Il campo indirizzo ha due sottocampi: un valore di base A e un registro R. L'indirizzo effettivo è la somma del contenuto di R e del valore A. Esistono tre varianti principali:

- **Relativo**: R è il Program Counter; l'indirizzo è A + (PC)
- **Registro-base**: R contiene l'indirizzo base, A lo spiazzamento
- **Indicizzazione**: A è la base, R contiene lo spiazzamento (usato per scorrere array)

Indirizzamento a Pila

La pila è una sequenza lineare di locazioni riservate in memoria. Il registro SP (stack pointer) punta alla cima. L'operando è sempre sulla cima della pila, rendendo questo un caso di indirizzamento registro indiretto.

Modi di Indirizzamento x86

L'architettura x86 utilizza registri di segmento che indicizzano una tabella di descrittori contenente indirizzi base, limiti e diritti di accesso. L'indirizzo lineare viene calcolato combinando indirizzo base del segmento, registro base, registro indice (moltiplicato per un fattore di scala 1, 2, 4 o 8) e displacement.

I modi disponibili sono: Immediato, Registro, Displacement, Base, Base con Displacement, Scaled Index con Displacement, Base con Index e Displacement, Base con Scaled Index e Displacement, Relativo.

Formato delle Istruzioni

Il formato definisce la struttura dei campi dell'istruzione, includendo codice operativo e operandi (espliciti o impliciti). La maggior parte dei linguaggi macchina prevede più formati.

Fattori che Influenzano la Lunghezza

- Dimensione e organizzazione della memoria
- Struttura del bus

- Complessità e velocità della CPU
- Compromesso tra potenza del repertorio e risparmio di spazio

Allocazione dei Bit

Dipende da: modi di indirizzamento supportati, numero di operandi (tipicamente 1 o 2), numero di registri (solitamente almeno 32), organizzazione in banchi, intervallo di indirizzi, granularità (byte o parola).

Esempi di Architetture

PDP-8: Istruzioni a 12 bit. Solo 6 operazioni base con istruzioni di riferimento a memoria. Codice operativo 7 definisce microistruzioni con singoli bit che controllano operazioni specifiche.

PDP-10: Formato fisso a 36 bit con campi per opcode (9 bit), registro (4 bit), bit indiretto, registro indice (4 bit) e indirizzo di memoria (18 bit).

PDP-11: Formato variabile con istruzioni da 1 a 3 parole. I campi sorgente e destinazione contengono ciascuno 3 bit per modo di indirizzamento e 3 bit per numero di registro.

VAX: Formato altamente variabile, da 1 a 37 byte. Opcode di 1 o 2 byte seguito da zero a sei specificatori di operando. Ogni specificatore ha 4 bit per il modo di indirizzamento, con eccezione del modo literal che usa 2 bit per il modo e 6 per il valore.

x86: Formato variabile composto da: prefissi opzionali (0-4 byte), opcode (1-3 byte), byte ModR/M opzionale (specifica modo e registri), byte SIB opzionale (scala, indice, base), displacement opzionale (0-4 byte), immediato opzionale (0-4 byte).