
Calcolatori Elettronici

Architettura i8086

Ing. Gestionale e delle Telecomunicazioni
A.A. 2008/09
Gabriele Cecchetti

Architettura i8086

■ Sommario:

- Registri
- Accesso alla memoria: indirizzi effettivo, indirizzo fisico
- Repertorio mnemonico delle istruzioni
- Modalità di indirizzamento degli operandi

■ Riferimenti

- G. Bucci “Architetture e organizzazione dei Calcolatori Elettronici – Fondamenti”, Cap. 9

Cenni storici al processore i8086

- La CPU Intel 8086 è la prima appartenente all'architettura x86. Essa risale al 1976.
- E' un processore a 16 bit, in tecnologia da 30um (oggi il processo produttivo si spinge fino a 45nm).
- La frequenza di clock dell'epoca era di 5Mhz (contro i Ghz dei processori odierni).
- Ogni istruzione macchina impiegava 15 cicli di clock per essere eseguita $\rightarrow 15 * 1/50000000 = 200 \text{ us} \rightarrow 0,33 \text{ milioni di istruzioni per secondo}$.

Registri generali, di segmento, IP e di stato

REGISTRI

I registri dell'i8086

8 registri generali:

- ❑ AX, BX, CX, DX detti registri dati,
- ❑ SP, BP, SI, DI detti registri indice.

4 registri di segmento (o registri selettori):

- ❑ CS, SS, DS, ES

1 registro IP usato come puntatore di istruzione in congiunzione con CS;

1 registro di stato (registro del Flag).

I registri dati

I registri AX, BX, CX e DX sono a 16 bit: ognuno di essi è diviso in due registri da 8 bit corrispondenti ai bit più significativi e meno significativi rispettivamente:

- ❑ AH, AL
- ❑ BH, BL
- ❑ CH, CL
- ❑ DH, DL

Questi registri possono essere utilizzati senza alcuna distinzione in operazioni aritmetiche e logiche.

Ruoli specifici per i registri dati

AX → registro accumulatore, usato anche per I/O, operazioni aritmetiche e su stringhe.

BX → registro base per il calcolo degli indirizzi.

CX → registro contatore.

DX → registro usato per I/O, moltiplicazione e divisione.

Registri indice e puntatori

SP → puntatore alla cima dello stack

BP → puntatore entro lo stack

SI → registro indice generico, usato per indicizzare la stringa sorgente.

DI → registro indice generico, usato per indicizzare la stringa di destinazione.

Registri di segmento e registro IP

CS → segmento codice corrente

SS → segmento stack

DS → segmento dati

ES → segmento dati

- Il registro IP contiene l'offset rispetto al registro CS dell'istruzione successiva da eseguire.

Registro di stato

Contiene 9 indicatori a 1 bit, detti anche *flag*:

■ Flag di stato

- ❑ CF – Carry flag → riporto
- ❑ AF – Auxiliary flag → riporto sul bit 3 (aritmetica BCD)
- ❑ OF – Overflow flag → trabocco
- ❑ SF – Sign flag → risultato negativo
- ❑ ZF – Zero flag → risultato zero
- ❑ PF – Parity flag → risultato con numero di bit a 1 pari

■ Flag di controllo

- ❑ DF – Direction flag → direzione manipolazione stringhe
- ❑ IF – Interrupt flag → abilita/disabilita le interruzioni
- ❑ TF – Trap flag → permette l'esecuzione *single step*

Memoria segmentata e memoria Fisica

ACCESSO ALLA MEMORIA

Memoria segmentata dell'i8086

- Ogni segmento è una unità logica di memoria che può avere una estensione massima di $2^{16} = 64\text{KB}$.
- Ogni segmento si compone di locazioni contigue di memoria e può essere posizionato in qualunque locazione di memoria.
- I segmenti possono essere adiacenti, disgiunti oppure parzialmente o totalmente sovrapposti.
- Gli indirizzi logici si esprimono nella forma:
SELETTORE : Offset

Calcolo dell'indirizzo fisico dell'i8086 (2/2)

- Il processore i8086 ha 20 piedini per indirizzare la memoria fisica: pertanto gli indirizzi fisici vengono ottenuti nel modo seguente:

$$\text{Indirizzo fisico} = \text{CS} * 16 + \text{IP}$$

- La CPU i8086 può indirizzare fino a 1MB (2^{20} Bytes) di memoria fisica.

Uso dei registri di segmento dell'i8086

Tipo di riferimento	Registro normale	Registro. alternativo	Offset
Fetch	CS		IP
Stack	SS		SP
Variab.	DS	CS, ES, SS	Ind. eff.
Stringhe sorg.	DS	CS, ES, SS	SI
Stringhe dest.	ES		DI
BP	SS	CS,DS,ES	Ind. eff.

- Esempio:

```
MOV AX, VAR          ; AX ← M[DS:Offset(Var)]  
MOV AX, ES:Var        ; AX ← M[ES:Offset(Var)]
```

Uso dello stack dell'i8086

- Lo stack ha ampiezza di parola (nell'8086 16 bit).
- Operazioni:
 - PUSH → immette i dati nello stack
PUSH AX ; M[SS:SP] ← AX
 - POP → preleva i dati dallo stack
POP AX ; AX ← M[SS:SP]
 - Accesso diretto allo stack
MOV AX, [BP] ; AX ← M[SS:BP]

Accesso alla memoria dell'i8086

- L'organizzazione è di tipo *little endian*.
- La CPU può leggere/scrivere 8 o 16 bit (sono permesse anche parole non allineate).
- Memoria segmentata che consente l'uso di aree separate per codice, stack e dati e facilita la rilocalizzazione dei programmi.

Partenza (boot/reset) dell'i8086

CS ← FFFFh

IP ← 0000h

- L'indirizzo effettivo della prima istruzione è FFFF0h.
- Questo indirizzo di solito contiene il salto alla prima effettiva istruzione del programma.
- Questa porzione di memoria di solito è contenuta in una ROM che avvia il firmware di inizializzazione del sistema (detto BIOS – Basic Input/Output System), che di solito si trova su memoria Flash (EEPROM).

Le istruzioni dell'i8086

REPERTORIO MNEMONICA DELLA ISTRUZIONI

Repertorio mnemonico delle istruzioni dell'i8086 (1/5)

■ Trasferimento dati

MOV	; Copia un byte/parola
PUSH/POP	; Immette/estrae una parola nello stack
IN/OUT	; Trasf. un byte/parola in ingresso/uscita
PUSHF/POPF	; Immette/estrae registri di stato nello stack
PUSHA/POPA	; Immette/estrae 8 registri generali a 16 bit
PUSHAD/POPAD	; Immette/estrae 8 registri generali a 32 bit
XCHG	; Scambia sorgente e destinatario

Repertorio mnemonico delle istruzioni dell'i8086 (2/5)

Istruzioni aritmetiche

ADD/SUB	; Somma/Sottrae un byte/parola
ADC/SBB	; Somma/Sottrae con riporto/prestito
INC/DEC	; Incrementa/Decrementa byte/parola
CMP	; Compara byte/parole
AAS	; Aggiusta ASCII per somma
MUL/IMUL	; Moltiplica interi senza segno/con segno
DIV/IDIV	; Divide interi senza segno/con segno
CBW/CWD/CDQ	; Converte byte a word / da word a double-word / da double-word a quad-word

Repertorio mnemonico delle istruzioni dell'i8086 (3/5)

Manipolazione dei bit

AND/OR/XOR ; Esegue l'AND/OR/XOR su byte/parole
NOT ; Esegue il NOT su byte/parola
SHL/SHR ; Scorrimento a sinistra/destra numeri nat.
SAL/SAR ; Scorrimento a sinistra/destra numeri int.
ROL/ROR ; Ruota a sinistra/destra numeri naturali
RCL/RCR ; Ruota a sinistra/destra con il riporto

Manipolazioni stringhe

MOVSB ; Sposta una stringa di byte
CMPS ; Confronta stringhe

Repertorio mnemonico delle istruzioni dell'i8086 (4/5)

Salti

JMP ; Salto incondizionato
JE/JNE/JZ/JNZ ; Salti condizionati (ZF)
JA/JAE/JB/JBE/JG/JGE/JL/JLE ; Salti cond. (ZF,CF)
JC/JNC ; Salti cond. (CF) - JO/JNO ; Salti cond. (OF)
JS/JNS ; Salti cond. (SF) - JPE/JPO ; Salti cond.(PF)
LOOP → DEC CX e JMP
LOOPE/LOOPNE/LOOPZ/LOOPNZ → DEC CX e JE/JNE/JZ/JNZ

Procedure

CALL ; Chiamata di sottoprogramma
RET ; Ritorno dal sottoprogramma

Interruzioni

INT ; Interruzione software
IRET ; Ritorno da interruzione

Repertorio mnemonico delle istruzioni dell'i8086 (5/5)

Controllo della CPU

STC/CLC ; Imposta/resetta il riporto
SLI/CLI ; Abilita/disabilita le interruzioni

Sincronizzazione con l'esterno

HLT ; Passa allo stato di halt
WAIT ; Passa allo stato di wait

Istruzione di non operazione

NOP ; Non fa niente

Modalità di indirizzamento operandi dell'i8086

Indirizzamento dei registri della CPU

MOV AH, BL ; $AH \leftarrow BL$

Indirizzamento degli operandi immediati

MOV AH, 2476 ; $AH \leftarrow 2476$

Indirizzamento diretto in memoria

MOV AX, Var ; $AX \leftarrow M[DS:Offset(Var)]$

Indirizzamento indiretto rispetto ai registri

MOV [BX], AL ; $M[BX] \leftarrow AL$

Indirizzamento relativo a un registro

MOV AX, Var[SI] ; $AX \leftarrow M[DS:Offset(Var)+SI]$

Indirizzamento indiretto attraverso un registro e indiciato

MOV [BX][DI], AX ; $M[BX+DI] \leftarrow AX$

Indirizzamento delle porte di I/O

IN AL, PORTA ; $AL \leftarrow \text{Porta in Ingresso}$

Indirizzamento delle porte di I/O con registro

OUT [DX], AX ; $\text{Porta}[DX] \leftarrow AX$