



Arsitektur dan Organisasi Komputer

7

Aditya Wikan Mahastama, S.Kom

Week 11



Set Instruksi

Apakah Set Instruksi itu?

- **Set instruksi (instruction set):** sekumpulan lengkap instruksi yang dapat dimengerti oleh sebuah CPU
- Disebut juga machine code (bahasa mesin), aslinya juga berbentuk biner → bahasa assembly
- Untuk konsumsi manusia (programmer), biasanya digunakan representasi yang lebih mudah dimengerti → bahasa yang dapat dimengerti manusia

Elemen-elemen Instruksi

- Operation code (Op code)
 - Kerjakan ini
- Source Operand reference
 - Terhadap isi alamat ini
- Result Operand reference
 - Letakkan hasilnya di alamat ini
- Next Instruction Reference
 - Kalau sudah selesai, selanjutnya kerjakan alamat ini

Dalam sebuah instruksi, tidak harus semua elemen ini dicantumkan, tergantung kebutuhan dan jenis instruksinya



Ke mana operand disimpan?

- Ingat: Semua instruksi dijalankan dalam CPU
- Rata-rata operasi hanya membutuhkan register sebagai tempat membaca / menyimpan operand
- Adakalanya juga register tidak berisi operand tapi menunjuk ke tempat penyimpanan lainnya (memory, cache, modul I/O) → nanti minggu depan dengan mode pengalamatan

Representasi Instruksi



- Pada bahasa mesin, setiap instruksi berbentuk pola bit biner yang unik
- Agar dapat dimengerti manusia, dibuatlah representasi simbolik instruksi, biasanya berupa singkatan (disebut *mnemonic*)
 - misal ADD, SUB, LOAD
- Sedangkan alamat operand direpresentasikan sebagai berikut:
 - ADD **A,B**
- Ada beberapa jenis representasi instruksi, tergantung dari jumlah alamat operand yang bisa diolah oleh satu instruksi

Format Instruksi (Biner)

- Misal Instruksi dengan 2 Alamat Operand:
- ADD A, B \rightarrow A & B suatu alamat register



4 bits

6 bits

6 bits

Opcode

Operand Reference

Operand Reference

16 bits

ADD

A

B

(dalam bentuk biner tentunya)

Contoh Simbolik Instruksi

- ADD: Add (Jumlahkan)
 - SUB: Subtract (Kurangkan)
 - MPY/MUL: Multiply (Kalikan)
 - DIV: Divide (Bagi)
 - LOAD: Load data dari register/memory
 - STOR: Simpan data ke register/memory
 - MOVE: pindahkan data dari satu tempat ke tempat lain
 - SHR: shift kanan data
 - SHL: shift kiri data
- dan lain-lain



Cakupan Jenis Instruksi

- Data processing: Aritmetik (ADD, SUB, dsb); Logic (AND, OR, NOT, SHR, dsb); konversi data
- Data storage (memory): Transfer data (STOR, LOAD, MOVE, dsb)
- Data movement: Input dan Output ke modul I/O
- Program flow control: JUMP, HALT, dsb.

Set Instruksi lengkap bisa dilihat di kitabnya William Stallings

Format Instruksi 3 Alamat

- Bentuk umum: [OPCODE] [AH], [AO1], [AO2]
- Satu alamat hasil, dua alamat operand
- Misal: **SUB Y, A, B**
 - Bentuk algoritmik: $Y \leftarrow A - B$
 - Arti: Kurangkan isi Reg A dengan isi Reg B, kemudian simpan hasilnya di Reg Y.
- Bentuk ini tidak umum digunakan di komputer
- Mengoperasikan banyak register sekaligus
- Program lebih pendek

Format Instruksi 2 Alamat

- Bentuk umum: [OPCODE] [AH], [AO]
- Satu alamat hasil merangkap operand, satu alamat operand
- Misal: **SUB Y, B**
 - Bentuk algoritmik: $Y \leftarrow Y - B$
 - Arti: Kurangkan isi Reg Y dengan isi Reg B, kemudian simpan hasilnya di Reg Y.
- Bentuk ini masih digunakan di komputer sekarang
- Mengoperasikan lebih sedikit register, tapi panjang program tidak bertambah terlalu banyak



Format Instruksi 1 Alamat

- Bentuk umum: [OPCODE] [AO]
- Satu alamat operand, hasil disimpan di accumulator
- Misal: **SUB B**
 - Bentuk algoritmik: $AC \leftarrow AC - B$
 - Arti: Kurangkan isi Acc. dengan isi Reg B, kemudian simpan hasilnya di Acc.
- Bentuk ini digunakan di komputer jaman dahulu
- Hanya mengoperasikan satu register, tapi program menjadi bertambah panjang



Format Instruksi 0 Alamat



- Bentuk umum: [OPCODE] [O]
- Semua alamat operand implisit, **disimpan dalam bentuk stack**. Operasi yang biasanya membutuhkan 2 operand, akan mengambil isi stack paling atas dan di bawahnya
- Misal: **SUB**
 - Bentuk algoritmik: $S[\text{top}] \leftarrow S[\text{top}-1] - S[\text{top}]$
 - Arti: Kurangkan isi Stack no.2 dari atas dengan isi Stack paling atas, kemudian simpan hasilnya di Stack paling atas
- Ada instruksi khusus Stack: **PUSH** dan **POP**

Contoh Format Instr 3 Alamat

- A, B, C, D, E, T, Y adalah register
- Program: $Y = (A - B) / (C + D \times E)$

SUB	Y, A, B	$Y \leftarrow A - B$
MPY	T, D, E	$T \leftarrow D \times E$
ADD	T, T, C	$T \leftarrow T + C$
DIV	Y, Y, T	$Y \leftarrow Y / T$

- Memerlukan 4 operasi





Contoh Format Instr 2 Alamat

- A, B, C, D, E, T, Y adalah register
- Program: $Y = (A - B) / (C + D \times E)$

MOVE	Y, A	$Y \leftarrow A$
SUB	Y, B	$Y \leftarrow Y - B$
MOVE	T, D	$T \leftarrow D$
MPY	T, E	$T \leftarrow T \times E$
ADD	T, C	$T \leftarrow T + C$
DIV	Y, T	$Y \leftarrow Y / T$

- Memerlukan 6 operasi

Contoh Format Instr 1 Alamat



- A, B, C, D, E, Y adalah register
- Program: $Y = (A - B) / (C + D \times E)$

LOAD	D	$AC \leftarrow D$
MPY	E	$AC \leftarrow AC \times E$
ADD	C	$AC \leftarrow AC + C$
STOR	Y	$Y \leftarrow AC$
LOAD	A	$AC \leftarrow A$
SUB	B	$AC \leftarrow AC - B$
DIV	Y	$AC \leftarrow AC / Y$
STOR	Y	$Y \leftarrow AC$

- Memerlukan 8 operasi

Contoh Format Instr 0 Alamat



- A, B, C, D, E, Y adalah register
- Program: $Y = (A - B) / (C + D \times E)$

PUSH A	$S[\text{top}] \leftarrow A$
PUSH B	$S[\text{top}] \leftarrow B$
SUB	$S[\text{top}] \leftarrow A - B$
PUSH C	$S[\text{top}] \leftarrow C$
PUSH D	$S[\text{top}] \leftarrow D$
PUSH E	$S[\text{top}] \leftarrow E$
MPY	$S[\text{top}] \leftarrow D \times E$
ADD	$S[\text{top}] \leftarrow C + S[\text{top}]$
DIV	$S[\text{top}] \leftarrow (A - B) / S[\text{top}]$
POP Y	$\text{Out} \leftarrow S[\text{top}]$

- Memerlukan 10 operasi

Yang Perlu Diperhatikan

- Semakin banyak register yang diolah dalam satu instruksi → semakin lambat
- Semakin banyak baris operasi untuk mengeksekusi sebuah program → juga semakin lambat
- Oleh karena itu, CPU masakini bisa dikatakan menggunakan perpaduan dari beberapa format instruksi, tergantung konteksnya



Latihan

- Kerjakan $X = (A + B \times C) / (D - E \times F)$
- Format Instruksi 3 Alamat \rightarrow Skor 2
- Format Instruksi 2 Alamat \rightarrow Skor 3
- Format Instruksi 1 Alamat \rightarrow Skor 3
- Format Instruksi 0 Alamat \rightarrow Skor 4

