Mode Pengalamatan (Bagian 2)

Sri Supatmi,S.Kom

5. Pengalamatan Register

- Format instruksi perngalamatan register



- Secara konseptual, pengalamatan register mirip dengan pengalamatan langsung kecuali lokasi memori digantikan dengan register untuk menyimpan operand.
- Instruksi berisi nomor register yang mempunyai operand.
- Mode pengalamatan ini sangat berguna untuk suatu program yang panjang dalam penyimpanan hasil-hasil sementara di dalam register daripada di dalam memori.

>> Lanjutan:

- Contoh berikut adalah ilustrasi mode pengalamatan register:
  - ✓ **ADD R1,R2** → Jumlah isi register R1 dan R2 dan hasilnya disimpan di R1. Kedua operand menggunakan pengalamatan register.
  - ✓ **STORE R1,MEM1** → isi dari register R1 disimpan/disalin ke alamat memori MEM1; operand pertama menggunakan pengalamatan register dan operand kedua menggunakan pengalamatan langsung.

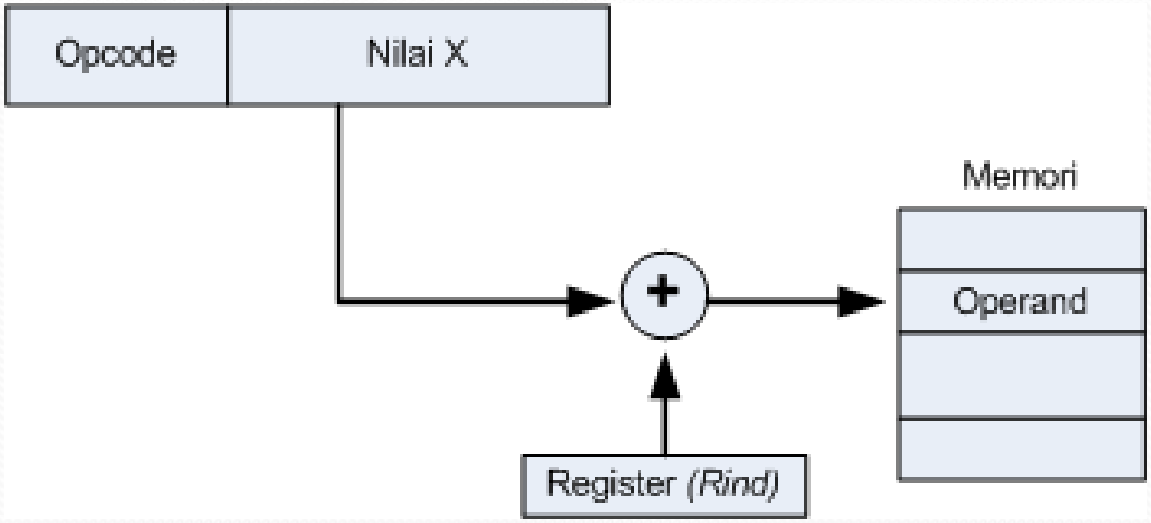
>> Keuntungan dan Kelemahan Pengalamatan Register

- ✓ **Keuntungan:**
  - ❑ pengambilan operand lebih cepat tanpa akses memori

- ✓ **Kelemahan:**
  - ❑ Jumlah register terbatas dan karena itu utilisasi efektif oleh programmer merupakan hal yang esensial.

6. Pengalamatan Indeks

- Format instruksi pengalamatan Indeks



- Pada mode pangalamatan indeks, alamat operand diperoleh dengan menambahkan sebuah konstanta ke suatu register, yang disebut register indeks.
- Instruksi ini mengisi register Ri dengan isi lokasi memori yang alamatnya adalah hasil jumlah isi register *Rind* dan nilai X
- *Contoh:*

  LOAD X(Rind),Ri → instruksi ini menyalin operand alamat hasil penjumlahan nilai X dengan nilai register *Rind* ke dalam register Ri

>> lanjutan 1:

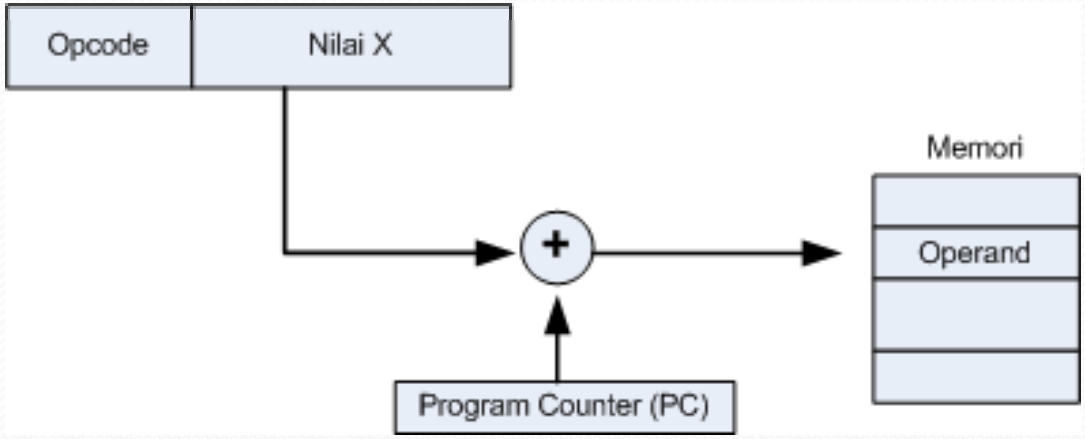
- Mode ini berbeda sedikti dengan mode pengalamatan *base register.*
- Register indeks berisi sebuah offset atau perpindahan (*displacement).*
- Instruksi berisi alamat yang akan ditambahkan pada offset dalam register indeks, untuk mendapatkan alamat operand efektif.
- Umumnya medan alamat dalam instruksi memberikan alamat awal array dalam memori.
- Register indeks berisi '*nilai indeks*' untuk operand yaitu selisih antara alamat awal dan alamat operand.
- Dengan mengubah register indeks, maka operand dalam array dapat diakses.
- Umumnya operand-operand(elemen-elemen array) berada dalam lokasi yang berurutan. Mereka diakses dengan *increment* yang sederhana pada register indeks.

>> lanjutan2:

- Beberapa CPU mendukung fitur '*autoindexing*', yang melibatkan *auto-increment* (dengan hardware) pada register indeks kapanpun sutu instruksi dengan pengalamatan indeks dieksekusi.
- Hal ini mengurangi penggunaan instruksi terpisah dalam menambah(increment) isi register indeks.
- Hal ini juga lebih mempercepat aksi serta lebih mengurangi ukuran program. Namun memberikan tanggungjawab tambahan '*autoindexing*' pada unit kontrol.

7. Pengalamatan Relatif

- Format instruksi pengalamatan relatif



- Pengalamatan relatif sama seperti pengalamatan indeks kecuali register indeks diganti dengan program counter (PC).
- Instruksi ini megisi Ri dengan kandungan lokasi memori yang alamatnya adalah hasil jumlah program counter (PC) dengan nilai X

>> Lanjutan3:

- Pada mode ini,instruksi menetapkan alamat operand (lokasi memori) sebagai posisi relatif dari alamat instruksi sekarang yaitu isi PC. Karena itu operand terletakpada ‘jarak pendek’ dari isi PC.
- Jumlah mode ini digunakan untuk menetapkan alamat pencabangan dalam instruksi branch, alamat pencabangan berada dekat dengan alamat instruksi.
- Contoh:
  - JUMP+8(PC)
  - JUMP-8(PC)

>> Keuntungan Pengalamatan Relatif

- ✓ **Keuntungan:**
  - ❑ Jumlah bit dalam medan alamat lebih sedikit

8. Pengalamatan Base Register

- Mode ini digunakan untuk relokasi program di dalam memori (dari satu area ke area lain).
- Pada mode pengalamatan *base register,* instruksi tidak berisi alamat. Dia memberikan perpindahan relatif terhadap area memori sekarang ke area memori yang lain, *base register* diisi dengan alamat *base* baru.
- Instruksi tidak perlu dimodifikasi/diubah. Dengan cara ini, keseluruhan program atau suatu segment dri program dapat dipindahkan dari satu area di memori ke yang lain tanpa mempengaruhi instruksi, dengan perubahan sederhana ini *base register.*
- Hal ini penting untuk sistem multiprogramming karena waktu yang berbeda (*run)*, area berbeda dari memori tersedia untuk sebuah program.
- Sebuah CPU dapat mempunyai lebih dari satu *base register.*

>>Format Instruksi & keuntungan Pengalamatan Base Register

- Format instruksi pengalamatan *base register*



Alamat operand=<base register>+offset

✓ ***Keuntungan:***

- medan alamat operand dalam instruksi sangat pendek karena dia hanya memberikan offset(perpindahan);alamat operand dikalkulasikan tanpa akses memori.

9. Pengalamatan Stack

- Pada pengalamatan stack(tumpukan), semua operand untuk suatu instruksi diambl dari bagian teratas stack.
- Instruksi tidak mempunyai medan operand.
- Misalnya, sebuah instruksi ADD hanya memberikan opcode (ADD). Kedua operand dalam stack, di dalam lokasi yang berurutan. Bila instruksi ADD dieksekusi, dua operand di-pop-off dari stack satu persatu. Setelah penjumlahan, hasilnya di-push ke dalam stack.
- **Keuntungan:**
  - Tidak ada menda operand dalam instruksi, karena itu instruksinya pendek.

Selesai