

Mode Pengalamatan

Pertemuan 6

Oleh :

Riyanto Sigit, S.T, M.Kom

Nur Rosyid Mubtada'i S.Kom

Setiawardhana , S.T

Hero Yudo Martono, S.T

Politeknik Elektronika Negeri Surabaya - ITS

2005

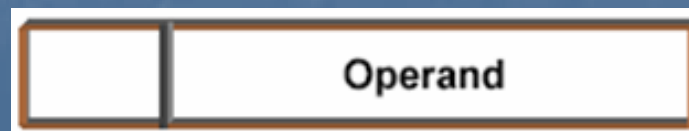
3.3. Mode Pengalamatan

- Mengatasi keterbatasan format instruksi
 - Dapat mereferensi lokasi memori yang besar
 - Mode pengalamatan yang mampu menangani keterbatasan tersebut
 - Masing – masing prosesor menggunakan mode pengalamatan yang berbeda – beda.
 - Memiliki pertimbangan dalam penggunaannya.
 - Ada beberapa teknik pengalamatan
 - Immediate Addressing
 - Direct Addressing
 - Indirect Addressing
 - Register Addressing
 - Register Indirect Addressing
 - Displacement Addressing
 - Stack Addressing

Immediate Addressing (1)

- Bentuk pengalamatan ini yang paling sederhana ?
 - Operand benar – benar ada dalam instruksi atau bagian dari instruksi = Operand sama dengan field alamat.
 - Umumnya bilangan akan disimpan dalam bentuk komplement dua.
 - Bit paling kiri sebagai bit tanda.
 - Ketika operand dimuatkan ke dalam register data, bit tanda akan digeser ke kiri hingga maksimum word data
 - Contoh :

ADD 5 ; tambahkan 5 pada akumulator



Immediate Addressing (+)&(-)

■ Keuntungan

- Mode ini adalah tidak adanya referensi memori selain dari instruksi yang diperlukan untuk memperoleh operand.
- Menghemat siklus instruksi sehingga proses keseluruhan akan cepat.

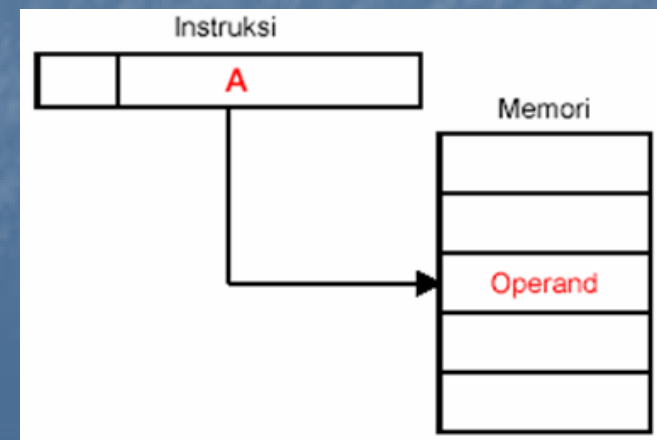
■ Kerugiannya

- Ukuran bilangan dibatasi oleh ukuran field alamat

Direct Addressing (2)

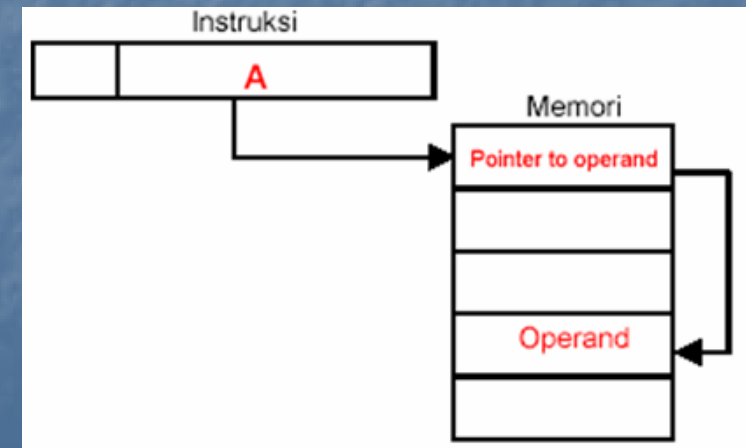
- Pengalamatan langsung
 - Kelebihan :
 - Field alamat berisi efektif address sebuah operand.
 - Teknik ini banyak digunakan pada komputer lama dan komputer kecil.
 - Hanya memerlukan sebuah referensi memori dan tidak memerlukan kalkulasi khusus.
 - Kelemahan :
 - Keterbatasan field alamat karena panjang field alamat biasanya lebih kecil dibandingkan panjang word
 - Contoh :

ADD A ; tambahkan isi pada lokasi alamat A ke akumulator



Indirect Addressing (3)

- Mode pengalamatan tak langsung
 - Field alamat mengacu pada alamat word di dalam memori, yang pada gilirannya akan berisi alamat operand yang panjang
 - Contoh :
ADD (A) ; tambahkan isi memori yang ditunjuk oleh isi alamat A ke akumulator



Indirect Addressing (+)&(-)

- Keuntungan

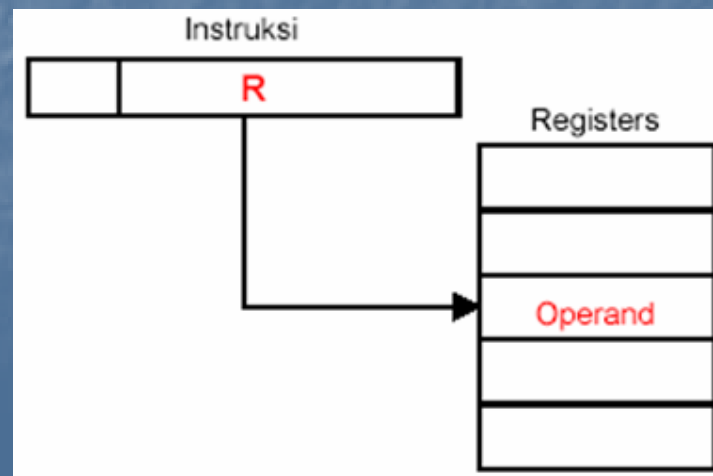
- Ruang bagi alamat menjadi besar sehingga semakin banyak alamat yang dapat referensi.

- Kerugian

- Diperlukan referensi memori ganda dalam satu fetch sehingga memperlambat proses operasi

Register Addressing (4)

- Metode pengalamatan register mirip dengan mode pengalamatan langsung.
- Perbedaannya terletak pada field alamat yang mengacu pada register, bukan pada memori utama.
- Field yang mereferensi register memiliki panjang 3 atau 4 bit, sehingga dapat mereferensi 8 atau 16 register general purpose.

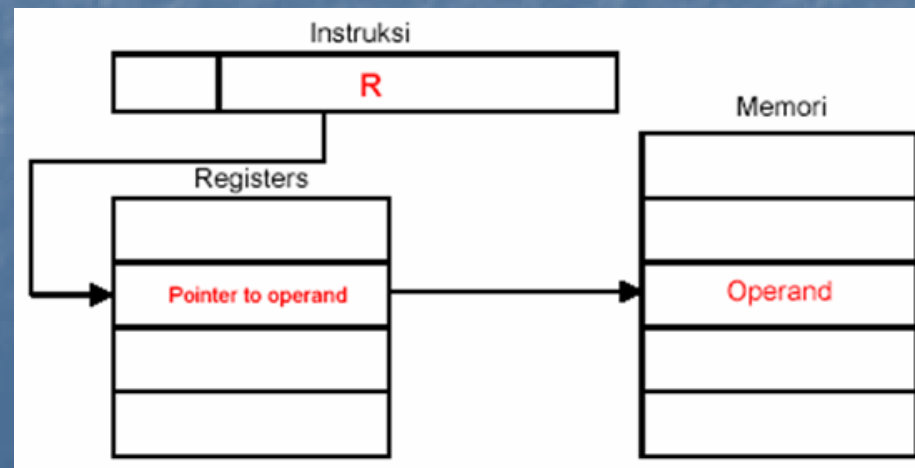


Register Addressing (+)&(-)

- Keuntungan pengalamatan register
 - Diperlukan field alamat berukuran kecil dalam instruksi dan tidak diperlukan referensi memori.
 - Akses ke register lebih cepat daripada akses ke memori, sehingga proses eksekusi akan lebih cepat.
- Kerugian
 - Ruang alamat menjadi terbatas

Register Indirect Addressing (5)

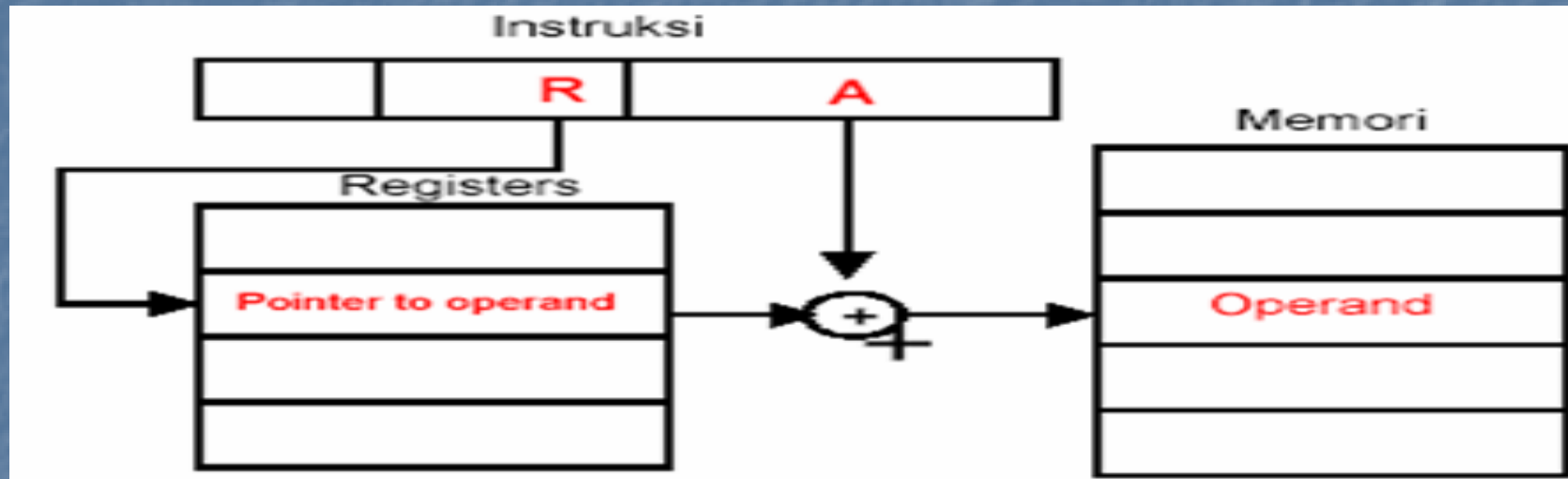
- Metode pengalamatan register tidak langsung mirip dengan mode pengalamatan tidak langsung.
- Perbedaannya adalah field alamat mengacu pada alamat register. Letak operand berada pada memori yang ditunjuk oleh isi register.
- Keuntungan dan keterbatasan pengalamatan register tidak langsung pada dasarnya sama dengan pengalamatan tidak langsung.
 - Keterbatasan field alamat diatasi dengan pengaksesan memori yang tidak langsung sehingga alamat yang dapat direferensi makin banyak.
 - Dalam satu siklus pengambilan dan penyimpanan, mode pengalamatan register tidak langsung hanya menggunakan satu referensi memori utama sehingga lebih cepat daripada mode pengalamatan tidak langsung



Displacement Addressing (6)

- Menggabungkan kemampuan pengalamatan langsung dan pengalamatan register tidak langsung.
- Mode ini mensyaratkan instruksi memiliki dua buah field alamat, setidaknya sebuah field yang eksplisit.
 - Field eksplisit bernilai A dan field implisit mengarah pada register

Displacement Addressing (6)



- Operand berada pada alamat A ditambah isi register.
- Tiga model displacement
 - Relative Addressing
 - Base Register Addressing
 - Indexing

Displacement Addressing

- Relative addressing, register yang direferensi secara implisit adalah program counter (PC).
 - Alamat efektif didapatkan dari alamat instruksi saat itu ditambahkan ke field alamat.
 - Memanfaatkan konsep lokalitas memori untuk menyediakan operand – operand berikutnya.
- Base register addressing, register yang direferensikan berisi sebuah alamat memori, dan field alamat berisi perpindahan dari alamat itu.
 - Referensi register dapat eksplisit maupun implisit.
 - Memanfaatkan konsep lokalitas memori.
- Indexing adalah field alamat mereferensi alamat memori utama, dan register yang direferensikan berisi pemindahan positif dari alamat tersebut.
 - Merupakan kebalikan dari model base register.
 - Field alamat dianggap sebagai alamat memori dalam indexing.
 - Manfaat penting dari indexing adalah untuk eksekusi program – program iteratif

Stack Addressing (7)

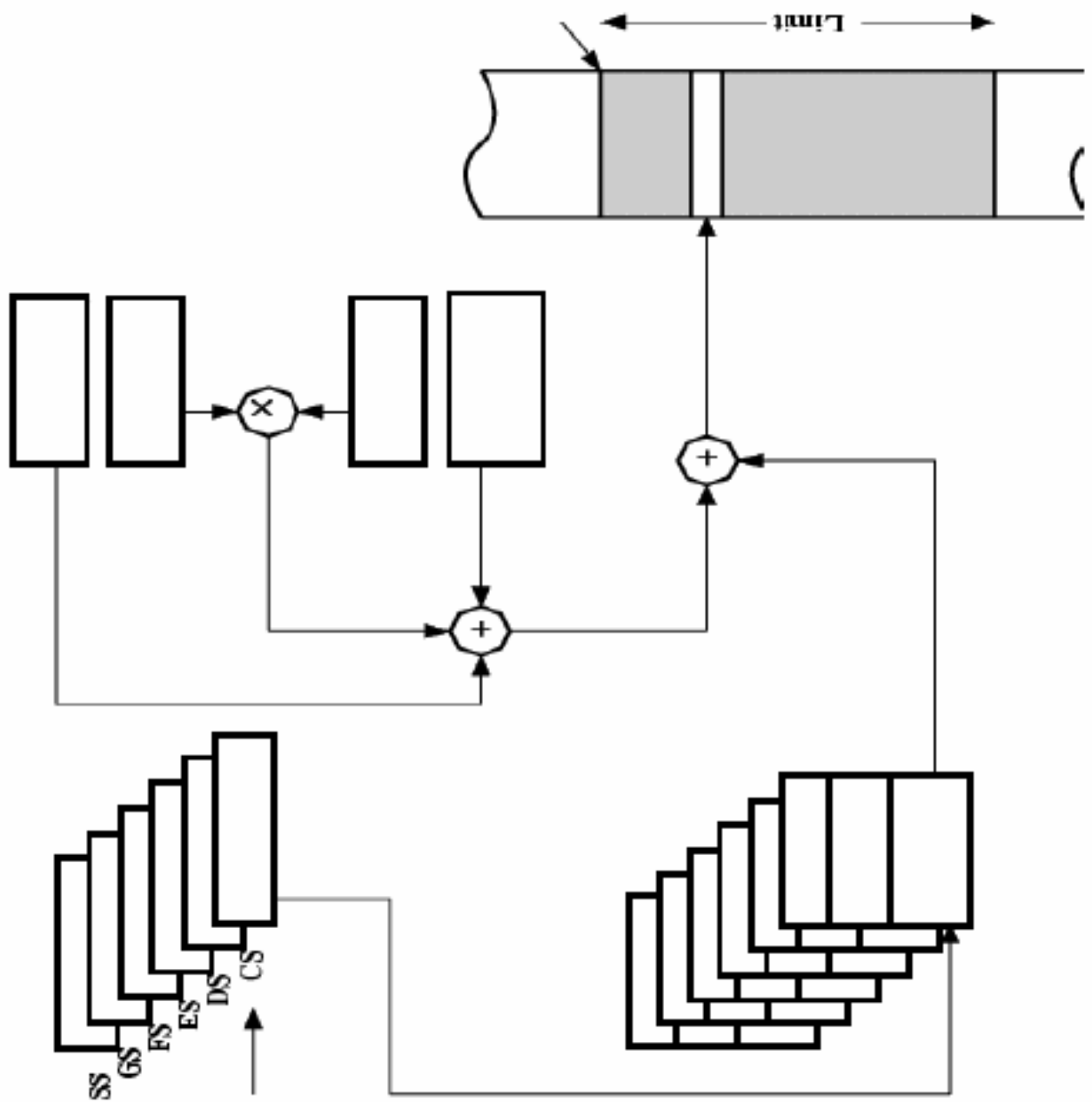
- Stack adalah array lokasi yang linier = pushdown list = last-in-first-out-queue.
- Stack merupakan blok lokasi yang terbalik.
 - Butir ditambahkan ke puncak stack sehingga setiap saat blok akan terisi secara parsial.
- Yang berkaitan dengan stack adalah pointer yang nilainya merupakan alamat bagian paling atas stack.
- Dua elemen teratas stack dapat berada di dalam register CPU, yang dalam hal ini stack pointer mereferensi ke elemen ketiga stack.
- Stack pointer tetap berada di dalam register.
- Dengan demikian, referensi – referensi ke lokasi stack di dalam memori pada dasarnya merupakan pengalamatan register tidak langsung

Mode pengalamatan (tabel)

Mode	Algoritma	Keuntungan	Kerugian
Immediate	Operand = A	Tidak ada referensi memori	Besaran operand terbatas
Direct	$eA = A$	Sederhana	Ruang alamat terbatas
Indirect	$eA = (A)$	Ruang alamat besar	Referensi memori ganda
Register	$eA = R$	Tidak ada referensi memori	Ruang alamat terbatas
Register Indirect	$eA = (R)$	Ruang alamat besar	Referensi memori ekstra
Displacement	$eA = A + (R)$	Fleksibel	Kompleks
Stack	$eA = \text{top of stack}$	Tidak ada referensi	Aplikasi memori terbatas

3.4. Mode Pengalamatan Pentium

- Pentium dilengkapi bermacam – macam mode pengalamatan untuk memudahkan bahasa – bahasa tingkat tinggi mengeksekusinya secara efisien.



Mode pengalamatan pentium

Mode	Algoritma
Immediate	Operand = A
Register	$eA = R$
Displacement	$eA = (SR) + A$
Base	$eA = (SR) + (B)$
Base with displacement	$eA = (SR) + (B) + A$
Scaled index with displacement	$eA = (SR) + (B) + (I) + A$
Base with scaled index and displacement	$eA = (SR) + (I) \times S + (B) + A$
Relative	$eA = (PC) + A$

Keterangan :

SR = register segment

PC = program counter

A = isi field alamat

B = register basis

I = register indeks

S = faktor skala

Mode pengalamatan pentium

- Mode immediate
 - Operand berada di dalam instruksi.
 - Operand dapat berupa data byte, word maupun doubleword
- Mode operand register, operand adalah isi register.
 - Beberapa macam jenis register
 - register 8 bit (AH, BH, CH, DH, AL, BL, CL, DL)
 - register 16 bit (AX, BX, CX, DX, SI, DI, SP, BP)
 - register 32 bit (EAX, EBX, ECX, EDX, ESI, EDI, ESP, EBP)
 - register 64 bit yang dibentuk dari register 32 bit secara berpasangan.
 - register 8, 16 dan 32 merupakan register untuk penggunaan umum (general purpose register).
 - register 64 bit biasanya untuk operasi floating point.
 - register segmen (CS, DS, ES, SS, FS, GS)

Mode pengalamatan pentium

- Mode displacement

- Alamat efektif berisi bagian – bagian instruksi dengan displacement 8, 16, atau 32 bit.
- Dengan segmentasi, seluruh alamat dalam instruksi mengacu ke sebuah offset di dalam segmen.
- Dalam Pentium, mode ini digunakan untuk mereferensi variabel – variabel global