

# **Mode Pengalamatan Pertemuan 7**

**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

# Mode pengalamatan pentium

- Mode base
  - Pengalamatan indirect yang menspesifikasi satu register 8, 16 atau 32 bit berisi alamat efektifnya.
  - Base with displacement mode
    - Instruksi mempunyai displacement yang akan ditambahkan ke register basisi.
    - Umumnya termasuk general purpose register.
    - Contoh penggunaan mode ini adalah digunakan kompiler untuk menunjuk awal daerah variabel, untuk mengindeks suatu larik, dan digunakan untuk mengakses field sebuah record

# Mode pengalamatan pentium

- Mode scaled index with displacement
  - Instruksi mengandung displacement yang akan ditambahkan ke register indeks.
  - Register indeks dapat berupa sembarang register kecuali ES yang umumnya untuk pengolahan stack.
  - Dalam perhitungan alamat efektif, isi register indeks dikalikan dengan 1, 2, 4, atau 8 dan kemudian ditambahkan ke displacement.
  - Mode ini sangat cocok untuk pengindekan larik.
  - Faktor skala 2 digunakan untuk larik integer 16 bit, skala 4 untuk larik integer 32 bit dan faktor skala 8 untuk bilangan floating point

# Mode Pengalamatan pentium

- Base with index and displacement mode
  - menjumlahkan isi register basis, register indeks, dan displacement untuk mendapatkan alamat efektifnya.
  - Register basis dan register indeks dapat berupa sembarang register, kecuali ESP.
  - Contoh :
    - Untuk mengakses larik lokal pada stack frame.
    - Mode ini juga dapat digunakan untuk mendukung larik dua dimensi, displacement menunjuk awal larik dan setiap register menangani satu dimensi larik

# Mode Pengalamatan pentium

- Base scaled index with displacement mode
  - Alamat efektif diperoleh dari penjumlahan isi register indeks yang dikalikan dengan faktor skala
  - Isi register basis, dan displacement.
  - Mode ini sangat berguna untuk pengaksesan larik pada stack frame

# Mode pengalamatan pentium

- Mode relative addressing
  - Digunakan dalam instruksi – instruksi tranfer kontrol.
  - Displacement ditambahkan ke program counter (PC), yang menunjuk ke instruksi berikutnya

## 3.5. Format Instruksi

- Format instruksi menentukan susunan dan tata letak bit suatu instruksi.
- Format instruksi harus mencakup opcode serta implisit dan eksplisit operand.
- Biasanya set instruksi memiliki lebih dari satu format instruksi.
- Inti dari format instruksi adalah menentukan panjang instruksi dan alokasi bit dalam instruksi tersebut



# Panjang Instruksi

- Penentuan panjang intruksi mempengaruhi dan dipengaruhi oleh
  - Ukuran memori
  - Organisasi memori
  - Struktur bus
  - Kompleksitas CPU
  - Kecepatan CPU

- Bahasan RISC -

# Penentuan panjang instruksi ?

Pertimbangan : (INSTRUKSI)

- Instruksi yang kompleks mempengaruhi perancangan perangkat keras prosesor, karena fungsi – fungsi yang disajikan CPU harus diimplementasikan dalam perangkat keras.
- Semakin kompleks perangkat keras, tentunya akan meningkatkan faktor biaya walau belum tentu meningkatkan kinerja komputer secara keseluruhan.

Penentuan panjang instruksi menjadi sangat essensi untuk mencapai kinerja komputer yang maksimal

# Penentuan panjang instruksi ?

Pertimbangan : (PROGRAMMER)

- Menginginkan opcode, operand, dan mode pengalamatan yang lebih banyak serta range alamat yang lebih besar karena semua itu akan mempermudah pemrogram mengimplementasikan keinginannya dalam program.
- Pertimbangannya bahwa opcode, operand dan mode pengalamatan yang lebih banyak akan membutuhkan ruang yang lebih besar.
- Instruksi 32 bit akan menempati ruang dua kali lebih banyak daripada instruksi 16 bit, namun kegunaannya mungkin tidak akan dua kali lebih banyak

# Penentuan panjang instruksi ?

Pertimbangan lain :

- Panjang instruksi harus sama dengan panjang perpindahan memori (pada sistem bus, panjang bus data) dan panjang instruksi seharusnya merupakan kelipatan panjang instruksi lainnya.
- Hal ini harus dipertimbangkan untuk mendapatkan optimalisasi proses eksekusi instruksi nantinya, baik kecepatan perpindahan maupun alokasi memorinya.
- Kecepatan perpindahan data tidak dapat diatasi dengan menambah kecepatan prosesor.
- Kecepatan prosesor hanya berhubungan dengan eksekusi internalnya, sedangkan kecepatan perpindahan tergantung bus, memori, dan data itu sendiri.
- Cara meningkatkan kecepatan perpindahan data adalah dengan menggunakan cache memori dan menggunakan instruksi – instruksi yang lebih pendek

# Penentuan panjang instruksi ?

- Panjang instruksi harus merupakan kelipatan panjang karakter, yang umumnya 8 bit, dan kelipatan panjang bilangan fixed point.
- Diabaikan ?
  - Terjadi pemborosan bit pada setiap word ketika sejumlah karakter disimpan di dalamnya
- Keputusan salah yang pernah diambil IBM
  - Mengeluarkan arsitektur prosesor 36 bit, terjadi banyak pemborosan karena ukuran karakter 8 bit.
  - Arsitektur tersebut diganti dengan arsitektur 32 bit

# Bagaimana dengan *Alokasi Bit* ?

- Inti dalam alokasi bit adalah berada pada untung – rugi antara jumlah opcode dengan kemampuan pengalamatannya.
- Opcode yang banyak akan menyebabkan bit yang lebih banyak pada field opcode, yang secara otomatis akan mengurangi jumlah bit untuk pengalamatan.
- Faktor yang merupakan hal – hal yang penting dalam menentukan penggunaan bit – bit pengalamatan :
  - Jumlah mode pengalamatan
  - Jumlah operand
  - Register vs memori
  - Jumlah set register
  - Jangkauan alamat
  - Granularitas alamat

- Jumlah mode pengalamatan
  - Mode pengalamatan dapat dilakukan secara implisit atau eksplisit, yang kesemuanya memerlukan jumlah bit yang berbeda
- Jumlah operand
  - Jumlah operand sangat mempengaruhi kemampuan instruksi.
  - Jumlah operand yang sedikit biasanya akan menjadikan instruksi yang panjang dalam suatu fungsi
- Register vs memori
  - Penggunaan register maupun memori membutuhkan jumlah bit yang berbeda.
  - Pada pengalamatan implisit dengan register akan dibutuhkan bit lebih kecil dari pada mode pengalamatan langsung ke memori

## ■ Jumlah set register

- Jumlah set register juga mempengaruhi penggunaan bit – bit instruksi.
- General purpose register yang umumnya dimiliki hampir seluruh arsitektur komputer dapat digunakan untuk register alamat maupun register instruksi

## ■ Jangkauan alamat

- untuk alamat – alamat yang mereferensi memori secara eksplisit, jangkauan ditentukan oleh jumlah bit yang digunakan untuk pengalamatan.
- Pertimbangan menggunakan mode displacement patut dipertimbangkan untuk memiliki jangkauan pengalamatan yang besar

## ■ Granularitas alamat :

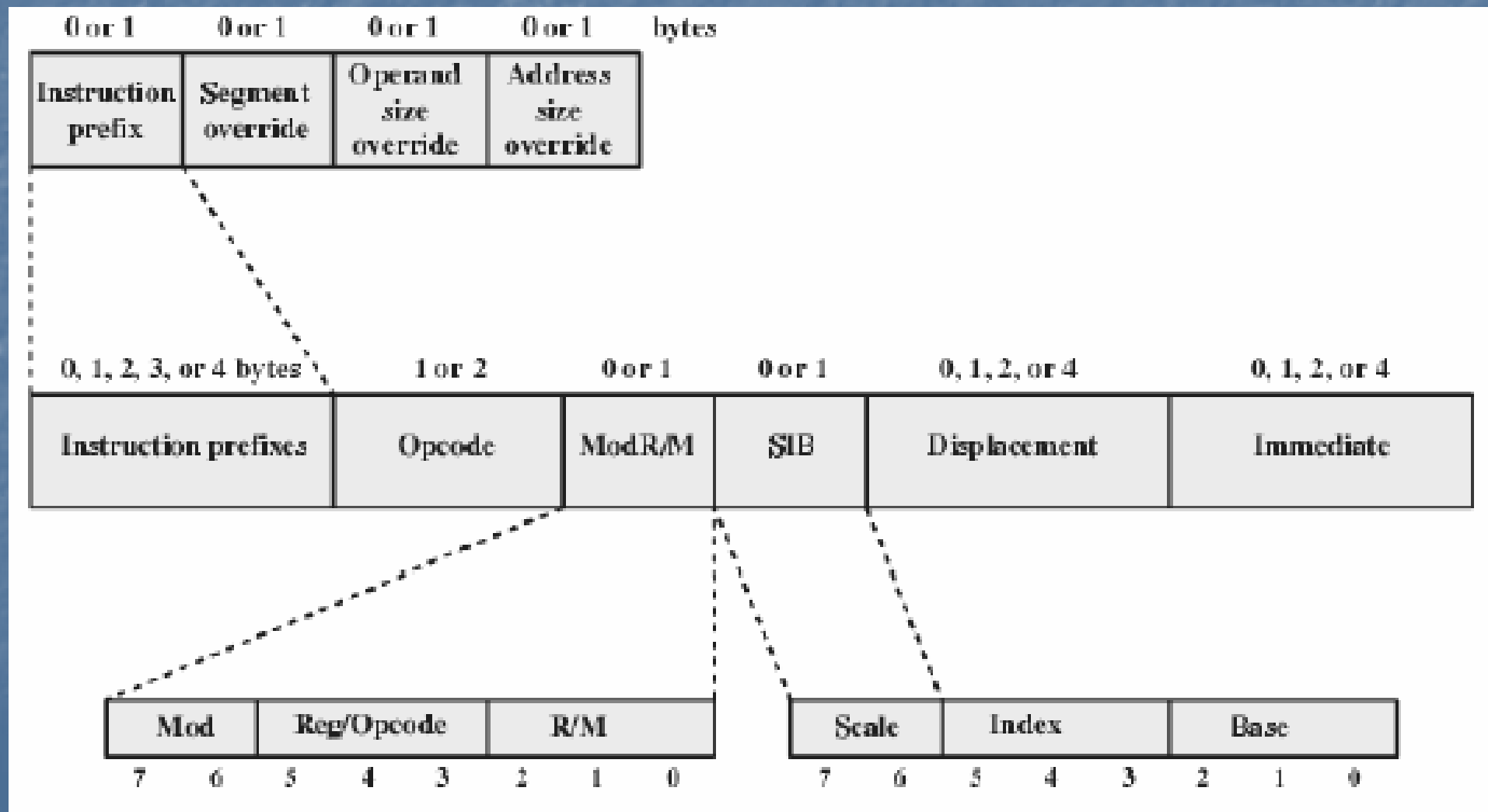
- Pengalamatan yang mereferensi memori dapat digunakan pengalamatan yang mereferensi word atau byte



## 3.6. Format Instruksi Pentium

- Arsitektur Pentium dilengkapi bermacam – macam format instruksi.
- Instruksi – instruksinya dibangun mulai dari nol hingga empat prefiks instruksi opsional, sebuah opcode satu atau dua byte, specifier alamat opsional, yang terdiri dari Mod r/m byte dan scale index byte (SIB), sebuah opsional displacement, dan opsional immediate

# Format Instruksi Pentium



# Bagian prefix byte

## ■ Instruction Prefixes Biasa

- Berisi : Prefiks Lock dan Prefiks perulangan.
- Prefiks Lock yang digunakan untuk keamanan penggunaan shared memory yang eksklusif dalam lingkungan multiprosesor.
- Prefiks perulangan berguna untuk operasi perulangan yang dapat diproses lebih cepat daripada menggunakan loop perangkat lunak biasa

## ■ Segment Override

- Menspesifikasi register segmen yang harus dipakai instruksi
- Mengesampingkan (override) pilihan register segmen default yang dihasilkan Pentium untuk instruksi tersebut

# Bagian prefix byte

## ■ Address Size

- Prosesor dapat mengamati memori dengan menggunakan alamat 16 bit atau 32 bit.
- Ukuran alamat menentukan ukuran displacement dalam instruksi dan ukuran offset alamat yang dihasilkan selama perhitungan alamat efektif berlangsung.
- Prefiks ukuran alamat digunakan untuk mengubah alamat 16 bit ke 32 bit dan sebaliknya

## ■ Operand Size

- Instruksi memiliki ukuran operand default 16 bit dan 32 bit
- Prefiks operand mengubah operand 16 bit ke 32 bit dan sebaliknya

# Field Instruksi

- Opcode
  - Opcode dapat mencakup bit – bit yang menspesifikasikan apakah suatu data merupakan byte atau full-size, arah operasi data, dan apakah immediate data field harus merupakan sign-extended
- Mod r/m
  - Memberikan informasi pengalamatan.
  - Byte Mod r/m menspesifikasikan apakah operand berada di dalam register atau berada di dalam memori. Apabila operand berada di dalam memori, maka field – field yang berada di dalam byte akan menspesifikasi mode pengalamatan yang akan dipakai
- SIB
  - Berisi skala indeks register dan base register
- Displacement
  - Bila mode pengalamatan menggunakan mode ini maka akan ditambahkan field displacement integer bertanda 8 bit, 16 bit atau 32 bit
- Immediate
  - Memberikan nilai operand 8 bit, 16 bit atau 32 bit

# Offset vs Flexibilitas

- Format Pentium memungkinkan menggunakan offset tidak hanya 1 byte saja tetapi dapat 2 byte atau 4 byte yang ditujukan untuk keperluan indexing.
- Walaupun pemakaian offset menyebabkan instruksi lebih panjang, tetapi fitur ini memberikan fleksibilitas yang dibutuhkan

# Kesimpulan

- Instruksi = biner
  - Bagian opcode
  - Bagian alamat
- Tipe data dan jenis instruksi digolongkan kebeberapa kelompok
- Panjang bit Opcode mempengaruhi jumlah jenis instruksi
- Jumlah bit Alamat mempengaruhi jangkauan alamat yang bisa digunakan
- Terdapat berbagai macam mode pengalamatan digunakan sesuai dengan kondisi

# Soal - Soal

- Jelaskan hubungan antara jumlah bit pada opcode dengan jumlah instruksi yang ada !
- Jelaskan hubungan antara jumlah bit pada Alamat yang ada di set instruksi dengan jumlah alamat yang bisa di jangkau !
- Bagaimana cara agar set instruksi jumlah dapat menambah jangkauan pada memori!
- Jelaskan kapan dan pada saat apa mode pengalamatan digunakan