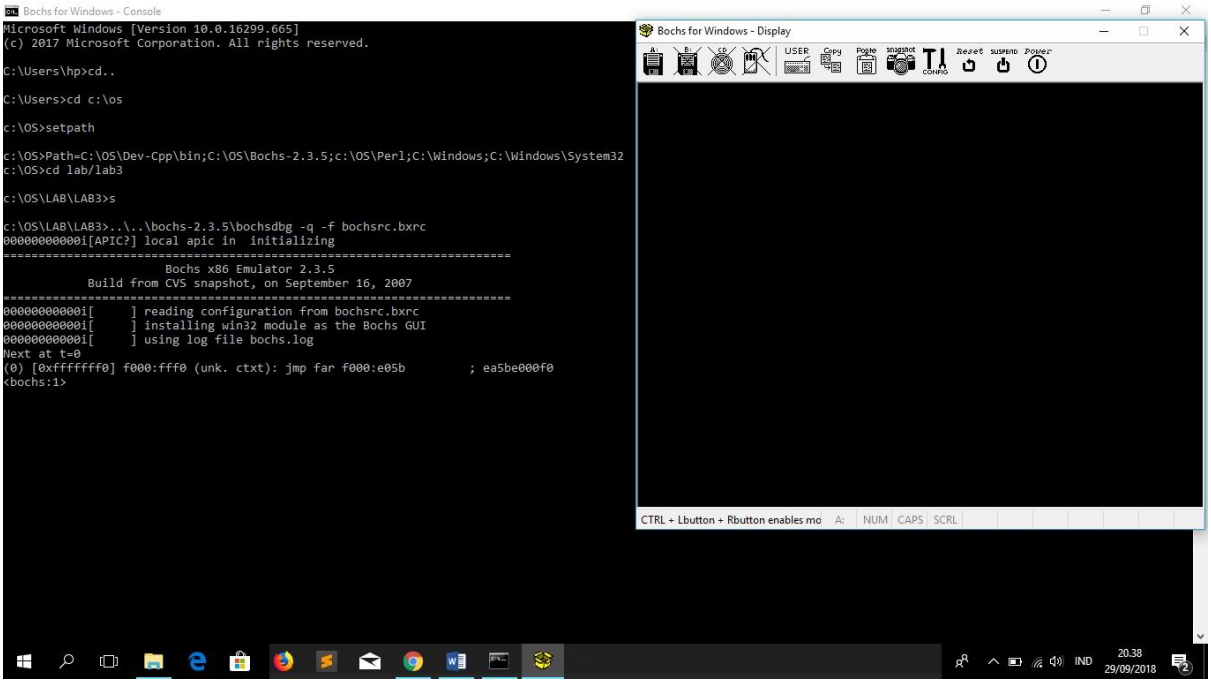
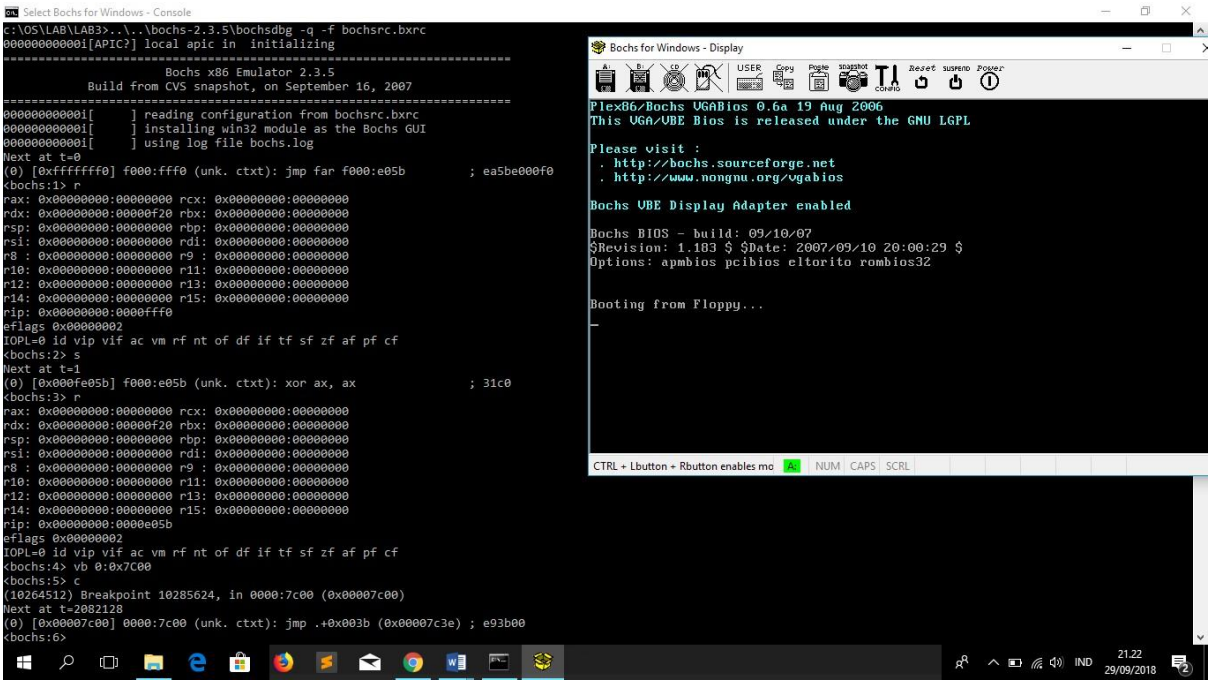
Lembar Kerja Praktikum Modul 3

Nama : Rizky Tri Setya W
NIM : L200170054

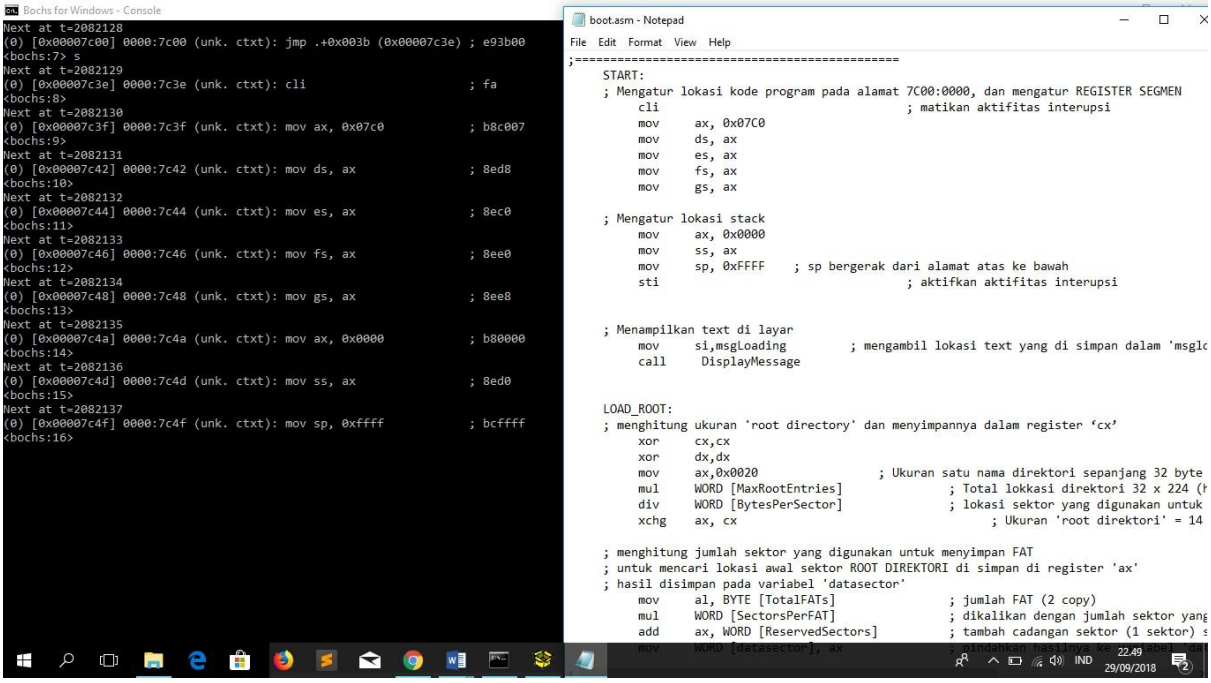
- Pada windows comman prompt ketik cd os
- Jalankan perintah setpath
- Kemudian ketikan “LAB/LAB3
- Mulai melakukan debugging : masukkan perintah ‘S’ <ENTER>



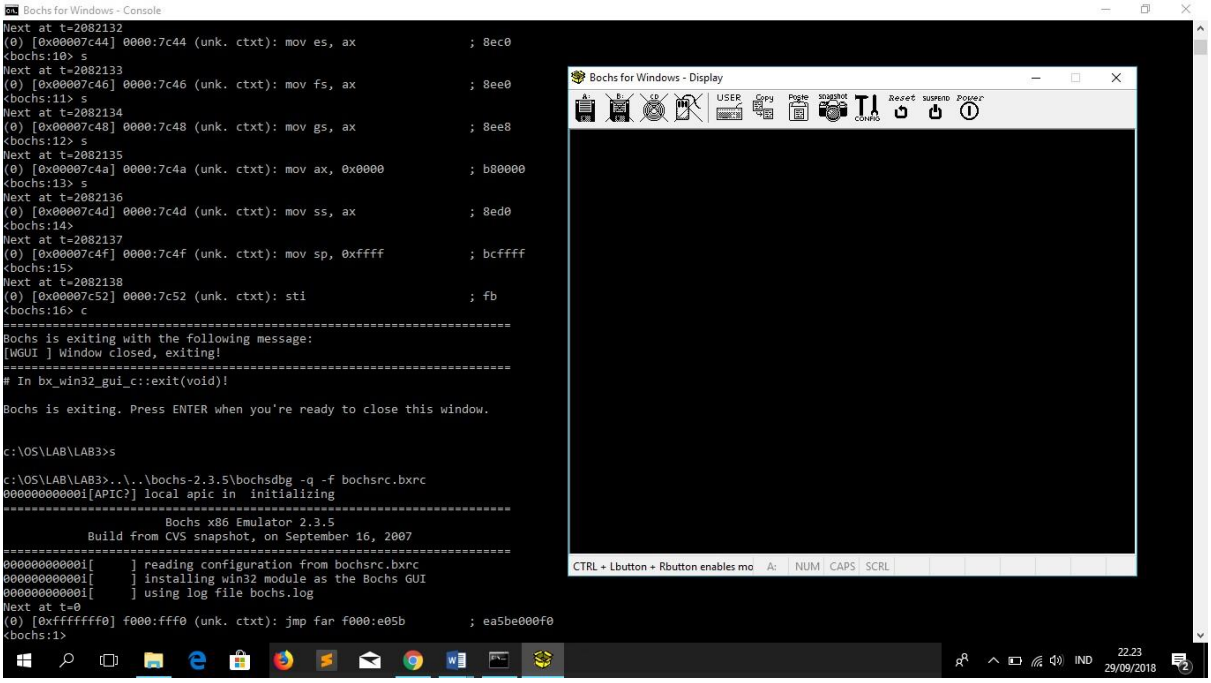
- Masukkan perintah ‘r’ <ENTER> untuk melihat isi register CS dan IP
- Selanjutnya, untuk mengeksekusi perintah tersebut, ketikan ‘s’ <ENTER>
- Kemudian lanjutkan dengan perintah ‘r’ <ENTER>
- Selanjutnya masukan perintah ‘vb 0:0x7C00
- Masukkan perintah ‘c’ <ENTER>

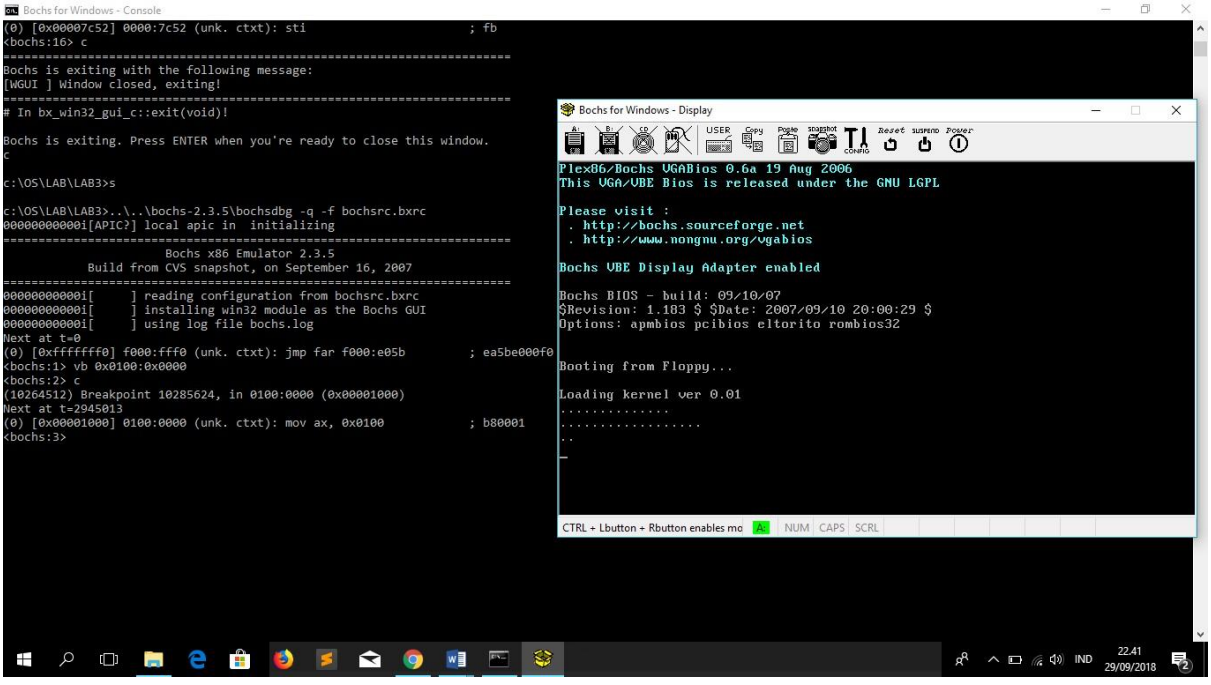

Select Bochs for Windows - Console
c:\OS\LAB\LAB3>..\..\bochs-2.3.5\bochsdbg -q -f bochsrc.bxrc
00000000000i[APIC?] local apic in initializing
Bochs x86 Emulator 2.3.5
Build from CVS snapshot, on September 16, 2007
00000000000i[   ] reading configuration from bochsrc.bxrc
00000000000i[   ] installing win32 module as the Bochs GUI
00000000000i[   ] using log file bochs.log
Next at t=0
(0) [0xfffffff0] f000:fff0 (unk. ctxt): jmp far f000:e05b   ; ea5be000f0
<bochs:1> r
rax: 0x00000000:00000000 rcx: 0x00000000:00000000
rdx: 0x00000000:00000f20 rbx: 0x00000000:00000000
rsp: 0x00000000:00000000 rbp: 0x00000000:00000000
rsi: 0x00000000:00000000 rdi: 0x00000000:00000000
r8 : 0x00000000:00000000 r9 : 0x00000000:00000000
r10: 0x00000000:00000000 r11: 0x00000000:00000000
r12: 0x00000000:00000000 r13: 0x00000000:00000000
r14: 0x00000000:00000000 r15: 0x00000000:00000000
rip: 0x00000000:0000fff0
eflags 0x00000002
IOPL=0 id vip vif ac vm rf nt of df if tf sf zf af pf cf
<bochs:2> s
Next at t=1
(0) [0x000fe05b] f000:e05b (unk. ctxt): xor ax, ax   ; 31c0
<bochs:3> r
rax: 0x00000000:00000000 rcx: 0x00000000:00000000
rdx: 0x00000000:00000f20 rbx: 0x00000000:00000000
rsp: 0x00000000:00000000 rbp: 0x00000000:00000000
rsi: 0x00000000:00000000 rdi: 0x00000000:00000000
r8 : 0x00000000:00000000 r9 : 0x00000000:00000000
r10: 0x00000000:00000000 r11: 0x00000000:00000000
r12: 0x00000000:00000000 r13: 0x00000000:00000000
r14: 0x00000000:00000000 r15: 0x00000000:00000000
rip: 0x00000000:0000e05b
eflags 0x00000002
IOPL=0 id vip vif ac vm rf nt of df if tf sf zf af pf cf
<bochs:4> vb 0:0x7C00
<bochs:5> c
(10264512) Breakpoint 10285624, in 0000:7c00 (0x00007c00)
Next at t=2082128
(0) [0x00007c00] 0000:7c00 (unk. ctxt): jmp .+0x003b (0x00007c3e) ; e93b00
<bochs:6>
Bochs for Windows - Display
A:
B:
CD
HD
USER
Copy
Paste
Snapshot
config
Reset
suspend
Power
Plex86/Bochs VGABios 0.6a 19 Aug 2006
This VGA/VBE Bios is released under the GNU LGPL
Please visit :
. http://bochs.sourceforge.net
. http://www.nongnu.org/vgabios
Bochs VBE Display Adapter enabled
Bochs BIOS - build: 09/10/07
$Revision: 1.183 $ $Date: 2007/09/10 20:00:29 $
Options: apmbios pcibios eltorito rombios32
Booting from Floppy...
CTRL + Lbutton + Rbutton enables mo
NUM
CAPS
SCRL
IND
21.22
29/09/2018

- Masukkan 10 intruksi ‘s’ yang akan dieksekusi oleh PC dengan program yang terdapat pada ‘boot.asm’
- Kemudian kita bisa membandingkan hasil yang di Command Prompt dg boot.asm

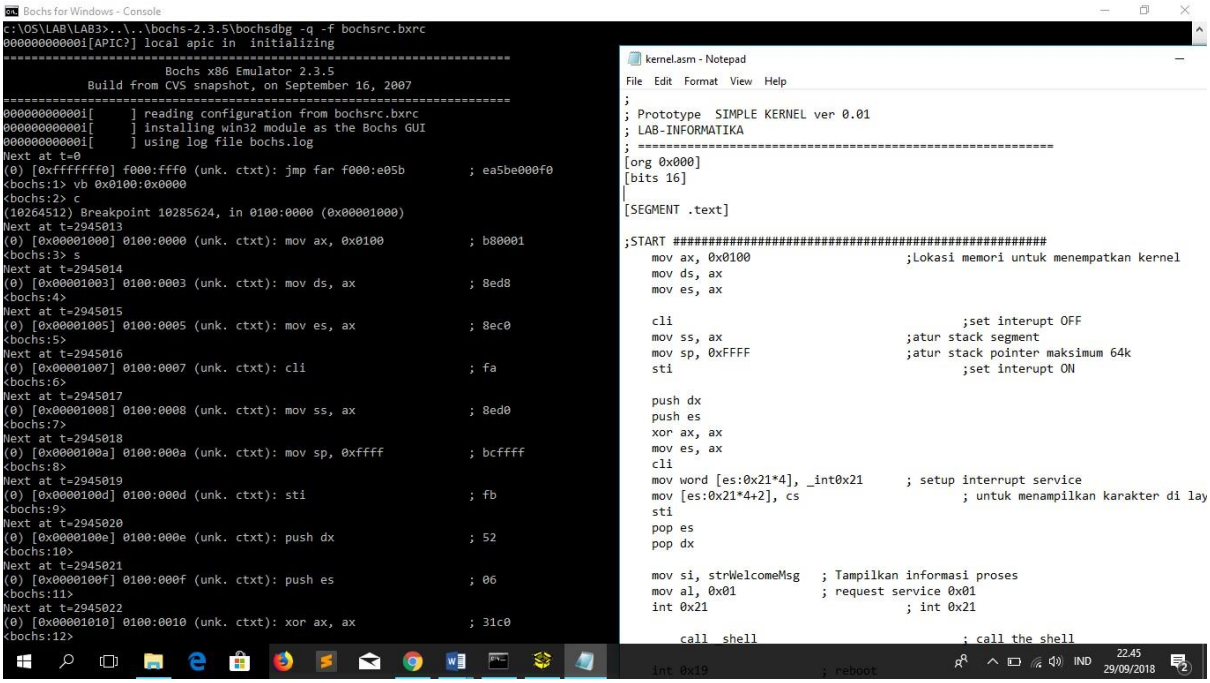


- Kemudian kita klik power pada bochs di windows
- Ketik ‘c’ untuk kembali ke command prompt
- Selanjutnya ketik ‘s’ <ENTER> dari command prompt
- Selanjutnya ketik ‘c’ pada command prompt untuk perintahkan PC melanjutkan pekerjaannya
- Kemudian masukan ‘vb 0x0100:0x0000’
- Dan selanjutnya ketikan ‘c’ pada command prompt





- Kemudian teruskan langkah PC Simulator step-by-step minimal sebanyak 10x, dengan ketikan ‘s’ <ENTER>, step berikutnya dapat dilakukan dengan cara menekan enter. - Selanjutnya badingkan source-code antara di command prompt dengan di kernel.asm (kernel.asm berada di lokal C folder OS kemudian pilih di LAB dan selanjutnya di LAB3)



Langkah-langkah di atas adalah merupakan salah satu cara yang banyak digunakan para pengembang perangkat lunak, termasuk pengembangan sistim operasi. Jika anda sebaiknya anda membiasakan dengan kegiatan di atas.

TUGAS :

1. Tabel pemetaan memori pada PC

Pemetaan Langsung (Direct Mapping)

Pemetaan langsung adalah teknik yang paling sederhana, yaitu teknik ini memetakan blok memori utama hanya ke sebuah saluran cache saja. Jika suatu block ada di cache, maka tempatnya sudah tertentu. Keuntungan dari direct mapping adalah sederhana dan murah. Sedangkan kerugian dari direct mapping adalah suatu blok memiliki lokasi yang tetap (jika program mengakses 2 blok yang di map ke line yang sama secara berulang-ulang, maka cachemiss sangat tinggi).

Berikut penjelasan lebih detail :

§ Setiap blok pada main memory dipetakan dengan line tertentu pada cache. i = j modulo C di mana i adalah nomor line pada cache yang digunakan untuk meletakkan blok main memory ke-j.

§ Jika M = 64 dan C = 4, maka pemetaan antara line dengan blok menjadi seperti

berikut :

Line 0 can hold blocks 0, 4, 8, 12, ...

Line 1 can hold blocks 1, 5, 9, 13, ...

Line 2 can hold blocks 2, 6, 10, 14, ...

Line 3 can hold blocks 3, 7, 11, 15, ...

§ Pada cara ini, address pada main memory dibagi 3 field atau bagian, yaitu:

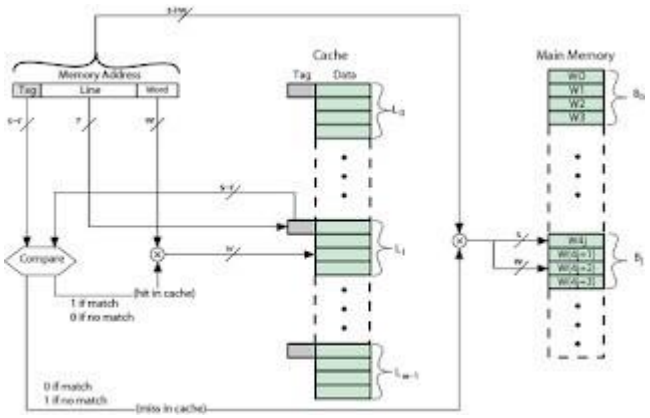
- Tag identifier.
- Line number identifier o Word identifier (offset)

§ Word identifier berisi informasi tentang lokasi word atau unit addressable lainnya dalam line tertentu pada cache.

§ Line identifier berisi informasi tentang nomor fisik (bukan logika) line pada chace §

Tag identifier disimpan pada cache bersama dengan blok pada line.

- Untuk setiap alamat memory yang dibuat oleh CPU, line tertentu yang menyimpan copy alamat tsb ditentukan, jika blok tempat lokasi data tersebut sudah dikopi dari main memory ke cache.
- Tag yang ada pada line akan dicek untuk melihat apakah benar blok yang dimaksud ada line tsb.



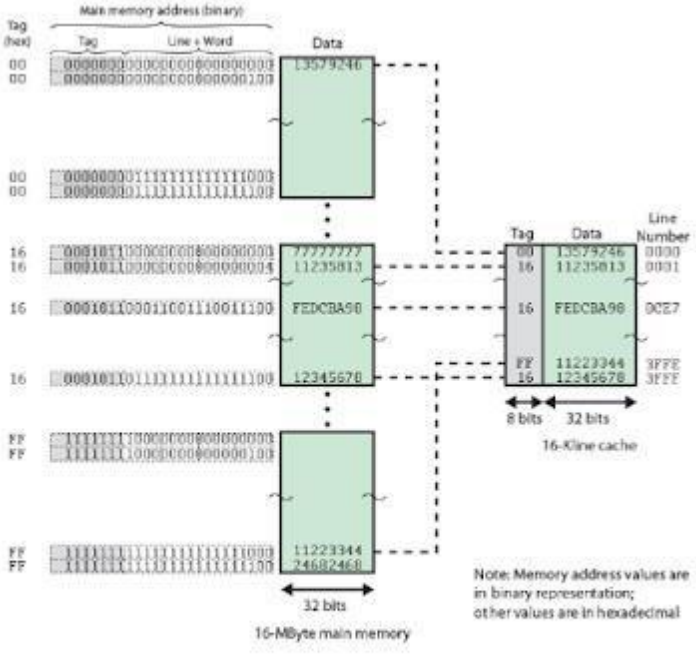
Gambar 2.1 : Gambar Organisasi Direct Mapping.

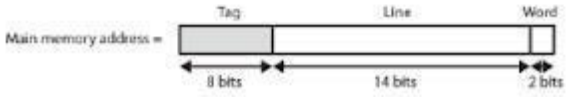
Keuntungan Menggunakan Direct Mapping antara lain :

1. Mudah dan Murah diimplementasikan
2. Mudah untuk menentukan letak salinan data main memory pada chace.

Kerugian menggunakan Direct Mapping antara lain :

1. Setiap blok main memory hanya dipetakan pada 1 line saja.
2. Terkait dengan sifat lokal pada main memory, sangat mungkin mengakses blok yang dipetakan pada line yang sama pada cache. Blok seperti ini akan menyebabkan seringnya sapu masuk dan keluar data ke/dari cache, sehingga hit ratio mengecil. Hit ratio adalah perbandingan antara jumlah ditemukannya data pada cache dengan jumlah usaha mengakses cache.





Gambar 2.2 : Gambar Contoh Pengalamatan Direct Mapping.

Ringkasan direct mapping nampak pada tabel berikut:

Item	Keterangan
Panjang alamat	(s+w) bits
Jumlah unit yang dapat dialamati	2s+w words or bytes
Ukuran Bloks sama dengan ukuran Line	2w words or bytes
Jumlah blok memori utama	2s+ w/2w = 2s
Jumlah line di chace	M = 2r
Besarnya tag	(s - r) bits

- **Pemetaan Asosiatif (Associative Mapping)**

Pemetaan asosiatif mengatasi kekurangan pemetaan langsung dengan cara mengizinkan setiap blok memori utama untuk dimuatkan ke sembarang saluran cache. Dengan pemetaan assosiatif, terdapat fleksibilitas penggantian blok ketika blok baru dibaca ke dalam cache. Kekurangan pemetaan asosiatif yang utama adalah kompleksitas rangkaian yang diperlukan untuk menguji tag seluruh saluran cache secara parallel, sehingga pencarian data di cache menjadi lama.

§ Memungkinkan blok diletakkan di sebarang line yang sedang tidak terpakai.

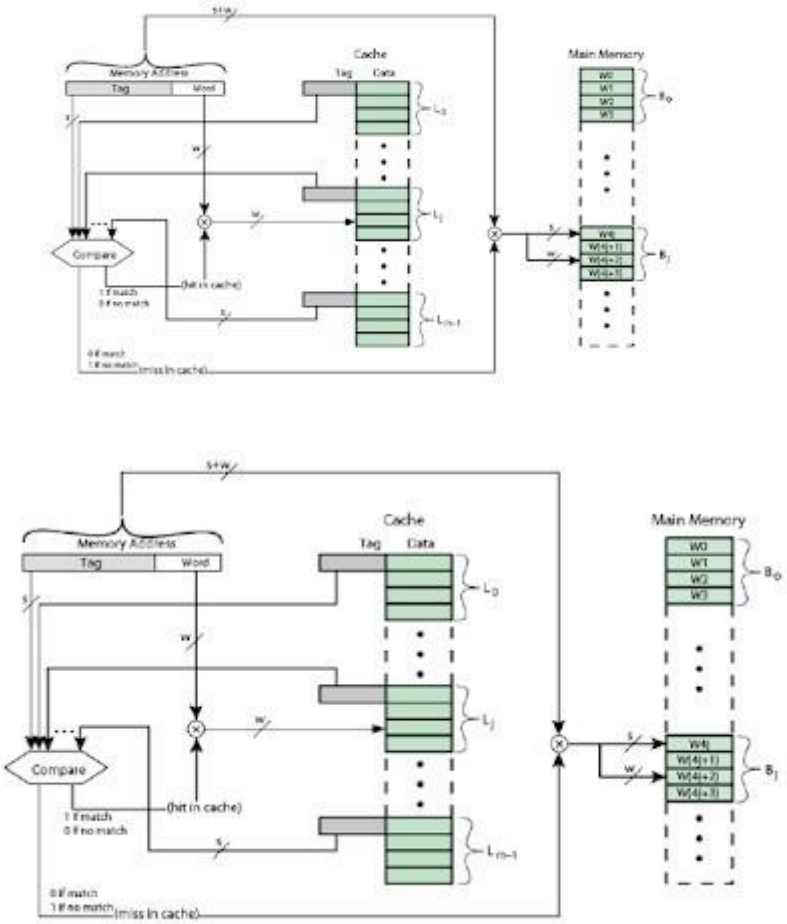
§ Diharapkan akan mengatasi kelemahan utama Direct Mapping.

§ Harus menguji setiap cache untuk menemukan blok yang diinginkan.

o Mengecek setiap tag pada line o Sangat

lambat untuk cache berukuran besar.

§ Nomor line menjadi tidak berarti. Address main memory dibagi menjadi 2 field saja, yaitu tag dan word offset.



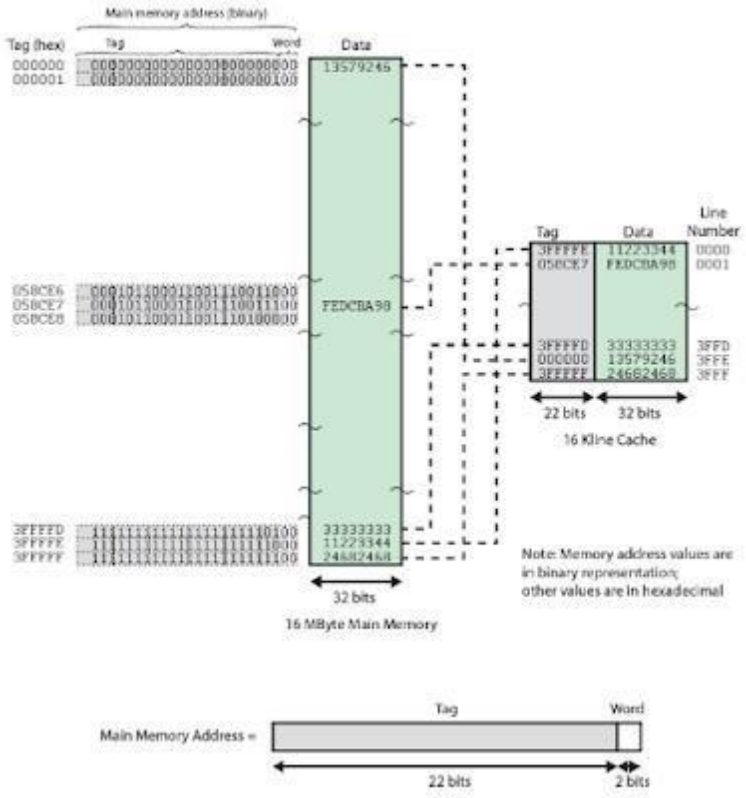
Gambar 2.3 : Gambar Organisasi Associative Mapping.

§ Melakukan pencarian ke semua tag untuk menemukan blok.

§ Cache dibagi menjadi 2 bagian : o

lines dalam SRAM o tag dalam

associative memory



Gambar 2.4 : Gambar Contoh Pengalamatan Associative Mapping

Keuntungan Associative Mapping : Cepat dan fleksibel.

Kerugian Associative Mapping : Biaya Implementasi, misalnya untuk cache ukuran 8 kbyte dibutuhkan 1024 x 17 bit associative memory untuk menyimpan tag identifier.

Ringkasan Associative Mapping nampak pada tabel berikut:

Item	Keterangan
Panjang alamat	(s+w) bits
Jumlah unit yang dapat dialamati	2s+w words or bytes

Ukuran Bloks sama dengan ukuran Line	2w words or bytes
Jumlah blok memori utama	2s+ w/2w = 2s
Jumlah line di chace	Undetermined
Besarnya tag	s bits

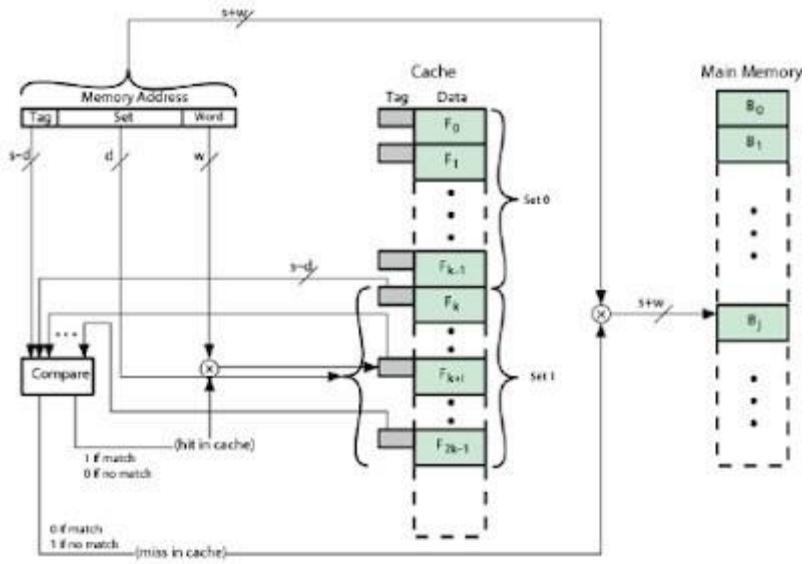
 Pemetaan Asosiatif Set (Set Associative Mapping)

Pada pemetaan ini, cache dibagi dalam sejumlah sets. Setiap set berisi sejumlah line. Pemetaan asosiatif set memanfaatkan kelebihan-kelebihan pendekatan pemetaan langsung dan pemetaan asosiatif.

§ Merupakan kompromi antara Direct dengan Full Associative Mapping.

§ Membagi cache menjadi sejumlah set (v) yang masing-masing memiliki sejumlah line (k)

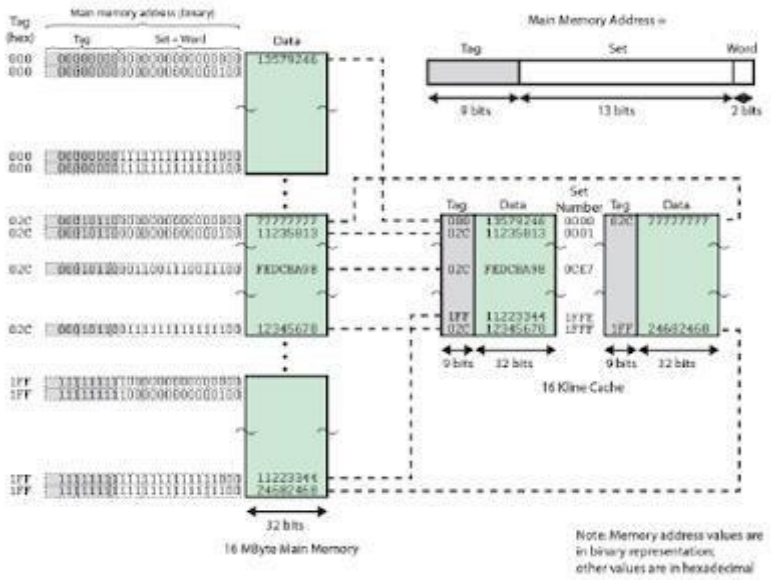
§ Setiap blok dapat diletakkan di sebarang line dengan nomor set: nomor set = j modulo v



Gambar 2.5 : Gambar Organisasi K-Way Set Associative Mapping.

§ Jika sebuah set dapat menampung X line, maka cache disebut memiliki X way set associative cache.

§ Hampir semua cache yang digunakan saat ini menggunakan organisasi 2 atau 4-way set associative mapping.



Gambar 2.6 : Gambar Contoh Pengalamatan 2-Way Associative Mapping.

Keuntungan menggunakan Set Associative Mapping antara lain:

Setiap blok memori dapat menempati lebih dari satu kemungkinan nomor line (dapat menggunakan line yang kosong), sehingga thrashing dapat diperkecil

Jumlah tag lebih sedikit (dibanding model associative), sehingga jalur untuk melakukan

perbandingan tag lebih sederhana.

Ringkasan Set Associative Mapping nampak pada tabel berikut:

Item	Keterangan
Panjang alamat	(s+w) bits
Jumlah unit yang dapat dialamati	2s+w words or bytes

Ukuran Bloks sama dengan ukuran Line	2w words or bytes
Jumlah blok memori utama	2d
Jumlah line dalam set	K
Jumlah set	V=2d
Jumlah line di chace	Kv = k*2d
Besarnya tag	(s – d)bits

2. Perbedaan mode kerja ‘Real-Mode’ dan mode kerja ‘Protect-Mode’ pada PC IBM Compatible

- Real-Mode

Real-Mode adalah sebuah modus di mana prosesor Intel x86 berjalan seolaholah dirinya adalah sebuah prosesor Intel 8085 atau Intel 8088, meski ia merupakan prosesor Intel 80286 atau lebih tinggi. Karenanya, modus ini juga disebut

sebagai modus 8086 (8086 Mode). Dalam modus ini, prosesor hanya dapat mengeksekusi instruksi 16-bit saja dengan menggunakan register internal yang berukuran 16-bit, serta hanya dapat mengakses hanya 1024 KB dari memori karena hanya menggunakan 20-bit jalur bus alamat. Semua program DOS berjalan pada modus ini.

Prosesor yang dirilis setelah 8085, semacam Intel 80286 juga dapat menjalankan instruksi 16-bit, tapi jauh lebih cepat dibandingkan 8085. Dengan kata lain, Intel 80286 benar-benar kompatibel dengan prosesor Intel 8086 yang didesain sebelumnya. Sehingga prosesor Intel 80286 pun dapat menjalankan program-program 16-bit yang didesain untuk 8085 (IBM PC), dengan tentunya kecepatan yang jauh lebih tinggi. Dalam Real-mode, tidak ada proteksi ruang alamat memori, sehingga tidak dapat melakukan multi-tasking. Inilah sebabnya, mengapa program-program DOS bersifat single-tasking. Jika dalam modus real terdapat multi-tasking, maka kemungkinan besar antara dua program yang sedang berjalan, terjadi tabrakan (crash) antara satu dengan lainnya.

- Protected Mode

Modus terproteksi (protected mode) adalah sebuah modus di mana terdapat proteksi ruang alamat memori yang ditawarkan oleh mikroprosesor untuk digunakan oleh sistem operasi. Modus ini datang dengan mikroprosesor Intel 80286 atau yang lebih tinggi. Karena memiliki proteksi ruang alamat memori, maka dalam modus ini sistem operasi dapat melakukan multitasking.

Prosesor Intel 80286 memang dilengkapi kemampuan masuk ke dalam modus terproteksi, tapi tidak dapat keluar dari modus tersebut tanpa harus mengalami reset (warm boot atau cold boot). Kesalahan ini telah diperbaiki oleh Intel dengan merilis prosesor Intel 80386 yang dapat masuk ke dalam modus terproteksi dan keluar darinya tanpa harus melakukan reset. Inilah sebabnya mengapa Windows 95/Windows 98 dilengkapi dengan modus Restart in MS-DOS Mode, meski sebenarnya sistem operasi tersebut merupakan sistem operasi yang berjalan dalam modus terproteksi.