

Contoh :

1110 bilangan desimalnya adalah

$$(1 * 2^3) + (1 * 2^2) + (1 * 2^1) + (0 * 2^0) = 14$$

c. Sistem Bilangan Oktal

Bilangan oktal merupakan bilangan berdasar 8, jadi bilangan ini hanya terdiri dari angka 0 sampai 7. Konversi bilangan oktal ke desimal mempunyai cara yang sama dengan konversi bilangan biner ke desimal, hanya saja untuk bilangan ini digunakan dasar delapan.

Contoh :

$$355 \text{ bilangan oktal ke desimal adalah } (3 * 8^2) + (5 * 8^1) + (5 * 8^0) = 237 \text{ desimal}$$

d. Sistem Bilangan Heksadesimal

Dalam mempelajari bahasa *Assembler*, bilangan heksadesimal mutlak harus dikuasai. Seperti bilangan biner yang berdasar 2 atau bilangan oktal yang berdasar 8, maka bilangan heksadesimal berdasar 16 sesuai dengan arti heksadesimal itu sendiri, yaitu heksa = 6, dan desimal = 10.

Sistem bilangan heksadesimal merupakan sistem dengan dasar 16, maka sebagai angka tambahan untuk nilai diatas 9 digunakan bantuan huruf abjad A sampai dengan F. Contoh urutan bilangan heksadesimal dapat dilihat sebagai berikut :

0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F, 10, 11, ...

Cara mengkonversi bilangan heksadesimal ke bilangan desimal sama dengan cara dari bilangan oktal dan biner.

Contoh :

3A bilangan desimalnya adalah :

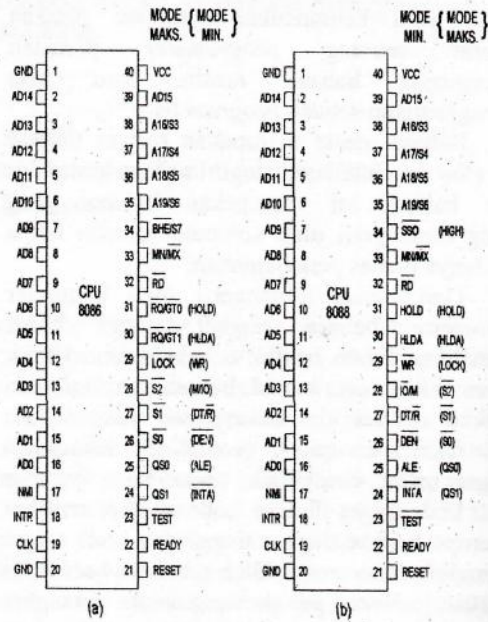
$$(2 * 16^1) + (10 * 16^0) = 58 \text{ desimal}$$

V. Jenis-Jenis Satuan Dalam Komputer

Dalam merepresentasikan nilai satuan dalam komputer, dibuat beberapa jenis satuan sebagai berikut :

1. Setiap 1 (satu) digit bilangan biner disebut satu *bit*.
2. Setiap 4 (empat) digit bilangan biner disebut satu *nibble*.
3. Setiap 8 (delapan) digit bilangan biner disebut satu *byte*.
4. Setiap 16 (enam belas) digit bilangan biner disebut satu *word*.
5. Setiap 32 (tiga puluh dua) digit bilangan biner disebut satu *double word*.
6. Setiap 128 (seratus dua puluh delapan) digit bilangan biner disebut satu *para*.
7. Setiap 256 (dua ratus lima puluh enam) byte atau 2048 (dua ribu empat puluh delapan) bit disebut satu *page*.

Perangkat Keras Mikroprosesor 8088/8086 i Pin-Out



(a) Pin-out mikroprosesor 8086

(b) pin-out mikroprosesor 8088

(b)

pin-out mikroprosesor 8086 dan 8088. mengilustrasikan sebagaimana terlihat jika dibandingkan dengan teliti, secara virtual sebenarnya tidak ada perbedaan antara kedua mikroprosesor ini, keduanya terkemas dalam *dual in-line package (DIP) 40-pin*.

8086 merupakan mikroprosesor 16-bit dengan bus data 16-bit dan 8088 merupakan mikroprosesor 16-bit dengan bus data 8-bit. Sebagaimana ditunjukkan oleh pin-out, 8086 memiliki koneksi pin AD0-AD15, dan 8088 memiliki koneksi pin AD0-AD7. Lebar bus data dengan demikian merupakan satu-satunya perbedaan utama antara kedua mikroprosesor ini.

Tetapi, ada satu perbedaan kecil pada salah satu sinyal kontrol. 8086 mempunyai pin M/IO, dan 8088 mempunyai pin M/IO. Perbedaan perangkat keras lainnya terdapat pada pin 34 dari kedua chip: pada 8088, terdapat pin SSO, sementara pada 8086, pin tersebut merupakan BHE/S7.

a. Fungsi Pin

Pin-pin yang terdapat dalam mikroprosesor 8088/8086 :

- AD7-AD0 : Jalur bus alamat/data 8088 terdiri dari bus data alamat yang dimultipleks pada 8088 dan berisi delapan bit paling kanan dari alamat memori atau nomor port I/O jika ALE aktif (logika 1) atau data ketika ALE aktif (logika 0). Pin-pin ini berada pada status impedansi tinggi selama *hold acknowledge*.
- A15-A8 : Bus alamat 8088 menyediakan bit-bit alamat memori paruh atas yang ada selama siklus bus. Hubungan alamat ini

menuju ke status impedansi tingginya selama *acknowledge hold*.

- AD15-AD8 : Jalur bus alamat/data 8086 terdiri dari bus alamat/data bagian atau pada 8086. Jalur-jalur ini berisi bit-bit alamat A15-A8 jika ALE pada logika 1, dan hubungan bus data D15-D8. Pin-pin ini memasuki status impedansi tinggi jika terjadi *hold acknowledge*.
- A19/S6-A16/S3 : Bit-bit bus alamat/status di-multipleks untuk memberikan sinyal alamat A19-A16 dan juga bit-bit status S6-S3. Pin-pin ini juga mempertahankan status impedansi tinggi selama *hold acknowledge*. Bit status S6 selalu tetap merupakan logika 0, bit S5 menunjukkan kondisi bit-bit flag IF, dan S4 dan S3 menunjukkan segmen mana yang diakses selama siklus bus pada saat itu.
- RD : Jika sinyal baca berupa logika 0, bus data bisa menerima data dari memori atau alat I/O yang dihubungkan ke sistem. Pin ini naik ke status impedansi tingginya selama *acknowledge hold*.
- READY : Input ini dikendalikan untuk menyisipkan status tunggu ke timing mikroprosesor. Jika pin READY diletakkan pada level logika 0, mikroprosesor memasuki status tunggu dan tetap idle (tidak bekerja). Jika pin READY diletakkan pada level logika 1, tidak memiliki efek pada operasi mikroprosesor.
- INTR : *Interrupt request* digunakan untuk meminta interrupt perangkat keras. Jika INTR dijaga tetap high ketika IF = 1, 8088/8086 memasuki siklus interrupt *acknowledge* (INTA menjadi aktif) setelah instruksi pada saat itu telah dijalankan sepenuhnya.
- TEST : Pin test merupakan input yang dipeson oleh instruksi WAIT. Jika TEST berupa logika 0, instruksi WAIT berfungsi sebagai NOP. Jika TEST berupa logika 1, instruksi WAIT menunggu sampai TEST menjadi logika 0. Pin ini biasanya dihubungkan dengan koprosesor numerik 8087.
- NMI : Input non-maskable interrupt sama dengan INTR kecuali interrupt NMI tidak memeriksa apakah bit flag IF merupakan logika 1. Jika NMI diaktifkan, input interrupt ini menggunakan vektor interrupt 2.
- RESET : Input reset menyebabkan mikroprosesor mereset dirinya sendiri jika pin ini tetap high selama minimum empat periode clock. Jika 8086 atau 8088 direset, akan mulai mengeksekusi instruksi pada lokasi memori FFFF0H dan mendisabel

interrupt berikutnya dengan meng-clearkan bit flag IF.

- CLK : Pin clock menyediakan sinyal waktu (timing) dasar ke mikroprosesor. Sinyal clock harus memiliki siklus kerja 33 persen untuk memberikan timing internal yang sesuai untuk 8088/8086.
- Vcc : *Input catu daya ini menyediakan sinyal +5,0 V, ± 10 % ke mikroprosesor.*
- GND : Hubungan ground merupakan jalur kembali catu daya.
- MN/MX : Pin mode minimum/maksimum memilih operasi mode minimum atau maksimum untuk mikroprosesor.
- BHE/S7 : Pin bus high enable digunakan pada 8086 untuk enable bit bus data yang paling signifikan selama operasi baca atau tulis.
- IO/M atau M/IO : Pin IO/M (8088) atau pin M/IO (8086) memilih memori atau I/O.
- WR : Jalur write merupakan strobe yang menunjukkan bahwa 8088/8086 sedang mengeluarkan data ke memori atau alat I/O.
- INTA : Sinyal *interrupt acknowledge* merupakan tanggapan terhadap pin input INTR.
- ALE : *Address latch enable* menunjukkan bahwa bus alamat/data 8088/8086 berisi informasi alamat.
- DT/R : Sinyal data transmit/receive menunjukkan bahwa bus data mikroprosesor sedang mengirim data atau menerima data.
- DEN : *Data bus enable* mengaktifkan buffer bus data eksternal.
- HOLD : *Input hold* meminta *direct memory access* (DMA).
- HLDA : *Hold acknowledge* menunjukkan bahwa 8088/8086 memasuki status *hold*.
- SS0 : Jalur status SS0 ekuivalen dengan pin S0 pada operasi mikroprosesor mode maksimum.
- S2, S1 dan S0 : Bit-bit status ini menunjukkan fungsi siklus bus saat itu.
- RO/GT1 dan RO/GT0 : Pin-pin *request/grant* ini meminta *direct memory access* (DMA) selama operasi mode maksimum.
- LOCK : *Output lock* digunakan untuk mengunci periferi dari sistem.
- QS1 dan QS0 : Bit *queue* status menunjukkan status antrian instruksi internal.

b. Persyaratan Catu Daya

Mikroprosesor 8086 dan 8088 membutuhkan +5,0 V dengan toleransi tegangan

catu ± 10 persen. 8086 menggunakan arus catu maksimum sebesar 360 mA, dan 8088 menarik maksimum 340 mA. Kedua mikroprosesor ini beroperasi pada temperatur lingkungan antara 32°F dan sekitar 180°F. Jangkauan ini tidak cukup besar untuk penggunaan di alam terbuka pada musim dingin atau bahkan pada musim panas, tetapi tersedia versi mikroprosesor 8086 dan 8088 dengan jangkauan temperatur yang diperluas. 80C88 dan 80C86 adalah versi CMOS yang hanya membutuhkan arus catu daya 10 mA dan berfungsi pada temperatur yang berkisar dari -40°F sampai +225°F.

c. Karakteristik Input dan Output

Karakteristik input mikroprosesor-mikroprosesor ini kompatibel dengan semua komponen logika standar yang tersedia saat ini. Tabel 2.1 berikut ini memberikan level tegangan input dan persyaratan arus input untuk semua pin input pada kedua mikroprosesor. Level arus input sangat kecil karena input merupakan koneksi gerbang MOSFET dan hanya merepresentasikan arus bocor.

Tabel Karakteristik input mikroprosesor 8086 dan 8088

Level Logika	Tegangan	Arus
0	0,8 V maksimum	± 10 µA maksimum
1	2,0 V minimum	± 10 µA maksimum

Tabel tersebut mengilustrasikan karakteristik semua pin output pada mikroprosesor-mikroprosesor ini. Level tegangan logika 1 pada 8088/8086 kompatibel dengan sebagian besar keluarga logika standar, tetapi logika 0 tidak. Rangkaian standar logika memiliki tegangan maksimum logika sebesar 0,4 V, dan 8088/8086 memiliki maksimum 0,45 V. Dengan demikian, ada perbedaan 0,05 V.

Tabel Karakteristik output mikroprosesor 8086 dan 8088

Level Logika	Tegangan	Arus
0	0,45 V maksimum	2,0 µA maksimum
1	2,4 V minimum	-400 µA maksimum

d. Bahasa Assembler 8088/8086

Bahasa Assembler 8088 merupakan salah satu bagian dari bahasa komputer tingkat rendah, dimana nilai 8088 ditujukan kepada mikroprosesor dengan arsitektur dasar 8088 buat Intel Corp. Penulis memilih bahasa assembler 8086 sebagai bahasa tujuan dibanding bahasa assembler seperti 6800, Zilog, dikarenakan perangkat komputer 8086

merupakan perangkat keras yang paling banyak dipakai dewasa ini.

Pengertian Register

Dalam mempelajari bahasa *Assembler*, kita harus menghafalkan semua *register* dan mengerti kegunaannya masing – masing. Yang dimaksud dengan *register* adalah sebagian kecil memori komputer yang dipakai untuk tempat penampungan data dengan ketentuan bahwa data yang terdapat dalam *register* dapat diproses dalam berbagai operasi dengan melihat berapa besar kemampuan menampung *register* tersebut.

Register dapat dibagi dalam lima golongan, yaitu :

1. General purpose register, yang terdiri dari
 AX (AH + AL) = Accumulator Register
 BX (BH + BL) = Base Register
 CX (CH + CL) = Counter Register
 DX (DH + DL) = Data Register
2. Segment Register, yang terdiri dari
 CS = Code Segment Register
 DS = Data Segment Register
 SS = Stack Segment Register
 ES = Extra Segment Register
3. Pointer Register
 IP = Instruction Pointer Register
 SP = Stack Pointer Register
 BP = Base Pointer Register
4. Index Register
 SI = Source Index Register
 DI = Destination Index Register
5. Flag Register

General Purpose Register

Register yang termasuk dalam kelompok ini adalah *register* AX, BX, CX, DX, yang masing – masing terdiri atas 16 bit. *Register – register* ini mempunyai satu ciri khas, yaitu dapat dipisah menjadi 2 bagian dimana masing – masing bagian terdiri atas 8 bit, seperti *register* AX dapat dipecah menjadi *register* AH, dan AL. Akhir H menunjukkan *High* dan akhiran L menunjukkan *Low*.

Secara umum *register-register* dalam kelompok ini dapat digunakan untuk berbagai keperluan, walaupun demikian ada pula penggunaan khusus dari masing-masing *register* ini, yaitu :

1. *Register* AX, secara khusus digunakan pada operasi aritmatika terutama dalam operasi pembagian dan pengurangan.
2. *Register* BX, biasanya digunakan untuk menunjukkan suatu alamat *offset* dari suatu segmen.
3. *Register* CX, digunakan secara khusus pada operasi *looping* dimana *register* ini menentukan berapa banyaknya *looping* yang akan terjadi.

4. Regi
 hasil
 5. TF
 pela
 6. SF (S
 bit i
 7. ZF (C
 nol,
 8. AF
 BCD
 9. PF (A
 pari
 bilan
 gena
 10. CF
 (bor
 baw
 berm
 VI.
 Assemb
 karena
 berupa
 perkalia
 ini hany
 saja.
 a. Pe
 berbag
 mikropr
 ADD, A
 penjum
 dalam
 instruks
 dalam p
 khusus
 bilangan
 macam
 1. AD
 me
 For
 AD
 BL
 2. AD
 me
 For
 AD
 BL
 3. INC
 mer
 Des
 mer
 dipakai,
 menjadi
 1. Pen
 Warta Te

4. Register DX, digunakan untuk menampung sisa hasil pembagian 16 bit.
5. TF (Trap Flag), digunakan untuk proses pelacakan langkah demi langkah.
6. SF (Sign Flag), jika digunakan bilangan bertanda bit ini akan bernilai 1.
7. ZF (Zero Flag), jika hasil operasi menghasilkan nol, maka bit ini akan bernilai 1.
8. AF (Auxiliary Flag), digunakan oleh operasi BCD, seperti pada perintah AAA.
9. PF (Parity Flag), digunakan untuk menunjukkan paritas bilangan, bit ini akan bernilai 1 bila bilangan yang dihasilkan merupakan bilangan genap.
10. CF (Carry Flag), jika terjadi peminjaman (*borrow*) pada operasi pengurangan atau bawaan (*carry*) pada penjumlahan, maka bit ini bernilai 1.

VI. Instruksi Aritmatika

Instruksi aritmatika dalam bahasa Assembly terdapat banyak variasinya, namun karena instruksi aritmatika yang diambil hanya berupa instruksi penjumlahan, pengurangan, perkalian dan pembagian, maka pembahasan kali ini hanya dibatasi pada beberapa instruksi tersebut saja.

a. Penjumlahan

Perintah penjumlahan tampil dalam berbagai format penulisan dalam beberapa jenis mikroprosesor. Instruksi penjumlahan dapat berupa ADD, ADC dan INC. ADD merupakan instruksi penjumlahan dimana *carry* tidak dimasukkan dalam perhitungan. Sedangkan ADC merupakan instruksi penjumlahan yang memasukkan *carry* dalam perhitungan. Inkremen (INC) adalah format khusus dari penjumlahan yaitu menambahkan suatu bilangan dengan satu.

Instruksi penjumlahan terdiri dari 3 macam yaitu :

1. ADD, yaitu instruksi penjumlahan yang tidak menyertakan nilai *carry* dalam perhitungan. Format : SUB *Source, Destination*. Contoh : ADD AL, BL akan menghasilkan $AL = AL + BL$.
2. ADC, yaitu instruksi penjumlahan yang menyertakan nilai *carry* dalam perhitungan. Format : SBB *Source, Destination*. Contoh : ADC AL, BL akan menghasilkan $AL = AL + BL + Carry$.
3. INC, yaitu instruksi penjumlahan yang menambahkan dengan 1. Format : INC *Destination*. Contoh : INC AL akan menghasilkan $AL = AL + 1$.

Berdasarkan mode pengalamatan yang dipakai, maka instruksi penjumlahan dapat dibagi menjadi :

1. Penjumlahan register (*register addition*).

Penjumlahan ini digunakan untuk menjumlahkan isi dari beberapa register.

Contoh :

ADD AL, BL akan menghasilkan $AL = AL + BL$.

ADC AL, BL akan menghasilkan $AL = AL + BL + Carry$.

2. Penjumlahan segera (*immediate addition*).

Penjumlahan ini digunakan untuk menjumlahkan isi register dengan nilai konstan.

Contoh :

ADD AL, 05H akan menghasilkan $AL = AL + 05H$

ADC AL, 05H akan menghasilkan $AL = AL + 05H + Carry$

b. Pengurangan

Ada banyak bentuk pengurangan yang ditampilkan dalam kumpulan instruksi. Format-format ini menggunakan mode pengalamatan data 8 bit dan 16 bit. Format khusus dari pengurangan DEC yaitu dekremen, digunakan untuk mengurangkan dengan 1. Seperti halnya instruksi penjumlahan, pengurangan juga memiliki nilai *borrow*, sehingga instruksi pengurangan terdiri dari 3 macam yaitu :

1. SUB, yaitu instruksi pengurangan yang tidak menyertakan nilai *borrow* dalam perhitungan. Format : SUB *Source, Destination*. Contoh : SUB AL, BL akan menghasilkan $AL = AL - BL$.
2. SBB, yaitu instruksi pengurangan yang menyertakan nilai *borrow* dalam perhitungan. Format : SBB *Source, Destination*. Contoh : SBB AL, BL akan menghasilkan $AL = AL - BL - Borrow$.
3. DEC, yaitu instruksi pengurangan yang mengurangkan dengan 1. Format : DEC *Destination*. Contoh : DEC AL akan menghasilkan $AL = AL - 1$.

Berdasarkan mode pengalamatan yang dipakai, maka instruksi penjumlahan dapat dibagi menjadi :

1. Pengurangan register (*register subtraction*).
Pengurangan ini digunakan untuk mengurangkan isi dari beberapa register.
Contoh :
SUB AL, BL akan menghasilkan $AL = AL - BL$.
SBB AL, BL akan menghasilkan $AL = AL - BL - Borrow$.
2. Pengurangan segera (*immediate subtraction*).
Pengurangan ini digunakan untuk mengurangkan isi register dengan nilai konstan.
Contoh :
SUB AL, 05H akan menghasilkan $AL = AL - 05H$

SBB AL, 05H akan menghasilkan AL = AL - 05H - Borrow

c. Perkalian

Perkalian dilakukan dalam format *byte* dan *word* dan dapat juga bilangan bulat bertanda (IMUL) atau bilangan bulat tak bertanda (MUL). Beberapa bit *flag* seperti O dan C berubah saat proses perkalian dijalankan dan memberikan hasil yang dapat diramalkan sebelumnya. *Flag* yang lain juga berubah, tetapi hasilnya tidak bisa diramalkan.

Namun pembahasan kali ini hanya dibatasi pada bilangan bulat tak bertanda yaitu operasi MUL yang memiliki format sebagai berikut : MUL *Source*. Contoh : MUL 05H akan menghasilkan AX = AX * 05H.

d. Pembagian

Seperti halnya perkalian, pembagian terjadi pada bilangan 8 bit atau 16 bit. Bilangan ini adalah bilangan bulat bertanda (IDIV) atau tak bertanda (DIV). Hasil pembagian selalu dua kali lebarnya dari bilangan yang dibagi. Artinya pembagian 8 bit membagi bilangan 16 bit dengan bilangan 8 bit.

Namun pembahasan kali ini hanya dibatasi pada bilangan bulat tak bertanda yaitu operasi DIV yang memiliki format sebagai berikut : DIV *Source*. Contoh : DIV 05H akan menghasilkan AX = AX / 05H.

VII. Algoritma dan Implementasinya

Algoritma perancangan perangkat lunak pembelajaran penerapan modul pengalaman dan operasi aritmatika pada mikroprosesor intel 8088/8086 dibagi menjadi 4 bagian yaitu,

1. Algoritma Eksekusi *Op-Code* MOV.
2. Algoritma Eksekusi *Op-Code* ADD dan ADC.
3. Algoritma Eksekusi *Op-Code* INC.
4. Algoritma Eksekusi *Op-Code* SUB dan SBB.
5. Algoritma Eksekusi *Op-Code* DEC.
6. Algoritma Eksekusi *Op-Code* MUL.
7. Algoritma Eksekusi *Op-Code* DIV.

Algoritma eksekusi *op-code* MOV adalah sebagai berikut,

Jika Karakter terakhir dari .Reg2 = "X" Or _

Karakter terakhir dari .Reg2 = "H" Or _
 Karakter terakhir dari .Reg2 = "L" Maka
 {Input merupakan pengalaman register}
 cHasil = GetIsiReg(.Reg2)

Else
 {Input merupakan pengalaman segera}
 Jika Karakter terakhir dari .Reg1 = "X" Maka

cHasil = FormatS(Hex2Biner(.Reg2), "0", 16)

Else

cHasil = FormatS(Hex2Biner(.Reg2), "0", 8)

End Jika

End Jika

{Tampilkan perubahan isi register}

cBaris = GetBaris(.Reg1)

{Atur warna register tujuan}

TRegister.Row = cBaris.Baris

TRegister.Col = cBaris.Kolom

TRegister.CellBackColor = &H66CCFF

Jika Karakter terakhir dari .Reg1 = "X" Maka

TRegister.Col = cBaris.Kolom + 1

TRegister.CellBackColor = &H66CCFF

End Jika

Delay nDelay

{Isi nilai register pada tabel}

Jika Karakter terakhir dari .Reg1 = "X" Maka

'AX

TRegister.TextMatrix(cBaris.Baris, cBaris.Kolom) =

BagiString(Mid(cHasil1, 1, 8), 4)

TRegister.TextMatrix(cBaris.Baris, cBaris.Kolom + 1) =

BagiString(Mid(cHasil1, 9, 8), 4)

Jika tidak, jika Karakter terakhir dari .Reg1 = "H" Or Karakter terakhir dari .Reg1 = "L" Maka

'AH / AL

TRegister.TextMatrix(cBaris.Baris, cBaris.Kolom) =

BagiString(cHasil1, 4)

End Jika

Delay nDelay

{Hapus warna}

TRegister.Row = cBaris.Baris

TRegister.Col = cBaris.Kolom

TRegister.CellBackColor = &HFFFFFF

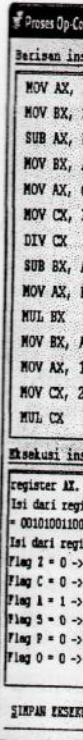
Jika Karakter terakhir dari (.Reg1, 1) = "X" Maka

TRegister.Col = cBaris.Kolom + 1

TRegister.CellBackColor = &HFFFFFF

End Jika

Input instruksi pada perangkat lunak yang dibuat diatas, menggunakan *output* program sebagai berikut :



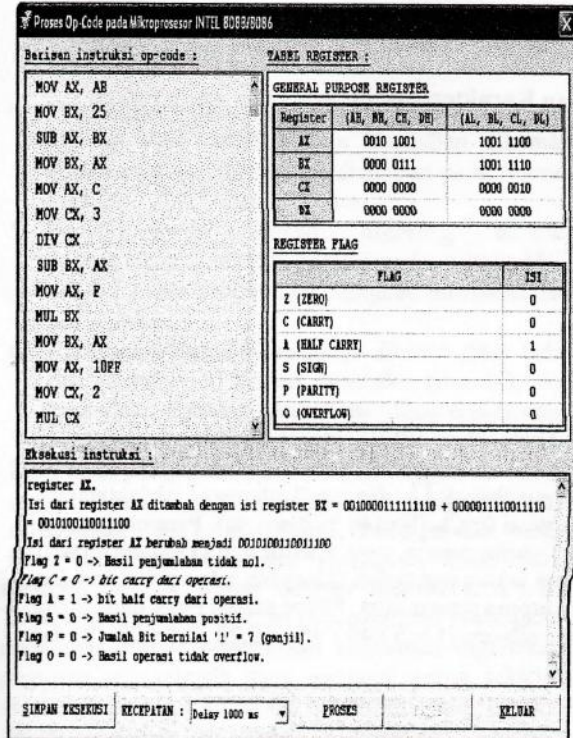
VIII
 Kesimp
 Ber
 pembaha
 perangkat
 pengalam
 mikropro
 kesimpul

1. Pera
 mem
 sege
 mikr
2. Pera
 pend
 mikr

Saran
 Setelah m
 aritmatika
 yaitu :
 disampaikan
 pengemba
 pembelaja

1. Mod
 dipen
 peng
 plus-
 bersl
2. Oper
 dapat
 opera
 perba

Warta Tekn



Tampilan Output Untuk Kasus – 1

VIII. Kesimpulan dan Saran

Kesimpulan

Berdasarkan hasil analisis data dan pembahasan setelah menyelesaikan perancangan perangkat lunak pembelajaran penerapan modus pengalamatan dan operasi aritmatika pada mikroprosesor intel 8088/8086 ini, penulis menarik kesimpulan sebagai berikut:

1. Perangkat lunak ini dapat digunakan untuk memahami modus pengalamatan *register* dan segera serta beberapa operasi aritmatika pada mikroprosesor intel 8088/8086.
2. Perangkat lunak dapat dijadikan sebagai media pendukung dalam proses pembelajaran mikroprosesor dasar.

Saran

Setelah melakukan penelitian pengalamatan operasi aritmatika pada mikroprosesor intel 8088/8086 yaitu : ini ada beberapa saran yang perlu disampaikan yang mungkin dapat membantu dalam pengembangan perancangan perangkat lunak pembelajaran penerapan modus yaitu :

1. Modus pengalamatan yang dibahas dapat diperbanyak seperti menambahkan modus pengalamatan langsung, tidak langsung, *base-plus-index*, *register* relatif dan indeks berskala.
2. Operasi yang dibahas dalam perangkat lunak dapat diperbanyak seperti menambahkan operasi aritmatika lainnya, operasi perbandingan, dan sebagainya.

Daftar Pustaka

- Albert Paul Malvino, *Elektronika Komputer Digital : Pengantar Mikrokomputer*, Edisi Kedua, Penerbit Erlangga, 1994.
- Ario Suryokusumo, *Microsoft Visual Basic 6.0*, PT. Elex Media Komputindo, 2001.
- Barry B. Brey, *Mikroprosesor Intel 8086/8088/80186/80188/80286/80386/80486 : Arsitektur Pemrograman Antarmuka*, Edisi 5, Jilid 2, Penerbit Erlangga, 2003.
- Hadi, Rahadian, *Pemrograman Microsoft Visual Basic dengan menggunakan Windows API*, PT. Elex Media Komputindo, Jakarta, 2001.
- Muchlas, *Rangkaian Digital*, Penerbit Gava Media, Yogyakarta, Januari 2005

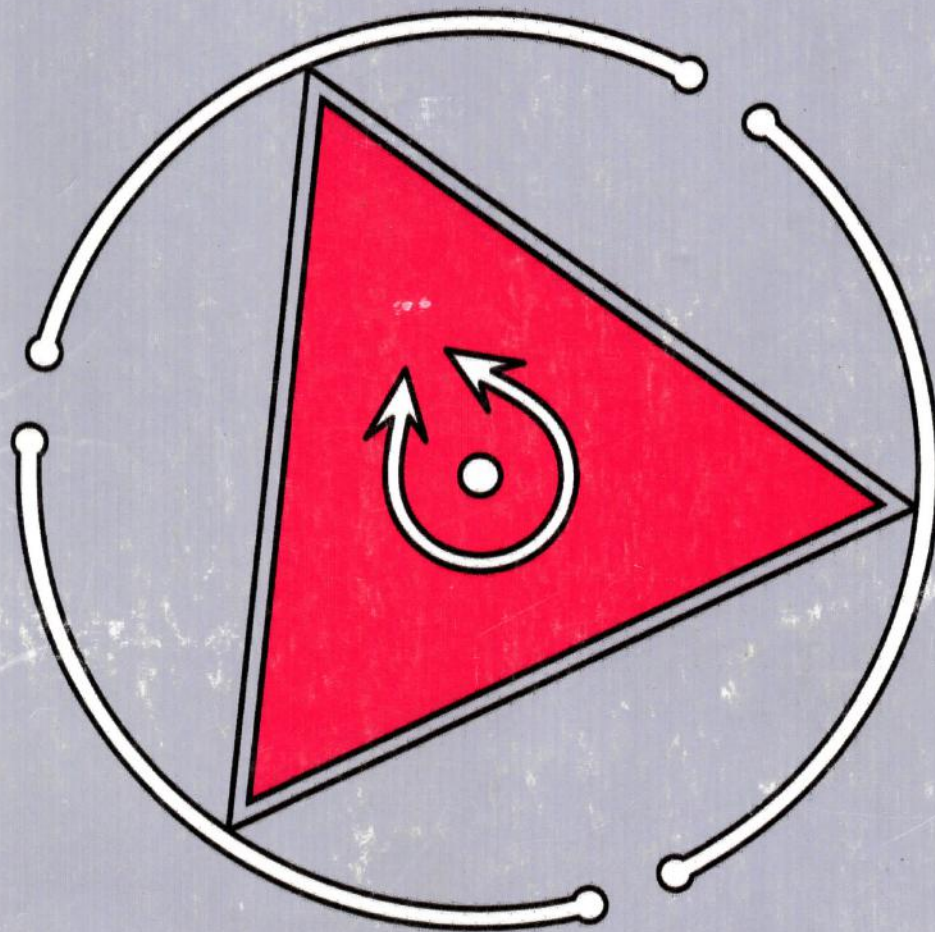
warta

ISSN 1410-2749

TEKNOLOGI INDUSTRI

Sarana Komunikasi Publikasi dan Informasi

Vol. XXVIII No. 4 / Desember / 2015



ALAMAT REDAKSI
KEMENTERIAN PERINDUSTRIAN RI
PENDIDIKAN TEKNOLOGI KIMIA INDUSTRI
JLN. MEDAN TENGGARA VII MEDAN 20228
TELP. 061-7867810 FAX. 061-7862439
<http://www.ptki.ac.id>

JUDUL	HALAMAN
Teknologi Pengolahan Limbah Industri Secara Biologis <i>(Rosmiati – PTKI Medan)</i>	63 - 65
Pengolahan Limbah Cair Industri Karet Dengan Kombinasi Sistem <i>Ponding</i> Dan Lumpur Aktif Dalam Menurunkan Kadar BOD dan COD <i>(Rumata Uli Tobing – PTKI Medan)</i>	66 - 68
Pemeliharaan Comminutor Sebagai Alat Pengolah Air Limbah <i>(M. J. Turnip – PTKI Medan)</i>	69 - 71
Perancangan Sistem Kendali Kecepatan Motor DC <i>(Nelson Silitonga – PTKI Medan)</i>	72 - 75
Proses Lumpur Aktif Untuk Pengolahan Limbah Cair Yang Mengandung Logam Merkuri <i>(Yunianto – PTKI Medan)</i>	76 - 78
Pengaruh Lama Pemasakan Terhadap Konsentrasi Total Alkali Aktif (Taa) Pada Cairan Pemasak (White Liquor) Dan Bilangan Kappa Pada Pengolahan Pulp <i>(Donda – PTKI Medan)</i>	79 - 81
Pengaruh Orientasi Pasar Pada Kinerja Bisnis <i>(Ido Arta – PTKI Medan)</i>	82 - 85
Sistem Pembelajaran Menggunakan Perangkat Lunak Dalam Operasi Aritmatika Menggunakan Mikroprosesor Intel 8088/8086 <i>(Muhammad Adam¹, Karti² – ¹Jurusan Teknik Elektro, Fakultas Teknik Universitas Muhammadiyah Sumatera Utara, ²Fungsional Dosen di PTKI Medan)</i>	86 - 93
Pengaruh Cara Blanching Dan Konsentrasi CaCl ₂ Terhadap Karakteristik <i>French Fries</i> Buah Sukun (<i>Artocarpus altilis</i>) <i>(Oksya Hikmawan – PTKI Medan)</i>	94 - 98
Identifikasi Pengaruh Komposisi Serat Pada Sound Damping Material Dari Serat Batang Pinang Raja <i>(Warman – PTKI Medan)</i>	99 - 103
Analisis Kejadian Keracunan Pada Petani Penyemprotan Pestisida Di Desa Aji Julu Kecamatan Tiga Panah Kabupaten Karo <i>(Jenny – PTKI Medan)</i>	104 - 106
Pemanfaatan Serat Batang Pinang Raja Sebagai Bahan Pengisi Pembuatan Papan Kotak Speaker <i>(Warman – PTKI Medan)</i>	107 - 111

Sistem Pembelajaran Menggunakan Perangkat Lunak Dalam Operasi Aritmatika Menggunakan Mikroprosesor Intel 8088/8086

Oleh

Muhammad Adam¹, Karti²

¹Jurusan Teknik Elektro, Fakultas Teknik Universitas Muhammadiyah Sumatera Utara,

²Fungsional Dosen di PTKI Medan

Abstract

Intel 8088/8086 processor is a microprocessor that became the engine for microcomputer-microcomputers IBM PC / XT and compatibles. This processor is often referred to briefly as "8088" or "8086" only. Although 8088 is slightly different hardware with 8086, but both can be considered equal in terms of programming. Intel 8088/8086 is a 16-bit processor, meaning that the data path and its registers are 16 bits. Most instructions in Assembly language programs require operands, the data to be processed. How to determine or obtain price is referred to as the operand addressing mode (addressing mode). Addressing mode is divided into several kinds, among others, addressing registers, addressing immediately, addressing direct and indirect addressing. Moreover, in studying assembly language, we have to memorize all the registers and understand the usefulness each - each. Register is a small portion of computer memory that is used to shelter the data provided that the data contained in the register can be processed in various operations by looking at how much the ability to accommodate these registers. Registers can be divided into five groups, namely general purpose registers (AX, BX, CX and DX), segment registers (CS, DS, SS and ES), a pointer register (IP, SP and BP), the index registers (SI and DI) and flag registers.

Keywords: Intel Microprocessors 8088/8086 "

I. Pendahuluan

Percobaan penggunaan komputer untuk proses belajar mengajar dimulai di Amerika Serikat pada akhir tahun 1950-an dan awal tahun 1960-an. Kemudian penelitian selanjutnya dilakukan oleh *Harvard University* bekerja sama dengan *IBM* pada tahun 1965. Setelah munculnya komputer mikro, sistem pengajaran dengan komputer menjadi semakin meluas pada pengembangan perangkat lunak pembelajaran yang dikenal dengan istilah perangkat lunak pembelajaran. Perangkat lunak pembelajaran dengan komputer muncul dari sejumlah disiplin ilmu, terutama ilmu komputer dan psikologi. Dari ilmu komputer dan matematika muncul program-program yang membuat semua perhitungan dan fungsi lebih mudah dan bermanfaat. Sedangkan dari ilmu psikologi muncul pengetahuan mengenai teori belajar, teknik belajar, serta motivasi yang baik.

Beberapa istilah yang digunakan untuk menyatakan perangkat lunak pembelajaran dengan komputer adalah *Computer Assisted Instruction (CAI)*, *Computer Based Instruction (CBI)*, *Computer Based Education (CBE)*, *Computer Assisted Learning (CAL)*, atau *Computer Based Training (CBT)*.

II. Metode Penelitian

Metode penelitian yang digunakan penulis dalam penelitian ini adalah sebagai berikut:

1. Perangkat lunak hasil penelitian dan rancangan ini dapat digunakan untuk membantu proses pembelajaran.

Mikroprosesor dan sekaligus menjadi perangkat pendukung dalam proses perkuliahan.

2. Mempelajari cara kerja dari mikroprosesor *Intel 8088/8086*.
3. Merancang proses kerja dari modus operasi aritmatika pada mikroprosesor *Intel 8088/8086* dan mengamati perubahan isi register yang terjadi.
4. Merancang perangkat lunak pembelajaran penerapan modus pengalamatan dan operasi aritmatika pada mikroprosesor *Intel 8088/8086*.
5. Melakukan pengujian dan pengetesan terhadap perangkat lunak hasil penelitian ini.

III. Bahasa-Bahasa Komputer

Pemakai sistem komputer modern sekarang ini bisa dibagi dalam dua kategori, yaitu :

1. Pemakai yang tidak pernah mengembangkan programnya dan hanya menggunakan program yang dikembangkan oleh pemakai lainnya.
2. Pemakai yang sangat prihatin akan pengembangan program dan berusaha mengembangkannya sendiri. Kategori pemakai ini biasa disebut ahli komputer (*programmer*). Kategori pemakai ini sangat beruntung karena sekarang ini telah banyak beredar bahasa pemrograman tingkat tinggi (*high level*) yang merupakan bahasa pemrograman dengan penggunaan bahasa yang mendekati bahasa manusia. Dibandingkan dengan *programmer*

pada awal kemunculan komputer dimana syarat seorang *programmer* haruslah menguasai bahasa mesin baru bisa menghasilkan sebuah *program* baru.

Bahasa mesin merupakan bahasa tingkat rendah (*low level*) dalam kategori bahasa komputer dimana bahasa ini merupakan bahasa yang langsung dimengerti oleh komputer sendiri tanpa perlu adanya proses penerjemahan.

Untuk bisa dimengerti oleh komputer maka suatu bahasa tingkat tinggi harus diterjemahkan dalam bentuk bahasa mesin. Untuk menerjemahkan suatu bentuk bahasa pemrograman diperlukan sebuah alat penerjemah yang umum disebut teknik kompilasi (*compiler*). Kompilasi berfungsi untuk mengubah suatu kode sumber menjadi kode target dimana kode sumber tersebut bisa berupa bahasa tingkat tinggi dan kode target bisa berupa bahasa mesin. Oleh sebab itu kompilasi merupakan jembatan penghubung antara perangkat keras dengan perangkat lunak.

a. Bahasa Tingkat Tinggi

Ciri – ciri bahasa tingkat tinggi adalah sebagai berikut :

1. Perintahnya mirip dengan bahasa manusia, seperti penggunaan bahasa Inggris.
2. Mudah dimengerti.
3. Mempunyai kemampuan yang terbatas (bila tidak dibantu dengan bahasa tingkat rendah).

Beberapa contoh bahasa tingkat tinggi adalah :

- BASIC (*Beginner's All – purpose Symbolic Interchange Code*).
- COBOL (*Common Bussiness Oriented Language*).
- PASCAL (Untuk mengenang jasa Blaise Pascal).

b. Bahasa Tingkat Menengah

Ciri khas dari bahasa tingkat menengah adalah kecepatan mengakses dan kemampuan yang cukup diandalkan. Keistimewaan lainnya adalah perintah yang digunakan hampir sama dengan bahasa manusia. Contoh bahasa tingkat menengah adalah bahasa C.

c. Bahasa Tingkat Rendah

Bahasa tingkat rendah cukup sulit dipelajari karena perintahnya tidak sama dengan bahasa manusia. Keistimewaan bahasa tingkat rendah adalah kecepatannya yang sangat diandalkan.

Contoh dari bahasa tingkat rendah adalah bahasa mesin dan bahasa *Assembler* (rakitan). Pada bahasa *Assembler* akan dijumpai beberapa program yang cukup panjang dan berbelit – belit. Hal ini tidak aneh karena pada bahasa *Assembler* kita harus memerintahkan satu per satu langkah dasar yang diambil. Sebagai contoh, akan diambil sebuah rutin menghapus layar monitor. Pada bahasa tingkat

tinggi seperti bahasa BASIC cukup diberikan perintah CLS maka otomatis semua tulisan dilayar akan terhapus dan bersih kembali. Lain halnya dengan bahasa *Assembler* dimana untuk melaksanakan rutin menghapus layar saja memerlukan 10 perintah dasar seperti dapat dilihat pada gambar table berikut ini :

Mov ah,06h
Xor cx,cx
Mov dh,24
Mov dl,79
Mov al,00
Mov bh,07h
Int 10h
Mov ah,2
Mov bh,00
Int 10h

barisan perintah dalam bahasa *Assembler*

IV. Sistem Bilangan

Untuk mempelajari bahasa *Assembler* hal pertama yang harus kita pelajari adalah mengenal beberapa sistem bilangan, karena hal itu sangat berguna dalam pengaksesan ke port ataupun menghafalkan kode ASCII (*American Standard Code for Information Interchange*) yang penting dan lain – lain.

Yang dimaksud dengan sistem bilangan adalah kumpulan angka yang membentuk suatu sistem peredaran tertentu. Sebagai contoh, sistem bilangan desimal yang biasa digunakan sehari – hari untuk segala keperluan, seperti angka untuk nomor, angka untuk menyatakan jumlah uang, banyaknya barang, dan lain sebagainya.

a. Sistem Bilangan Desimal

Bilangan yang digunakan dalam kehidupan sehari – hari adalah sistem bilangan desimal. Sistem bilangan desimal terdiri dari angka 0 hingga 9 dengan menggunakan basis/dasar 10.

Contoh :

Angka 321 dengan dasar 10, maka :
 $(3 * 10^2) + (2 * 10^1) + (1 * 10^0) = 321$

b. Sistem Bilangan Biner

Komputer memproses data maupun program dari memori komputer berupa sejumlah bilangan biner yang menyatakan keadaan hidup atau mati dengan angka 1 dan 0, sehingga dapat disimpulkan bahwa sebenarnya semua yang diproses komputer hanya angka 0 dan 1.

Karena bilangan biner hanya terdiri dari 1 dan 0, maka dapat disimpulkan bahwa bilangan biner itu berdasar 2. Adapun cara mengkonversi ke bilangan desimal adalah dengan cara mengalikan dua dengan pangkat N (suku ke – N) seperti yang dilakukan pada bilangan desimal, yaitu mengalikan 10 engan pangkat N (suku ke – N).

Contoh :

1110 bilangan desimalnya adalah

$$(1 * 2^3) + (1 * 2^2) + (1 * 2^1) + (0 * 2^0) = 14$$

c. Sistem Bilangan Oktal

Bilangan oktal merupakan bilangan berdasar 8, jadi bilangan ini hanya terdiri dari angka 0 sampai 7. Konversi bilangan oktal ke desimal mempunyai cara yang sama dengan konversi bilangan biner ke desimal, hanya saja untuk bilangan ini digunakan dasar delapan.

Contoh :

355 bilangan oktal ke desimal adalah $(3 * 8^2) + (5 * 8^1) + (5 * 8^0) = 237$ desimal

d. Sistem Bilangan Heksadesimal

Dalam mempelajari bahasa *Assembler*, bilangan heksadesimal mutlak harus dikuasai. Seperti bilangan biner yang berdasar 2 atau bilangan oktal yang berdasar 8, maka bilangan heksadesimal berdasar 16 sesuai dengan arti heksadesimal itu sendiri, yaitu heksa = 6, dan desimal = 10.

Sistem bilangan heksadesimal merupakan sistem dengan dasar 16, maka sebagai angka tambahan untuk nilai diatas 9 digunakan bantuan huruf abjad A sampai dengan F. Contoh urutan bilangan heksadesimal dapat dilihat sebagai berikut :

0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F, 10, 11, ...

Cara mengkonversi bilangan heksadesimal ke bilangan desimal sama dengan cara dari bilangan oktal dan biner.

Contoh :

3A bilangan desimalnya adalah :

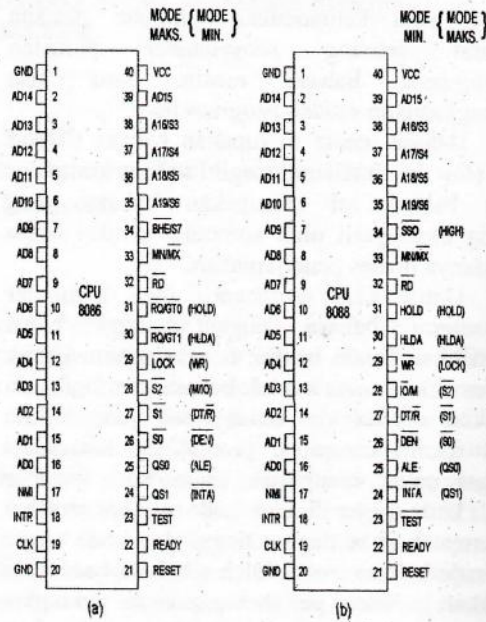
$$(2 * 16^1) + (10 * 16^0) = 58 \text{ desimal}$$

V. Jenis-Jenis Satuan Dalam Komputer

Dalam merepresentasikan nilai satuan dalam komputer, dibuat beberapa jenis satuan sebagai berikut :

1. Setiap 1 (satu) digit bilangan biner disebut satu *bit*.
2. Setiap 4 (empat) digit bilangan biner disebut satu *nibble*.
3. Setiap 8 (delapan) digit bilangan biner disebut satu *byte*.
4. Setiap 16 (enam belas) digit bilangan biner disebut satu *word*.
5. Setiap 32 (tiga puluh dua) digit bilangan biner disebut satu *double word*.
6. Setiap 128 (seratus dua puluh delapan) digit bilangan biner disebut satu *para*.
7. Setiap 256 (dua ratus lima puluh enam) byte atau 2048 (dua ribu empat puluh delapan) bit disebut satu *page*.

Perangkat Keras Mikroprosesor 8088/8086 i Pin-Out



(a) Pin-out mikroprosesor 8086

(b) pin-out mikroprosesor 8088

(b)

pin-out mikroprosesor 8086 dan 8088. mengilustrasikan sebagaimana terlihat jika dibandingkan dengan teliti, secara virtual sebenarnya tidak ada perbedaan antara kedua mikroprosesor ini, keduanya terkemas dalam *dual in-line package* (DIP) 40-pin.

8086 merupakan mikroprosesor 16-bit dengan bus data 16-bit dan 8088 merupakan mikroprosesor 16-bit dengan bus data 8-bit. Sebagaimana ditunjukkan oleh pin-out, 8086 memiliki koneksi pin AD0-AD15, dan 8088 memiliki koneksi pin AD0-AD7. Lebar bus data dengan demikian merupakan satu-satunya perbedaan utama antara kedua mikroprosesor ini.

Tetapi, ada satu perbedaan kecil pada salah satu sinyal kontrol. 8086 mempunyai pin M/IO, dan 8088 mempunyai pin M/IO. Perbedaan perangkat keras lainnya terdapat pada pin 34 dari kedua chip: pada 8088, terdapat pin SSO, sementara pada 8086, pin tersebut merupakan BHE/S7.

a. Fungsi Pin

Pin-pin yang terdapat dalam mikroprosesor 8088/8086 :

- AD7-AD0 : Jalur bus alamat/data 8088 terdiri dari bus data alamat yang dimultipleks pada 8088 dan berisi delapan bit paling kanan dari alamat memori atau nomor port I/O jika ALE aktif (logika 1) atau data ketika ALE aktif (logika 0). Pin-pin ini berada pada status impedansi tinggi selama *hold acknowledge*.
- A15-A8 : Bus alamat 8088 menyediakan bit-bit alamat memori paruh atas yang ada selama siklus bus. Hubungan alamat ini