Pengalamatan Mikroprosesor 8086/8088 Menggunakan Operasi Aritmatika

Iqbal Mubaroq⁽¹⁾, Walidatul Isna⁽²⁾, Ike Wahyu Septiani⁽³⁾, Abd. Charis Fauzan⁽⁴⁾

^{1,2,3,4}Program Studi Ilmu Komputer Universitas Nahdlatul Ulama Blitar, Indonesia

Email: ¹iqbalmubaroq30@gmail.com, ²Walidatulisna15@gmail.com, ³ikewahyu0509@gmail.com, ⁴abdcharis@unublitar.ac.id

Tersedia Online di

http://www.jurnal.unublitar.ac.id/index.php/briliant

Sejarah Artikel

Diterima pada 17 Juli 2019 Disetujui pada 30 Agustus 2019 Dipublikasikan pada 31 Agustus 2019 Hal. 298-305

Kata Kunci:

Pengalamatan, Mikroprosesor, Operasi Aritmatika

DOI

http://dx.doi.org/10.28926/briliant .v3i4.342

Abstrak: Intel Corporation mulai memperkenalkan Mikroprosesor 8088 sekitar tahun Mikroprosesor 8088/8086 memiliki 4 register, yaitu Counter Register CX, Accumulator Register AX, Base Register BX, dan Data Register DX menggunakan perintah MOV (perpindahan), ADD (penjumlahan), SUB (pengurangan), dan MUL (perkalian). Berdasarkan data tersebut, digunakan metode manual dan ekskusi program untuk menganalisis set instruksi mikroprosesor 8086/8088. Kedua metode diatas menghasilkan pengalamatan yang berbeda. Penelitian bertujuan untuk memahami cara operasi aritmatika menggunakan mikroprosesor 8086/8088. Penelitian dilakukan dengan menjelaskan permasalahan dengan menggunakan metode manual dalam pembuatan Op-Code serta eksekusi instruksi. Berdasarkan penelitian yang dilakukan, dihasilkan 2 metode yaitu bisa lewat perangkat lunak maupun secara manual agar lebih mudah dalam pengaplikasiannya.

PENDAHULUAN

Pengalamatan mikroprosesor adalah bagian yang tidak terpisahkan dari ranah penelitian arsitektur dan organisasi komputer (Fahri & Riadi, 2013). Pengalamatan ini juga membahas tentang komponen penyusun sistem komputer antar interkoneksi serta tentang unit-unit operasional sedangkan di arsitektur komputer akan membahas tentang atribut sistem komputer yang berterkaitan dengan seorang programmer. (S & Wijaya, 2012). Penelitian ini diangkat untuk meneliti peran operasi aritmatika menggunakan microprosessor 8086/8088. Peran sangat tidak lepas dari kata prosesor yang biasa disebut dengan otaknya computer. Apabila suatu komponen pC biasanya memiliki fungsi sebagai pentransmisi data,dan professornya memiliki fungsi untuk menghitung dan menentukan segala aktifitas (S & Wijaya, 2012). Microprosesor telah menjadi suatu mesin bagi microcomputer kompatibelnya serta IBM PC/XT merupkan pengertian dari Prosesor Intel 8086/8088. Prosesor Intel 8088 memiliki perbedaan dari 8086, tapi bisa juga memiliki kesamaan jika dilihat misalnya dari pemrogramannya Intel 8086/8088 yaitu prosesor yang memiliki 16 bit, yang berarti data lintasannya serta

registernya memiliki ukuran hampir sama dengan prosesornya yaitu 16 bit (Mustawa, 2013).

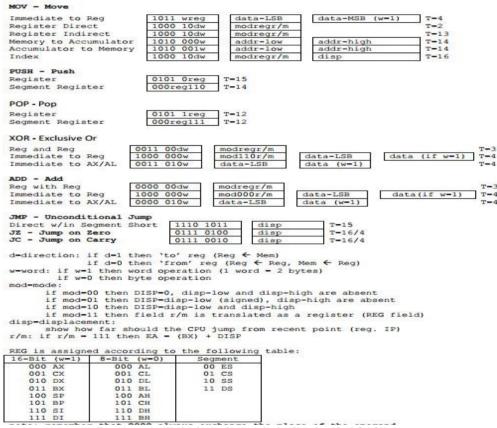
Mikroprosesor menurut teknologi bisa dikumpulkan berdasarkan jumlah dari bit data,berdasarkan kemampuan atau karakteristik mikroposesor serta struktur dan fungi dari mikroprosesor itu sendiri.Memiliki jumlah bit data yang bisa do mulai dari 1 bit,4bit,8bit,lalu 16bit kemudian 32bit selanjutnya 64 bit (Hambali, 2017). Jalur alamat mempunyai pin yang hampir sama dengan jalur data, yang memiliki arti apabila pada saat-saat tertentu akan dipakai sebagai jalur alamat dan juga bisa dilakukan pada waktu-waktu yang lain. 8088 diharapkan memiliki system yang relatif ekonomis, serta banyak diikuti oleh para pengguna desain missal 8 bit. Sircuit Boards yang memiliki jalur bus yang lumayan lebar bisa dikatakan masih mahal jika nanti diterbitkan. (Hendra Gunawan, 2010). Prosessor biasanya mempunyai alamat bus sebanyak kira-kira 20 bit, memiliki arti bahwa bisa mencapai alamat sampai 1.048.575 lokasi. Berdasarkan Heksadesimal, angka seperti itu dikatakan 00000 hingga dengan FFFFF merupakan alamat fisik dari sebuah microprosessor (Najah, Yaqin, Angreani, & Fauzan, 2019).. Jika 8086 serta 8088 memiliki alamat bus yang terdiri dari 20 bit, maka secara otomatis penulisan fisik alamatnya yang ada 5 bit dari Heksadesimal. (Panjaitan & Sagala, 2018). Berdasarkan latar belakang diatas akan diperlukan penelitian tentang pengalamatan mikroprosesor 8086/8088 menggunakan operasi aritmatika.

METODE

Metode penelitian yang digunakan adalah metode manual dan eksekusi program. Berikut metode manual dalam pembuatan Op-Code:

- a. Tentukan perintah program nya yang akan di buat kode operasi. dibuat Contoh : MOV AX,BX.
- b. Pada Gambar 1 tabel set instruksi (lembar data Intel 8086/8088) kita lihat perintah programnya. Kita ambil salah satu dari pilihan data sheet di tabel yang sejalur dengan perintah program yang akan dibuat. Contoh program diatas menunjukkan bahwa program tersebut menyalin data dari register BX ke AX.
- c. Tuliskan kode instruksinya. Contoh: **Register/Memory from/to Register** yaitu **1000 10dw mod reg r/m**.
- d. Variabel-variabel seperti d,w,mod dll kita ganti dengan angka biner yang cocok. Dibawah ini pengertian dari masing-masing variabel tersebut.
 - i. **d(direction)** merupakan pernyataan arah perpindahan data, d bernilai 1 jika data berpindah ke register, d bernilai 0 apabila data beralih dari register. contoh yang tertera, kita bisa menginputkan 0 ataupun 1(satu). Apabila kita pilih 1, berarti data tersebut berpindah ke register yakni AX.
 - ii. **w(word)** untuk menunjukkan operasi diatas termasuk 16 bit atau yang lain. Apabila tergolong 16 bit maka w=1 dan jika bukan maka w=0. Mov AX, BX tergolong word, jadi w=1 karena AX dan BX termasuk register 16 bit.
 - iii. **mod(mode)** disini merupakan lintas data. Mod: bernilai 11 karena membuktikan operasii antar register.

- iv. **reg(register)** yaitu menjelaskan register yang dipakai. sebab d bernilai 1/ d=1 maka kita cari kode binernya dari register AX, yakni 000.
- v. **r/m(register/memory)** merupakan perhitungan alamat fisik dari memori. Perintah program diatas tidak ada tentang memori, jadi kita bebas memilih. Sebagai contoh kia memilih r/m = 111, maka Kode binernya 100010 1111000111.
- e. Setelah selesai semua, kemudian dilanjutkan ke pengkonversian biner ke heksadesimal. Hasilnya 100010 1111000111 heksademilanya menjadi 8BC7. Jadi kode operasinya adalah 8BC7 (Ardi panjaitan & Sagala, 2018).



Gambar 1. Tabel set instruksi (Syahrul, 2010)

Sedangkan untuk eksekusi instruksi merujuk pada contoh dari ekspresi aritmatika berikut: AFD9h - ((40h+30h+2A6h)*5h) dituliskan dalam op-code seperti ini:

- a. MOV AX 0040
- b. MOV BX 0030
- c. MOV BX 02A6
- d. ADD AX BX
- e. ADD AX CX
- f. MOV BX 0005
- g. MUL BX
- h. MOV BX AX
- i. MOV AX AFD9

j. SUB AX BX

Ekskusi op-code dan register flag akan menjadi seperti ini:

1. MOV AX, 0040

Instruksi MOV menulis ulang 0040H ke register AX Register AX = 0040H = 000000001000000

2. MOV BX, 0030

Instruksi MOV menulis ulang 0030H ke register BX Register BX = 0030H = 000000000110000

3. MOV CX. 02A6

Instruksi MOV menulis ulang 02A6H ke register CX Register CX = 02A6H = 000001010100110

4. ADD AX, BX

Instruksi ADD menambahkan ke register AX & BX. Hasilnya ditambahkan lagi ke register AX.

AX + BX = 0000000001000000 + 000000000110000 = 0000000001110000

Register AX berubah menjadi = 0000000001110000

Flag Z = 0 Hasil penjumlahan tak O(nol).

Flag C = 0 carry/sisa bit dari operasi

Flag A = 0 bit half carry operasi

Flag S = 0 penjumlahan positif

Flag P = 0 bit bernilai 1 = 3(bilangan ganjil)

Flag O = 0 operasi ini tak Overflow

5. ADD AX, CX

Instruksi ADD menambahkan ke register AX dan CX. Hasilnya ditambahkan lagi ke register AX.

AX + CX = 0000000001000000 + 0000001010100110 = 000000110001010

0000000001000000 0000001010100110 + 0000001100010110

Register AX = 0000001100010110

Flag Z = 0 Hasil penjumlahan tak O(nol).

Flag C = 0 carry/sisa bit dari operasi

Flag A = bit half carry operasi

Flag S = 0 penjumlahan positif

Flag P = 0 bit bernilai 1 = 5(bilangan ganjil)

Flag O = 0 operasi ini tak Overflow

6. MOV BX, 0005

Instruksi MOV menulis ulang 0005H ke register BX

Register BX = 0005H = 000000000000101

7. MUL BX

Instruksi MUL mengalikan register BX dan AX. Hasilnya di muat ke register DX-AX

 $0000000000000101\\0000001100010110$

 $0000000000000101\\0000000000000101$

000000000000111101101110

Flag Z = 0 Hasil dari Penjumlahan Tidak 0.

Flag C = 1 MSB(most Significant bit) 16 bit = 1(satu)

Flag S = 0 Perkalian Positif

Flag P = 0 bit bernilai 1 = 9(bilangan ganjil)

Flag O = 0 operasi ini tak Overflow

8. MOV BX, AX

Instruksi MOV mengalihkan data AX ke BX

Register BX sekarang = Register AX = 000011110110110 = 0F6E

9. MOV AX, AFD9

Instruksi MOV menulis ulang AFD9 ke register AX

Isi AX = 1010111111011001

10. SUB AX, BX

Instruksi SUB yaitu mengurangi register AX & BX. Hasilnya balikkan lagi ke AX.

AX - BX = 10101111111011001 - 0000111101101110 = 1010000001101011

 $\frac{1010111111011001}{0000111101101110} - \\ 1010000001101011$

FLAG

Z = 0 => Hasil Penjumlahan Tidak nol

 $C = 0 \Rightarrow MSB 16 Bit = 1$

S = 0 => Pengurangan Positif

P = 0 => bit bernilai 1(satu) = 7 (bilangan ganjil)

O = 0 => operasi ini tidal overflow

Hasil akhir operasinya dimasukkan ke register AX, 1010000001101011(biner) atau 41067(desimal) atau A06B(heksadesimal) (Panjaitan & Sagala, 2018).

HASIL DAN PEMBAHASAN

Hasil dari metode diatas yaitu kita dapat menemukan proses atau cara untuk eksekusi instruksi. Kita juga bisa melihat contoh ekspresi aritmatika yang kemudian di tuliskan dalam bentuk Op Code. Pada bagian ini diterangkan dahulu pengertian Op Code. Op Code diartikan menjelaskan atau mendeskripsikan jenis operasi yang akan dikerjakan. Kode Operasi tersebut berbentuk Biner. Pada suatu Bidang, Op Code merupakan suatu bagian dari instruksi bahasa mesin dan menentukan jenis operasi yang akan dilaksanakan. Selain itu, intruksi tersebut menentukan data yang akan memproses, berupa Operand (Hayati, 2017).

Contoh dari Op Code:

 $0001 = 1 \Rightarrow Load AC from Memory$

 $0010 = 2 \Rightarrow$ Save AC at Memory

 $0101 = 5 \Rightarrow$ Add to AC from Memory.

AC atau Accumulator disini diartikan sebagai penyimpanan Sementara (Stallings, n.d.). Operasi Set Instruksi yang digunakan untuk transfer data yaitu:

- a. Mov: Memindahkan. Maksudnya ialah memindahkan word dari sumbernya ke tujuan.
- b. Load: Memindahkan word dari memory ke Prosessor.
- c. Set: Memindahkan word 1 ke tujuannya.
- d. Pop: menyalin kata (word) dari bagian yang paling utama.
- e. Clear/Reset: Memindahkan word 0 ke tujuannya.

Untuk melakukan operasi aritmatik yang perlu dilakukan CPU ialah:

- a. Meng Set Kode Flag dan kondisi.
- b. Transfer data setelah nya ataupun sebelumnya.
- c. Melaksanakan fungsi ALU.

Kemudian, digunakan Set Operasi Instruksi dalam aritmatika, yaitu: ADD (Penjumlahan), SUBTRACT/ SUB (Pengurangan), MULTIPLY/ MUL (Perkalian), DIVIDE/DIV (Pembagian). Ada juga Increment, Decrement, Absolute, Negative yang merupakan instruksi operand tunggal (Akbar, 2017). Pengertian dari masing-masing Flag register:

- Overflow Flag (OF), apabila overflow di operasi aritmatika, akan bernilai 1 pada bit ini.
- Carry flag (CF), di setting/reset oleh operasi aritmatika. Kemudian statusnya diuji oleh program. Overflow mikroprosesor akan menambahkan 16-bit kemudian membuat CF menjadi *set*.
- Zero flag (ZF), me-set apabila hasil dari instruksi ialah nol(0) dan clear apabila hasilnya tidak sama dengan nol(0).
- Sign flag (SF), ialah set di MSB(most significant bit) akumulator sesuai dengan logika instruksi. Instruksi ini memakai MSB data agar bisa dinyatakan sebagai tanda bilangan pada akumulator.
- Parity flag (PF), digunakan agar memperlihatkan paritas bilangan. Jika bilangan menunjukkan genap, maka biner bernilai 1.

Penjumlahan dalam Biner

```
Pertama-tama kita cermati aturan pasangan biner
```

```
0 + 0 = 0
        0 + 1 = 1
        1 + 0 = 1
1 + 1 = 0 => menyimpan 1
```

Sebagai Contoh:

```
Penyelesaiannya:
   111110
   100110 +
   1000100
```

0 + 0 = 0 carry/sisa 0

1 + 1 = 0 sisa 1

1 + 1 + 1 = 1 sisa1

1 + 0 + 1 = 0 sisa 1

1 + 0 + 1 = 0 sisa 1

1 + 1 + 1 = 0 sisa 1

= 1(satu) sisa satu di taruh di awal bilangan biner Carry adalah sisa dari perhitungan bilangan biner.

KESIMPULAN

Berdasarkan hasil dari penelitian dan analisis yang sudah dilakukan, maka dapat disimpulkan bahwa pengalamatan mikroprosesor 8086/8088 dengan metode manual dan juga eksekusi instruksi berbeda. Pada metode manual kita harus membuat tabel set instruksi terlebih dulu dan pada metode ekskusi program dibuatlah persoalan yang kemudian di instruksikan pada perintah MOV (perpindahan), ADD (penjumlahan), SUB (pengurangan), dan MUL (perkalian). Dalam penerapan 2 metode tersebut bisa lewat perangkat lunak (software). Modus pengalamatan di berikan sebuah perangkat lunak agar lebih mudah untuk pengaplikasiannya.

SARAN

Penelitian ini menyarankan untuk membandingkan beberapa jenis mikroprosesor yang berbeda menggunakan operasi aritmatika. Komparasi ini bertujuan untuk mendapatkan hasil yang optimal. Disamping itu, operasi aritmatika juga perlu ditelaah terhadap relevansinya pada jenis mikroprosesor yang berbeda.

DAFTAR RUJUKAN

Akbar, M. (2017). JURNAL IT Perangkat Lunak Pembelajaran Modus Pengalamatan Dan Operasi Aritmatika Pada Mikroprosesor Intel 8088 / 8086. 8(1), 44–52.

Ardi panjaitan, & Sagala, M. (2018). PENERAPAN MODUS PENGALAMATAN DANOPERASI ARITMATIKA PADA MIKROPROSESOR INTEL 8088/8086. 03. 144–151.

Fahri, & Riadi, I. (2013). MEDIA PEMBELAJARAN ARITMATIKA

- KOMPUTER (Studi Kasus Materi Konversi Bilangan). *Jurnal Sarjana Teknik Informatika*, *I*(1), 191–200. Retrieved from http://www.jogjapress.com/index.php/JSTIF/article/view/1745
- Hambali. (2017). IMPLEMENTASI PENGISIAN REGISITER DATA DAN OPERASI ARITMATIKA MENGGUNAKAN PROGRAM COMMAND PROMPT, BAHASA ASSEMBLER. IV(1), 51–57.
- Hayati, N. (2017). ORGANISASI DAN ARSITEKTUR KOMPUTER.
- Hendra Gunawan. (2010). PERANCANGAN PERANGKAT LUNAK PEMBELAJARAN MODUS PENGALAMATAN DAN OPERASI ARITMATIKA PADA MIKROPROSESOR INTEL 8088/8086 DENGAN MENGGUNAKAN MS. VISUAL BASIC 6.0". 13–29.
- Mustawa, S. (2013). OPERASI ARITMATIKA PADA MIKROPROSESOR INTEL 8088 / 8086 DENGAN METODE CBT (Computer Based Training). 1(1), 17–25.
- Najah, S., Yaqin, M. A., Angreani, L. S., & Fauzan, A. C. (2019). The User Interface (UI) Discovery Application to Measure Query Accuracy on Interface Repository. 3(May), 1–6.
- Panjaitan, A., & Sagala, M. (2018). PENERAPAN MODUS PENGALAMATAN DAN OPERASI ARITMATIKA PADA MIKROPROSESOR INTEL 8088/8086. 03, 144–151.
- S, T. F. P., & Wijaya, N. (2012). PERANCANGAN PERANGKAT LUNAK UNTUK SIMULASI PEMBUATAN PETA ALAMAT MEMORI PADA SISTEM MIKROPROSESSOR. (April), 1–10.
- Stallings, W. (n.d.). William Stallings Computer Organization and Architecture 8.