

6

INSTRUKSI DEBUG. COM

Kita dapat menjalankan / membuat program dalam 2 cara, yaitu:

1. DEBUG
2. MACRO ASSEMBLER (side Kick / Edlin)

Untuk program-program yang pendek dan sederhana sebaiknya menggunakan program debug. Semua program Debug dibuat dalam sistem bilangan heksadesimal dan programnya/ filenya diakhiri dengan extension .COM

Sekarang memulai dengan Debug dengan mengetik nama Debug setelah prompt DOS (yaitu A> dalam contoh ini):

A> DEBUG, tekan enter

Hyphen (garis sambung) yang Saudara lihat dalam jawaban atas perintah Saudara adalah simbol prompt debug. Hal ini berarti menunggu perintah atau instruksi.

Adapun perintah atau instruksi debug adalah:

1. A – Assemble

Format : A [address]

address = alamat untuk memulai membuat program, nilai default-nya CS:0100.

Tujuan : untuk membuat program-program secara langsung ke memori.

Contoh : -A100
0768:100

Catatan : tombol **Enter** akan meneruskan baris berikutnya dari program, jika sudah selesai membuat program, tekan Enter pada baris yang kosong untuk kembali ke prompt DEBUG.

2. C - Compare

Tujuan : membandingkan data pada alamat tertentu.

Format : C range address

range = daerah yang akan dibandingkan.

address = awal alamat untuk memulai melakukan perbandingan, default-nya DS.

Contoh : C CS:100 L0F CS:110 melakukan perbandingan sebanyak 16 byte, bermula dari alamat CS:100 dengan 16 byte berikutnya.

3. D - Dump

Tujuan : melihat data yang ada dalam memori

Format : D [address] atau D [range]

[address] = alamat yang akan ditampilkan, jika kita tulis D tanpa alamat, maka default-nya menjadi CS:100, sedangkan jika kita tulis alamat tertentu, maka segmentnya DS.

[range] = daerah yang akan ditampilkan

Contoh : D 0 L50 tampilkan 80 byte dari memori, berawal dari DS:0

5. E - Enter

Tujuan : memasukkan atau mengganti data ke dalam memori.

Format : E address [list]

address = dimana kita akan memasukkan data (dalam byte), default-nya DS.

[list] = daftar byte atau data yang akan di masukkan.

Contoh : E DS:10 'DATA PRO' 0 masukan kata DATA PRO diikuti angka 0.

6. F - Fill

Tujuan : isi daerah tertentu dengan data yang ditentukan.

Format : F range list

range = daerah yang akan diisi data.
list = data yang akan dimasukkan
Contoh : F B800:0 LFA0 'X' 07 isi alamat color graphic adapter dengan karakter 'X' pada putih di atas hitam.

7. G – Go

Tujuan : untuk menjalankan program yang ada dalam memori.
Format : G [=address][address[address...]]
[=address]= jika didefinisikan, program yang akan dijalankan dimulai dari CS:address.
[address[address...]] = alamat penghentian jalannya program (minimal 10).

Contoh : 1. G =200 240 jalankan program mulai dari CS:200, dan hentikan pada alamat CS:240.
2. G
A
Program terminated normally

8. H – Heksadesimal

Tujuan : untuk menjumlah dan mengurangi dua bilangan heksadesimal.
Format : H value value
Contoh : H 4200 21A2, hasilnya 63A2 205E

9. I – Input

Tujuan : membaca dan menampilkan data dari port.
Format : I portaddress

10. L – Load

Tujuan : masukkan data dari sektor ke memori
Format : L address drive sektor sektor
address = alamat yang akan dipakai, default-nya CS.
Contoh : L DS:0 0 5 1 masukkan sektor direktori drive A ke memori.

L – Load

Tujuan : baca file, setelah menentukan nama file.
Format : L [address]
address = alamat yang akan dipakai, default-nya CS : 100.

11. M - Move

Tujuan : Copy data tertentu ke alamat tertentu.
Format : M range address
range = daerah yang akan di copy
address = alamat tujuan data

Contoh : M B000 :A0 copy tiga baris pertama dari RAM video monochrome ke baris berikutnya.

12. N - Name

Tujuan : Ubah atau definisikan nama file
Format : N [d:][path] nama file [ext]
[d:] [path]nama file [ext] = nama file yang akan di akses ke drive.

13. Output

Tujuan : kirim data ke port
Format : O portaddress byte

14. Proceed

Tujuan : sama seperti Trace, hanya menjalankan fungsi call, loop, interrupt, atau gabungan beberapa intruksi menjadi satu intruksi.
Format : p [=address][value]
[=address]=awal pencarian kesalahan (trace), default-nya CS:IP
[value]=banyaknya intruksi yang akan dijalankan.

15. Q-Quit

Tujuan : keluar dari DEBUG tanpa melakukan perekaman.
Format : Q
Contoh : -Q
A>

16. R-Register

Tujuan : menampilkan dan ubah isi dari register.
Format : R [nama register]
Contoh : RCX tampilkan dan perbaiki isi register CX.
Untuk mengetahui isi masing-masing register maka ketik R, lalu tekan enter.

Contoh :

-R

AX=0000 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=3970 ES=3970 SS=3970 CS=3970 IP=0100 NV UP DI PL NZ AC PE NC
3970:0100 CD20 INT20

17. S-Search

Tujuan : cari suatu data dalam memori.

Format : S range list

range = data yang dicari.

Contoh : S CS:100 L 100 'DATA PRO' mencari kata 'DATA PRO' pada 256 byte pertama.

18. T-Trace

Tujuan : menjalankan program sambil menampilkan register yang dipakai.

Format : T [=address][value]

carian kesalahan, default-nya CS:IP

[value]=banyaknya intruksi yang akan di jalankan.

Contoh: T=102 3 periksa tiga intruksi pertama, dimulai dari alamat CS:102.

Contoh lain lagi untuk memproses sebaris program saja dengan mengetik T, yaitu :

- T

- T

AX=0CD1 BX=092A CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=3756 ES=3756 SS=3756 CS=3756 IP=0102 NV UP DI PL NZ AC PE NC
3756:0102 AC LODSB

19. U-Unassemble

Tujuan : tampilkan program yang sudah ada.

Format : u [address] atau U [range]

[address] = alamat awal program yang akan ditampilkan, alamat defaultnya adalah setelah perintah U sebelumnya.

Jika belum pernah menulis perintah U, maka defaultnya CS:0100.

[range] = daerah yang akan dilihat, defaultnya CS

Contoh : U CS:100 tampilkan program mulai dari alamat awal CS:100.

Catatan : Setiap kali kita mengetik U dan menekan Enter, maka yang akan ditampilkan adalah program sepanjang 21 heksadesimal.

Contoh :

```

-U 100
3970:0100 CD21      INT      21
3970:0102 CD20      INT      20
3970:0104 D98D460250B8 ESC     09,[DI+0246][DI+B850]
3970:010A 8D00      LEA      AX,[BX+SI]
3970:010C 50        PUSH     AX
3970:010D E82A23     CAL      243A
3970:0110 8BES      MOV      SP,BP
3970:0112 83C41A     ADD      SP,+1A
3970:0115 5D        POP      BP
3970:0116 C3        RET
3970:0117 55        PUSH     BP
3970:0118 83EC02     SUB      SP,+02
3970:011B 8BEC      MON      BP,SP
3970:011D 823E0E0000 CMP      BYTE PTR [000E], 00

```

Untuk menentukan sendiri panjang program yang ingin Saudara lihat dapat digunakan perintah L (length) dibelakang U dan format penulisannya adalah:

U [address awal] L [panjang program]

Contoh : -U 100 L 0A

```

3985:0100 B241      MOV      DL,41
3985:0102 B90A00     MOV      CX,000A
3985:0105 E8F800     CALL     0200
3985:0108 E2FB      LOOP     0105
3985:010A CD20      INT      20

```

20. W - Write

Tujuan : menulis program dan data ke dalam disket.

Format : W address drive sektor sektor

address = alamat awal yang akan di tulis ke dalam disket.

Contoh : W DS: 0 0 5 1 tulis sektor direktori ke drive A

W - Write

Tujuan : tulis program yang sudah dibuat ke dalam disket.

Format : W address

address = alamat awal yang akan ditulis, default-nya CS