

Nama : Muhammad Faqih
NIM : 2020230032
Date : Jum'at, 22-04-2022

Example 5-16

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG _
To adjust the emulated CPU speed, use ctrl-F11 and ctrl-F12.
To activate the keymapper ctrl-F1.
For more information read the README file in the DOSBox directory.

HAVE FUN!
The DOSBox Team http://www.dosbox.com

:\>SET BLASTER=A220 I7 D1 H5 T6

:\>MOUNT C "/home/faqih"
Drive C is mounted as local directory /home/faqih/

:\>C:

:\>DEBUG.EXE
A100
73F:0100 MOV AX,F
73F:0103 MOV BL,B
73F:0105 DIV BL
73F:0107 ADD AH,AH
73F:0109 CMP AH, BL
73F:010B JB 010F
73F:010D INC AL
73F:010F INT 20
73F:0111
```

DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG



```
AX=000F BX=0000 CX=0000 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=073F IP=0103  NV UP EI PL NZ NA PO NC
073F:0103 B30B          MOV     BL,0B
```

-T

```
AX=000F BX=000B CX=0000 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=073F IP=0105  NV UP EI PL NZ NA PO NC
073F:0105 F6F3          DIV     BL
```

-T

```
AX=0401 BX=000B CX=0000 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=073F IP=0107  NV UP EI PL NZ NA PO NC
073F:0107 00E4          ADD     AH,AH
```

-T

```
AX=0801 BX=000B CX=0000 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=073F IP=0109  NV UP EI PL NZ NA PO NC
073F:0109 38DC          CMP     AH,BL
```

-T

```
AX=0801 BX=000B CX=0000 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=073F IP=010B  NV UP EI NG NZ AC PO CY
073F:010B 7202          JB      010F
```

-

```
AX=0801 BX=000B CX=0000 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=073F IP=010F  NV UP EI NG NZ AC PO CY
073F:010F CD20          INT     20
```

-T

```
AX=0801 BX=000B CX=0000 DX=0000 SP=00F7 BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=F000 IP=1480  NV UP DI NG NZ AC PO CY
F000:1480 FE3B          ???     [BX+SI]          DS:000B=03
```

-

mov ax, F → memberi nilai F ke register ax

mov bl, B → memberi nilai B ke register bl

div bl → membagi nilai ax dengan bl (F : B)

- Pembagian ax dengan bl, F dengan B atau 15 dibagi dengan sebelas menghasilkan 1 dengan sisa 4
- Maka isi register ax adalah 0401. 1 disimpan di register AL dan sisanya 4 di register AH

add ah,ah → menambahkan nilai register ah dengan ah sendiri dan disimpan di register ah. Maka nilai AH adalah 0801

cmp ah, bl → membandingkan nilai ah dengan bl yaitu mengecek untuk pembulatan pada nilai pembagi.

jb 010F → jika komparasi oke maka lanjut perintah selanjutnya. Jika tidak maka menuju ke int 20 atau interupsi 20 untuk keluar

inc al → membulatkan nilai pembagi

int 20 → selesai

Pada program di atas menampilkan urutan instruksi membagi AX dengan BL dan membulatkan hasil *unsigned*. Program ini menggandakan sisa sebelum dibandingkan dengan DL untuk menentukan apakah untuk membulatkan pembagi. Instruksi INC membulatkan isi dari AL setelah komparasi. Misalkan sisa pecahan dibutuhkan daripada sisa integer. Pecahan didapat dari hasil simpan pembagi. Selanjutnya, register AL dibersihkan ke 0. Angka sisa AX sekarang dibagi dengan operand asli to menghasilkan sisa pecahan.

Example 5-23

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG - x
For more information read the README file in the DOSBox directory.
HAVE FUN!
The DOSBox Team http://www.dosbox.com

Z:\>SET BLASTER=A220 I7 D1 H5 T6

Z:\>MOUNT C "/home/faqih"
Drive C is mounted as local directory /home/faqih/

Z:\>C:

C:\>DEBUG.EXE
-A100
073F:0100 MOV AX,245
073F:0103 XOR DX,DX
073F:0105 MOV CX,100
073F:0108 DIV CX
073F:010A AAM
073F:010C ADD AX,3030
073F:010F XCHG AX,DX
073F:0111 AAM
073F:0113 ADD AX,3030
073F:0116 INT 20
073F:0118 _
```

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG - x

AX=0245 BX=0000 CX=0000 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=073F IP=0103 NU UP EI PL NZ NA PO NC
073F:0103 31D2 XOR DX,DX
-T

AX=0245 BX=0000 CX=0000 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=073F IP=0105 NU UP EI PL ZR NA PE NC
073F:0105 B90001 MOV CX,0100
-T

AX=0245 BX=0000 CX=0100 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=073F IP=0108 NU UP EI PL ZR NA PE NC
073F:0108 F7F1 DIV CX
-T

AX=0002 BX=0000 CX=0100 DX=0045 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=073F IP=010A NU UP EI PL ZR NA PE NC
073F:010A D40A AAM
-T

AX=0002 BX=0000 CX=0100 DX=0045 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=073F IP=010C NU UP EI PL NZ NA PO NC
073F:010C 053030 ADD AX,3030
_
```

```

DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG - x
AX=3032 BX=0000 CX=0100 DX=0045 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=073F IP=010F  NU UP EI PL NZ NA PO NC
073F:010F 87C2          XCHG    AX,DX
-T
AX=0045 BX=0000 CX=0100 DX=3032 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=073F IP=0111  NU UP EI PL NZ NA PO NC
073F:0111 D40A          AAM
-T
AX=0609 BX=0000 CX=0100 DX=3032 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=073F IP=0113  NU UP EI PL NZ NA PE NC
073F:0113 053030       ADD     AX,3030
-T
AX=3639 BX=0000 CX=0100 DX=3032 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=073F IP=0116  NU UP EI PL NZ NA PE NC
073F:0116 CD20        INT     20
-T
AX=3639 BX=0000 CX=0100 DX=3032 SP=00F7 BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=F000 IP=1480  NU UP DI PL NZ NA PE NC
F000:1480 FE3B       ???     [BX+SI]          DS:0000=CD
-

```

mov ax, 245 → memberi nilai 245 ke register ax
 xor dx, dx → membersihkan register dx
 mov cx, 100 → memberi nilai 100 ke cx
 div cx → membagi ax-dx dengan 100
 aam → konversi ke bcd
 add ax,3030 → konversi ke ASCII
 xchg ax,dx → ulangi untuk sisa dengan menukar ax dengan dx
 aam → konversi ke BCD
 add ax, 3030 → konversi ke ASCII
 int 20 → keluar program

AX dikonversi ke empat digit karakter string ASCII dengan menggunakan pembagian dan instruksi AAM. Pertama-tama DX dibersihkan lalu membagi DX-AX dengan 100. Sebagai contoh, jika ax=245, ax = 2 dan dx =45 setelah pembagian. Bagian yang terpisah ini dikonversi ke BCD menggunakan AAM, dan kemudian 3030 ditambahkan untuk mengonversi ke kode ascii.