



# MİKROİŞLEMCİLER

Dr. Meltem KURT PEHLİVANOĞLU

W-14

# MİKROİŞLEMCİLER

**Digital Logic +**

**Digital Design +**

**Computer Architecture +**

**Microprocessors +**

**Microcontrollers +**

**Assembly Language**

**Programming(8086)**

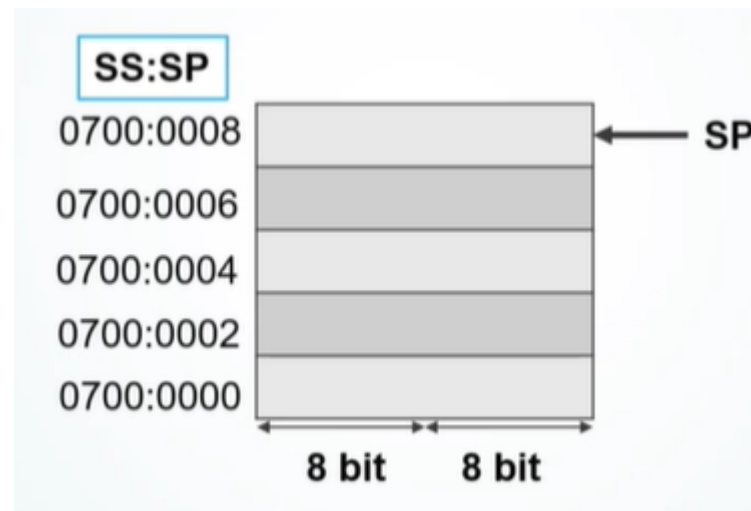
# 8086 16-Bit Mikroişlemci

- Segment ve adres register çiftleri:
- CS:IP
- SS:SP SS:BP
- DS:BX DS:SI
- DS:DI
- ES:DI

# 8086 16-Bit Mikroişlemci

## EMU 8086-MICROPROCESSOR EMULATOR

- Stack (yığın): Geçici verileri tutmak için kullanılan bellek alanıdır.
- LIFO (Last In First Out) mantığı ile çalışır. Yani son giren ilk çıkar
- Normalde her RAM hücresi 8 bit (1 byte) yer kaplıyor ancak Stack içinde her eleman 16 bit (2 byte) olarak tutuluyor. Diğer bir ifadeyle ardışık 8 bitlik 2 RAM hücresi işgal eder.



# 8086 16-Bit Mikroişlemci

## EMU 8086-MICROPROCESSOR EMULATOR

- CF(carry flag): Elde varsa 1 olur.
- ZF(zero flag): Herhangi bir işlem sonucunda 0 elde ediliyorsa ZF 1 olur
- SF(sign flag): ALU tarafından gerçekleştirilen bir işlemin sonucu eğer negatif çıkıyorsa SF 1 olur
- OF(overflow flag): İşaretli sayılarda işlem sonucu işaretli sayı aralığını aşıyorsa taşma bayrağı 1 olur (8-bitlik işaretli sayılar için en küçük değer -128, en büyük değer +127)
- PF(parity flag): İşlem sonucunda bulunan '1' bitlerinin sayısı çift ise PF 1 olur. **Sonuç 16-bit olsa bile düşük değerlikli 8-bit ele alınır.**
- AF(auxiliary flag): İşaretsiz sayılarda yapılan işlemlerdeki düşük değerlikli 4 bitte taşma meydana gelirse AF 1 olur.
- DF(direction flag): Diziler gibi ardışık verilerde özellikle string işlemlerinde kullanılan komutların ileri yönlü mü yoksa geri yönlü mü çalışacağını belirlemek için kullanılır. DF=0 iken ileri yönlü (düşük adresten yüksek adrese) işlem yapılır, DF=1 iken geri yönlü (yüksek adresten düşük adrese). Varsayılan 0 değeridir.
- IF (interrupt flag): Varsayılan olarak aktif bu sayede kesmelere izin veriyor. Örneğin klavyeden değer okuma, ekrana metin yazdırma vb.

# 8086 16-Bit Mikroişlemci

## **EMU 8086-MICROPROCESSOR EMULATOR**

### **DS VE ES KULLANIMI**

- COM dosyalarında tek bir bölüt vardır ve bu bölüt 64KB ile sınırlıdır
- Segment registerları başlangıçta aynı adresi işaret ederler

# 8086 16-Bit Mikroişlemci

## **EMU 8086-MICROPROCESSOR EMULATOR**

### **SREG: OFFSET**

DS : SI

ES : DI

SS : SP

CS : IP

# 8086 16-Bit Mikroişlemci

## EMU 8086-MICROPROCESSOR EMULATOR

- **KESMELER(INTERRUPTS)**

- **Donanım Kismesi:** İşlemcinin kesme bacağına elektrik sinyali gönderilir ve olağan akışından çıkması sağlanır. Örn. Uyku moduna geçme
- **Yazılım Kismesi:** Komut kümesinde ve interrupt vector table da yer alan önceden belirlenmiş değerler kullanılarak işlemcinin normal akışından çıkıp istenilen işlem yapılır. Örn. Klavyeden bir değer okumak, ekrana yazı yazmak



# 8086 16-Bit Mikroişlemci

## EMU 8086-MICROPROCESSOR EMULATOR

---

[Documentation Index](#) | [Licence](#) | [Tutorials](#) | [8086 Instruction Set](#) | [Interrupts](#) |

---

Quick reference:

[INT 10h/00h](#)  
[INT 10h/01h](#)  
[INT 10h/02h](#)  
[INT 10h/03h](#)  
[INT 10h/05h](#)  
[INT 10h/06h](#)  
[INT 10h/07h](#)  
[INT 10h/08h](#)  
[INT 10h/09h](#)  
[INT 10h/0Ah](#)  
[INT 10h/0Ch](#)  
[INT 10h/0Dh](#)  
[INT 10h/0Eh](#)  
[INT 10h/13h](#)

[INT 10h/1003h](#)  
[INT 11h](#)  
[INT 12h](#)  
[INT 13h/00h](#)  
[INT 13h/02h](#)  
[INT 13h/03h](#)  
[INT 15h/86h](#)  
[INT 16h/00h](#)  
[INT 16h/01h](#)  
[INT 19h](#)  
[INT 1Ah/00h](#)  
[INT 20h](#)

[INT 21h](#)  
[INT 21h/01h](#)  
[INT 21h/02h](#)  
[INT 21h/05h](#)  
[INT 21h/06h](#)  
[INT 21h/07h](#)  
[INT 21h/09h](#)  
[INT 21h/0Ah](#)  
[INT 21h/0Bh](#)  
[INT 21h/0Ch](#)  
[INT 21h/0Eh](#)  
[INT 21h/19h](#)  
[INT 21h/25h](#)  
[INT 21h/2Ah](#)  
[INT 21h/2Ch](#)

[INT 21h/35h](#)  
[INT 21h/39h](#)  
[INT 21h/3Ah](#)  
[INT 21h/3Bh](#)  
[INT 21h/3Ch](#)  
[INT 21h/3Dh](#)  
[INT 21h/3Eh](#)  
[INT 21h/3Fh](#)  
[INT 21h/40h](#)  
[INT 21h/41h](#)  
[INT 21h/42h](#)  
[INT 21h/47h](#)  
[INT 21h/4Ch](#)  
[INT 21h/56h](#)

[INT 33h/0000h](#)  
[INT 33h/0001h](#)  
[INT 33h/0002h](#)  
[INT 33h/0003h](#)

---

# 8086 16-Bit Mikroişlemci

## EMU 8086-MICROPROCESSOR EMULATOR

### Yazılım Kesmesi Türleri;

1. **DOS**: MS-DOS işletim sistemi temelli programlar tarafından kullanılan kesmelerdir. Örn. INT 21h
2. **BIOS**: Herhangi bir işletim sistemine ihtiyaç duymayan direk donanım üzerinde gerçekleştirilen fonksiyonları barındıran kesmelerdir. Örn. INT 10h , INT 11h

# 8086 16-Bit Mikroişlemci

## EMU 8086-MICROPROCESSOR EMULATOR

INT	immediate byte	<p>Interrupt numbered by immediate byte (0..255).</p> <p>Algorithm:</p> <p>Push to stack:</p> <ul style="list-style-type: none"><li>◦ flags register</li><li>◦ CS</li><li>◦ IP</li></ul> <ul style="list-style-type: none"><li>• IF = 0</li><li>• Transfer control to interrupt procedure</li></ul> <p>Example:</p> <p>MOV AH, 0Eh ; teletype. MOV AL, 'A' INT 10h ; BIOS interrupt. RET</p> <table><tr><td>C</td><td>Z</td><td>S</td><td>O</td><td>P</td><td>A</td><td>I</td></tr><tr><td colspan="6">unchanged</td><td>0</td></tr></table>	C	Z	S	O	P	A	I	unchanged						0
C	Z	S	O	P	A	I										
unchanged						0										

### INT(interrupt) komutu:

IF=0 olur çünkü bir kesme gelip bitmeden başka bir kesmenin gelmesi engellenir.

AH: Fonksiyon numarası

BH,BL,CH,CL.. : Parametre listesi

INT x: Kesme Numarası

# 8086 16-Bit Mikroişlemci

## EMU 8086-MICROPROCESSOR EMULATOR

- BIOS KESMELERİ (VIDEO KESMELERİ)

### Video Kesmeleri

BIT COLOR TABLE (STİL TABLOSU)					
HEX	BIN	COLOR	HEX	BIN	COLOR
0	0000	siyah	8	1000	koyu gri
1	0001	mavi	9	1001	açık mavi
2	0010	yeşil	A	1010	açık yeşil
3	0011	camgöbeği	B	1011	açık camgöbeği
4	0100	kırmızı	C	1100	açık kırmızı
5	0101	eflatun	D	1101	açık eflatun
6	0110	kahverengi	E	1110	sarı
7	0111	açık gri	F	1111	beyaz
Yüksek değerli 4 bit arka plan rengini, düşük değerli 4 bit yazı rengini verir. (11110000b beyaz üzerine siyah yazı)					

# 8086 16-Bit Mikroişlemci

## EMU 8086-MICROPROCESSOR EMULATOR

- BIOS KESMELERİ:**

Kesme Numarası	10h	Kesme Açıklaması	Video modu ayarlama
Fonksiyon Numarası	AH=00h	Parametre Kaydedicisi	AL
00h-metin modu. 40x25 karakter. 16 renk. 8 sayfa. (sütun x satır)			
03h-metin modu. 80x25 karakter. 16 renk. 8 sayfa. (sütun x satır)			
13h-grafik modu. 40x25 karakter. 256 renk. 320x200 piksel. 1 sayfa. (sütun x satır)			

Kesme Numarası	10h	Kesme Açıklaması	İmleç pozisyonu ayarlama.
Fonksiyon Numarası	AH=02h	Parametre Kaydedicisi	DH,DL,BH
DH=satır.			
DL=sütun.			
BH=sayfa numarası (0...7).			

Kesme Numarası	10h	Kesme Açıklaması	İmleç pozisyonunu okuma.
Fonksiyon Numarası	AH=03h	Parametre Kaydedicisi	-
DH=satır.			
DL=sütun.			
BH=sayfa numarası (0...7).			

Kesme Numarası	10h	Kesme Açıklaması	Aktif sayfayı seçme.
Fonksiyon Numarası	AH=05h	Parametre Kaydedicisi	AL
AL=aktif olacak sayfa numarası (0...7)			

org 100h

```
mov ah,00h ; video modu ayarlama
mov al, 00h
;mov al, 03h
;mov al, 13h
int 10h
```

```
mov ah,02h ; imlec pozisyonunu ayarlama
mov dh, 13h ;satir
mov dl, 45h ;sutun
mov bh, 0
int 10h
```

```
mov ah,03h ; imlec pozisyonunu okuma
; bu durumda imlecin pozisyonunu DH: satir DL:sutun
int 10h
```

```
mov ah,05h ; aktif sayfayı secme
mov al,1
int 10h
```

ret

# 8086 16-Bit Mikroişlemci

## EMU 8086-MICROPROCESSOR EMULATOR

- BIOS KESMELERİ:**

<b>Kesme Numarası</b>	10h	<b>Kesme Açıklaması</b>	İmleç Karakter Okuma
<b>Fonksiyon Numarası</b>	AH=08h	<b>Parametre Kaydedicisi</b>	BH
BH=sayfa numarası AL=okunan karakter AH=karakter stil			

<b>Kesme Numarası</b>	10h	<b>Kesme Açıklaması</b>	İmleç Karakter Yazma
<b>Fonksiyon Numarası</b>	AH=09h	<b>Parametre Kaydedicisi</b>	AL,BH,CL,CX
AL=görünecek karakter BH=sayfa numarası BL=karakter stil CX=tekrar sayısı			

<b>Kesme Numarası</b>	10h	<b>Kesme Açıklaması</b>	İmleç Karakter Yazma
<b>Fonksiyon Numarası</b>	AH=0Ah	<b>Parametre Kaydedicisi</b>	AL,BH,CX
AL=görünecek karakter BH=sayfa numarası CX=tekrar sayısı			

<b>Kesme Numarası</b>	10h	<b>Kesme Açıklaması</b>	Tek Piksel Renk Değiştirme
<b>Fonksiyon Numarası</b>	AH=0Ch	<b>Parametre Kaydedicisi</b>	AL,BH,CX
Video modu grafik modu olarak ayarlanmalıdır. AL=piksel rengi CX=sütun DX=satır			
<b>Kesme Numarası</b>	10h	<b>Kesme Açıklaması</b>	Tek Piksel Renk Alma
<b>Fonksiyon Numarası</b>	AH=0Ch	<b>Parametre Kaydedicisi</b>	CX,DX
Video modu grafik modu olarak ayarlanmalıdır. CX=sütun DX=satır AL=piksel rengi			

# 8086 16-Bit Mikroişlemci

## **EMU 8086-MICROPROCESSOR EMULATOR**

- SORU1: metin db 'ekranayazdirma' dizisindeki elemanları tek tek ekrana yan yana olacak şekilde yazdıran 8086 Assembly kodunu yazınız
  1. İmleç (imlecin bulunduğu yere) karakter yazma
  2. İmleç pozisyonunu okuma
  3. İmleç pozisyonunu ayarlama

# 8086 16-Bit Mikroişlemci

## EMU 8086-MICROPROCESSOR EMULATOR

org 100h

lea SI,metin

mov CX, sayac-offset metin

dongu:

PUSH CX ; metin icinde kac kez donecegimizi tutuyoruz

mov AH,0Ah ; imlec karakter yazma

MOV AL,[SI]

MOV BH,0

MOV CX,1 ; karakteri 1 kez yazdirma

INC SI

int 10h

MOV AH,03h ; imlec pozisyonunu okuma

int 10h ; DH satir DL sutun degeri var

MOV AH,02h ; imlec pozisyonunu ayarlama

INC DL ; yan yana yazacagi icin DL degerini arttirmek gerekiyor imleci saga kaydirarak yazdiriyoruz

int 10h

POP CX

loop dongu

ret

metin db "ekranayazdirma"

sayac db 0



# 8086 16-Bit Mikroişlemci

## EMU 8086-MICROPROCESSOR EMULATOR

### • BIOS KESMELERİ:

#### Klavye Kesmeleri

<b>Kesme Numarası</b>	16h	<b>Kesme Açıklaması</b>	Klavyeden Karakter Okuma
<b>Fonksiyon Numarası</b>	AH=00h	<b>Parametre Kaydedicisi</b>	-
AH=BIOS tarama kodu (scan code) AL=ASCII karakteri  Klavye ara belleğinde bir tuş vuruşu varsa klavye ara belleğinden kaldırılır.			

<b>Kesme Numarası</b>	16h	<b>Kesme Açıklaması</b>	Klavye Tampon Kontrolü
<b>Fonksiyon Numarası</b>	AH=01h	<b>Parametre Kaydedicisi</b>	-
ZF=1 ise ara bellekte tuş vuruşu yoktur. ZF=0 ise ara bellek tuş vuruşu vardır. AH=BIOS tarama kodu (scan code) AL=ASCII karakteri  Klavye ara belleğinde bir tuş vuruşu varsa klavye ara belleğinden <u>kaldırılmaz</u> .			

# 8086 16-Bit Mikroişlemci

## EMU 8086-MICROPROCESSOR EMULATOR

org 100h	; ZF=1 aktifken tamponda veri yok demek org 100h
don:	don:
MOV AH,00h	MOV AH,01h
int 16h	int 16h
jmp don	jmp don
ret	ret

# 8086 16-Bit Mikroişlemci

## **EMU 8086-MICROPROCESSOR EMULATOR**

- SORU2: Kullanıcıdan alacağı 7 karakteri ekrana yazdıran 8086 Assembly kodunu yazınız
  1. Klavyeden karakter okuma
  2. İmleç (imlecin bulunduğu yere) karakter yazma
  3. İmleç pozisyonunu okuma
  4. İmleç pozisyonunu ayarlama

# 8086 16-Bit Mikroişlemci

## EMU 8086-MICROPROCESSOR EMULATOR

```
org 100h
```

```
MOV CX,7
```

```
don:
```

```
PUSH CX
```

```
MOV AH,00h
```

```
int 16h ; klavyeden veri al
```

```
MOV AH,0Ah ; imlec karakter yazma
```

```
MOV BH,0 ;AL de karakter gorunecek
```

```
MOV CX,1 ; karakteri 1 kez yazsin
```

```
int 10h
```

```
MOV AH,03h ; imlec pozisyon okuma
```

```
int 10h ; DL de sutun degeri var
```

```
MOV AH, 02h ;imlec pozisyon ayarlama
```

```
INC DL
```

```
int 10h
```

```
POP CX
```

```
loop don
```

```
ret
```

# 8086 16-Bit Mikroişlemci

## EMU 8086-MICROPROCESSOR EMULATOR

- BIOS KESMELERİ:

Kesme Numarası	10h	Kesme Açıklaması	Otomatik Karakter Yazdırma
Fonksiyon Numarası	AH=0Eh	Parametre Kaydedicisi	AL
Bu kesme ekranda bir karakter görüntüler, imleci ilerletir ve ekran gerektiği gibi kaydırır. Yazdırma her zaman geçerli etkin sayfaya yapılır. AL=karakter			

Kesme Numarası	10h	Kesme Açıklaması	Dizi Yazdırma
Fonksiyon Numarası	AH=13h	Parametre Kaydedicisi	AL,BH,BL,CX,DL,DH,ES:BP
AL=yazdırma modu (0=yazdıktan sonra imleç güncelle veya 1=dizi nitelikleri içerir) BH=sayfa numarası BL=stil (nitelikler) CX=dizi uzunluğu DL=sütun DH=satır ES:BP=dizinin bulunduğu adres			

org 100h

MOV AH,0Eh

MOV AL,'A' ; ekrana A karakterinin yazılması  
int 10h

ret

org 100h

MOV AH, 13h

MOV CX, sayac-offset mesaj

MOV AL,0

MOV BH,0

MOV DH,0

MOV DL,0

MOV BP,offset mesaj

MOV BL,11110000b ;

int 10h

ret

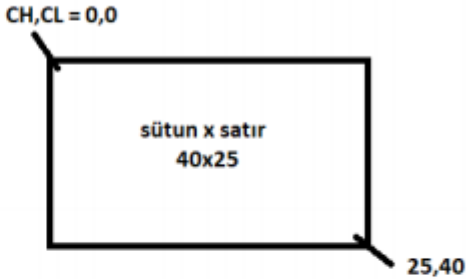
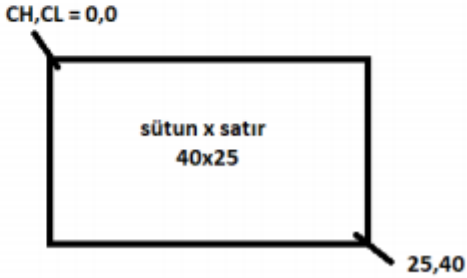
mesaj db "yeni dizi"

sayac db 0

# 8086 16-Bit Mikroişlemci

## EMU 8086-MICROPROCESSOR EMULATOR

- BIOS KESMELERİ:

Kesme Numarası	10h	Kesme Açıklaması	Yukarıya doğru sayfa temizleme
Fonksiyon Numarası	AH=06h	Parametre Kaydedicisi	AL, BH, CH, CL, DH, DL
<b>AL</b> =temizlenecek (kayacak) satır sayısı (00h=seçili olan tüm alanı temizler(boyar)) <b>BH</b> =temizlemede kullanılacak stil (stil tablosuna bakın) <b>CH, CL</b> =sol üst köşe koordinatları (CH=satır, CL=sütun) <b>DH, DL</b> =sağ alt köşe koordinatları (DH=satır, DL=sütun)			
Kesme Numarası	10h	Kesme Açıklaması	Aşağıya doğru sayfa temizleme
Fonksiyon Numarası	AH=07h	Parametre Kaydedicisi	AL, BH, CH, CL, DH, DL
<b>AL</b> =temizlenecek (kayacak) satır sayısı (00h=seçili olan tüm alanı temizler(boyar)) <b>BH</b> =temizlemede kullanılacak stil (stil tablosuna bakın) <b>CH, CL</b> =sol üst köşe koordinatları (CH=satır, CL=sütun) <b>DH, DL</b> =sağ alt köşe koordinatları (DH=satır, DL=sütun)			

# 8086 16-Bit Mikroişlemci

## **EMU 8086-MICROPROCESSOR EMULATOR**

- **DOS KESMELERİ**

# 8086 16-Bit Mikroişlemci

## EMU 8086-MICROPROCESSOR EMULATOR

- MS-DOS KESMELERİ

### Video ve Klavye Kesmeleri

Kesme Numarası	21h	Kesme Açıklaması	Karakter Okuma ve Yazdırma
Fonksiyon Numarası	AH=01h	Parametre Kaydedicisi	-
Klavye tamponunda herhangi bir karakter yoksa karakter girilene kadar bekler. Karakter AL içerisinde saklanır. AL içerisindeki karakterde ekrana yazdırılır.			

Kesme Numarası	21h	Kesme Açıklaması	Karakter Yazdırma
Fonksiyon Numarası	AH=02h	Parametre Kaydedicisi	DL
DL içerisindeki karakter ekrana yazdırılır ve yazdırılan karakter AL içerisine de aktarılır. İmleç otomatik olarak sağa kaydırılacaktır.			

Kesme Numarası	21h	Kesme Açıklaması	Direkt Okuma ve Yazdırma
Fonksiyon Numarası	AH=06h	Parametre Kaydedicisi	DL
DL içerisine 0-254 arasında değerleri girilirse ekrana karakter yazdırılır. Karakter yazdırma sonucunda AL=DL olur. DL içerisine 255 değeri girilirse karakter okuma yapar.  Karakter okuma esnasında eğer karakter yoksa <b>ZF=1</b> olur ve <b>AL=00h</b> olur. Eğer karakter varsa <b>ZF=0</b> olur ve <b>AL=okunan karakter</b> şeklinde atama yapılır. Aynı zamanda karakter tampondan temizlenir.			

Kesme Numarası	21h	Kesme Açıklaması	Karakter Okuma
Fonksiyon Numarası	AH=07h	Parametre Kaydedicisi	-
Klavyeden okunan karakter <b>AL</b> içerisine aktarılır. Klavye arabelleğinde (tampon) karakter yoksa, işlev herhangi bir tuşa basılana kadar bekler.			



# 8086 16-Bit Mikroişlemci

## EMU 8086-MICROPROCESSOR EMULATOR

- MS-DOS KESMELERİ

Kesme Numarası	21h	Kesme Açıklaması	Dizi Yazdırma
Fonksiyon Numarası	AH=09h	Parametre Kaydedicisi	-
DS:DX ikilisinin gösterdiği adresteki “\$” işareti ile biten string ifadeyi ekrana yazdırır.			

org 100h

MOV DX, offset msg  
MOV AH,09h  
int 21h ; kesme numarası

ret

msg db 'hello world\$'  
msglen db 0

# 8086 16-Bit Mikroişlemci

## EMU 8086-MICROPROCESSOR EMULATOR

- MS-DOS KESMELERİ**

<b>Kesme Numarası</b>	21h	<b>Kesme Açıklaması</b>	Girdi Durum Kontrolü
<b>Fonksiyon Numarası</b>	AH=0Bh	<b>Parametre Kaydedicisi</b>	-
AL=00h ise karakter yok, AL=0FFh ise karakter var. NOT: Emulator içerisindeki screen penceresi açılarak test edilebilir.			

org 100h

<b>Kesme Numarası</b>	21h	<b>Kesme Açıklaması</b>	Tampon Temizleme
<b>Fonksiyon Numarası</b>	AH=0Ch	<b>Parametre Kaydedicisi</b>	-
Tampon temizlenir AL de bulunan fonksiyon numarasına göre işlem gerçekleştirilir. AL=01h, 06h, 07h, 08h veya 0Ah değerlerini alabilir. Diğer değerler için sadece tampon temizlenir fonksiyon çalıştırılmaz.			

MOV AH,2Ah

int 21h

; CX=YIL DH=AY DL=GÜN (1-31)

;AL=HAFTA GÜNÜ (00H:PAZAR)

<b>Kesme Numarası</b>	21h	<b>Kesme Açıklaması</b>	İşletim Sistemine Dönüş
<b>Fonksiyon Numarası</b>	AH=4Ch	<b>Parametre Kaydedicisi</b>	-
Programları durdurmak için kullanılmaktadır.			

ret

### Tarih ve Saat İşlemleri

<b>Kesme Numarası</b>	21h	<b>Kesme Açıklaması</b>	Sistem Tarihini Alma
<b>Fonksiyon Numarası</b>	AH=2Ah	<b>Parametre Kaydedicisi</b>	-
CX=yıl (1980-2099) DH=ay DL=ay günü (1-31) AL= hafta günü (00h=pazar)			
<b>Kesme Numarası</b>	21h	<b>Kesme Açıklaması</b>	Sistem Zamanı Alma
<b>Fonksiyon Numarası</b>	AH=2Ch	<b>Parametre Kaydedicisi</b>	-
CH=saat CL=dakika DH=saniye DL=salise (1/100)			

# 8086 16-Bit Mikroişlemci

## **EMU 8086-MICROPROCESSOR EMULATOR**

- **DOSYA İŞLEMLERİ**

# 8086 16-Bit Mikroişlemci

## EMU 8086-MICROPROCESSOR EMULATOR

- **DOSYA İŞLEMLERİ**

DOS dosya sistemi C:\emu8086\vdribe\x (x sürücü harfi) şeklindedir.

Sürücü harfi belirtilmezse ve geçerli izin ayarlanmazsa, varsayılan olarak C:\emu8086\MyBuild\ yolu kullanılır.

NOT: DOS dosya sisteminde dosya adı sınırı vardır.

Dosya adı olarak 8 karakter, dosya uzantısı olarak 3 karakter kullanılabilir(noktadan sonrası). Örneğin, Dosyam.txt geçerli bir dosya adıdır.

# 8086 16-Bit Mikroişlemci

## EMU 8086-MICROPROCESSOR EMULATOR

- DOSYA İŞLEMLERİ**

<b>Kesme Numarası</b>	21h	<b>Kesme Açıklaması</b>	Varsayılan Sürücüyü Seçme
<b>Fonksiyon Numarası</b>	AH=0Eh	<b>Parametre Kaydedicisi</b>	DL
<b>DL</b> =sürücü numarası (0=A, 1=B, vb.) <b>AL</b> =potansiyel geçerli sürücü harflerinin sayısı <b>NOT:</b> Varsayılan sürücüyü seçmek yeterli değildir. Klasör ve dosyaları oluştururken sürücü adını bildirmek gereklidir.			

<b>Kesme Numarası</b>	21h	<b>Kesme Açıklaması</b>	Geçerli Varsayılan Sürücü Alma
<b>Fonksiyon Numarası</b>	AH=19h	<b>Parametre Kaydedicisi</b>	-
<b>AL</b> =sürücü adı (0=A, 1=B, vb.)			

org 100h

; varsayılan surucu ayarlama

MOV AH,0Eh

MOV DL,2 ; C:\ ayarlandı

int 21h

; varsayılan surucu alma

mov ah,19h

int 21h ; AL de surucu adi olacak (0=A , 1=B , 2=C ..)

ret

# 8086 16-Bit Mikroişlemci

## EMU 8086-MICROPROCESSOR EMULATOR

- DOSYA İŞLEMLERİ**

<b>Kesme Numarası</b>	21h	<b>Kesme Açıklaması</b>	Klasör(Dizin) Oluşturma
<b>Fonksiyon Numarası</b>	AH=39h	<b>Parametre Kaydedicisi</b>	DS:DX
<b>DS:DX</b> --> klasorAdi db "C:\klasor",0 şeklinde dosya yolu ve adı verilmelidir.			
Klasör oluşturma işlemi başarılı olursa <b>AX</b> temizlenir ve <b>CF=0</b> olur. Klasör oluşturma işlemi başarısız olursa <b>AX=hata kodu</b> ve <b>CF=1</b> olarak ayarlanır.			
<b>NOT:</b> Yukarıdaki klasöre kendi bilgisayarınızda varsayılan olarak C:\emu8086\vdribe\C\mydir yolunu takip ederek ulaşabilirsiniz.			

org 100h

; klasor olusturma

LEA DX, klasoradi  
MOV AH,39h  
int 21h

ret

; C:\emu8086\vdribe\C\mydir içinde olusacak klasor1 adli klasor  
klasoradi db 'C:\klasor1',0 ; burada varsayılan surucu olarak C ayarlasak bile C:\ yazmayip klasor1 yazarsanız MyBuild altina ekler

# 8086 16-Bit Mikroişlemci

## EMU 8086-MICROPROCESSOR EMULATOR

- DOSYA İŞLEMLERİ**

<b>Kesme Numarası</b>	21h	<b>Kesme Açıklaması</b>	Klasör(Dizin) Silme
<b>Fonksiyon Numarası</b>	AH=3Ah	<b>Parametre Kaydedicisi</b>	DS:DX
<b>DS:DX</b> --> klasorAdi db "C:\klasor",0 şeklinde dosya yolu ve adı verilmelidir.			
Klasör silme işlemi başarılı olursa <b>AX</b> temizlenir ve <b>CF=0</b> olur. Klasör silme işlemi başarısız olursa <b>AX=hata kodu</b> ve <b>CF=1</b> olarak ayarlanır.			

org 100h

; klasor silme

LEA DX, klasoradi

MOV AH,3Ah

int 21h

ret

; C:\emu8086\vdribe\C\mydir icindeki klasor1 adli klasor silinir

klasoradi db 'C:\klasor1',0 ; klasor silinmesi icin klasorun ici bos olmalı

# 8086 16-Bit Mikroişlemci

## EMU 8086-MICROPROCESSOR EMULATOR

- DOSYA İŞLEMLERİ

Kesme Numarası	21h	Kesme Açıklaması	Geçerli Klasör(Dizin) Alma
Fonksiyon Numarası	AH=47h	Parametre Kaydedicisi	DL, DS:SI
DL=sürücü numarası (00h=varsayılan, 01h=A, vb.)			
DS:SI -->geçerli dizini tutacak dizi (maksimum 64 byte)			
Başarılı olursa CF=0 olur. Başarısız olursa CF=1 ve AX=hata kodu (0Fh) alır.			

org 100h

; gecerli dizin alma

LEA SI, gecerlidizin

MOV DL,00h ;00: varsayılan 01:A, 02:B, 03:C

MOV AH,47h

int 21h

ret

gecerlidizin db 64 dup(0)

end



# 8086 16-Bit Mikroişlemci

## EMU 8086- MICROPROCESSOR EMULATOR

- DOSYA İŞLEMLERİ**

Kesme Numarası	21h	Kesme Açıklaması	Geçerli Klasör(Dizin) Ayarlama
Fonksiyon Numarası	AH=3Bh	Parametre Kaydedicisi	DS:DX
DS:DX --> klasorAdi db "C:\klasor",0 şeklinde dosya yolu ve adı verilmelidir. (En fazla 64 bayt).			
Başarılı olursa CF=0 ve AX temizlenir. Başarısız olursa CF=1 ve AX=hata kodu döndürülür.			

org 100h

; varsayılan surucu ayarlama

MOV AH,0Eh

MOV DL,0 ; A ayarladık B olarak degistirirsek varsayılan surucu B ayarladığımız için dosyayolu: klasor2 olur  
int 21h

; A için geçerli izin ayarlama

LEA DX, varsdizinA

MOV AH,3Bh

int 21h

; B için geçerli izin ayarlama

LEA DX, varsdizinB

MOV AH,3Bh

int 21h

; varsayılan surucudeki geçerli izin alma

LEA SI, dosyayolu

MOV DL,00h ;00: varsayılan 01:A, 02:B, 03:C ; varsayılan surucu A ayarladığımız için dosyayolu: klasor1 olur

MOV AH,47h

int 21h

ret

dosyayolu db 64 dup(0)

varsdizinA db 'A:\klasor1',0

varsdizinB db 'B:\klasor2',0

end

# 8086 16-Bit Mikroişlemci

## EMU 8086-MICROPROCESSOR EMULATOR

### • DOSYA İŞLEMLERİ

org 100h

;dosya olusturma

LEA DX, dosyaadi

MOV CX,0

MOV AH,3Ch

int 21h

ret

dosyaadi db 'dosyam.txt',0 ;C:\emu8086\MyBuild

end

org 100h

;dosya silme

LEA DX, dosyaadi

MOV CX,0

MOV AH,41h

int 21h

ret

dosyaadi db 'dosyam.txt',0 ;C:\emu8086\MyBuild

end

Kesme Numarası	21h	Kesme Açıklaması	Dosya Oluşturma
Fonksiyon Numarası	AH=3Ch	Parametre Kaydedicisi	CX, DS:DX
<b>DS:DX</b> --> dosyaAdi db "Dosyam.txt",0 şeklinde dosya adı verilmelidir. <b>CX</b> = dosya özellikleri <ul style="list-style-type: none"><li>• 0=normal</li><li>• 1=yalnızca okunabilir (read-only)</li><li>• 2=gizli (hidden)</li><li>• 4=sistem (system)</li><li>• 7=gizli, sistem ve yalnızca okunabilir (hidden, system and read-only)</li><li>• 16=arşiv (archive)</li></ul> Başarılı olursa <b>CF=0</b> olur ve <b>AX=dosya tanıtıcısı</b> alır. Başarısız olursa <b>CF=1</b> olur ve <b>AX=hata kodu</b> alır.  <b>NOT:</b> Eğer belirtilen dosya varsa uyarı yapılmaksızın silinir ve yeni dosya oluşturulur. Dosya tanıtıcısı sistem tarafından verilen bir değerdir. Bu değeri kaybetmemek için bir değişkende tutmakta fayda vardır. Çünkü açık olan dosyalar kapatılmak istendiği zaman file handle (dosya tanıtıcısı) kullanılmaktadır. Aynı zamanda anlık dosya oluşturup silmek için de file handle gereklidir.			

Kesme Numarası	21h	Kesme Açıklaması	Dosya Silme (Bağlantıyı Kaldırma)
Fonksiyon Numarası	AH=41h	Parametre Kaydedicisi	DS:DX
<b>DS:DX</b> -->dosya adı yazılır  Başarılı olursa <b>CF=0</b> ve <b>AX</b> temizlenir. <b>AL</b> silinen dosyanın sürücüsünü tutar. Başarısız olursa <b>CF=1</b> ve <b>AX= hata kodunu</b> tutar.  <b>NOT:</b> Aslında dosya silinmez sadece FAT dosya sistemi bağlantısı kopartılır ve dosya erişilemez hale gelir. Dosya açıkken silme işlemi tutarsızlıklara sebebiyet verebilir.			

# 8086 16-Bit Mikroişlemci

## EMU 8086-MICROPROCESSOR EMULATOR

### • DOSYA İŞLEMLERİ

Kesme Numarası	21h	Kesme Açıklaması	Dosya Açma
Fonksiyon Numarası	AH=3Dh	Parametre Kaydedicisi	AL, DS:DX
<p>DS:DX --&gt; dosyaAdi db "Dosyam.txt",0 şeklinde dosya adı verilmelidir.</p> <p>AL=erişim modu</p> <ul style="list-style-type: none"><li>• okuma (read)-0</li><li>• yazma (write)-1</li><li>• okuma/yazma (read/write)-2</li></ul> <p>Başarılı olursa <b>CF=0</b> olur ve <b>AX=dosya tanıtıcısı</b> alır. Başarısız olursa <b>CF=1</b> olur ve <b>AX=hata kodu</b> alır.</p> <p><b>NOT:</b> Dosya işaretçisi dosyanın başına ayarlanmıştır. Ayrıca dosyanın var olması gerekmektedir.</p>			
Kesme Numarası	21h	Kesme Açıklaması	Dosya Kapatma
Fonksiyon Numarası	AH=3Eh	Parametre Kaydedicisi	BX
<p><b>BX</b>=dosya tanıtıcısı (file handle)</p> <p>Başarılı olursa <b>CF=0</b> ve <b>AX</b> temizlenir. Başarısız olursa <b>CF=1</b> ve <b>AX=hata kodu (06h)</b> değerlerini alır.</p>			

# 8086 16-Bit Mikroişlemci

## EMU 8086-MICROPROCESSOR EMULATOR

```
org 100h
```

```
;dosya olusturma
```

```
LEA dx,dosyaadi  
MOV CX,0  
MOV AH,3Ch  
int 21h  
MOV [fileHandle],AX
```

```
; dosya kapatma  
MOV AH,3Eh  
MOV BX,[fileHandle]  
int 21h
```

```
;dosya silme
```

```
LEA DX, dosyaadi  
MOV AH,41h  
int 21h
```

```
ret  
dosyaadi db 'dosyam.txt',0 ;C:\emu8086\MyBuild  
fileHandle dw ? ; dosya taniticisi
```

```
end
```

# 8086 16-Bit Mikroişlemci

## EMU 8086-MICROPROCESSOR EMULATOR

### • DOSYA İŞLEMLERİ

org 100h

;dosya olusturma

```
LEA dx,dosyaadi
MOV CX,0
MOV AH,3Ch
int 21h
mov [fileHandle],AX
```

; dosya yazma

```
mov bx, [fileHandle]
mov dx, offset data
MOV CX,sonraki-offset data ; $ - offset data
MOV AH,40h
int 21h
```

```
mov al,0
mov bx, [fileHandle]
mov cx,0
mov dx,5 ; isaretciji 5 byte ilerleterek baslatiriz
mov ah,42h
int 21h
```

; dosyadan oku

```
mov bx, [fileHandle]
mov dx,offset buffer
MOV CX,4
MOV AH,3fh
int 21h
```

```
; dosya kapatma
MOV AH,3Eh
MOV BX,[fileHandle]
int 21h
```

```
ret
dosyaadi db 'dosyam.txt',0 ;C:\emu8086\MyBuild
fileHandle dw ? ; dosya taniticiisi
data db 'merhaba dosya'
sonraki db 0
buffer db 4 dup ('')
```

end

Kesme Numarası	21h	Kesme Açıklaması	Dosya İçeriği Yazma
Fonksiyon Numarası	AH=40h	Parametre Kaydedicisi	BX, CX, DS:DX
<b>BX</b> =dosya tanıtıcısı (file handle) <b>CX</b> =yazılacak bayt sayısı <b>DS:DX</b> -->yazılacak veri  Başarılı olursa <b>CF=0</b> olarak ayarlanır. <b>AX=yazılan bayt</b> sayısını tutar. Başarısız olursa <b>CF=1</b> olarak ayarlanır ve <b>AX=hata kodunu</b> tutar.			

Kesme Numarası	21h	Kesme Açıklaması	İşaretçi Ayarlama
Fonksiyon Numarası	AH=42h	Parametre Kaydedicisi	AL, BX, CX:DX
<b>AL=0</b> (dosya başlangıcı) veya <b>1</b> (geçerli dosya pozisyonu) veya <b>2</b> (dosya sonu) <b>BX</b> =dosya tanıtıcısı (file handle) <b>CX:DX</b> =kaç bayt ilerleyeceği bilgisini taşır  Örneğin; AL=0 iken CX=0 ve DX=7 ayarlanırsa işaret edilen dosyanın başından itibaren 7 karakter ileriye atlar.  Başarılı olursa <b>CF=0</b> , <b>DX:AX</b> =dosya başından itibaren yeni işaretçi konumu bayt sayısı olarak alır. Başarısız olursa <b>CF=1</b> olur ve <b>AX=hata kodu</b> değerini içerir.			

Kesme Numarası	21h	Kesme Açıklaması	Dosya İçeriğini Okuma
Fonksiyon Numarası	AH=3Fh	Parametre Kaydedicisi	BX, CX, DS:DX
<b>BX</b> =dosya tanıtıcısı (file handle) <b>CX</b> =okunacak bayt sayısı <b>DS:DX</b> -->okunan verinin yazılacağı dizi  Başarılı olursa <b>CF=0</b> olarak ayarlanır. <b>AX=okunan bayt</b> sayısını tutar. Ancak çağrıdan önce dosya sonu (end of file) 0 ise bu işlem gerçekleşir. Başarısız olursa <b>CF=1</b> olarak ayarlanır ve <b>AX=hata kodunu</b> tutar.  <b>NOT:</b> Veriler geçerli dosya konumundan başlayarak okunur ve başarılı bir okumadan sonra dosya konumu güncellenir. Kısmi bir okuma gerçekleştirilmişse AX içerisindeki değer CX içerisindeki değerden küçük olabilir.			

# 8086 16-Bit Mikroişlemci

## EMU 8086-MICROPROCESSOR EMULATOR

- DOSYA İŞLEMLERİ

Kesme Numarası	21h	Kesme Açıklaması	Yeniden Adlandırma ve Taşıma
Fonksiyon Numarası	AH=56h	Parametre Kaydedicisi	DS:DX, ES:DI
DS:DX -->Var olan dosya adı/klasör adı ES:DI -->Yeni dosya adı/klasör adı  Başarılı olursa <b>CF=0</b> olarak ayarlanır. Başarısız olursa <b>CF=1</b> ve <b>AX=hata kodu</b> alır.			

[DOS error codes \(stanislavs.org\)](http://stanislavs.org)

# 8086 16-Bit Mikroişlemci

## EMU 8086-MICROPROCESSOR EMULATOR

data segment

```
first db 'C:\klasor1\dosyam.txt',0
;first db 'C:\klasor1\dosyam.txt',0
ends
```

extra segment

```
new db 'C:\klasor2\dosyam.txt',0
;new db 'C:\klasor1\dosyam2.txt',0
ends
```

code segment

start:

; set segment registers:

```
mov ax, data
mov ds, ax
mov ax, extra
mov es, ax
```

;dosya tasima

```
LEA DX, first
LEA DI, new
MOV AH,56h
int 21h
```

```
mov ax, 4c00h ; exit to operating system.
int 21h
ends
```

end start ; set entry point and stop the assembler.