

4.B.2.2 – BIOS VE DOS KESMELERİ İLE İLGİLİ PROGRAMLAR

BIOS ve DOS kesmeleri "3.B.4.3– KESME KOMUTLARI" bölümünde açıklanıldığı gibi ASSEMBLY DİLİNÉ büyük özellikler kazandıran iki önemli işlem yaptırıcılarıdır.

BIOS ve DOS kesmeleri değişik komutlardan oluşmaktadır. Bu komutların listesi daha önce verildiği için burada tekrarlanmayacaktır.

Ancak, hatırlatma bakımından genel özellikleri bir kere daha açıklanacak ve sonradan KESMELİ PROGRAMLARDAN örnekler verilecektir.

4.B.2.2.1 – BIOS KESMELERİ

BIOS şu cumlenin kısaltılmışıdır:

BASIC INPUT-OUTPUT SYSTEM (Temel Giriş Çıkış Sistemi)

Adından anlaşıldığı gibi, BIOS kesme komutları bilgisayardaki pek çok GİRİŞ ve ÇIKIŞ işlemlerinin daha verimli şekilde yürütülmesini sağlar.

Kesme komutları, BELİRLİ BİR PROGRAM içerisinde İŞLERLİK kazanlığı gibi, bir veya daha fazla TUŞA basılarakta işlerlik kazanır. Örneğin, "PRINT SCREEN" ile "05H" EKRAN YAZDIRMA kesmesi işlerlik kazanır.

6 adet Kesme Komutu vardır:

INT, INT0, IRET, BOUND, CLI ve STI

Bunların içinde en çok kullanılan kesme komutu "INT" dır.

INT; INTERRUPT (kesme) kelimesinin kısaltılmışıdır.

Komut Formu : INT n

n; İşlem türlerini gösteren Vektör sıra numarasıdır.

İşlem türleri; 00H'dan 70H'ya kadar değişmektedir.

En çok kullanılan işlem komutları, BIOS KESMELERİNE ait INT 10H ve DOS KESMELERİNE ait INT 21H'dır.

Bu kesme komutlarının etkilediği değişik **FONKSİYONLAR** vardır.

Her bir fonksiyonun yine bir NUMARASI vardır.

İşlenecek **fonksiyona ait NUMARA**, AH veya AL kaydedicisine (genelde AH) kaydedilerek, hemen bir sonraki satıra yazılan INT 10 veya INT 21 komutu ile işleme sokulur.

Aşağıda bunlara ait örnek programlar verilmiştir.

Bu kesmelerin kullanılacağı makinaların IBM uyumlu olmaları gereklidir.

Her mikroişlemcinin kendisine has BIOS ve DOS komutları vardır. Fakat çoğunluğu ortak kullanılmıştır.

Kesme komutları INT ile başlamaktadır ve daha sonra sıra numarası gelmektedir.

Kesme numaraları HEXADECIMAL olarak verilmiştir. Bu nedenle sonlarına "H" harfi konmaktadır.

Bellekten bu sıra numaraları ile çağrılmaktadır.

En çok kullanılan KESME KOMUTU olan INT 10H Ekran Kontrolü Yapan Bir Kesmedir.

Değişik EKRAN İŞLEMLERİNİN yaptırılabilmesi için, değişik **fonksiyon** numaraları kullanılır. Bu numaraların kaydı için "AH" kaydedicisi kullanılır. Bu nedenle numaralar AH olarak tanımlanır. AH=0, AH=1 gibi.

Her fonksiyona bağlı ikincil fonksiyonlar vardır. Bunlara ait giriş bilgileri de, önceden belirlenmiş olan kaydedicilere yüklenir.

Aşağıdaki TABLODA, INT 10H ile ilgili olarak "3.B.4.3 – KESME KOMİTLARI" bölümünde özet olarak verilmiş olan FONKSİYONLAR ve bunlarla ilgili KAYDEDİCİLERİN hangi İŞLEMLER için kullandıkları verilmiştir.

INT 10H EKRAN KONTROL FONKSİYONLARI

AH No	Fonksiyonu	Giriş	Cıkış
EKRANIN GÖRÜNTÜ KONTROLU (Interface Control of the CRT)			
AH=0	Set the mode of display (Ekran modu düzenleme)	AL= 0 AL= 1 AL= 2 AL= 3 AL= 4 AL= 5 AL= 6 AL= 10	40 X 25 B/W 40 X 25 color 80 X 25 B/W 80 X 25 color 320 X 200 graphics color 320 X 200 graphics B/W 640 X 200 graphics B/W 640 X 530 graphics E.G.A.
AH= 1	Set cursor type (kürsör tipi düzenleme)	CH =	Bits 4-0 start of line for cursor (kürsör başlama)
		CL =	Bits 4-0 end of line for cursor (kürsör bitiş satırı)
AH= 2	Set corsur position (kürsör poz. düzenleme)	DH = DL = BH =	Row (satır) Column (sütun) Page number of display (ekran sayfa No)

AH= 3	Read cursor position (kürsör poz. okuma)	DH = DL = CH = CL = BH =	Row Columun Cursor mode (kürsör modu) Cursor mode (kürsör modu) Page number of display (ekran sf.no)
AH= 4	Get light pen position (İşik kalemi poz. alma)	AH = 0	Switch not down/triggered (çalışmıyor)
		AH = 1	Valid answers: follow (çalışıyor)
		DH =	Row
		DL =	Column
		CH =	Graph line (grafik satırı) (0-99)
		BX =	Graphic columun (graf. sütunu) (0-319/639)
AH= 5	Set active display page (aktif ekran sayfa. düz.)	AL =	New page value (yeni sayf.değeri) Modes 0 and 1 (mod 0 ve 1) (0-7) Modes 2 and 3 (mod 2 ve 3) (0-3)
AH= 6	Scroll active page up (aktif sayfayı. yukarı kaydır)	AL = CH = CL = DH = DL = BH =	Number of lines (Kayacak satır sayısı) Row, upper left corner (sol üst köşe satır no) Column upper left corner (Sol üst köşe sütün no) Row, lower right corner (Sağ alt köşe satır no) Column, lower right corner (Sağ alt köşe sütün no) Attribute to be used (Kullanım alanı özelliği)
AH= 7	Scroll active page down (ak. sayf. aşağı kayd.)	AL = CH = CL = DH = DL = BH =	Number of line Row, upper left corner Column, upper left corner Row, lower right corner Columun, lower right corner Attribute to be used

KARAKTERLERİN İŞLENMESİ

(Handling Characters)

AH= 8	Read attribute/character at cursor position (kürsördeki kar-ASCII- oku)	BH = AL = AH =	Display page (ekr. sayf.) Character read (kar. okuma) Attribute of character (kar. türü)
AH= 9	Write attribute/characater at cursor position (kürt. karekt. görüntüleme)	BH = CX = AL = BL =	Display page (ekran sayf) Count of characters to write (yazılacak karakter sayısı) Character to write (yazılacak kar.) Attribute of character (kar. özelliği)

AH= 10 (0AH)	Write character at cursor position (kür. kar. görüntüleme)	BH = CX =	Display page (AH9 gibi) Count of characters to write
GRAFİK DÜZENLEME (Graphics Interface)			
AH= 11 (0BH)	Select color palette (renk seçme)	BH = BL =	Palette ID (palet no) (0-127) Color for palette ID (palet rengi): 0 – Background (fon) (0-15) 1 – Palette (palet) 0 – Green (yeşil) (1), red (kırmızı) (2) yellow (sarı) (3) 1-Cyan (1), magenta (2) white (bez) (3)
AH= 12 (0CH)	Draw dot on screen (piksel görüntüleme)	DX = CX = AL =	Row (0-199) Column (0-319/639) Color of dot (piksel rengi)
AH= 13 (00H)	Read dot information (pikseli okuma)	DX = CX = AL =	Row (0-199) Column (0-319/639) Value of dot (piksel değeri)
UZAK YAZIM ÇIKIŞI (ASCII Teletyp Output)			
AH= 14 (0EH)	Write to active page (aktif sayfa yazma)	AL = BL =	Character to write (kar. yazımı) Foreground color (ön zemin rengi)
AH= 15 (0FH)	Video state returned (Sayfa görüntüleme)	AL = AH = BH =	Current mode (geçerli mod) Number of screen columuns (Ekran sütun sayısı) Current display page (geçerli ekran sayfası)
AH= 16 (10H)	Reserved (yedek)		
AH= 17 (11H)	Reserved (yedek)		
AH= 18 (12H)	Reserved (yedek)		
AH= 19 (13H)	Write string (Dizin görüntüleme)	ES:EP = CX = DX = BH = AL = 0	Point to string (dizin noktası) Length of string (dizin uzunluğu) Cursor position for start (start için kürsör poz.) Page number (sayfa No) BL = attribute (özellik) (char,char,char...char)
		AL = 1	Cursor not moved (kürsör hareketsız) BL = attribute (char, char, char...char)
		AL = 2	Cursor is moved (kür. hareketli) (char,attr,char,attr...)
		AL = 3	Cursor not moved (char,attr,char,attr...)
			Cursor is moved

Not : 1 – Ek bilgi için "IBM Technical Reference Manual for the AT computer"e bakılabilir.

2 – Program yazılrken burada verilen kaydediciler ve fonksiyonlar kullanıldıktan sonra INT 10H kesme komutu verilerek gerekli EKRAN İŞLEMLERİ yaptırılacaktır.

BIOS KESMELERİ İLE İLGİLİ ÖRNEK PROGRAMLAR:

1 – EKRAN SİLME

Yeni bir programa başlamadan ekranı silmek gerekir.

Silme, "Assembly'de DOS CLS (clear screen) ile yapılamaz. Bunun için, ya "10H BIOS SİLME" programı ile ekran kaydırılarak silinir. Ya da direk ekran giriş'i programlanarak silinir. Burada her iki örnekte verilecektir.

1. a–

**;8088/80286/80386/80486 MİKROİŞLEMÇİLERİ İÇİN
;BIOS INT 10H KESMESİ İLE EKRAN SİLME PROGRAMI**

```
STACK SEGMENT PARA 'DATA'
        DB      64 DUP ('MYSTACK')
STACK ENDS
MYCODE SEGMENT PARA 'DATA'
MYPROC PROC FAR
ASSUME CS:MYCOD, DS:MYDATA, SS:STACK
PUSH    DS
SUB     AX,AX
PUSH    AX
```

;RENKLİ EKRANI SİLMEK İÇİN GEREKLİ KOD

MOV CX,0000	;ROW, COLUMN UPPER LEFT CORNER (Satır, sütun üst sol köşe)
MOV DX,2479H	;ROW, COLUMN LOWER RIGHT CORNER (Satır, sütun alt sağ köşe)
MOV BH,07	;NORMAL ATTRIBUTE (Normal özellik)
MOV AH,06	;SCROLL ACTIVE PAGE UP (Aktif sayfayı yukarı kaydır)
MOV AL,00	;SCROLL ENTIRE WINDOW (Tüm ekranı kaydır)
INT 10H	;CALL BIOS INTERRUPT (BIOS kesmeyi çağır)

;EKRANI SİLME YORDAMININ SONU

```
RET
MYPROC ENDP
MYCODE ENDS
END
```

1.b –

;8088/80286/80386/80486 (RGB-RED, GREEN, BLUE- MONİTÖRLÜ İÇİN)
;BIOS KESMESİ KULLANMADAN RENKLİ EKRANI SİLME

```
STACK SEGMENT PARA STACK
      DB      64 DUP ('MYSTACK')
STACK ENDS
MYCODE SEGMENT PARA 'DATA'
MYPROC PROC FAR
ASSUME CS:MYCODE, DS:MYDATA, SS:STACK
PUSH    DS
SUB     AX,AX
PUSH    AX
```

;RENKLİ EKRANI SİLMEK İÇİN GEREKLİ KOD

```
MOV     AX,0B800H ;POINT OF ENTRY TO COLOR RAM
                  ;(Renkli RAM'a giriş)
MOV     ES,AX  ;NOVE VALUE TO ES
                  ;(Değeri ES'ye yükle)
MOV     DI,00H ;STARTING ADDRESS
                  ;(Başlama adresi)
MOV     AL,00H ;CHARACTER TO PRINT
                  ;(Görüntülenecek karakter)
MOV     AH,07H ;SET NORMAL ATTRIBUTE
                  ;(Normal öz. düz.)
MOV     CX,7D0H ;WRITTE 2000 TIMES
                  ;(2000 defa yaz)
REP    STOSW ;DO IT (Onu yap)
```

;EKRANI SİLME YORDAMININ SONU

```
RET
MYPROC ENDP
MYCODE ENDS
END
```

**2 – ;80286/80386/80486 MİKROİŞLEMCİLERİ İÇİN
;BIOS KESMESİ KULLANARAK SAAT VE TARİHİ GÖRÜNTÜLEME
PROGRAMI**

```
STACK SEGMENT PARA STACK
      DB      64 DUP ('MYSTACK')
STACK ENDS
MYDATA SEGMENT PARA 'DATA'
TIME   *DD      0H ;STORAGE FOR TIME AS DOUBLE
                  ;WORD
                  ;(Saat için iki kelimelek yer ayırımı)
```

DATE DD 0H ;STORAGE FOR DATE AS DOUBLE WORD
 (Tarih için iki kelime lik yerayırmı)

MYDATA ENDS
 MYCODE SEGMENT PARA 'CODE'
 MYPROC PROC FAR
 ASSUME CS:MYCODE,DS:MYDATA,SS:STACK
 PUSH DS
 SUB AX,AX
 PUSH AX
 MOV AX,MYDATA
 MOV DS,AX

;BIOS KESMESİ İLE SAAT YORDAMI

MOV	AH,02	;PARAMETER FOR TIME (Saat parametresi)
INT	1AH	;TIME/DATA INTERRUPT (Saat / tarih kesmesi)
MOV	BYTE PTR TIME+3,DL	;MOVE SECONDS INTO TIME (Time'a saniyeyi gir)
MOV	BYTE PTR TIME+2,DH	;MINUTES INTO TIME (Time'a dakikayı gir)
MOV	BYTE PTR TIME+1,CH	;HOURS INTO TIME (Time'a saatü gir)

;BIOS KESMESİ İLE ZAMAN YORDAMI

MOV	AH,04	;PARAMETER FOR DATE
INT	1AH	;TIME/DATA INTERRUPT
MOV	BYTE PTR DATE+3,DL	;MOVE DAY INTO DATE (günü gir)
MOV	BYTE PTR DATE+2,DH	;MONTH INTO DATE (ayı gir)
MOV	BYTE PTR DATE+1,CH	;YEAR INTO DATE (yıtı gir)
MOV	BYTE PTR DATE,CH	;CENTRUY INTO DATE (asırı gir)
RET		;RETURN CONTROL TO DOS
MYPROC	ENDP	;END PROCEDURE NAMED MYPROC
MYCODE	ENDS	;END CODE SEGMENT NAMED MYCODE
END		;END WHOLE PROGRAM

"INT 1AH" kesmesi ile, AH içeriğine göre şu işlemler yapılabilmektedir:

AH=0: set edilen saat okunur, **AH=01:** saat kaydedilir,

AH=02: saat okunur, **AH=03:** Saat, dakika, saniye kaydedilir,

AH=04: tarih okunur, **AH=05:** tarih kaydedilir, **AH=06:** alarm kaydedilir (23:59:59'a kadar), **AH=07:** alarm reset edilir.

Bu işlemler için aşağıdaki kaydediciler kullanılmaktadır:

Saat okumak için "AH, 02" ile kodu çözülen kaydediciler		
DH	Saniye	(00...59)
CL	Dakika	(00...59)
CH	Saat	(00...23)

Tarihi okumak için "AH,04" ile kodu çözülen kaydediciler		
DL	Gün	(1...31)
DH	Ay	(1...12)
CL	Yıl	(1994...)
CH	Yüzyıl	(20...)

Kodu çözülen bilgiler, BYTE PTR komutu kullanılarak, "TIME" (saat) ve "DATE" (tarih) değişkeni adı ile, "DD" tarafından bellekte ayrılmış olan yere yerleştirilir.

DD (Defined double word) bir PSEUDO işlemcisi olup, işlevi, tanımı yapılan bir değişken için bellekte 32 bit'lik bir yer ayırmaktır.

Saat (TIME) ve Tarih (Data), her ikisinin de görüntülenebilmesi için, "DEBUG-D" komutundan yararlanılır.

3 – ;80286/80386/80486 MİKROİŞLEMCİLERİ İÇİN

;BIOS KESMESİ KULLANILARAK BELLEK KAPASİTESİNİN BELİRLENMESİ

```

STACK SEGMENT PARA STACK
        DB      64 DUP ('MYSTACK')
STACK ENDS

MYDATA SEGMENT PARA 'DATA'
SYSMEM DW      ?
EXTMEM DW      ?
;ANA BELLEK (640K'ya kadar)
;UZATILMIŞ BELLEK (640K'nın üstü)
MYDATA ENDS

MYCODE SEGMENT PARA 'CODE'
MYPROC PROC FAR
        ASSUME CS:MYCODE, DS:MYDATA, SS:STACK
        PUSH  DS
        SUB   AX,AX
        PUSH  AX
        MOV   AX,MYDATA
        MOV   DS,AX

```

;ANA BELLEK KAPASİTESİNİ BELİRLİYEN YORDAM

INT	12H	;INTERRUPT PARAMETER (Kesme parametresi)
MOV	SYSTEM,AX	;MOVE VALUE TO MAIN MEMORY VAR (Değeri ana bellek VAR'a yükle)

;UZATILMIŞ BELLEK KAPASİTESİNİ BELİRLİYEN YORDAM

MOV	AH,88H	;INTERRUPT PARAMETER (Kesme parametresi)
INT	15H	;INTERRUPT CALL FOR EXT MEMORY (Dış bellek için kesme çağrıısı)
MOV	EXTMEM,AX	;AX'ı dış belleğe yükle
RET		

MYPROC ENDP
MYCODE ENDS
END

4 - BIOS KESMESİ KULLANILARAK ORTA REZOLÜSYONLU RENKLİ EKRANA NOKTA ÇIKARILMASI

Orta ve yüksek olmak üzere iki rezolutyon tipi vardır. Orta rezolutyonda, ekrana 4 renkli çapraz 320, aşağıya doğru 200 nokta çıkarılabilir. Yüksek rezolutyonda bu değerler 640 ve 200 dır. Daha önce verilmiş olan INT 10H tablosunda grafik değerler AH11, 12, 13 olarak verilmiştir.

8088/80286/80486 (-RGB MONİTÖRLÜĞÜN)**BIOS KESMESİ KULLANARAK ORTA REZOLÜSYONLU RENKLİ EKRANA NOKTA ÇIKARILMASI PROGRAMI**

(Renkli Grafik kartının kullanılması gerekmektedir)

STACK	SEGMENT PARA 'DATA'	STACK
	DB 64 DUP ('Mystack')	
STACK	ENDS	
MYCODE	SEGMENT PARA 'DATA'	
MYPROC	PROC FAR	
	ASSUME CS:MYCODE,SS:STACK	
	PUSH DS	
	SUB AX,AX	
	PUS AX	

;EKRAN MERKEZİNE YAKIN ÜÇ NOKTANIN ÇIKARILMASI İÇİN YORDAM:

MOV	AH,00	;PREPARE TO SET SCREEN MODE (Ekran modunu hazırla)
-----	-------	---

MOV	AL,04X	;SET 320X200 COLOR MODE (320X200 renk modunu düzenle)
INT	10H	;CALL INTERRUPT (Kesmeyi çağır)
MOV	AH,11	;SET COLOR PALETTE (Renk paletini düzenle)
MOV	BH,00	;SET BACKGROUND COLOR (fon rengini düzenle)
MOV	BL,01	;SET IT TO BLUE (maviye düzenle)
INT	10H	;CAL INTERRUPT (kesmeyi çağır)
MOV	AH,11	;SET COLOR PALETTE (renk paletini düzenle)
MOV	BH,01	;SELECT FOREGROUND PALETTE (ön zemin paletini seç)
MOV	BH,00	;GREEN/RED/YELLOW (yeşil/kırmızı/sarı)
INT	10H	;CALL INTERRUPT (kesmeyi çağır)
MOV	AL,02	;SET DOT COLOR TO RED (kırmızı noktayı düzenle)
MOV	AH,12	;WRITE DOT PARAMETER (nokta par. yaz)
MOV	DX,64H	;SET FOR 100 ROWS DOWN (VERTI- CAL) (düsey 100 satır için düzn.)
MOV	CX,9EH	;SET FOR 158 COLUMNS OVER (HORI- ZONTAL) (158 sütun için düzn.)
INT	10H	;CALL INTERRUPT (kesmeyi çağır)
MOV	AH,12	;WRITE DOT PARAMETER (nokta parametresini yaz)
MOV	CX,0A0H	;PLOT ANOTHER (Diğer bir nokta çıkarması)
INT	10H	;CALL INTERRUPT (kesmeyi çağır)
MOV	AH,12	;WRITE DOT PARAMETER (nokta par. yaz)
MOV	CX,0A2H	;PLOT ANOTHER (diğer bir nokta çıkış)
INT	10H	;CALL INTERRUPT (kesmeyi çağır)
RET		
MYPROC	ENDP	
MYCODE	ENDS	
	END	

5 - ;BIOS KESMESİ KULLANILARAK YÜKSEK REZOLÜSYONLU EKRANA ÇİZGİ ÇİZİLMESİ

;8088/80286/80386/80486 (RGB MONİÖRLÜ İÇİN)

(Renkli Grafik kartının kullanılması gerekmektedir)

```
STACK SEGMENT PARA STACK
DB      64 DUP ('MYSTACK')
STACK ENDS
```

```
MYCODE SEGMENT PARA 'CODE'
MYPROC PROC FAR
        ASSUME CS:MYCODE,SS:STACK
        PUSH DS
        SUB AX,AX
        PUS AX
```

;BİR ÇİZGİ ÇİZMEK İÇİN YORDAM

	MOV AH,00	;PREPARE TO SET SCREEN MODE (Ekran mod. hazırl.)
	MOV AL,06	;SET 640X200 GRAPHICS MODE (640X200 grafik mod. hazırl.)
	INT 10H	;CALL INTERRUPT (kesmeyi çağır)
	MOV AL,01	;SET DOT COLOR (nokta rengini düzenle)
	MOV AX,00	;SET VERTICAL START TO LEFT (sola düşey startı düz)
	MOV CX,00	;SET HORIZONTAL START TO TOP (yatay startı düz)
AGAIN: (tekrar)	MOV AH,12	;WRITE DOT PARAMETER (nokta par.yaz)
	INT 10H	;CALL INTERRUPT (kesmeyi çağır)
	INC DX	;INCREMENT VERTICAL POSITION (düşey poz. 1 artır)
	INC CX	;INCREMENT HORIZONTAL POSITION (Yatay poz. 1 artır)
	INC CX	
	CMP DX,0C8H	;IS LINE AT SCREEN BOTTOM (200)? (satır sonumu?)
	JE ENDO	;IF SO, END PROGRAM (evet ise bitir)
	JMP AGAIN	;IF NOT, PLOT ANOTHER DOT (Hayır ise, diğer bir noktayı çıkart)

ENDO:

```
MYPROC ENDP
MYCODE ENDS
END
```

4.B.2.2.2 – DOS KESMELERİ

DOS kesmeleri ile de, BIOS kesmelerinde olduğu gibi bilgisayardan daha etken şekilde yararlanılması mümkün olabilmektedir.

DOS şu cümlenin kısaltılmıştır:

Disk Operatign System (Disk İşletim Sistemi)

Disk İşletim Sisteminin donanım elemanları olan DİSK ve DİSKET, RAM belleği takviye eden YARDIMCI BELLEK'tir. Bunlara kaydedilen programlarla hem bellek kapasitesi büyütülmekte, hem de bilgisayarın çok yönlü ve daha verimli işlem yapması sağlanmaktadır.

"DOS KESMELERİ" de bu işlemlerden ASSEMBLY DİLİ ile ilgili olan bölümünü oluşturmaktadır.

DOS KESMELERİ'de "INT" komutu ile sağlanır.

DOS KESMELERİNDEN **en geniş uygulama alanı olanı "21H" Nolu, FONKSİYON ÇAĞIRMA** (Function Call) kesmesidir.

"21H" DOS kesmesi, DOS işletim sistemine bir işlem yaptırırmak ve **DOS fonksiyonlarından** yararlanmak için kullanılmaktadır. Gerekli ara işlemler tamamlandıktan sonra tekrar asıl programa dönülerek kalınan yerden devam edilmektedir.

DOS fonksiyonlarının özet listesi "**3.B.4.3 – KESME KOMUTLARI**" bölümünde verilmiştir.

Bu fonksiyonlar "AH" veya "AL" kaydedicisine "MOV" komutu ile kaydedilir ve "INT 21H" komutu ile işleme sokulur.

Bu bölümde BIOS ve DOS kesmeleri ile ilgili örnek programlar verilecektir.

Başlıca DOS kesmeleri şunlardır:

20H Programı sonlandırma

21H Dos fonksiyonlarını çağrıma

22H İşlemi bitirme ve adresine aktarma

23H Ctrl+C kontrolü

24H Hata kontrolu

25H Disk veya Disketten belleğe kayıt

26H Disk veya Diskete bellekten kayıt

27H Sonlandırma ve belleğe kayıt

28H –3FH arası yedek (rezery)

Aşağıdaki TABLODA, INT 21H ile ilgili olarak "3.B.4.3 – KESME KOMUTLARI" bölümünde özet halinde verilmiş olan FONKSİYONLAR ve bunlar ile ilgili kaydedicilerin hangi işlemler için kullanıldıkları verilmiştir.

INT 21H FONKSİYONLARI			
AH N0	Fonksiyonu	Giriş	Çıkış
AH = 1	Bekle ve Klavye karakterini CTRL-BREAK çeki ile görüntüle	=	AL-Girilen karakter
AH = 2	"Bekle" hariç, "AH=1" gibi	DL =	Görüntülenecek karakter
AH = 3	Asenkron karakter girişi		AL- Girilen kar.
AH = 4	Asenkron karakter çıkışı	DL =	Gönderilecek kar.
AH = 5	Yazılan karakter	DL =	Yazılan kar.
AH = 6	Klavyeden kar. girişi	DL = OFFH	Girilen Kar. (0 değilse)
AH = 7,8	AH = 1 gibi (Görüntü yok)		AL – Girilen karakter
AH = 9	Dizin görüntüleme	DS:DX =	Dizin adresi, \$ ile sonlanmalı
AH = A	Klavye dizini "buffer"a	DS:DX =	Buffer adresi. 1.byte= boyut, 2. byte= okunan kar. sayısı
AH = B	Klavye durumu girişi		AL– Karakter yok = OFFH
AH = C	Klavye "buffer"ını boşalt ve fonksiyonu çağır	AL =	1,6,7,8,0A- fonksiyon No
AH = D	Disk sürme (reset)	Yok	Yok
AH = E	Disk sürme (seçimli)	DL =	AI - "Sürücü No 0= A, 1 = B
AH = 19	Sürücü kodu (default)		AL - 0=A, 1-B, 2 = C
AH = 25	Kesme (set)	DS:DX=	Kesme vektörü adresi
		AL =	Kesme numarası
AH = 2A	Tarih (okuma)		CX= Yıl, DH = Ay D1= gün
AH = 2B	Tarih (set etme)	CX:DX	Yukarıdaki gibi, AL=0 iken
AH = 2C	Saat (okuma)		CH-Saat CL-Dakika
AH = 2D	Saat (set etme)	CX:DX	Yukarıdaki gibi AL = 0 iken
AH = 2E	Durum kontrolu (set)	DL=	0
		AL =	0 (0ff), 1 (on)
AH = 35	Kesme adresi (okuma)	AL =	Kesme No, ES:BX- (Vekt ads)
AH = 36	Disk boşluğu (uygun)	DL =	Sürücü (0-öz, 1-A, 2-B v.s.)
AH = 39	Kütük oluşturma	DS:DX =	Kütük kaydolacak dizini adresi

INT 21H FONKSİYONLARI (devam)			
AH N0	Fonksiyonu	Giriş	Çıkış
AH = 3A	Kütük silme	DS:DX =	Kütükteki dizin adresi
AH = 3B	Kütüğün değiştirilmesi	DS:DX =	Yeni kütük için dizin adresi
AH = 3C	Dosya yaratma (file creat)	DS:DX = CX =	Dosya için dizin adresi Dosyanın özelliği
AH = 3D	Dosya açma (file open)	DS:DX= AL =	Dosyaya girecek dizin ad. 0 – Okuma, 1 – Yazmak için açık, 2 – Her ikisi, AX - Dönüş
AH = 3E	Dosya kaydı (kapama)	BX =	Dosya
AH = 3F	Dosya veya devre (okuma)	BX = CX = DS:DX	Dosya Okunacak byte sayısı Buffer adresi AX – Okunan byte sayısı
AH = 40	Dosya veya devre (yazma)	BX = CX = DS:DX =	Dosya kaydı Yazılacak byte sayısı Yazılacak VERİ ads. AX- yazılan byte sayısı
AH = 41	Dosya (silme)	DS:DX =	Dizin dosyasının adresi
AH = 42	Dosya (özellik alma)	CX:DX = AL = 0 AL = 1	Byte olarak offset byte sayısı göstericisi "mevcut yer + offset göst.
AH = 43	Dosya (özellik verme)	DS:DX =	Dosya dizinin adresi AL = 0 ise, özellik CX'a, AL = 1 ise, CX'dan dosyaya geçiş olur.
AH = 47	Mevcut kütük	DL = DS:SI =	Sürücü No (0-öz, 1=A 2-B) Buffer adresi DS:SI- Dizinin dönüş adresi
AH = 54	Doğrulama hali	yok	AL - 0 doğrulama yok AL – 1 doğrulama var.
AH = 56	Dosya (yeniden adlandırma)	DS:DX = ES:DI =	Eksi bilgi için dizin adresi Yeni bilgi için dizin adresi

DOS KESMELERİ İLE İLGİLİ ÖRNEK PROGRAMLAR:

1 – DOS INT 21H KESMESİ İLE KAYDEDİCİ İÇERİĞİNİN EKRANDA GÖRÜNTÜLENMESİ

Program uygulaması olarak küçük bir aritmetik işlem yapalım. 01234H ile 02299H'yi toplayıp sonucu geçici bir kaydediciye kaydedelim.

Toplama sonucu olan "34CDH"nin kaydedileceği geçici kaydedicinin değişken adı "TMPNUM" olsun.

"TMPNUM" 16 bit'lik dört adet Hex rakamı taşıyan bir kaydediciyi göstermektedir. DOS INT 21H ile ekranda görüntü alabilmek için TMPNUM içeriğinin, bir ASCII dizini haline getirilmesi gereklidir.

ASCII karakterleri "TMPCHAR" dizin değişkenine yüklenir.

Bu durum karakterlerin, VERİ SEGMANINA yüklenmesi demektir.

INT 21H, VERİ SEGMANINDAKİ bilgilerin ekranda görüntülenmesini sağlar. Bu işlem "\$" işaretini görülmeye kadar devam eder.

Bu işlemler için hazırlanan Program aşağıda görüldüğü gibi olacaktır:

;8088/80286/80486 MİKROİŞLEMÇİLERİ İÇİN

;DOS INT 21H KESMESİ İLE KAYDEDİCİ İÇERİĞİNİN EKRANDA GÖRÜNTÜLENMESİ

STACK	SEGMENT	PARA	STACK
		DB 64 DUP	(MYSTACK')
STACK	ENDS		
MYDATA	SEGMENT	PARA 'DATA'	
TMPNUM	DW	?	:STORAGE FOR REG TO BE DISPLAYED (Görüntülenecek kaydediciye depolama)
TMPCHAR	DB	8 DUP (' '), \$:CHARACTER DATA FROM REG (Kaydediciden VERİ karakteri)
MYDATA	ENDS		
MYCODE	SEGMENT	PARA 'CODE'	
MYPROC	PROC	FAR	
	ASSUME	CS:MYCODE,DS:MYDATA,SS:STACK	
	PUSH	DS	
	SUB	AX,AX	
	PUSH	AX	
	MOV	AX,MYDATA	
	MOV	DS,AX	

;TOPLAMA İŞLEMİ

MOV	AX,001234H	;FIRST NUMBER (1. sayı)
ADD	AX,02299H	;SECOND NUMBER (2. sayı)
MOV	TMPNUM,AX	;STORE SUM IN TEMP LOCATION (Toplamı geçici yerine yükle)

;16 BİT KAYDEDİCİNİN EKRANDA GÖRÜNTÜLENMESİNİ SAĞLIYACAK OLAN
PROGRAM BÖLÜMÜ

AGAIN: (tekrar)	MOV	CX,04H	;NUMBER OF DIGITS TO CONVERT (çevrilecek digit adedi)
	MOV	AX,TMPNUM	;GET 16 BITS OF DATA INTO AX (16 bit veriyi AX'a al)
	AND	AX,0FH	;KEEP ONLY LOWER 4 BITS (en düşük 4 bit'i al)
	ADD	AL,30H	;MAKE IT ASCII (onu ASCII'ya çevir)
	CMP	AL,39H	;IS IT A LETTER (O bir harf mi)?
JL	ASCII	;IF A NUMBER, JUMP TO ASCII (sayı ise ASCII'ya dallan)	
ASCII:	ADD	AL,07H	;MAKE LETTER CORRECT ASCII CHAR (harf ise ASCII karakterine dönüştür)
	MOV	SI,CX	;SET INDEX EQUAL TO CHARACTER (indeksi karaktere eşitle)
	TMPCHAR[SI], AL		
	MOV		;SAVE CONVERTED NUMBER AS CHARACTER (çevrilen sayıyı karakter olarak sakla)
	ROR	TMPNUM,1	;ROTATE NEXT DIGIT TO LSB POSITION (Bir sonraki basamağı LSB pozisyonuna döndür.)
ROR	TMPNUM,1		
ROR	TMPNUM,1		
LOOP	AGAIN	;IF CX>0, THEN ANOTHER DIGIT (eğer CX>0 ise, diğer basamağa geç)	

;OLUŞAN DİZİNİ GÖRÜNTÜLEYECEK YORDAM

LEA	DX,TMPCHAR	;DOS STRING DISPLAY ROUTINE (DOS dizini ekran yordamı)
MOV	AH,9	;DOS PARAMETER (DOS parametresi)
INT	21H	;DOS INTERRUPT (DOS kesmesi)
RET		;RETURN CONTROL TO DOS (DOS'un kontroluna dön)
MYPROC	ENDP	
MYCODE	ENDS	
	END	

2 – DOS INT 21H KESMESİ İLE BİR KLAVYE KARAKTERİNİN OKUNMASI

Programdaki, Y = YES (evet), N = NO (hayır) anlamındadır. Eğer karakter bir "Y" veya "N" ise bazı değerler, "TEST" olarak gösterilen değişkene yüklenmektedir.

;8088/80286/80386/80486 MİKROİŞLEMÇİLERİ İÇİN

;DOS INT 21H KESMESİ İLE BİR KLAVYE KARAKTERİNİN OKUNMASI PROGRAMI

```

STACK      SEGMENT PARA 'DATA'           STACK
          DB 100 DUP ('MYSTACK')
STACK      ENDS
MYDADA    SEGMENT PARA 'DATA'
TEST      DW ?
MYDATA    ENDS

MYCODE    SEGMENT PARA 'CODE'
MYPROC    PROC FAR
          ASSUME CS:MYCODE,DS:MYDATA,SS:STACK
          PUSH DS
          SUB AX,AX
          PUSH AX
          MOV AX,MYDATA
          MOV DS,AX

```

;KLAVYE KARAKTERİNİN OKUNMASI VE GÖRÜNTÜLENMESİ

MOV	AH,01H	;PARAMETER FOR READING A CHARACTER (Bir karakteri okuma parametresi)
INT	21H	;DISPLAY ONE CHARACTER (Bir karakteri görüntüle)
CMP	AL,'Y'	;IS IT A "Y" (o bir "Y" midir)
JNE	HERE	;IF NOT, JUMP TO "HERE" (Eğer değilse, "HERE"e dallan)
MOV	BX,9999H	;IF YES, MOVE A 9999H INTO BX (Eğer "EVET" ise BX'a 9999H'yi yükle)
JMP	ENDO	;THEN END PROGRAM (o zaman "ENDO" ya dallan)
HERE:	CMP AL,'N'	;IS IT A "N" (O bir "N" midir)
JNE	ENDO	;IF NOT, JUMP TO "ENDO" (Değilse "ENDO" ya dallan)
MOV	BX,5555H	;IF YES, MOVE A 5555H INTO BX (EVET ise, BX'a 5555H'yi yükle)
ENDO:	MOV TEST,BX	;SAVE THE VALUE IN TEST (Değeri "TEST"e aktar)
	RET	
MYPROC	ENDP	
MYCODE	ENDS	
	END	