

ج) مکان نما را به سطر ۱۰ ستون ۲۰ ببرد.

د) پیغام های زیر را روی مانیتور نمایش دهد:

MY NAME IS :

WHAT IS YOUR NAME :

```
PAGE 100,120
TITLE 'TEST_2.ASM'
.MODEL SMALL
.STAK DW 32 DUP(0)
.DATA
MESSAGE DB 'MY NAME IS :'
          'WHAT IS YOUR NAME :' , '$'
.CODE
START PROC FAR
    MOV AX,@DATA
    MOV DS,AX
;-----
    MOV AH,07H
    MOV AL,00
    MOV BH,16H
    MOV CX,0000
    MOV DX,184FH
    INT 10H
;-----
    MOV AH,00
    MOV AL,1
```

```

        INT 10H
;-----
        MOV AH,02
        MOV BH,00
        MOV DH,10
        MOV DL,20
        INT 10H
;-----
        MOV BH,09H
        LEA DX,MESSAGE
        INT 21H
;-----
        MOV AH,4CH
        INT 21H
START ENDP
        END START

```

مثال: برنامه ای بنویسید که ابتدا مانیتور را پاک کند. سپس مکان نما را در مختصات $x=30$ و $y=40$ قرار دهد. سپس عبارت good را روی مانیتور نمایش و سرانجام مکان نما را در سطر بعدی قرار دهد.

پاک کردن مانیتور CLEAR
قرار دادن مکان نما در نقطه مورد نظر SETCUR1
نوشتن عبارت WRITE
مکان نما در سطر بعدی SETCUR2

PAGE 70,100

TITLE 'TEST_B.ASM'

.MODEL SMALL

.STACK DW 128 DUP(?)

```
.DATA
MS DB 'GOOD BYE','$'

.CODE

MAIN PROC FAR

MOV AX,@DATA
MOV DS,AX
MOV ES,AX

CALL CLEAR

CALL SETCUR1

CALL WRITE

CALL SETCUR2

MOV AX,4C00H
INT 21H

MAIN ENDP

;-----
CLEAR PROC NEAR

MOV AX,600H
MOV BH,00
MOV CX,0000
MOV DX,184FH
INT 10H
RET

CLEAR ENDP

;-----
SETCUR1 PROC NEAR

MOV AH,02
MOV DX,281EH
INT 10H
RET
```

```
SETCUR1 ENDP  
;-----  
WRITE PROC NEAR  
MOV AH,09H  
LEA DX,MS  
INT 21H  
RET  
WRITE ENDP  
;-----  
SETCUR2 PROC NEAR  
MOV AH,02  
MOV DL,0AH  
INT 21H  
RET  
SETCUR2 ENDP  
;-----  
END MAIN
```

رسم خطوط با استفاده از سرویس 09H دستور 10H و INT 21H

کدهای اسکی 80H الی FFH جهت حروف و علامات خاصی بکار می روند.

جدول (۱۲-۶) نمایشی کدهای اسکی خطها و علامات

مثال: زیر برنامه ای بنویسید که بتواند یک خط افقی با ۲۰ خط تیره به رنگ زرد با زمینه سفید رسم کند. کد اسکی خط تیره را C4H در نظر بگیرید.

کد اسکی	طرز نمایش روی مانیتور
C4H	—
B3H	
DAH	∩
BFH	┌
C0H	└
D9H	┘
C3H	┐
B4H	┌
C2H	└
C1H	┘
C5H	+
B0H	⋮
B1H	⋮
B2H	⋮
DBH	⋮
D6H	∩
B7H	┌
D3H	└
BDH	┘
C7H	┐
B6H	┌
D2H	└
D0H	┘
D7H	#
CDH	=
BAH	
C9H	∩
BBH	┌
C8H	└
BCH	┘
CCH	┐
B9H	┌
CBH	└
CAH	┘
CEH	+
DFH	■
DDH	■
DEH	■
DCH	■
D5H	∩
B8H	┌
D4H	└
BEH	┘
C6H	┐
B5H	┌
D1H	└
CFH	┘
D8H	+

```
MOV AH,09H
```

```
MOV AL,0C4H
```

```
MOV BL,0EH
```

```
MOV BH,00
```

```
MOV CX,20
```

```
INT 10H
```

مثال: مساله قبل را با دستور INT 21H تکرار کنید.

```
.DATA
```

```
LINE DB 20 DUP ('-'), '$'
```

```
.
```

```
.
```

```
.CODE
```

```
.
```

```
.
```

```
MOV AH,09H
```

```
LEA DX,LINE
```

```
INT 21H
```

مثال: برنامه ای بنویسید که بتواند یک مستطیل با طول ۲۵ کاراکتر و عرض ۸ کاراکتر رسم کند. این مستطیل از سمت چپ ۱۰ کاراکتر با لبه مانیتور فاصله دارد. کد 0DH مکان نما را به سطر بعدی و 0AH مکان نما را به آغاز همان سطر باز می گرداند.

.DATA

```
MOST DB 10 DUP(' '), 0DAH, 25 DUP(0C4H), 0BFH, 0DH, 0AH
      DB 10 DUP(' '), 0B3H, 25 DUP(' '), 0B3H, 0DH, 0AH
      DB 10 DUP(' '), 0B3H, 25 DUP(' '), 0B3H, 0DH, 0AH
      DB 10 DUP(' '), 0B3H, 25 DUP(' '), 0B3H, 0DH, 0AH
      DB 10 DUP(' '), 0B3H, 25 DUP(' '), 0B3H, 0DH, 0AH
      DB 10 DUP(' '), 0B3H, 25 DUP(' '), 0B3H, 0DH, 0AH
      DB 10 DUP(' '), 0B3H, 25 DUP(' '), 0B3H, 0DH, 0AH
      DB 10 DUP(' '), 0C0H, 25 DUP(0C4H), 0D9H, 0DH, 0AH, '$'
```

تمرینات شکل با نوشتن در داخل آنها

❖ کاربرد سرویس های دستور INT 10H (سرویس های BIOS) برای حالت گرافیک مانیتور

در مد گرافیک مکان نما حذف می گردد

تعداد پیکسل ها بسته به وضوح و کارت گرافیک برای انواع مانیتورها متفاوت است .

سرویس 00 دستور int 10H: قرار دادن مانیتور در حالت گرافیک

AH=00

AL=شماره مد

INT 10H

سرویس 0C دستور int 10H: روشن کردن یک پیکسل

AH=0CH

AL=شماره رنگ

CX=(X) شماره ستون

DX=(Y) شماره سطر

BH=شماره صفحه فعال

INT 10H

شماره رنگ		رنگ
هگزا	دسیمال	پیکسل
0	0000	سیاه
1	0001	آبی
2	0010	سبز
3	0011	آبی آسمانی
4	0100	قرمز
5	0101	زرشکی
6	0110	قهوه ای
7	0111	سفید
8	1000	خاکستری
9	1001	آبی کم رنگ
0A	1010	سبز روشن
0B	1011	آبی روشن
0C	1100	قرمز روشن
0D	1101	زرشکی روشن
0E	1110	زرد
0F	1111	سفید پراق

نکته : در مد سیاه و سفید جهت روشن کردن یک پیکسل از مد فوق با $AL=1$ پیکسل روشن و با $AL=0$ پیکسل خاموش می شود .

مثال : پیکسل به مختصات $\begin{cases} 20 \\ 30 \end{cases}$ را با رنگ آبی کم رنگ روشن کنید .

```
MOV AH, 0CH
MOV AL, 09H
MOV CX, 20
MOV DX, 30
MOV BH, 00H
INT 10H
```

مثال : زیر برنامه ای بنویسید که در مد گرافیک بتواند یک خط افقی از مختصات $\begin{cases} 15 \\ 30 \end{cases}$ و به طول ۱۰۰ پیکسل با رنگ قهوه ای رسم کند .

```
MOV CX, 15
MOV DX, 30
BACK : MOV AH, 0C
MOV AL, 06
INT 10H
INC CX
CMP CX, 115
JNZ BACK
```

مثال : زیر برنامه ای بنویسید که در مد گرافیک بتواند یک خط عمودی از مختصات $\begin{cases} 70 \\ 120 \end{cases}$ الی $\begin{cases} 70 \\ 220 \end{cases}$ با رنگ قرمز روشن رسم کند .

```
MOV CX, 70
MOV DX, 120
BACK : MOV AH, 0C
MOV AL, 0C
INT 10H
INC DX
CMP DX, 220
JNZ BACK
```


مثال : زیر برنامه ای بنویسید که در مد گرافیک بتواند ده خط افقی از مختصات $\begin{cases} 15 \\ 40 \end{cases}$ و به طول ۲۰ پیکسل زیر هم با رنگ قهوه ای رسم کند

```
MOV CX, 15
MOV DX, 40
BACK : MOV AH, 0C
MOV AL, 06
INT 10H
INC CX
CMP CX, 35
JNZ BACK
MOV CX, 15
SHL DX, 1
CMP DX, 60
JNZ BACK
```

توضیح برنامه :

جهت رسم یک خط افقی کافیت پیکسل های ستونهای یک ردیف را روشن کنیم (INC CX)

انتهای یک خط به طول ۲۰ پیکسل که از ستون ۱۵ شروع شده است ستون ۳۵ می باشد .

فاصله خطوط موازی از هم به فاصله دو پیکسل دو پیکسل می باشد دستور SHL DX, 1 ثبات DX را در عدد ۲ ضرب می کند .

تمرین : رسم خط ۴۵ درجه با طول و مختصات اختیاری

سرویس 0BH دستور INT 10H : تغییر صفحه نمایش به رنگ آبی

مثال : زیر برنامه ای بنویسید که مانیتور با کارت گرافیکی CGA را در مد گرافیک با وضوح بالا تنظیم کند . سپس تمام صفحه را آبی کند .

```
MOV AH, 00
MOV AL, 06 ; 640*200 High resolution
INT 10H
;-----
```

```
MOV AH, 0BH
MOV BH, 00
MOV BL, 01
INT 10H
```

سرویس 0DH دستور 10H INT : خواندن مشخصات یک پیکسل در مد گرافیک
این سرویس دقیقاً عکس سرویس 0CH می باشد .

AH=0DH

BH= شماره صفحه فعال

CX= ستون

DX= سطر

INT 10H

با اجرای دستورات فوق شماره رنگ پیکسل مورد نظر در ثبات AL ذخیره می گردد .

تمرین ۱ : تابع AH=03 از وقفه INT 10H بکار رفته است . پس از آن نتایج DH=05 و DL=34 بدست آمده است . این به چه معناست ؟

تمرین ۲ : هدف از برنامه زیر برای مانیتور تک رنگ چیست ؟

```
MOV AH, 02
MOV BH, 00
MOV DX, 0000
INT 10H
MOV AH, 09
MOV BH, 00
MOV AL, 2AH
MOV CX, 80
MOV BL, 0F0H
INT 10H
```

تمرین ۳: برنامه زیر کل صفحه نمایش را در کارت CGA پاک می کند . خطاها را اصلاح کنید .

```
MOV AX,0600H
```

```
MOV BH,07
```

```
MOV CX,0000
```

```
MOV DX,184F
```

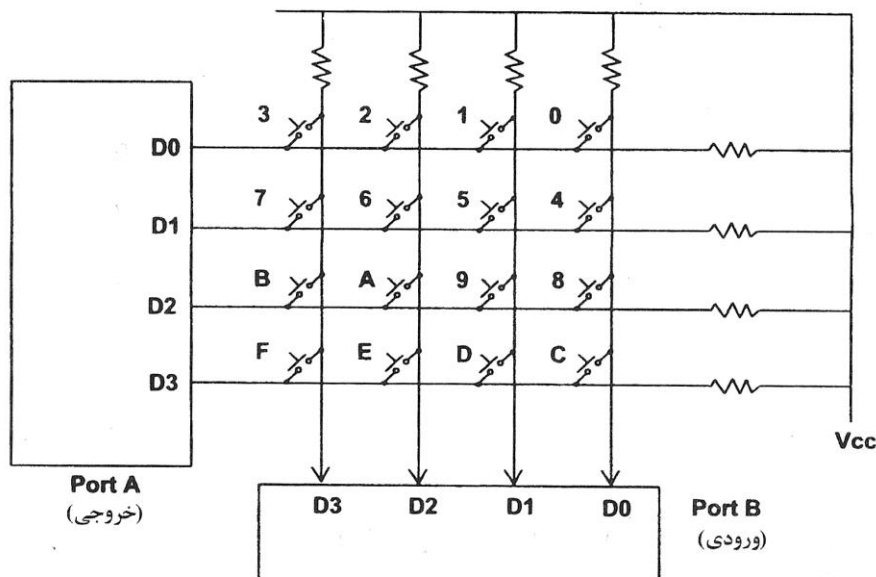
```
INT 10H
```

تمرین ۴:

کاربردهای زبان اسمبلی جهت عملیات بر روی صفحه کلید

مقدمه سخت افزاری

اسکن و شناسایی کلید: شکل زیر یک صفحه کلید ماتریسی 4×4 می باشد که به دو پورت متصل است را نشان می دهد. ریزپردازنده صفحه کلید را دائماً برای تشخیص و شناسایی کلید فشرده شده اسکن می کند.



برای تشخیص کلید فشرده شده ریزپردازنده همه سطرها را صفر کرده و سپس ستون ها را می خواند. اگر داده خوانده شده از ستون $D3-D0=1111$ باشد کلیدی فشرده نشده است.

اگر یکی از بیت های ستون حاوی صفر باشد مثلاً $D3-D0=1101$ کلیدی در ستون $D1$ فشرده شده است. پس از تشخیص فشردن کلید ریزپردازنده وارد فرایند شناسایی کلید می شود. ریزپردازنده ابتدا با صفر کردن سطر $D0$ و یک کردن مابقی سطرها و سپس خواندن ستون ها به دنبال پیدا کردن کلید فشرده شده در سطر می گردد. اگر داده خوانده شده تماماً یک باشد کلید در آن سطر فشرده نشده و فرایند به سطر بعدی منتقل می شود. این عمل تا شناسایی سطری که کلید در آن فشرده شده ادامه می یابد. پس از شناسایی سطر یافتن ستونی که کلید به آن متعلق است کار بعدی می باشد.

مثال: در هر یک از حالات زیر کلید فشرده شده را بیابید. (کدهای زیر کد اسکن می باشند)

الف: $D3-D0=1110$ برای سطر و $D3-D0=1011$ برای ستون

ب: $D3-D0=1101$ برای سطر و $D3-D0=0111$ برای ستون

جواب: کلید ۲ و کلید ۷

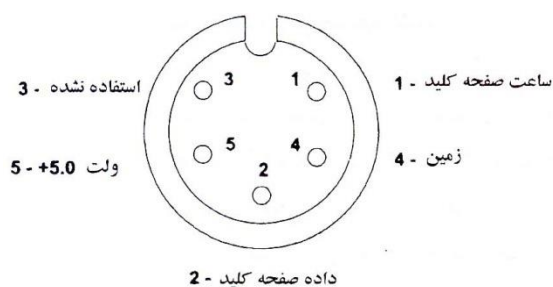
در IBM PC از یک میکروکنترلر (Intel 8042) برای تشخیص و شناسایی کلیدها استفاده می شود. در IBM PC فشردن کلید یک کد اسکن و رها کردن کلید کد اسکن دیگری دارد (کد رها کردن همیشه 128 + کد اسکن کلید فشرده شده است) مثلاً اگر کلیدی دارای کد اسکن 06 باشد و فشرده شود کد قطع آن برابر $06+80H=86H$ خواهد شد.

صفحه کلیدهای امروزی دارای 101 یا 104 کلید هستند:

الف - کلیدهای A-Z و ارقام 0-9 و علاماتی مانند ؟ > # % \$

ب - کلیدهای تابع F1 - F12 بعلاوه Ctrl Shift Alt و کلیدهای Page UP و...

ج - کلیدهای عملیاتی Enter, Esc و ...



رخداده فشردن کلید در IBM PC:

پس از فشردن کلید کد اسکن آن توسط میکروکنترلر صفحه کلید تولید می شود. همزمان بطور خودکار دستور 9 INT فعال می شود. این وقفه کد اسکن کلید فشرده شده را خوانده و کد اسکی لازم را تولید می کند و سپس کد اسکی و کد اسکن را در مکانی از حافظه به نام بافر صفحه کلید که در محدوده داده BIOS قرار دارد ذخیره می کند. سپس می توان بوسیله دستورات INT 16H و INT 21H به این دو کد دست یافت.

ESC 76	F1 05	F2 06	F3 04	F4 0C	F5 03	F6 0B	F7 83	F8 0A	F9 01	F10 09	F11 78	F12 07	↑ E0 75	
~ 0E	1! 16	2@ 1E	3# 26	4\$ 25	5% 2E	6^ 36	7& 3D	8* 3E	9(46	0) 45	-_ 4E	=+ 55	Back Space ← 66	→ E0 74
TAB 0D	Q 15	W 1D	E 24	R 2D	T 2C	Y 35	U 3C	I 43	O 44	P 4D	[{ 54]} 5B	\\ 5D	← E0 6B
CapsLock 58	A 1C	S 1B	D 23	F 2B	G 34	H 33	J 3B	K 42	L 4B	;;: 4C	'' 52	Enter ↵ 5A	↓ E0 72	
↑ Shift 12	Z 1Z	X 22	C 21	V 2A	B 32	N 31	M 3A	,< 41	>. 49	/? 4A	↑ Shift 59			
Ctrl 14	Alt 11	Space 29					Alt E0 11	Ctrl E0 14						

Keyboard Scan Codes

توابع مربوط به INT 21H	توابع مربوط به INT 16H
01H خواندن یک حرف بدون نمایش	00H خواندن یک حرف بدون نمایش
06H تشخیص فشار دادن کلید	01H بررسی فشار دادن کلید
07H ورودی مستقیم از صفحه کلید بدون نمایش آن	02H گزارش وضعیت برخی کلیدها
08H ورودی از صفحه کلید بدون نمایش آن	03H تنظیم نرخ کارکترهای تایپ (سرعت تایپ)
0AH ورودی از بافر صفحه کلید	05H نوشتن در بافر صفحه کلید
0BH تست وضعیت صفحه کلید	10H خواندن کاراکتر از صفحه کلید
0CH پاک کردن بافر صفحه کلید و اجرای عمل ورودی	11H تعیین وجود کاراکتر در بافر
	12H بازگرداندن وضعیت کلیدهای شیفت صفحه کلید

سرویس های دستور INT 21H مربوط به صفحه کلید :

سرویس 01H دستور INT 21H : خواندن یک حرف از صفحه کلید و نمایش آن روی مانیتور (حساس به ctrl+break)

AH=01H

INT 21H

پس از اجرای دستورات فوق کامپیوتر منتظر می ماند تا کلیدی فشار داده شود . به محض فشار کلیدی کد اسکی آن در ثبات AL ذخیره می شود و همزمان حرف مربوطه روی مانیتور نمایش داده می شود . این سرویس با فشار ctrl+break از دستور فوق جلوگیری می کند .

سرویس 06H دستور INT 21H : تشخیص فشار دادن کلید

از این سرویس می توان برای تشخیص اینکه آیا کلیدی فشار داده شده است یا خیر استفاده نمود (بدون نمایش حرف)

AH=06H

DL=FFH

INT 21H

پس از اجرای دستورات فوق اگر کلیدی فشار داده شده باشد کد اسکی آن در ثبات AL ذخیره می شود (ZF=0) ولی اگر کلیدی فشار داده نشده باشد AL=0 (ZF=1) می شود .

سرویس 07H دستور INT 21H : خواندن یک حرف از صفحه کلید بدون نمایش آن روی مانیتور

این سرویس مانند سرویس 01 می باشد اما حرف را نشان نمی دهد معمولاً از این سرویس جهت وارد کردن Password که نباید روی مانیتور نمایش داده شود استفاده می شود . پس از اجرای دستورات زیر کد اسکی کلید فشار داده شده در ثبات AL ذخیره می شود .

```
AH=07H
```

```
INT 21H
```

مثال : برنامه ای بنویسید که ابتدا مانیتور را پاک کند سپس مکان نما را در مرکز صفحه نمایش قرار دهد سپس پیغام "This is a test of the display" را روی مانیتور نمایش دهد سپس اگر کلید M روی کیبورد فشرده شود ابتدا * را با استفاده از سرویس 02 دستور INT 21H نمایش دهد سپس * را با استفاده از سرویس 0EH دستور INT 10H نمایش دهد و سرانجام کامپیوتر بوق بزند .

```
.DATA
```

```
DATAS DB 'This is a test of display','$'
```

```
.CODE
```

```
;-----
```

```
MOV AH,06
```

```
MOV CX,0000
```

```
MOV DX,184F
```

```
INT 10H
```

```
;-----
```

```
MOV AH,02
```

```
MOV BH,00
```

```
MOV DH,12
```

```
MOV DL,39
```

```
INT 10H
```

```
;-----
```

```
MOV AH,09H
```

```
LEA DX,DATAS
```

```
INT 21H
```

```
;-----
```

```
MOV AH,07H
```

```

INT 21H

CMP AL, 'M'

JNZ QUIT ; end of program

;-----

MOV AH, 02

MOV DL, '*'

INT 21H

;-----

MOV AH, 0EH

MOV AL, '*'

INT 10H

;-----

MOV AH, 0EH

MOV AL, 07H

INT 10H ; beep

```

مثال: برنامه ای بنویسید که ابتداء مانیتور را پاک کند سپس پیغام HELLO را روی مانیتور نمایش دهد سپس اگر کلید C فشرده شود ابتداء حرف D را با استفاده از سرویس 02 دستور INT 21H نمایش دهد سپس حرف K را با استفاده از سرویس 0EH دستور INT 10H نمایش دهد در غیر اینصورت به سیستم عامل برگردد .

```

PAGE 70,100

TITLE 'salam.ASM'

.MODEL SMALL

.STACK DW 64 DUP(?)

.DATA

SM DB 'HELLO, '$'

.CODE

MAIN PROC FAR

MOV AX, @DATA

MOV DS, AX

```



```
MOV ES,AX

CALL CLR MON

CALL WRT MES

CALL C DETECT

MOV AX,4C00H

INT 21H

MAIN ENDP

;-----

CLR MON PROC NEAR

MOV AX,600H

MOV BH,00

MOV CX,0000

MOV DX,184FH

INT 10H

RET

CLR MON ENDP

;-----

WRT MES PROC NEAR

MOV AH,09

LEA DX,SM

INT 21H

RET

WRT MES ENDP

;-----

C DETECT PROC NEAR

MOV AH,07

INT 21H

CMP AL,43H

JNZ QUIT
```

```

CALL DISPLAY

QUIT : RET

C DETECT ENDP

;-----

DISPLAY PROC NEAR

MOV AH,02

MOV DL,'D'

INT 21H

MOV AH,0EH

MOV AL,'K'

INT 10H

RET

DISPLAY ENDP

;-----

END MAIN

```

سرویس 0AH دستور INT 21H : خواندن یک رشته اطلاعات از صفحه کلید و قرار دادن آن در محلی در بافر حافظه و نشان دادن آن روی مانیتور

با فشار دادن کلیدهای صفحه کلید (مثلاً تایپ) کدهای اسکی آن ها در محلی از حافظه که در سگمنت داده تعریف می شود قرار می گیرد و حرف های نظیر کلیدهای فشار داده شده نیز بر روی مانیتور نشان داده می شود

AH= 0AH

DX= آفست رشته اطلاعات

INT 21H

سرویس 0BH دستور INT 21H : تست بافر صفحه کلید (حساس به Ctrl+Break)

AH=0BH

INT 21H

اگر در بافر صفحه کلید حرفی وجود داشته باشد آنگاه $AL=FFH$ در غیر اینصورت $AL=0$

سرویس $0CH$ دستور $INT\ 21H$: پاک کردن بافر صفحه کلید و فراخوانی یک سرویس صفحه کلید

این سرویس با سرویس های $0AH$, 07 , 06 , 01 دستور $INT\ 21H$ کار می کند

$AH=0CH$

شماره سرویس = AL

آفست آدرس بافر ورودی DX

$INT\ 21H$

با اجرای این دستور بافر صفحه کلید پاک و سرویس وقفه ای که شماره آن در AL قرار دارد اجرا می شود .

سرویس های دستور $INT\ 16H$ مربوط به صفحه کلید

سرویس $00H$ دستور $INT\ 16H$: خواندن یک حرف از صفحه کلید بدون نمایش آن بر روی مانیتور

$AH=00$

$INT\ 16H$

اگر در بافر حرفی باشد کد اسکی آن در AL و کد اسکن آن در AH ذخیره می شود . اگر در بافر حرفی نباشد منتظر فشار کلید می ماند . جهت کلیدهای $F1-F12$ که کد اسکی ندارند $AL=0$ است .

سرویس $01H$ دستور $INT\ 16H$: بررسی فشار دادن کلید کیبرد

$AH=01$

$INT\ 16H$

اگر کلیدی فشار داده شود $ZF=0$ می شود در غیر اینصورت $ZF=1$

می توان بعد از این سرویس از سرویس 00 استفاده کرد که اگر کلیدی فشار داده شده باشد آنرا پیدا کند .

سرویس 02H دستور 16H INT : گزارش وضعیت بعضی کلیدها (قبل از اجرا کلید بایستی فشار داده شود)

AH=02

INT 16H

در ثبات AL کدی قرار می گیرد که بیتهای آن بصورت زیر است :

اگر یک باشد Right+Shift فشار داده شده	بیت ۰
اگر یک باشد Left+Shift فشار داده شده	بیت ۱
اگر یک باشد Ctrl فشار داده شده	بیت ۲
اگر یک باشد Alt فشار داده شده	بیت ۳
اگر یک باشد Scroll Lock فشار داده شده	بیت ۴
اگر یک باشد Num LOCK فشار داده شده	بیت ۵
اگر یک باشد Caps Lock فشار داده شده	بیت ۶
اگر یک باشد Insert فشار داده شده	بیت ۷

مثال : برنامه ای بنویسید که تست کند آیا کلید Caps Lock فشار داده شده است یا خیر ؟ اگر کلید موردنظر فشار داده شده باشد به برچسب next پرش انجام شود .

```
MOV AH, 02
```

```
INT 16H
```

```
TEST AL, 00000010B
```

```
JNZ next
```

سرویس 11H دستور 16H INT : مشخص می کند آیا حرفی در بافر صفحه کلید وجود دارد یا خیر .

```
AH=11H
```

```
INT 16H
```

اگر حرفی در بافر وجود نداشته باشد $ZF=1$ ولی اگر حرفی وجود داشته باشد $ZF=0$

مثال : زیر برنامه ای بنویسید که تست کند آیا کلید فشار داده شده حرف B است یا خیر ؟

```
MOV AH, 00
```

```
INT 16H
```

```
CMP AL, 'B'
```

```
JZ NEXT
```