

LẬP TRÌNH XỬ LÝ MÀN HÌNH & BÀN PHÍM

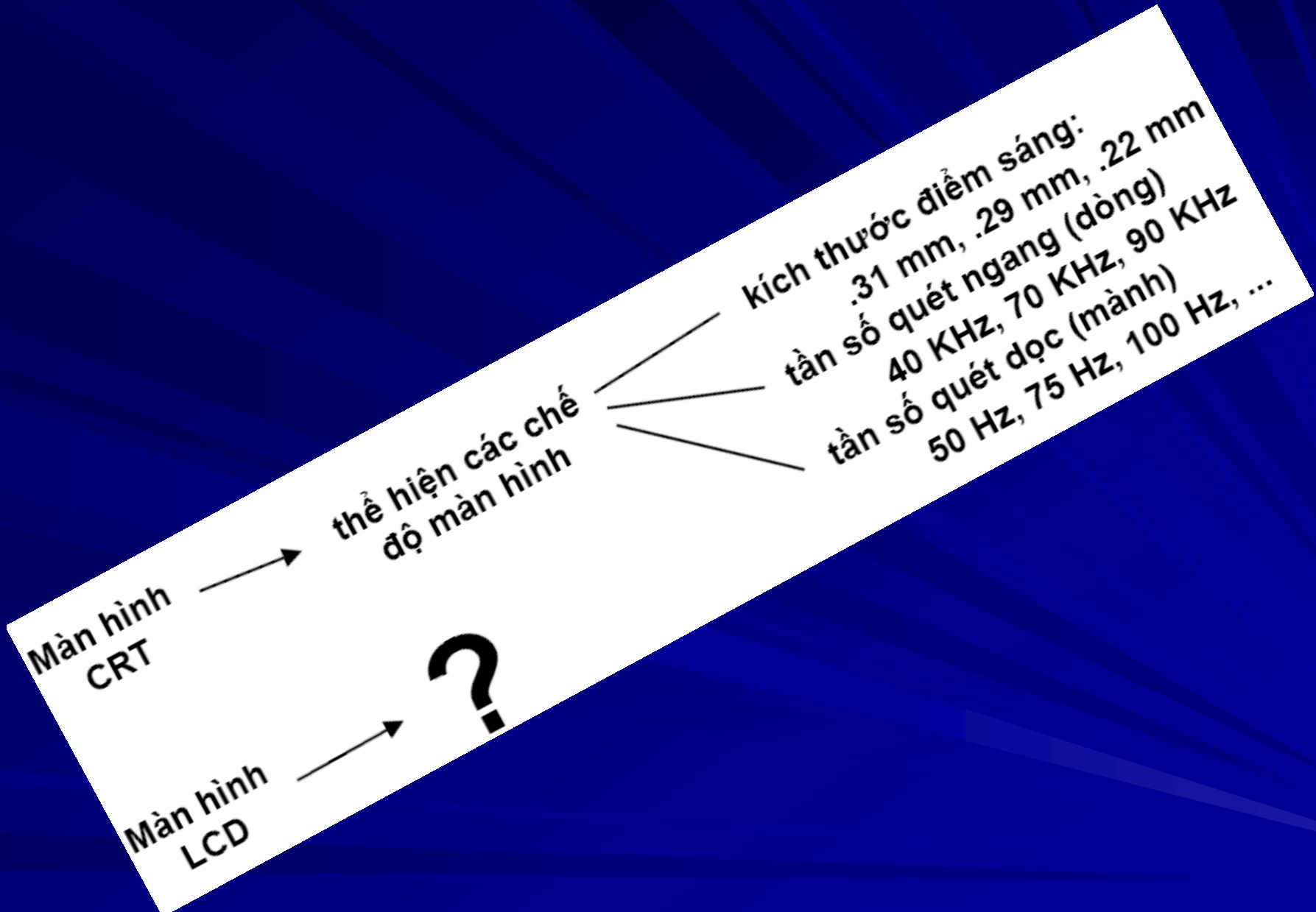
- Giới thiệu màn hình & việc quản lý màn hình
- Hiểu được tổ chức của màn hình.
- So sánh chức năng điều khiển màn hình của INT 10h của ROM BIOS với chức năng của INT 21h.
- Biết cách lập trình quản lý màn hình trong ASM.
- Biết cách lập trình xử lý phím và 1 số ứng dụng của nó.

MÀN HÌNH

ĐẶC TRƯNG CỦA MÀN HÌNH

ĐỘ PHÂN GIẢI

■ **Độ phân giải** : số điểm trong màn hình. Hình ảnh ma trận gồm 1 lưới hình chữ nhật các điểm (thí dụ 640×480). Độ phân giải thường cho dưới dạng $h \times v$ trong đó h là số lượng pixel theo dòng và v là số lượng pixel theo cột.



CARD MÀN HÌNH



Cung cấp các chế độ MH

Độ phân giải



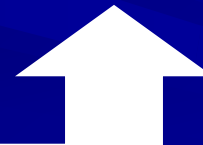
Số điểm ngang x số điểm dọc x số màu (số bit màu)

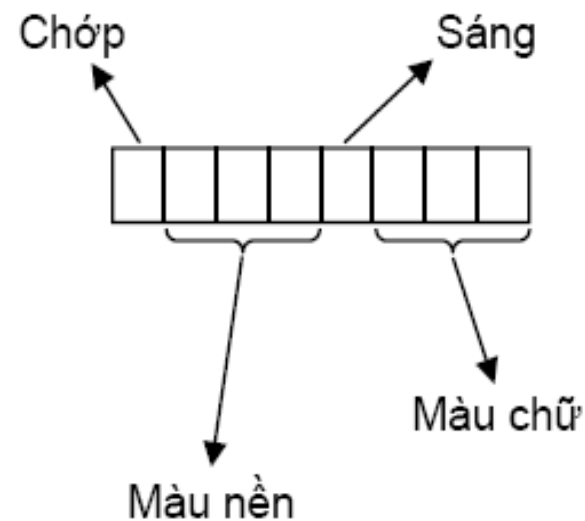
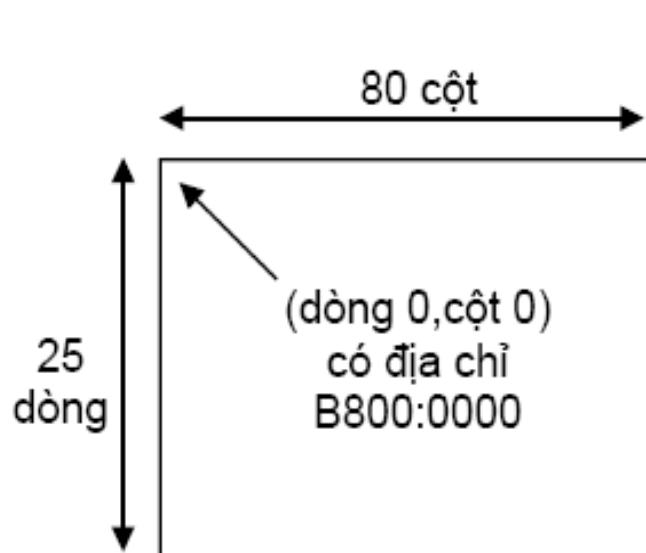
CHẾ ĐỘ
ĐỒ HỌA

$800 \times 600 \times 16 \text{ bits} = 960\,000 \text{ bytes} \rightarrow 1\text{Mb}$
 $1024 \times 768 \times 32 \text{ bits} \rightarrow 3.145.728 \text{ bytes} \rightarrow 4\text{Mb}$

RAM MÀN HÌNH

Dung lượng





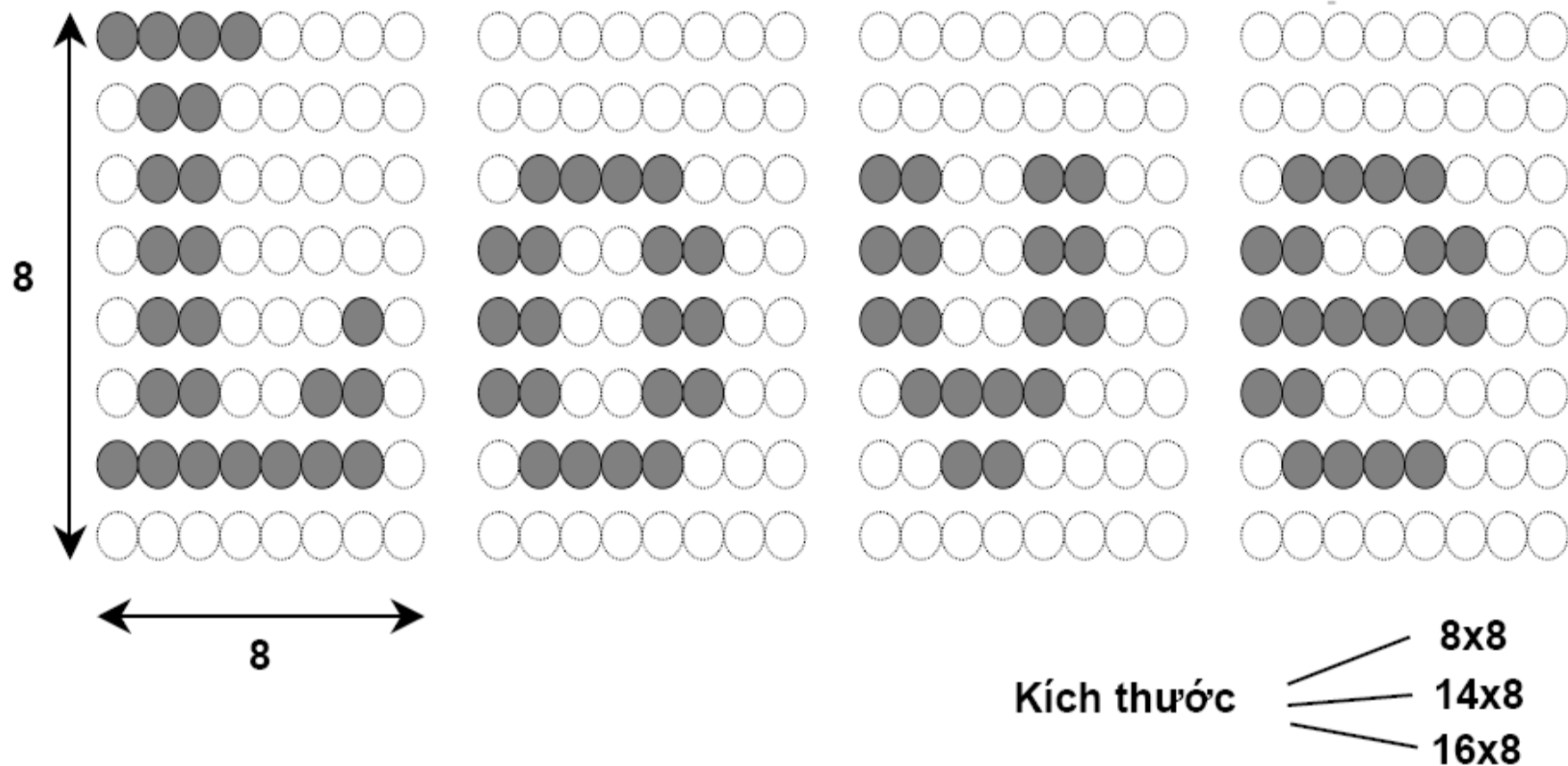
- Mỗi ký tự lưu bằng 2 byte.
- Byte địa chỉ thấp chứa mã ASCII.
- Byte địa chỉ cao chứa mã màu

$$\text{Địa chỉ (i,j)} = \text{B800:0000} + (i * 160 + j * 2)$$

Ma trận ký tự trên màn hình

Thiết bị
ngoại vi

Chế độ văn bản



QUẢN LÝ MÀN HÌNH

■ Màn hình được điều khiển hiệu quả nhờ các chức năng của **INT 10H** trong Rom Bios. Các chức năng này quản lý màn hình tốt hơn các chức năng của INT 21h của Dos.

■ Bên cạnh 1 số chức năng do INT 21h của Dos cung cấp, 1 số tác vụ được thực hiện trên màn hình nhờ các chức năng trong INT 10h như xoá màn hình, định vị con trỏ, thiết lập màn hình ...

■ IBM PC hỗ trợ 3 loại màn hình cơ bản có tên tùy thuộc vào loại Card màn hình cắm trên Bus mở rộng trên Mainboard như : **Monochrome** chỉ hiển thị text đơn sắc; **CGA** (Color Graphic Adaptor) cho phép hiển thị text và đồ họa; **EGA** (Enhanced Graphics Adaptor) hiển thị text và đồ họa với độ phân giải cao hơn. Ngoài ra còn có card **VGA** (Video Graphics Array), **SVGA** ...

THUỘC TÍNH MÀN HÌNH

- Mỗi ký tự hiển thị (văn bản) chiếm 2 bai trong RAM.
- Bai 1: mã ASCII của ký tự
- Bai 2: bai thuộc tính xác định màu ký tự

7	6	5	4	3	2	1	0
b	R	G	B	i	R	G	B

Màu nền

Màu chữ

Chóp

Sáng

THUỘC TÍNH MÀN HÌNH

- Mỗi vị trí trên màn hình có thể lưu 1 ký tự đơn cùng với thuộc tính riêng của ký tự này chẳng hạn như đảo màu, nhấp nháy, chiếu sáng, gạch dưới ...
- Thuộc tính của ký tự được lưu trong 1 byte gọi là byte thuộc tính.

HIGHT/LOW

BIT 7 6 5 4 3 2 1 0

0	0	0	0	0	1	1	1
---	---	---	---	---	---	---	---

BLINK

BACKGROUND BITS

FOREGROUND BITS

THUỘC TÍNH MÀN HÌNH

■ Ex : các ký tự màu vàng chanh nhấp nháy trên nền màu nâu

BLINK = 10000000B

RED = 100B

MOV BH, (RED SHL 4) +YELLOW+BLINK

ĐỂ TẠO 1 BYTE THUỘC TÍNH VIDEO TỪ 2 MÀU , TA DÙNG SHL CHUYỂN CÁC BIT MÀU NỀN SANG TRÁI 4 VỊ TRÍ.

VÙNG HIỂN THỊ MÀN HÌNH

- **Vùng hiển thị của màn hình đơn sắc ở địa chỉ B000h trong Bios.**
- **Vùng hiển thị video đồ họa màu cơ bản bắt đầu từ vị trí B800h của Bios.**

THUỘC TÍNH MÀN HÌNH

■ Các thuộc tính chuẩn của màn hình Monochrome :

HEX VALUE	ATTRIBUTE
07H	Normal – thường
87H	Blinking – nhấp nháy
0FH	Bright – sáng
70H	Reverse – đảo thuộc tính
01H	Underline
09H	Bright Underline

THUỘC TÍNH MÀN HÌNH (tt)

- Bất kỳ 1 thuộc tính nào cũng có thể thêm thuộc tính nhấp nháy bằng cách cho bit 7 có trị là 1. Thí dụ normal blinking 87H, bright linking 8Fh.

Card màn hình CGA và EGA không hỗ trợ thuộc tính Underline nhưng cho phép sử dụng màu trong text mode. Các màu được chia làm 2 loại : màu chữ (Foreground) và màu nền (Background).

Bit 6,5,4 : màu nền

Bit 2,1,0 : màu chữ

Bit 3 : độ sáng

BẢNG MÀU (COLOR PALETTE)

FOREGROUND OR

BACKGROUND COLOR

FOREGROUND COLOR ONLY

000	BLACK	1000	GRAY
001	BLUE	1001	LIGHT BLUE
010	GREEN	1010	LIGHT GREEN
011	CYAN	1011	LIGHT CYAN
100	RED	1100	LIGHT RED
101	MAGENTA	1101	LIGHT MAGENTA
110	BROWN	1110	YELLOW
111	WHITE	1111	BRIGHT WHITE

EX : 01101110 : 06EH nền Brown, chữ Yellow, không nhấp nháy.

EX : 11010010 : 0D2H nền Magenta, chữ Green, nhấp nháy.

CÁC MODE MÀN HÌNH

- Các Card màn hình CGA, EGA, VGA cho phép chuyển đổi Video mode nhờ INT 10h.
- Các trình ứng dụng thường dùng INT 10h để tìm Video mode hiện hành.

Ex: 1 ứng dụng thường muốn thể hiện đồ họa với độ phân giải cao (640x200) phải kiểm tra chắc chắn rằng MT hiện đang sử dụng đang dùng Card màn hình CGA, VGA hoặc EGA.

CÁC MODE MÀN HÌNH

- Có 2 chế độ làm việc của màn hình : text và đồ họa.

Màn hình là hình ảnh của Video Ram.

- Chế độ màn hình :
25 dòng và 80 cột
25 dòng và 40 cột.



*Ở chế độ text một trang màn hình
cần tối thiểu bao nhiêu byte của
VIDEO Ram*



25X80X2 = 4000BYTES RAM VIDEO

VÙNG NHỚ NÀY NẪM TRÊN CARD MH

CÁC MODE MÀN HÌNH

Các Video mode thông dụng :

Mode	Mô tả
<i>02h</i>	<i>80x25 black and white text</i>
<i>03h</i>	<i>80x25 color text</i>
<i>04h</i>	<i>320x400 4 color graphics</i>
<i>06h</i>	<i>640x200 2 color graphics</i>
<i>07h</i>	<i>80x25 black and white text, monochrome adaptor only</i>
<i>0Dh</i>	<i>320x200 16 color graphics</i>
<i>0Eh</i>	<i>640x200 16 colors graphics, EGA, VGA only</i>
<i>0Fh</i>	<i>640x350 monochrome graphics, EGA, VGA only</i>
<i>10h</i>	<i>640x350 16 colors graphics, EGA, VGA only</i>

TRANG MÀN HÌNH (VIDEO PAGE)

Tất cả các Card CGA đều có khả năng lưu trữ nhiều màn hình text gọi là các trang màn hình (video page) trong bộ nhớ. Riêng card mono chỉ hiển thị 1 trang – trang 0. Số trang phụ thuộc vào mode màn hình.

Trong card màn hình màu, ta có thể ghi vào 1 trang này trong khi hiển thị trang khác hoặc chuyển đổi qua lại vị trí giữa các trang. Các trang được đánh số từ 0 đến 7.

TRANG MÀN HÌNH (VIDEO PAGE)

số trang	mode	adaptor
0	07h	monochrome
0-7	00h – 01h	CGA
0-3	02h-03h	CGA
0-7	02h-03h	EGA
0-7	0Dh	EGA
0-3	0Eh	EGA
0-1	0Fh, 10h	EGA

THÍ DỤ VỀ TRANG MH

ĐỂ HIỂN THỊ 1 KÝ TỰ VỚI THUỘC TÍNH CỦA NÓ TẠI 1 VỊ TRÍ BẤT KỲ → CHỨA KÝ TỰ VÀ THUỘC TÍNH VÀO TỪ TƯƠNG ỨNG TRONG TRANG HIỂN THỊ HOẠT ĐỘNG.

EX : Lấp đầy màn hình bằng chữ 'A' màu đỏ trên nền xanh

CHẾ ĐỘ ĐỒ HỌA

- MOV AH, 0
MOV AL, CheDo
INT 10h
- CheDo=

4	320 x 200	4 màu CGA
6	640 x 200	2 màu CGA
D	320 x 200	16 màu EGA
E	640 x 200	16 màu EGA
10	640 x 350	16 màu EGA
11	640 x 480	2 màu VGA
12	640 x 480	16 màu VGA
13	640 x 480	256 màu VGA

Truy xuất thiết bị xuất chuẩn (màn hình)

. 1. Chọn chế độ hiển thị :

Chức năng AH = 0, ngắt 10H

Vào : AH = 0, AL = kiểu

Ví dụ : thiết lập chế độ văn bản màu

```
XOR    AH, AH
```

```
MOV    AL, 3    ; chế độ văn bản màu 80 x 25
```

```
INT    10H
```

THAY ĐỔI SIZE CON TRỎ MÀN HÌNH

Chức năng AH = 1, ngắt 10H

Vào : AH = 1,

CH = định qũt đầu, CL = định qũt cuối

Ví dụ : thiết lập con trỏ với kích thước lớn nhất

```
MOV    AH, 1
```

```
MOV    CH, 0 ; định bắt đầu
```

```
MOV    CL, 13 ; định kết thúc
```

```
INT    10H
```

DỊCH CHUYỂN CON TRỎ

Chức năng AH = 2, ngắt 10H
Vào : AH = 2,
DH = dòng mới (0-24),
DL = cột mới (0-79)
BH = số hiệu trang

Ví dụ : Di chuyển con trỏ đến giữa màn hình 80
x 25 của trang 0

```
MOV    AH, 2
XOR    BH, BH        ; trang 0
MOV    DX, 0C27H    ; dòng 12 cột 39
INT    10H
```


LẤY VỊ TRÍ KÍCH THUỐC CON TRỎ HIỆN HÌNH

Chức năng AH = 3, ngắt 10H

Vào : AH = 3, BH = số hiệu trang

Ra : DH = dòng, DL = cột,

CH = dòng quét đầu, CL = dòng quét cuối

Ví dụ : Di chuyển con trỏ ln một dòng nếu nĩ không ở dòng tĩn ếng

```
MOV AH, 3
```

```
XOR BH, BH ; trang 0
```

```
INT 10H
```

```
OR DH, DH ; dòng tĩn ếng DH = 0 ?
```

```
JZ exit
```

```
MOV AH, 2 ; chức năng dịch con trỏ
```

```
DEC DH ; giảm một dòng
```

```
INT 10H
```

exit :

CUỘN MÀN HÌNH

Chức năng AH = 6, ngắt 10H

Vào : AH = 6,

AL = số dòng cuộn (= 0 ÷ tổng màn hình)

Ra : BH = thuộc tính các dòng trống ,

CH, CL = dòng, cột gốc trái trên

DH, DL = dòng, cột gốc phải dưới của cửa sổ

Ví dụ : Xóa đen màn hình 80 x 25

```
MOV AH, 6
```

```
XOR AL, AL
```

```
XOR CX, CX
```

```
MOV DX,
```

184FH ; gốc phải dưới

```
MOV BH, 7
```

```
INT 10H
```

Ví dụ tổng hợp : Viết chương trình thực hiện như sau:

- . Lập chế độ hiển thị màu 80 x 25
- . Xoá cửa sổ tại góc trái trên : cột 26 dòng 8 và góc phải dưới tại cột 52 dòng 16 thành màu đỏ. . Sau đó hiển thị kí tự A màu cam tại vị trí con trỏ.

CÁC HÀM XỬ LÝ MÀN HÌNH

Các chức năng xử lý màn hình nằm trong INT 10h

Chức năng (để trong AH)

nhiệm vụ

- 0 *set video mode chọn mono, text, graphic hoặc color mode*
- 1 *Set cursor line thiết lập 1 dòng quét tạo dạng cho cursor.*
- 2 *Set cursor position định vị cursor*
- 3 *get cursor position lấy vị trí cursor*
- 4 *đọc vị trí và trạng thái của bút vẽ light pen.*
- 5 *chọn trang muốn hiển thị.*
- 6 *cuộn cửa sổ hiện hành lên, thế các dòng cuộn bằng ktrống.*
- 7 *cuộn cửa sổ hiện hành xuống.*
- 8 *đọc ký tự và thuộc tính ký tự tại vị trí con trỏ hiện hành.*
- 9 *ghi ký tự và thuộc tính ký tự tại vị trí con trỏ hiện hành.*

CÁC HÀM XỬ LÝ MÀN HÌNH

Các chức năng xử lý màn hình nằm trong INT 10h

Chức năng (để trong AH)

nhiệm vụ

- 0Ah** Ghi ký tự bỏ qua thuộc tính ký tự vào vị trí con trỏ hiện hành.
- 0Bh** Chọn palette màu
- 0Ch** Ghi 1 điểm graphic trong graphics mode.
- 0Dh** Đọc giá trị màu của 1 pixel có vị trí đã biết.
- 0Eh** Ghi ký tự ra màn hình và cập nhật con trỏ sang phải 1 vtrí.
- 0Fh** Lấy mode màn hình hiện hành để xem đang ở chế độ text hay graphics.

- Thiết lập video mode.

AH = 0

AL = mode.

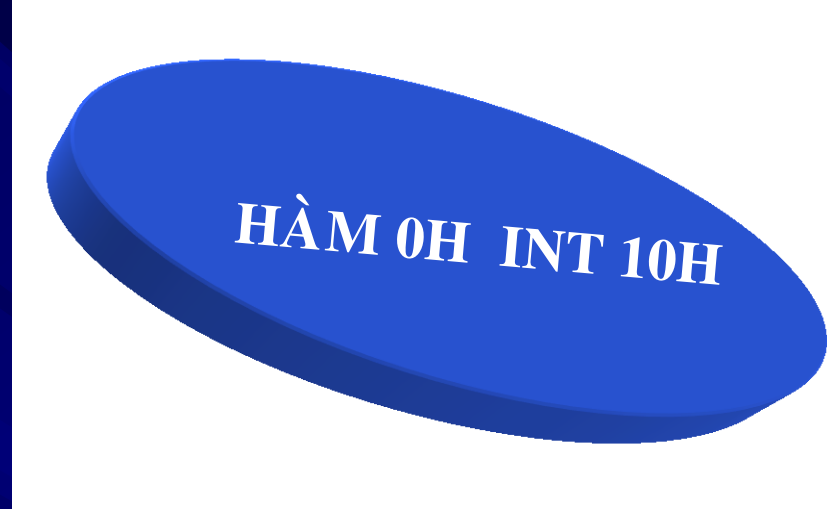
Nếu bit cao của AL = 0 sẽ tự động xoá màn hình.
Nếu bit cao của AL = 1 không xoá màn hình.

- Ex : thiết lập 80x25 color text mode

```
MOV AH, 0
```

```
MOV AL, 3 ; mode 3 , có xoá màn hình
```

```
INT 10h
```



LƯU Ý : Không muốn xoá màn hình thì AL = 83H



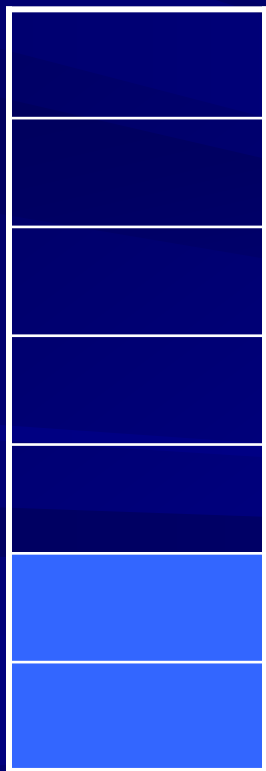
HÀM 0H INT 10H

■ Ex : đoạn chương trình sau
high resolution graphics, đợi gõ 1 phím sau đó reset lập
video mode là color text mode.

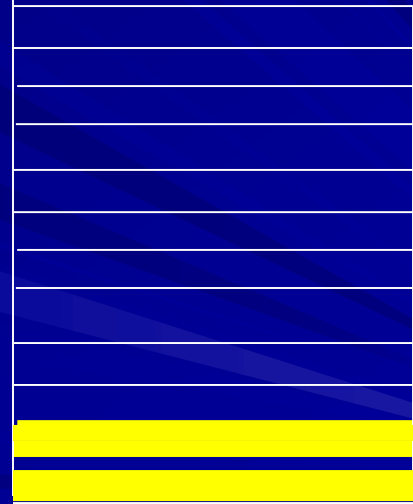
```
MOV AH, 0 ; set video mode
MOV AL, 6 ; 640x200 color graphics mode
INT 10h
MOV AH, 1 ; đợi gõ 1 phím
INT 21H
MOV AH, 0 ; set video mode
MOV AL, 3 ; color text mode
INT 10H
```



- Dạng màn hình được tạo ra bằng cách chỉ định số dòng quét.
Việc thay đổi dạng con trỏ chính là thay đổi số lượng và vị trí dòng quét này.
- Màn hình monochrome dùng 13 dòng (từ 0 – 0Ch)
- Màn hình CGA, VGA dùng 8 dòng (từ 0-7).



CGA/EGA



MONOCHROME

Ex: Minh họa

Thiết lập con trỏ có hình khối đặc.

```
MOV AH, 1
```

```
MOV CH, 0
```

```
MOV CL, 0CH
```

```
INT 10H
```

Để thay đổi dạng con trỏ :
AH = 1
CH = TOP (dòng đầu)
CL = BOTTOM (dòng cuối)

Ex2: trả kích thước con trỏ về dạng mặc định trước khi thoát.

```
MOV AH, 1
MOV CX, 0607H
INT 10H
MOV AX, 4C00H
INT 21H
```

```
MOV AH, 3
MOV BH, 0
INT 10H
MOV SAVECURSOR, CX
OR CH, 00100000
INT 10H
.....
MOV AH, 1
MOV CX, SAVECURSOR
INT 10H
```

Ex2: lưu kích thước con trỏ hiện hành vào 1 biến trước khi thay đổi kích thước con trỏ để sau này phục hồi lại.

**HÀM 02H
INT 10H**

*Thiết lập vị trí hiện hành
SET CURSOR POSITION*

**AH = 2 ; DH = CHỈ SỐ DÒNG ; DL = CHỈ SỐ CỘT ;
BH = TRANG MÀN HÌNH CHỨA CURSOR**

Ex : THIẾT LẬP CURSOR TẠI TỌA ĐỘ (DÒNG 10, CỘT 20) CỦA TRANG 0

MOV AH, 2 ; Chức năng set cursor
MOV DH, 10 ; dòng 10
MOV DL, 20 ; cột 20
MOV BH, 0 ; trang 0
INT 10H ; gọi BIOS

**HÀM 03H
INT 10H**

*Lấy vị trí cursor hiện hành
GET CURSOR POSITION*

AH = 3 ;

BH = TRANG MÀN HÌNH MUỐN LẤY CURSOR

Gía trị trả về :

CH = Dòng quét đầu của cursor

CL = Dòng quét cuối của cursor

DH = vị trí dòng màn hình

DL = vị trí cột màn hình



**HÀM 03H
INT 10H**

*Lấy vị trí cursor hiện hành
GET CURSOR POSITION*

Ex : lấy vị trí của cursor lưu vào biến.

Thường dùng trong các tác vụ menu.

```
MOV AH, 3  
MOV BH, 0  
INT 10H  
MOV SAVECURSOR, CX  
MOV CURRENT_ROW , DH  
MOV CURRENT_COL , DL
```

**HÀM 05H
INT 10H**

THIẾT LẬP TRANG MÀN HÌNH SET VIDEO PAGE

AH = 5 ; AL = TRANG MÀN HÌNH SẼ LÀ TRANG HIỆN HÀNH

EX : THAY ĐỔI TRANG MH KHI GỎ 1 PHÍM BẤT KỲ

DOSSEG

.MODEL SMALL

.STACK 100H

.CODE

MAIN PROC

MOV AX, @DATA

MOV DS,AX

MOV DX, OFFSET TRANG0

INT 21H

MOV AH, 1

INT 21H

SANG_TRANG_1 :

MOV AH, 5

MOV AL, 1

**HÀM 05H
INT 10H**

***THIẾT LẬP TRANG MÀN HÌNH
SET VIDEO PAGE***

```
INT 10H  
MOV AH, 9  
MOV DX, OFFSET  
TRANG1  
INT 21H  
MOV AH, 1  
INT 21H  
SANG_TRANG_0 :  
MOV AH, 5  
MOV AL, 0
```

```
INT 10H  
MOV AX, 4C00H  
INT 21H  
MAIN ENDP  
.DATA  
    TRANG0 DB ' DAY LA TRANG 0,$'  
    TRANG1 DB ' DAY LA TRANG 1,$'  
END MAIN
```

**HÀM 06H, 07H
INT 10H**

CUỘN MÀN HÌNH

SCROLL WINDOW UP AND DOWN

CUỘN MÀN HÌNH LÀ TÁC VỤ LÀM CHO DỮ LIỆU TRƯỢT LÊN HOẶC XUỐNG.

CÁC DÒNG DỮ LIỆU BỊ CUỘN SẼ ĐƯỢC THAY THẾ BẰNG CÁC DÒNG TRỐNG
TẠ ĐỊNH NGHĨA WINDOWS NHỜ HỆ TỌA ĐỘ HÀNG CỘT VỚI GỐC TỌA ĐỘ LÀ GÓC TRÊN TRÁI CỦA MÀN HÌNH.

HÀNG SẼ THAY ĐỔI TỪ 0 ĐẾN 24 TỪ TRÊN XUỐNG.

CỘT SẼ THAY ĐỔI TỪ 0 ĐẾN 79 TỪ TRÁI SANG PHẢI.

TÀ CÓ THỂ CUỘN 1 VÀI DÒNG HOẶC CẢ WINDOWS.

TOÀN BỘ WINDOWS BỊ CUỘN → MÀN HÌNH BỊ XOÁ.

**HÀM 06H, 07H
INT 10H**

***CUỘN MÀN HÌNH
SCROLL WINDOW UP AND DOWN***

CÁC THÔNG SỐ :

CUỘN LÊN AH = 6 ; CUỘN XUỐNG AH = 7

AL = SỐ DÒNG SẼ CUỘN (= 0 NẾU CUỘN TOÀN BỘ MÀN HÌNH)

CH, CL = TỌA ĐỘ HÀNG, CỘT CỦA GÓC TRÊN TRÁI CỦA WINDOWS

DH, DL = TỌA ĐỘ HÀNG, CỘT CỦA GÓC DƯỚI PHẢI CỦA WINDOWS

**BH = THUỘC TÍNH MÀN HÌNH CỦA CÁC DÒNG TRỐNG
KHI MÀN HÌNH ĐÃ CUỘN.**

**HÀM 06H, 07H
INT 10H**

CUỘN MÀN HÌNH

SCROLL WINDOW UP AND DOWN

**EX : XÓA MÀN HÌNH BẰNG CÁCH CUỘN LÊN TOÀN BỘ MÀN HÌNH
VỚI THUỘC TÍNH NORMAL**

```
MOV AH, 6  
MOV AL, 0  
MOV CH, 0  
MOV CL, 0  
MOV DL, 24  
MOV DH, 79  
MOV BH, 7  
INT 10H
```



```
MOV AX, 0600H  
MOV CX, 0000H  
MOV DX, 184FH  
MOV BH, 7  
INT 10H
```



HÀM 06H, 07H
INT 10H

CUỘN MÀN HÌNH

SCROLL WINDOW UP AND DOWN

EX : CUỘN WINDOWS TỪ (10,20) TỚI (15,60), CUỘN XUỐNG 2 DÒNG, 2 DÒNG CUỘN SẼ CÓ THUỘC TÍNH VIDEO ĐẢO.

```
MOV AX, 0702H
```

```
MOV CX, 0A14H
```

```
MOV DX, 0F3CH
```

```
MOV BH, 70H
```

```
INT 10H
```

**HÀM 08H
INT 10H**

***ĐỌC 1 KÝ TỰ VÀ THUỘC TÍNH KÝ TỰ.
READ CHARACTER AND ATTRIBUTE***

AH = 8 ; BH = TRANG MÀN HÌNH

TRI TRẢ VỀ :

AL = KÝ TỰ ĐÃ ĐỌC ĐƯỢC ; AH = THUỘC TÍNH CỦA KÝ TỰ

EX : THIẾT LẬP CURSOR TẠI HÀNG 5 CỘT 1 SAU ĐÓ NHẬN 1 KÝ TỰ NHẬP. LƯU KÝ TỰ ĐÃ ĐỌC ĐƯỢC VÀ THUỘC TÍNH CỦA KÝ TỰ NÀY.

LOCATE :

MOV AH, 2

MOV BH, 0

MOV DX, 0501H

INT 10H

GETCHAR :

MOV AH, 8

MOV BH, 0

INT 10H

MOV CHAR, AL

MOV ATTRIB, AH

HÀM 09H
INT 10H

*GHI 1 KÝ TỰ VÀ THUỘC TÍNH KÝ TỰ.
WRITE CHARACTER AND ATTRIBUTE*

CHỨC NĂNG 09H INT 10H :

XUẤT (GHI) 1 HOẶC NHIỀU KÝ TỰ CÙNG VỚI THUỘC TÍNH CỦA CHÚNG LÊN MÀN HÌNH. CHỨC NĂNG NÀY CÓ THỂ XUẤT MỌI MÃ ASCII KỂ CẢ KÝ TỰ ĐỒ HỌA ĐẶC BIỆT CÓ MÃ TỪ 1 ĐẾN 31

AH = 9 ; BH = TRANG VIDEO

AL = KÝ TỰ SẼ XUẤT ;

BL = THUỘC TÍNH CỦA KÝ TỰ SẼ XUẤT

CX = HỆ SỐ LẶP

HÀM 0AH
INT 10H

*GHI 1 KÝ TỰ VÀ THUỘC TÍNH KÝ TỰ.
WRITE CHARACTER AND ATTRIBUTE*

CHỨC NĂNG 0AH INT 10H :

XUẤT (GHI) 1 HOẶC NHIỀU KÝ TỰ CÙNG VỚI THUỘC TÍNH CỦA CHÚNG LÊN MÀN HÌNH. CHỨC NĂNG NÀY CÓ THỂ XUẤT MỌI MÃ ASCII KỂ CẢ KÝ TỰ ĐỒ HỌA ĐẶC BIỆT CÓ MÃ TỪ 1 ĐẾN 31

AH = 9 ; BH = TRANG VIDEO

AL = KÝ TỰ SẼ XUẤT ;

BL = THUỘC TÍNH CỦA KÝ TỰ SẼ XUẤT

CX = HỆ SỐ LẶP

HÀM 0AH
INT 10H

*GHI 1 KÝ TỰ VÀ THUỘC TÍNH KÝ TỰ.
WRITE CHARACTER AND ATTRIBUTE*

CHỨC NĂNG 0AH INT 10H :

XUẤT (GHI) 1 HOẶC NHIỀU KÝ TỰ CÙNG VỚI THUỘC TÍNH CỦA CHÚNG LÊN MÀN HÌNH. CHỨC NĂNG NÀY CÓ THỂ XUẤT MỌI MÃ ASCII KỂ CẢ KÝ TỰ ĐỒ HỌA ĐẶC BIỆT CÓ MÃ TỪ 1 ĐẾN 31

AH = 9 ; BH = TRANG VIDEO

AL = KÝ TỰ SẼ XUẤT ;

BL = THUỘC TÍNH CỦA KÝ TỰ SẼ XUẤT

CX = HỆ SỐ LẶP

**HÀM 0FH
INT 10H**

***LẤY VIDEO MODE
GET VIDEO MODE***

CHỨC NĂNG 0FH INT 10H : LẤY VIDEO MODE

**AH = 0F ;
BH = TRANG HIỆN HÀNH**

**AH = SỐ CỘT MÀN HÌNH ;
AL = MODE MÀN HÌNH HIỆN HÀNH**

EX : MOV AH,0FH ; Get Video Mode Function

INT 10H ; gọi BIOS

MOVE VIDEO_MODE, AL ; lưu Video Mode vào biến bộ nhớ

MOV PAGE, BH ; lưu trang hiện hành.

LẬP TRÌNH XỬ LÝ PHÍM

1. Đọc phím nhấn :

Chức năng $AH = 0$, ngắt $16H$

Vào : $AH = 0$

Ra : $AL =$ mã ASCII nếu một phím ASCII được nhấn
 $= 0$ nếu phím điều khiển được nhấn

$AH =$ mã scan của phím nhấn

LẬP TRÌNH XỬ LÝ PHÍM

BÀN PHÍM

- Gồm 2 nhóm phím:
 1. ASCII: chữ (a..z), số (0..9), dấu (+-*/...), Esc, Enter (↵), BackSpace (←), Tab (⇠).
 2. ASCII mở rộng: Shift, CTrl, Alt, Caps Lock, Num Lock, Scroll Lock (thường dùng với phím khác); phím hàm (F1..F12), hướng (←, ↑, →, ↓, Home, End, Page Up, Page Down), Insert, Delete và các phím còn lại.

BÀN PHÍM

INT 21h, AH = 1 (DOS)

- Nhập ký tự bàn phím → AL (hiển thị)
- AL = ASCII (nhóm 1), AL = 0 (nhóm 2)
- INT 21h → AL = mã quét

BÀN PHÍM

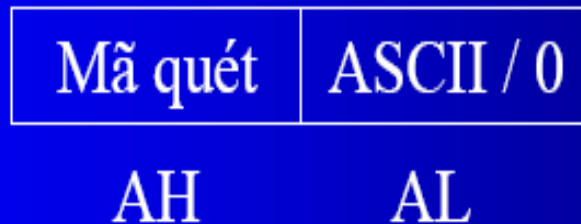
INT 21h, AH = 8 (DOS)

- Như INT 21h, AH = 1 nhưng không hiển thị.
- Ví dụ sau chờ nhập “secret” và Enter. Chỉ kết thúc khi nhập đúng. Chuỗi nhập không hiển thị.
- → PASSWORD.ASM

BÀN PHÍM

INT 16h, AH = 0 (BIOS)

- ASCII / 0 → AL , mã quét → AH
- Phím nhập không hiển thị
- → INT16-00.ASM



BÀN PHÍM

INT 16h, AH = 1 (BIOS)

- Kiểm tra vùng đệm bàn phím
- ZF = 1, rỗng
- ZF = 0, không rỗng
- Xoá vùng đệm: + INT 16h, AH = 0

BÀN PHÍM

INT 16h, AH = 2 (BIOS)

- Xác định trạng thái các phím điều khiển
- AL(b) = 1: đã nhấn bit b
- AL(b) = 0: chưa nhấn bit b

LẬP TRÌNH XỬ LÝ PHÍM

Kiểm tra trạng thái các phím Ctrl, Alt, Shift :

Chúng ta có thể đọc trực tiếp từ địa chỉ 0:0417 hoặc lấy trong AL thông qua hàm AH = 2 ngắt 16H.

Cách đọc trực tiếp

```
XOR AX, AX
```

```
MOV ES, AX
```

```
MOV AL, ES:[417H]
```

```
TEST AL, 01H ; kiểm tra phím Shift
```

```
JNZ SHIFT_DANGNHAN
```

```
TEST AL, 04H ; kiểm tra phím Ctrl
```

```
JNZ CTRL_DANGNHAN
```

```
TEST AL, 08H ; Kiểm tra phím Alt
```

```
JNZ ALT_DANGNHAN
```


3. Kiểm tra về thiết lập trạng thái các phím Caps/Num/Scroll Lock tương tự nhưng với mã scan khác Scroll = 10H, Num = 20H, Cap = 40H.

4. Đặt lại các trạng thái đèn Caps/Num/Scroll Lock, ta chỉ cần đặt lại giá trị ở địa chỉ 0:0417.

Vd, để bật đèn Caps Lock và đổi trạng thái đèn Num Lock ta sẽ làm như sau:

```
XOR    AX, AX
MOV    ES, AX          ; ES = 0
MOV    AL, ES:[417H]  ; đọc trạng thái đèn
OR     AL, 40H        ; bật đèn Caps Lock
XOR    AL, 20H        ; đảo đèn Num Lock
MOV    ES:[417H], AL ;
MOV    AH, 2H
INT    16H
```

BÀI TẬP LẬP TRÌNH

BÀI 1 : VIẾT ĐOẠN CHƯƠNG TRÌNH LÀM CÁC VIỆC SAU :

- CUỘN WINDOW TỪ HÀNG 5, CỘT 10 TỚI HÀNG 20 CỘT 70 VỚI THUỘC TÍNH MÀN HÌNH ĐẢO.
- ĐỊNH VỊ CURSOR TẠI HÀNG 10, CỘT 20
- HIỂN THỊ DÒNG TEXT “ DAY LA 1 DONG TEXT TRONG WINDOW ”
- SAU KHI XUẤT TEXT ĐỢI NHẤN 1 PHÍM.
- CUỘN WINDOW TỪ HÀNG 5, CỘT 15 TỚI HÀNG 18 CỘT 68 VỚI THUỘC TÍNH THƯỜNG.
- XUẤT KÝ TỰ A VỚI THUỘC TÍNH NHẤP NHÁY TẠI GIỮA WINDOW.
- ĐỢI GỎ 1 PHÍM, XÓA TOÀN BỘ MÀN HÌNH..

BÀI TẬP LẬP TRÌNH

BÀI 2 : VIẾT CHƯƠNG TRÌNH LÀM CÁC VIỆC SAU :

- XUẤT CHUỖI “GO VAO 1 KY TU THUONG : ‘ .
- KHI USER GỎ 1 KÝ TỰ (KHI GỎ KHÔNG HIỂN THI KÝ TỰ GỎ RA MÀN HÌNH) ,ĐỔI KÝ TỰ NÀY THÀNH CHỮ HOA RỒI XUẤT RA MÀN HÌNH.GIẢ SỬ CHỈ NHẬP CÁC KÝ TỰ HỢP LỆ.
- KHI GỎ KÝ TỰ MỞ RỘNG SẼ THOÁT VỀ DOS, NHƯNG CÓ LẼ BẠN CÒN NHÌN THẤY 1 KÝ TỰ XUẤT THÊM TRÊN MÀN HÌNH. GIẢI THÍCH.

1. Viết chương trình để :

a. Xổ màn hình, tạo kích thước to nhất cho con trỏ về di chuyển
nữ đến gốc trái trên

b. Nếu nhấn phím Home : chuyển con trỏ đến gốc trái trên, End :
chuyển đến gốc trái dưới, Page Dn : chuyển con trỏ đến gốc
phải dưới, Esc : kết thúc chương trình.

2. Dịch chuyển con trỏ đến gốc trái trên màn hình nếu phím F1
được nhấn, gốc trái dưới nếu phím F2 được nhấn. Chương
trình sẽ bỏ qua các kí tự thông thường.

3. Viết chương trình soạn thảo văn bản như sau :

a. Xổ màn hình, định vị con trỏ tại đầu dòng 12

b. Để người sử dụng nhập vào các kí tự. Con trỏ dịch chuyển đi
sau khi hiển thị kí tự nếu nữ không ở tại lề phải của màn hình

c. Phím mũi trên, phải, xuống, xuống dịch con trỏ tương ứng

d. Phím Insert : chèn kí tự, Delete : Xổ một kí tự , Esc : kết thúc
chương trình.

BÀI TẬP LẬP TRÌNH

BÀI 3 : VIẾT CHƯƠNG TRÌNH LÀM CÁC VIỆC SAU :

CHO PHÉP VẼ ĐƠN GIẢN NHỜ CÁC PHÍM MŨI TÊN TRÊN BÀN PHÍM ĐỂ DI CHUYỂN THEO HƯỚNG MONG MUỐN.

PHẢI BẢO ĐẢM XUẤT CÁC KÝ TỰ GÓC THÍCH HỢP.

BIẾT RẰNG MÃ ASCII CỦA 1 SỐ KÝ TỰ : xem bảng mã ASCII

MÃ SCAN CODE CỦA CÁC PHÍM MŨI TÊN :

TRÁI 4BH PHẢI 4DH LÊN 48H XUỐNG 50H