Microprocessors & Interfacing

# Programming Model

BITS Pilani

Dr. Gargi Prabhu
Department of CS & IS

# Int 10H



- INT 10H subroutines are burned into ROM BIOS and are used to communicate with the computer's screen video
- Provides a set of functions to control the video display such as setting the video mode, moving the cursor, scrolling the screen, displaying characters, and more
- Serves as an interface between the software (assembly language programs) and the video hardware of the computer
- Chosen by putting a specific value in register AH

# Modes



**INT 10h / AH = 00h** - set video mode.

*input:*

**AL** = desired video mode.

## Text Modes:

- **Mode 03h**: 80x25 characters, monochrome (black and white).
- **Mode 02h**: 80x25 characters, CGA-compatible colors.
- **Mode 00h**: 40x25 characters, CGA-compatible colors.
- **Mode 13h**: 320x200 pixels, 256 colors (VGA mode, but often used for graphics).

## Graphics Modes:

- **Mode 13h**: 320x200 pixels, 256 colors. Often used for games and graphics programming due to its ease of use and compatibility.
- **Mode 12h**: 640x480 pixels, 16 colors.
- **Mode X (Mode 10h)**: A set of high-resolution graphics modes available on VGA-compatible cards. These modes provide various resolutions and color depths, commonly used for games and multimedia applications.

# Text mode of 80 × 25 characters.

innovate achieve lead

- A total of 2K (80×25 = 2000) for characters, plus 2K for attributes, as each character has one attribute byte.
- In this mode, 16 colors are supported.
- To select this mode, use AL = 03 for mode selection in INT 10H option AH = 00.

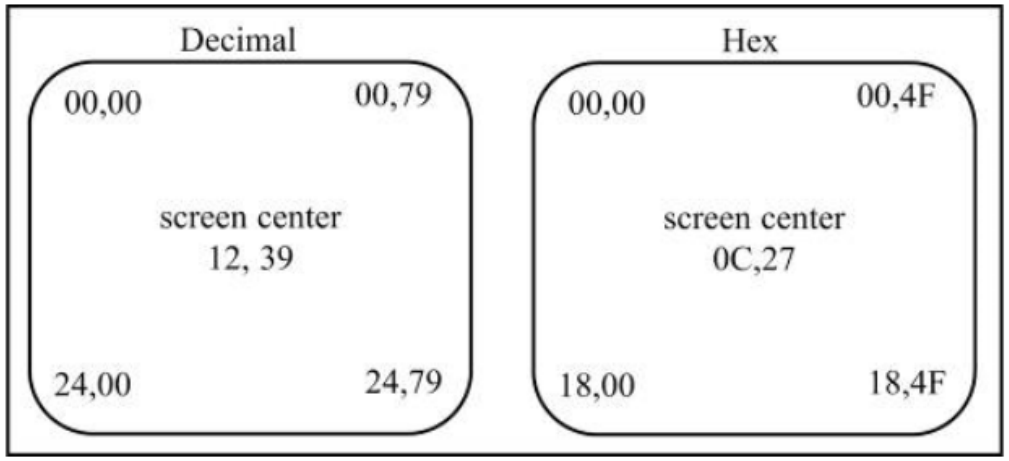
# Screen

The monitor screen in the x86 PC is divided into 80 columns and 25 rows in normal text mode.

– Columns are numbered from 0 to 79.

– Rows are numbered 0 to 24.

The top left corner has been assigned 00,00, the top right 00,79. Bottom left is 24,00, bottom right 24,79



# Clearing the Screen

```
MOV AH,06  ; SELECT SCROLL FUNCTION
MOV AL,00 ;  ENTIRE PAGE
MOV BH,07 ; NORMAL ATTRIBUTES
MOV CH,00 ; ROW VALUE OF START POINT
MOV CL,00 ; COLUMN VALUE OF START POINT
MOV DH,24;  ROW VALUE OF END POINT
MOV DL,79; COLUMN VALUE OF END POINT
INT 10H ; INVOKE THE INTERRUPT
```

# Clearing the Screen



```
MOV AX, 0600H ; SCROLL ENTIRE SCREEN
MOV BH,07
MOV CX,0000
MOV DX,184FH
INT 10H
```

# Cursor Position



**INT 10h** / **AH = 02h** - set cursor position.
*input:*
**DH** = row.
**DL** = column.
**BH** = page number (0..7).

Write the code to set the cursor position to row = 15 = 0FH and column = 25 = 19H.

```
MOV AH,02
MOV BH,00
MOV DL,25
MOV DH,15
INT 10H
```

# Example



```
.model small
.stack 100h

.data
msg1 db 'Hello World$'

.code
main proc
        mov ax,@data
        mov ds,ax

        mov ah,9
        mov dx, offset(msg1)
        int 21h

        mov ah, 4ch
        int 21h
        main endp
end main
```

# Example

```
main proc
        mov ax,@data
        mov ds,ax

        mov ah,9
        mov dx, offset(msg1)
        int 21h

        mov ah,2        ; interrupt function
        mov bh,0        ; Page number
        mov dl,20       ; column number
        mov dh,15       ; row number

        mov ah,9
        mov dx, offset(msg1)
        int 21h

        mov ah, 4ch
        int 21h
        main endp
end main
```

# Get cursor position

innovate achieve lead

```
MOV AH, 03 ; option 03 for INT 10H
MOV BH,00 ; page 00
INT 10H
```

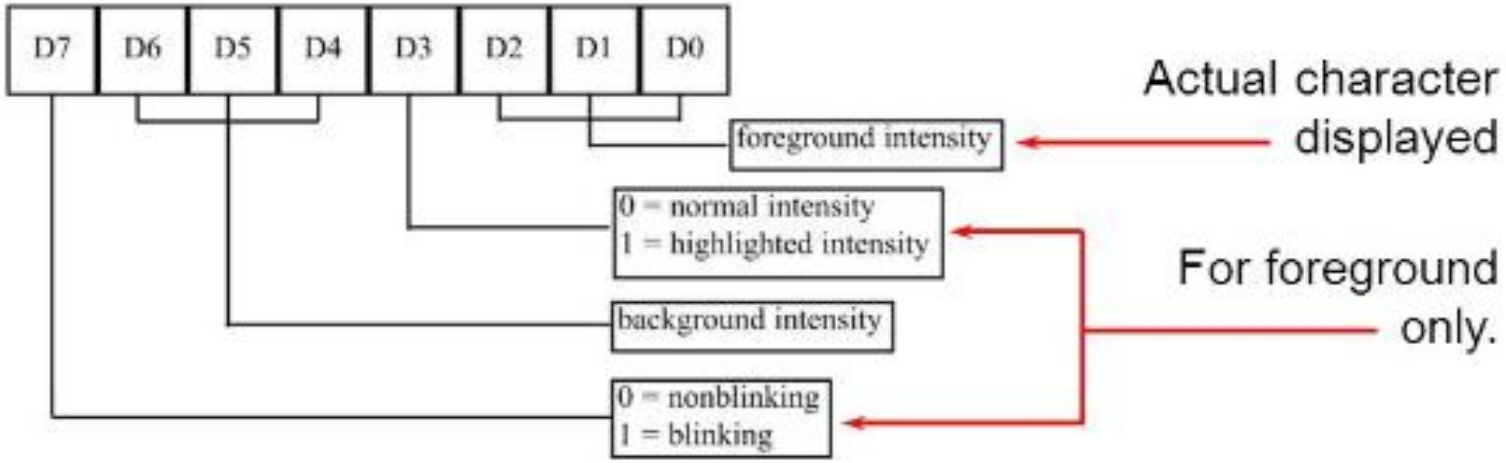
-> register DH and DL will have the current row and column position

-> cx provides information about the shape of the cursor

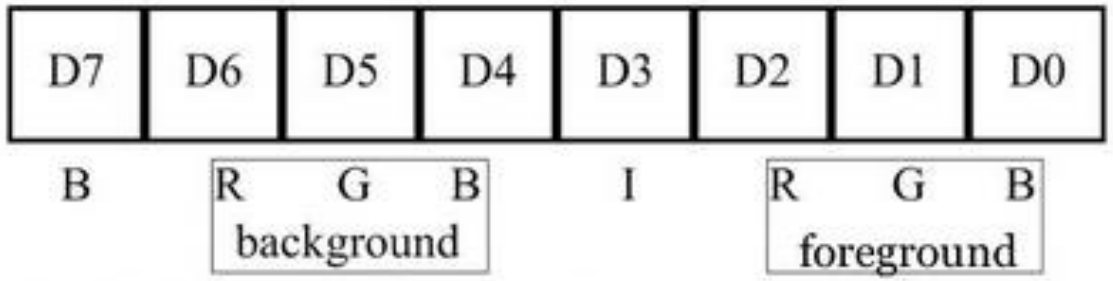
# Attribute byte for Monochrome Monitors



- An attribute associated with each character on the screen provides information to the video circuitry.
  - Character (foreground) & background color/intensity.
- The attribute byte for each character on the monochrome monitor is limited.



# Attribute byte in CGA mode




B = blinking I = foreground intensity

Blinking and intensity apply to foreground only.

# Colors

Character attribute is 8 bit value, low 4 bits set foreground color, high 4 bits set background color. Background blinking not supported.

**Table 4-1: The 16 Possible Colors**

| I | R | G | B | Color |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | black |
| 0 | 0 | 0 | 1 | blue |
| 0 | 0 | 1 | 0 | green |
| 0 | 0 | 1 | 1 | cyan |
| 0 | 1 | 0 | 0 | red |
| 0 | 1 | 0 | 1 | magenta |
| 0 | 1 | 1 | 0 | brown |
| 0 | 1 | 1 | 1 | white |
| 1 | 0 | 0 | 0 | gray |
| 1 | 0 | 0 | 1 | light blue |
| 1 | 0 | 1 | 0 | light green |
| 1 | 0 | 1 | 1 | light cyan |
| 1 | 1 | 0 | 0 | light red |
| 1 | 1 | 0 | 1 | light magenta |
| 1 | 1 | 1 | 0 | yellow |
| 1 | 1 | 1 | 1 | high intensity white |

Some possible CGA colors and variations.

| Binary | Hex | Color effect |
|---|---|---|
| 0000 0000 | 00 | Black on black |
| 0000 0001 | 01 | Blue on black |
| 0001 0010 | 12 | Green on blue |
| 0001 0100 | 14 | Red on blue |
| 0001 1111 | 1F | High-intensity white on blue |

# Example

Write a program using INT 10H to

a) Change the video mode

```
MOV AH,00
MOV AL,03
INT 10H
```

# Example

Write a program using INT 10H to

b) Display the letter “D” in 200H locations with attributes black on white blinking( blinking letters “D” are black and the screen background is white)

```
MOV AH, 09 ; Display Option
MOV BH,00 ; Page 0
MOV AL,44H ; ASCII Character for letter D
MIV CX, 200H; Repeat it 200H times
MOV BL,0F0H; Black on white blinking
INT 10H
```

# Draw Box

Ah, 06 Function/Routine to request to Scroll lines up
Al number of lines to be scrolled, lines to be filled
BH: color attribute
CH: Top row of window
CL: Left most column of window
DH: Bottom row of window
DL: right most column of window

# References

- https://www.ic.unicamp.br/~celio/mc404-2004/service_interrupts#attrib

- The x86 PC Assembly Language, Design and Interfacing by Muhammad Ali Mazidi, Janice Gillispie Mazidi, Danny Causey

innovate achieve lead

BITS Pilani

Pilani Campus

# Thank You