III. STRUCTURED ASSEMBLY LANGUAGE PROGRAMMING TECHNIQUES

Memory-mapped Controls
and Direct Port Access

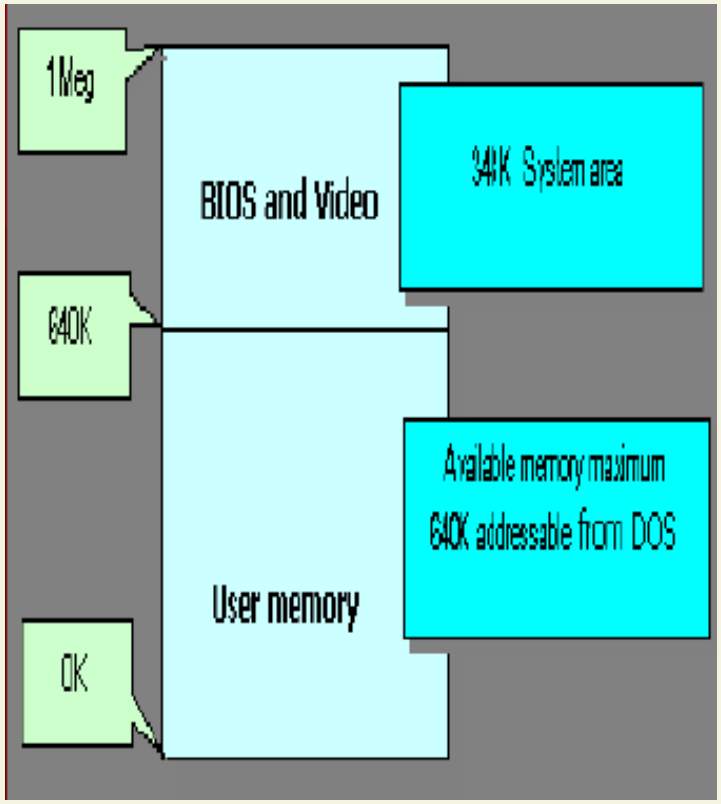
Memory-mapped Control

- One of the various interactions between hardware and software in a working computer system involves mapping device attributes to certain memory locations.

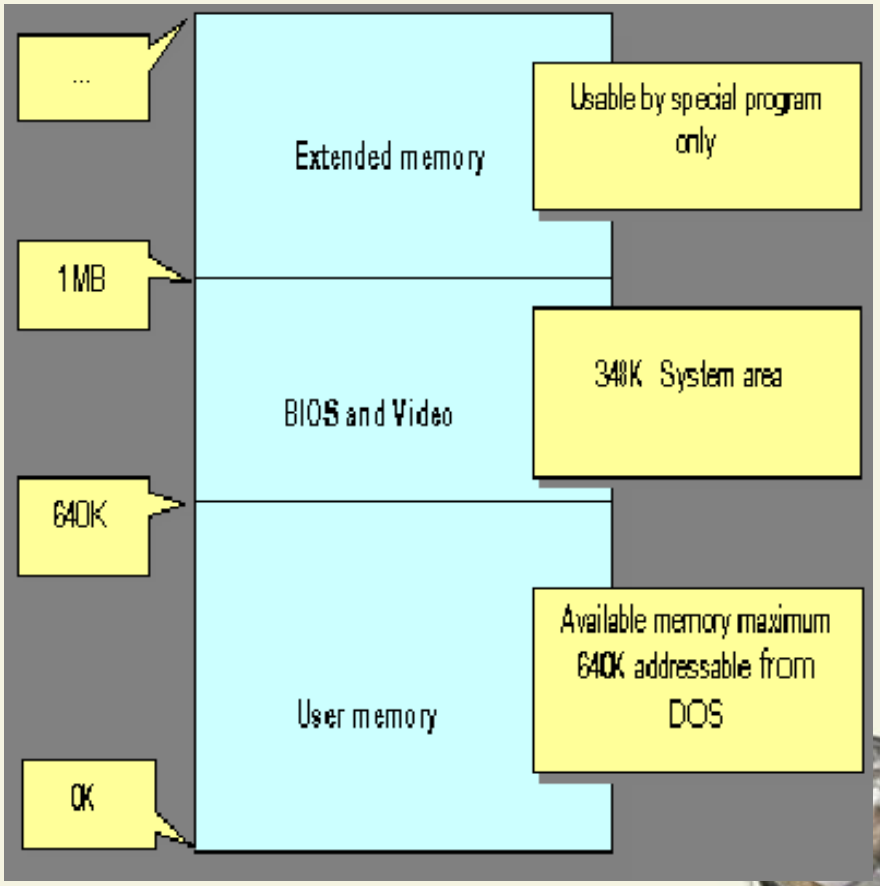
- Controlling the data stored in such memory locations would directly affect the device.

DOS Memory Layout

DOS Conventional Memory



DOS with Extended Memory



DOS Video Buffer

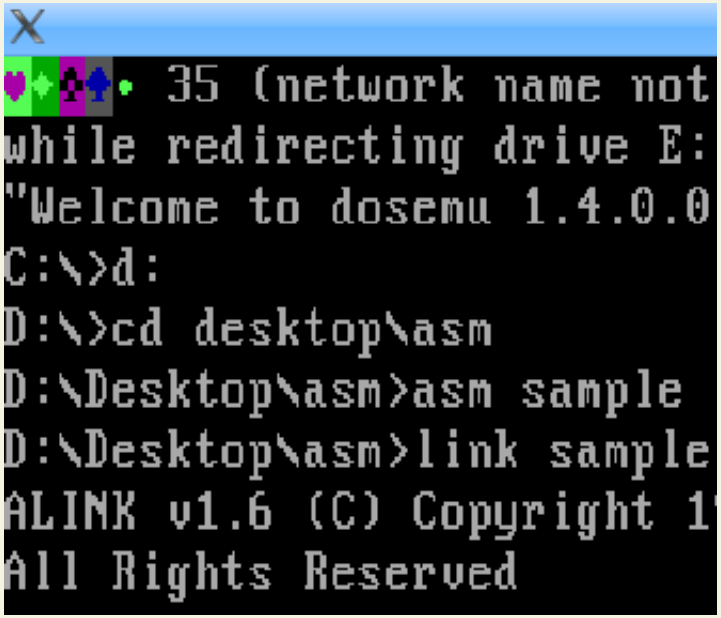
- Located at segment B800h

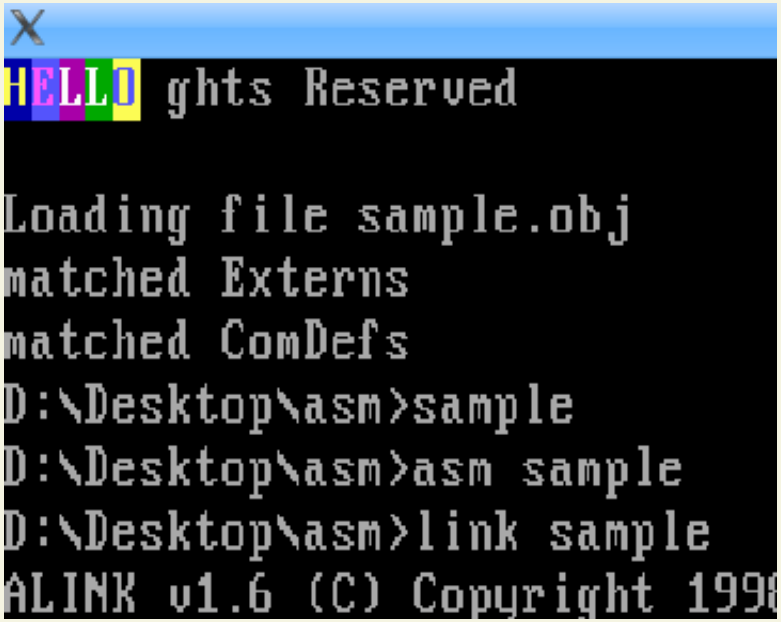
```
..start:

        mov ax, 0B800h
        mov ds, ax

        mov bx, 0

        mov ah, 0A5h
        mov al, 03h

        mov cx, 5
```

```
aloop:
        mov [bx], ax
        add bx, 2
        rcl ah, 3
        inc al

        loop aloop

mov ah,4Ch
int 21h
```

DOS Video Buffer

- Located at segment B800h





Direct Memory Access

- Video buffer is directly accessed by the video/monitor.

- Direct memory access (DMA) allows certain hardware subsystems within a computer to access system memory for reading and/or writing independently of the CPU.

Direct Port Access

- At the lowest level, a program communicates with a device by reading and writing through a port, a connection through which data passes to or from a device.
- Each port has a 16-bit identification number in the range 0h to FFFFh.
- Some devices use several numbers, each for a different purpose.

Direct Port Access

- To read from a port, the IN instruction can be used. To write to a port, we may use the OUT instruction.

- The formats are:

  IN register, port

  OUT port, register

  where register used is AL and port is a port number.

  Ex: OUT 61H, AL -write to speaker port

Direct Port Access

```
segment code

..start:

;beep 523 pulses per second for 655.35 seconds
.       mov ax, 523
.       mov bx, 65535
.       call beep

.       mov ah, 4ch
.       int 21h

beep:

;save requested frequency in CX
.       mov cx, ax

;prepare timer to accept frequency setting
.       mov al, 0B6H
.       out 43h, al

;calculate frequency setting from requested frequency
.       mov dx, 0012h
.       mov ax, 34dch
.       div cx

;write frequency setting to timer
.       out 42h, al
.       jmp short $+2
.       mov al, ah
.       out 42h, al

;get the current speaker setting and save in ah
.       in al, 61h
.       mov ah, al

;turn on the speaker
.       or al, 03h
```