Lecture # 26

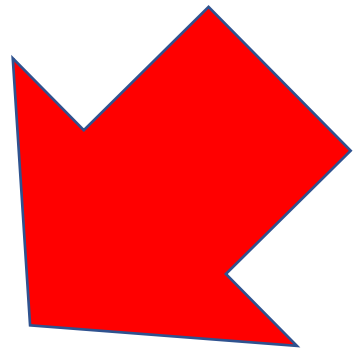
Hardware Control using I/O Ports

Textbook chapter #17, section 17.5

Download Textbook (7th Edition) from Slate

17

Expert MS-DOS Programming



IN and OUT Instructions The IN instruction inputs a byte, word, or doubleword from a port. Conversely, the OUT instruction outputs a value to a port. The syntax for both instructions is

```
IN    accumulator,port
OUT   port,accumulator
```

Port may be a constant in the range 0 to FFh, or it may be a value in DX between 0 and FFFFh. *Accumulator* must be AL for 8-bit transfers, AX for 16-bit transfers, and EAX for 32-bit transfers. Examples are as follows:

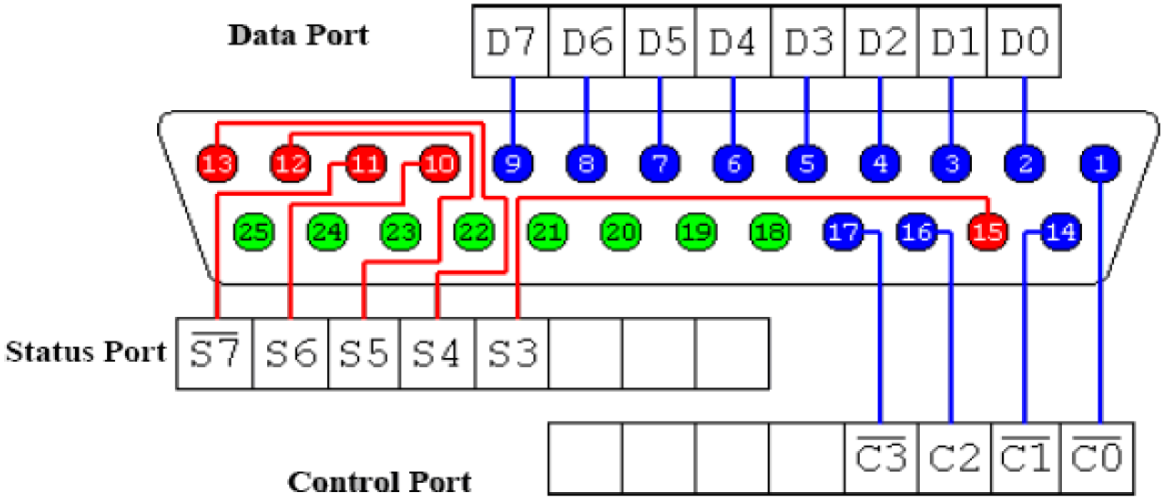
```
in    al,3Ch                ; input byte from port 3Ch
out   3Ch,al                ; output byte to port 3Ch
mov   dx, portNumber        ; DX can contain a port number
in    ax,dx                 ; input word from port named in DX
out   dx,ax                 ; output word to the same port
in    eax,dx                ; input doubleword from port
out   dx,eax                ; output doubleword to same port
```

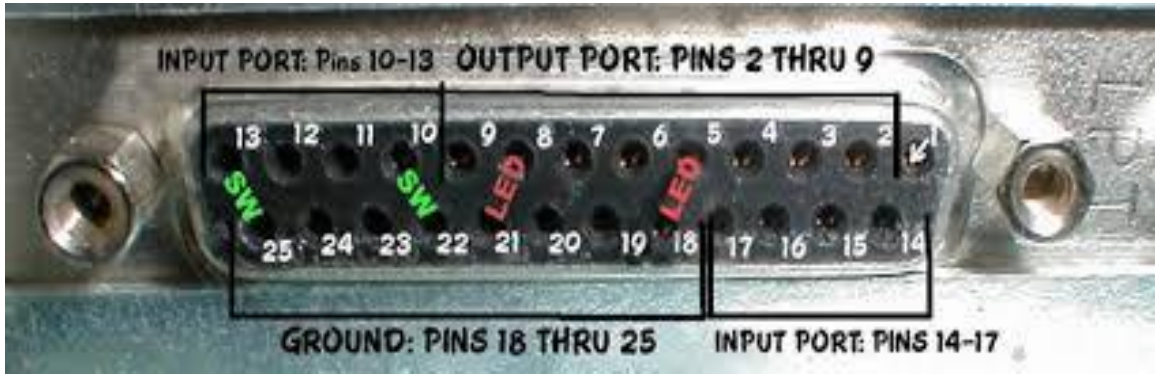
Mnemonic	Description
`OUT imm8, AL`	Output byte in AL to I/O port address imm8.
`OUT imm8, AX`	Output word in AX to I/O port address imm8.
`OUT imm8, EAX`	Output doubleword in EAX to I/O port address imm8.
`OUT DX, AL`	Output byte in AL to I/O port address in DX.
`OUT DX, AX`	Output word in AX to I/O port address in DX.
`OUT DX, EAX`	Output doubleword in EAX to I/O port address in DX.

Mnemonic	Description
`IN AL,imm8`	Input byte from imm8 I/O port address into AL.
`IN AX,imm8`	Input byte from imm8 I/O port address into AX.
`IN EAX,imm8`	Input byte from imm8 I/O port address into EAX.
`IN AL,DX`	Input byte from I/O port in DX into AL.
`IN AX,DX`	Input word from I/O port in DX into AX.
`IN EAX,DX`	Input doubleword from I/O port in DX into EAX.

The original IBM-PC's parallel port had a total of 12 digital outputs and 5 digital inputs accessed via 3 consecutive 8-bit ports in the processor's I/O space.

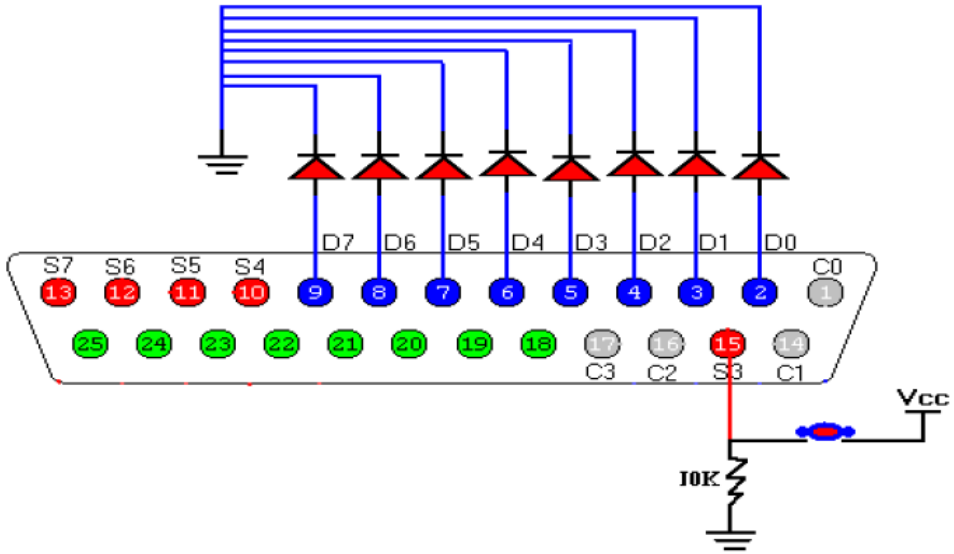
Port	Direction	address	# of used bits	Used Bits
Data	output	378h	8	D0-D7
Status	input	379h	5	S3-S7
control	output	37ah	4	C0-C3





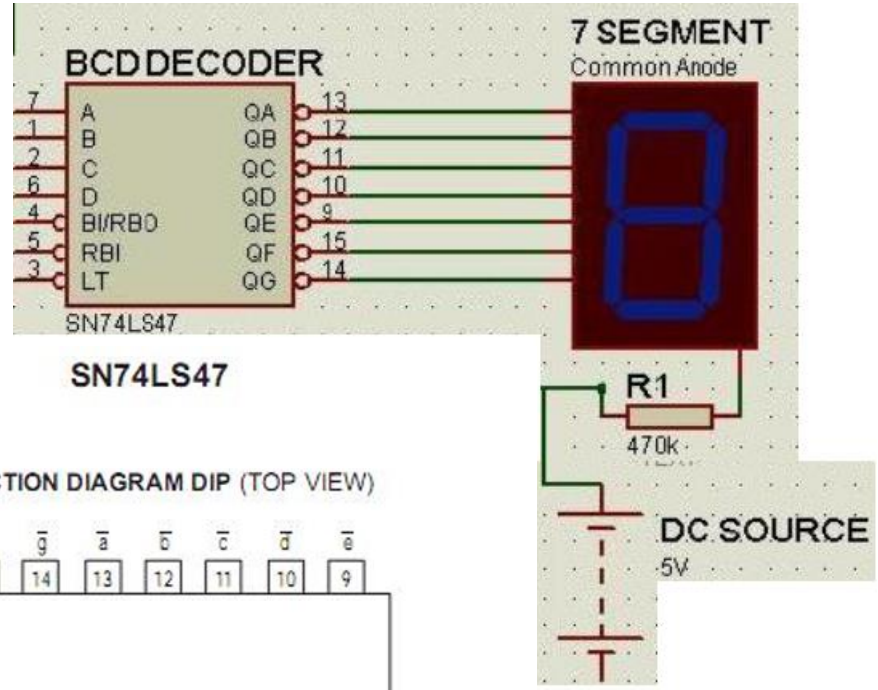
```
                .code
x1:             mov dx,379h ; Input port
                in al,dx
                and al,00001000b
                cmp al,08H
                 je light
                mov al,00000000b
                jmp end1
light:          mov al,11111111b
end1:           mov dx,378H ; Output port
                out dx,al
                 jmp x1
end:            mov ah,4ch
                int 21h
                END
```

DOS way to exit a program

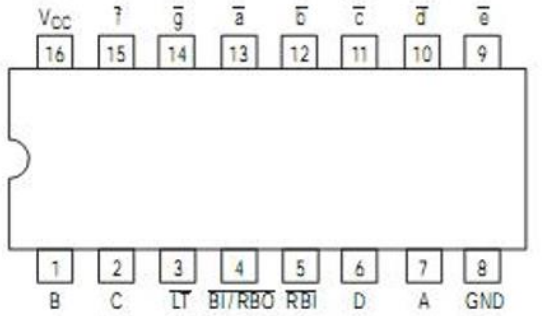


Direction	address
output	378h
input	379h
output	37ah

Question: Suppose EAX = 87345629. Display this number to to port 378h with a delay of one second. Use Irvine library for delay call Delay with EAX value of 1000 (for 1000 millisec=1 sec).



CONNECTION DIAGRAM DIP (TOP VIEW)

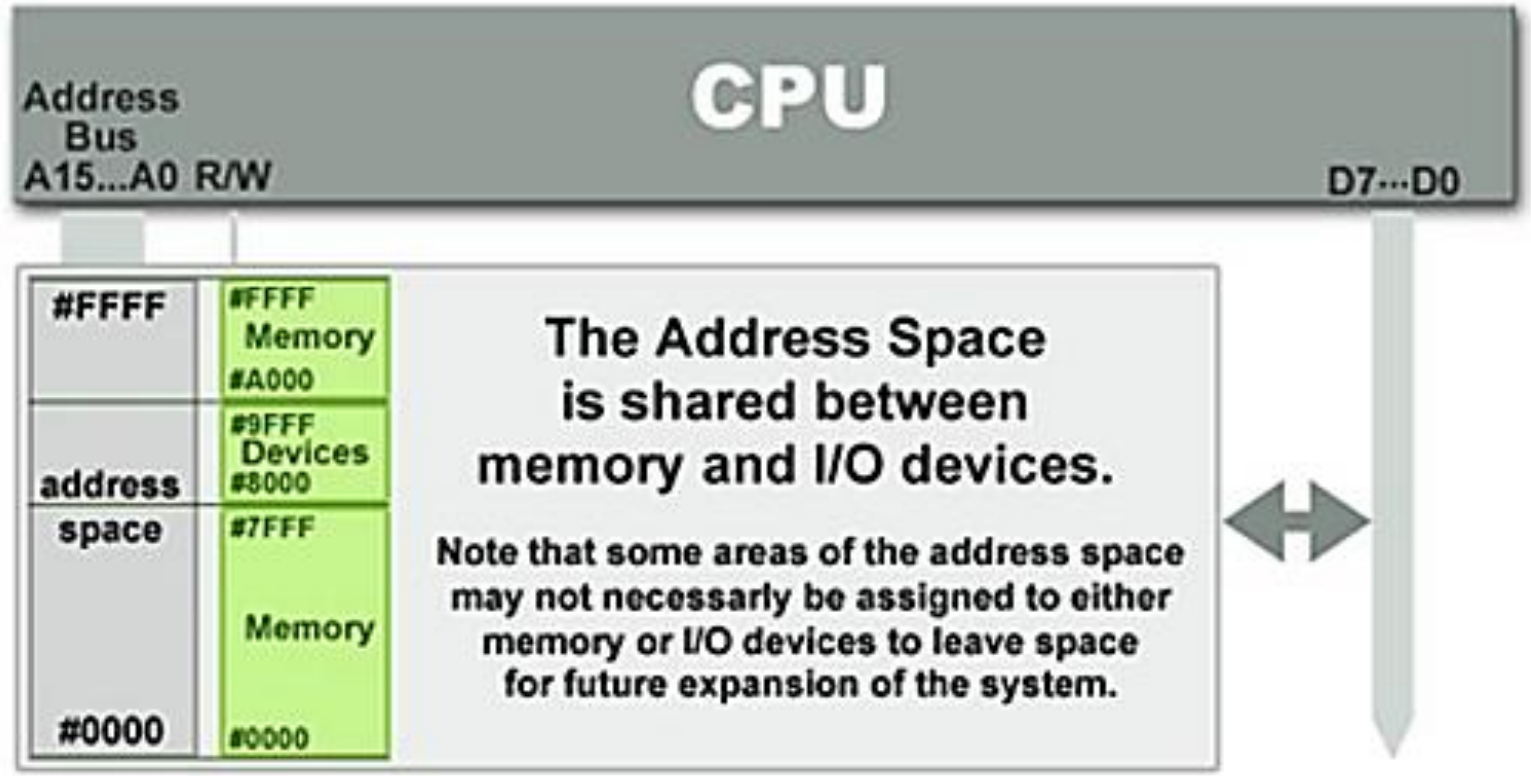


PIN NAMES		LOADING (Note a) HIGH	LOW
A, B, C, D	BCD Inputs	0.5 U.L.	0.25 U.L.
$\overline{RBI}$	Ripple-Blanking Input	0.5 U.L.	0.25 U.L.
$\overline{LT}$	Lamp-Test Input	0.5 U.L.	0.25 U.L.
$\overline{BI/RBO}$	Blanking Input or Ripple-Blanking Output	0.5 U.L. 1.2 U.L.	0.75 U.L. 2.0 U.L.
$\overline{a}$ to $\overline{g}$	Outputs	Open-Collector	15 U.L.

NOTES:
a) 1 Unit Load (U.L.) = 40 μA HIGH, 1.6 mA LOW.
b) Output current measured at V_{OUT} = 0.5 V
The Output LOW drive factor is 15 U.L. for Commercial (74) Temperature Ranges.

Memory mapped I/O

- Some processors have only one address space.
- Use memory address space for I/O devices. How?
- **Reserve Address space slots for Devices.**



Port mapped I/O

- Separate address spaces for Memory and I/O Devices.
- Processor generate R/W for Memory and IORQ to access memory