Microprocessor System Design
Input / Output
Peripheral Interfacing

Omid Fatemi

(omid@fatemi.net)

Outline

- Peripheral devices
  - Input devices
  - Output devices
- 8 bit / 16-bit IO
- Simple Output device - interfacing LEDs
- Simple Input device - interfacing switches

Peripheral

- is an input and/or output device
- like a memory chip, it is mapped to a certain location (called the port address)
- unlike a memory chip, a peripheral is usually mapped to a single location

Output Device

- like a memory chip, you can write to an output device
- You can write to a memory chip using the command `mov [bx], al`
- You can write to an output device using the command `out dx, al`

Input Device

- like a memory chip, you can read from an input device
- You can read from a memory chip using the command `mov al, [bx]`
- You can read from an input device using the command `in al, dx`

Memory mapped vs. peripheral

- Same instruction vs. independent instruction
- Entire address bus vs. part of address bus
- Same control signals vs. independent
- More IO ports vs. 65536 ports
- More commands and operations
- Uses memory space

Two formats for IN / OUT

Format 1

- **IN AL, port#**

Or

- **OUT port#, AL**
- **Example:**
  - **BACK:** **IN AL,22H**
    CMP AL, 100
    JNZ BACK

Format 2

- **MOV DX,port#**
  IN AL, DX

Or

- **MOV DX, port#**
  OUT DX, AL

8bit vs 16bit IO

- 8088 case:
- MOV DX, 648H
  OUT DX, AX ;AX = 76A9H

- Address bus and ALE
- Low byte (A9), IOW
- Setup time
- Address (649) and ALE
- High byte (76), IOW
- Setup time

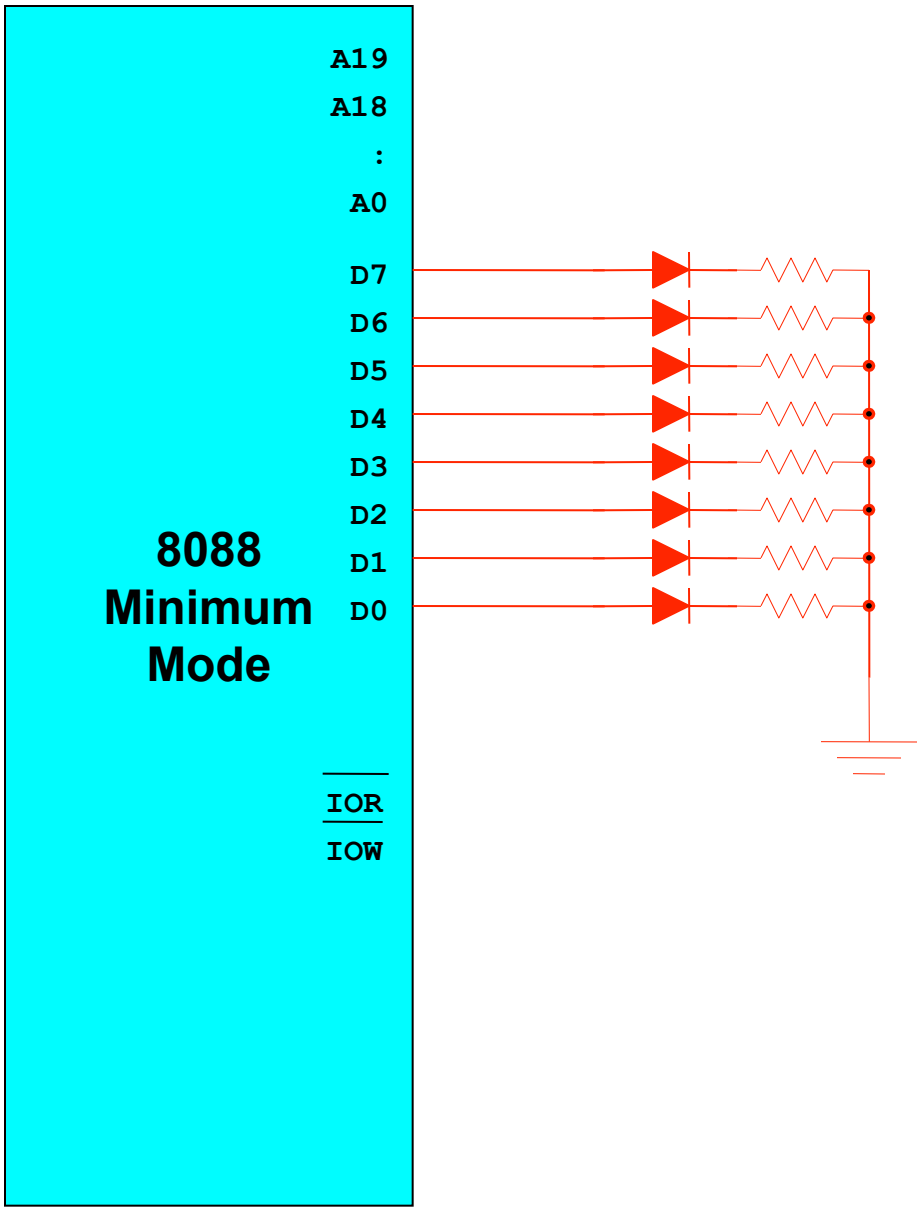
- 8086 case:
- MOV DX, 648H
  OUT DX, AX ;AX = 76A9H

- Address bus and ALE
- Word (76A9), IOW
- Setup time

Creating a Simple Output Device

- **Use 8-LED's**

Use 8 LED's

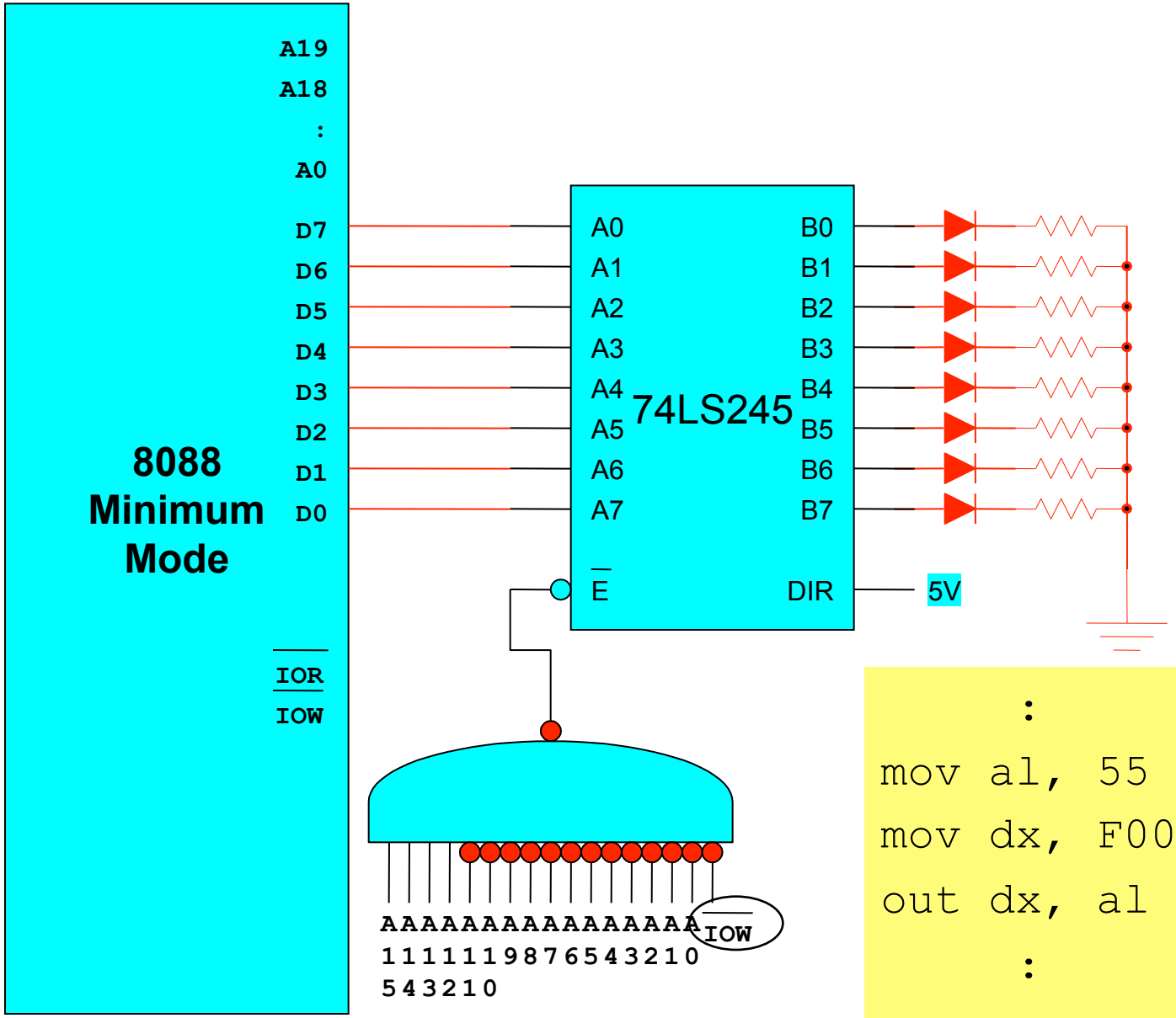



```
        :
mov al, 55
out dx, al
        :
        :
        :
```

Creating a Simple Output Device

- **Use 8-LED's**
- **Use a chip and an address decoder such that the LED's will respond only to the command `out` and a specific address (let's assume that the address is `F000`)**

Use of 74LS245 and Address Decoder

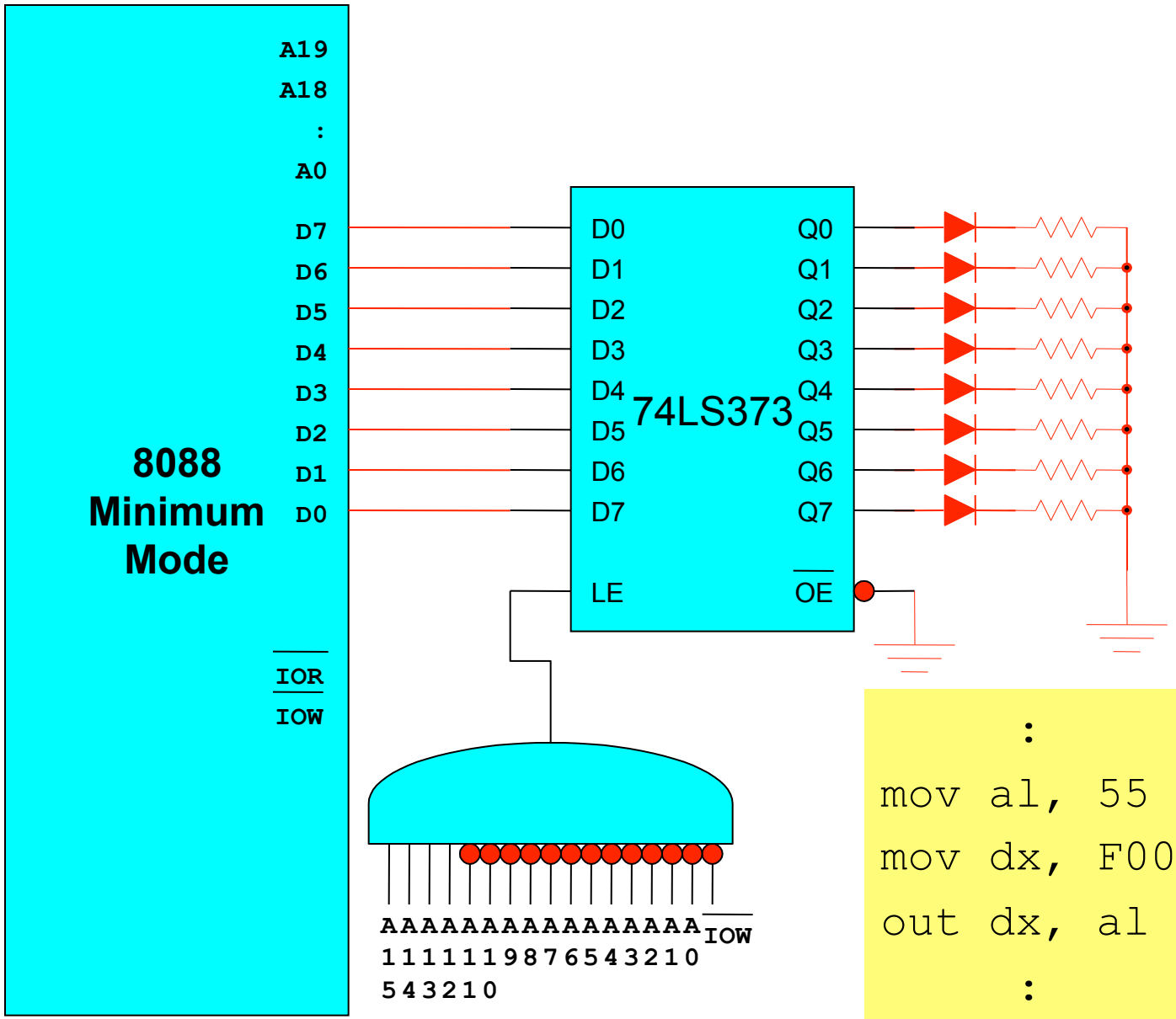



```
      :
mov al, 55
mov dx, F000
out dx, al
      :
```

Creating a Simple Output Device

- Use 8-LED's
- Loses the data
- Solution?
- Use a chip and an address decoder such that the LED's will not only respond to the command `out` and a specific address (let's assume that the address is `F000`) but will also latch the data

Use of 74LS373 and Address Decoder

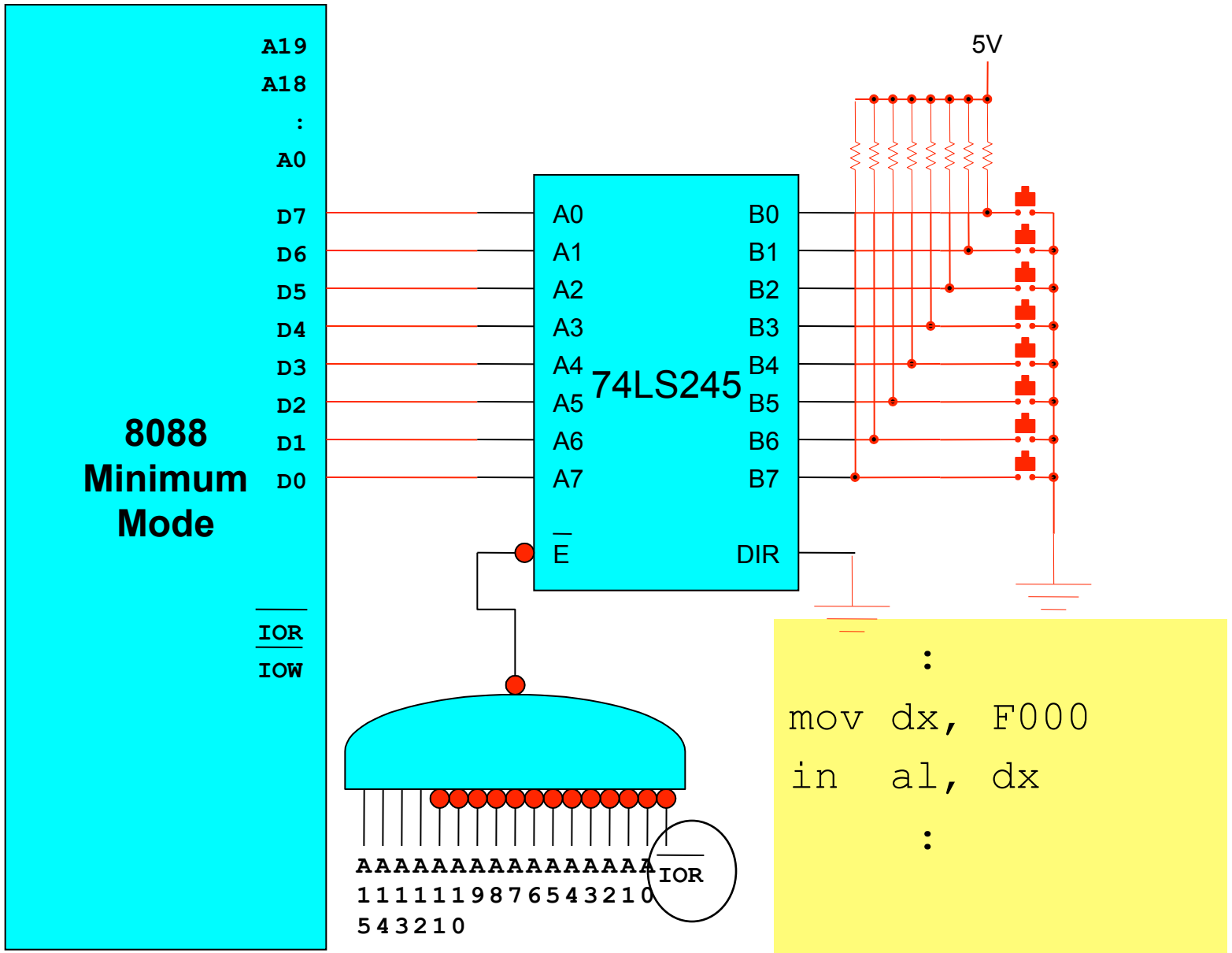



```
        :
mov al, 55
mov dx, F000
out dx, al
        :
```

Creating a Simple Input Device

- **Use 8-Switches (keys)**
- **Use a chip and an address decoder such that the keys will be read only to the command `in` and a specific address (let's assume that the address is `F000`)**
- **How to interface a switch to computer?**

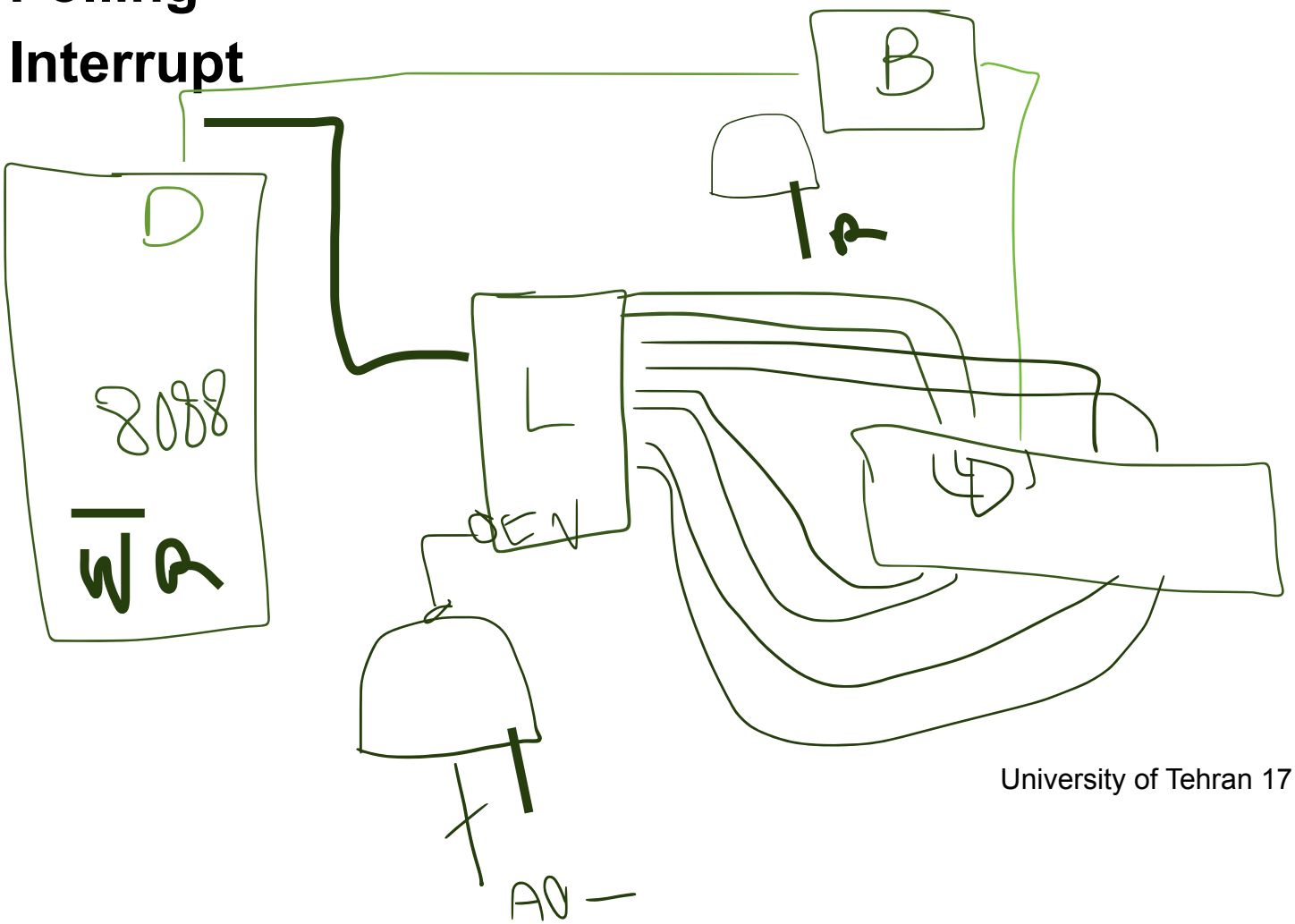
Use of 74LS245 and Address Decoder



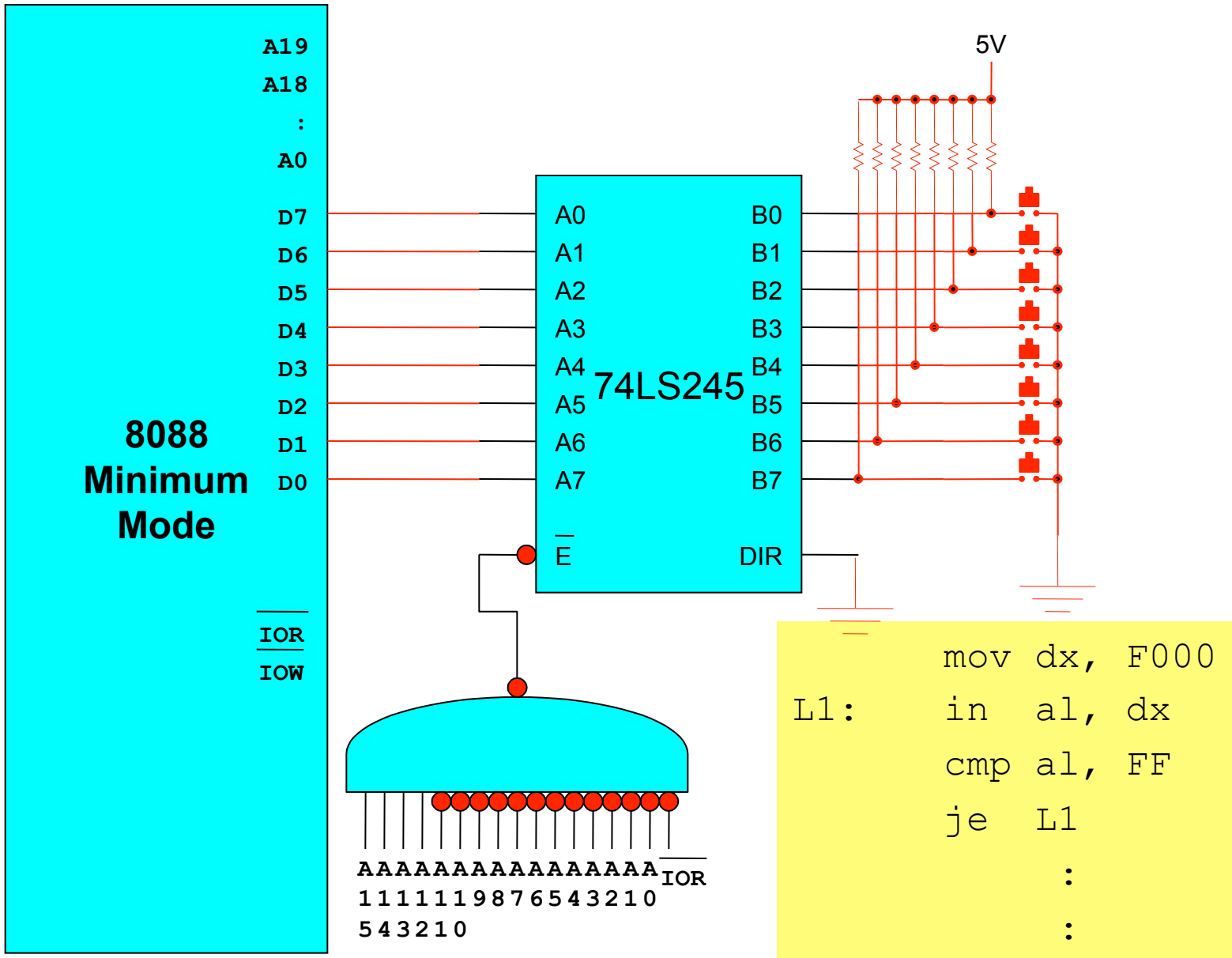
Same address for input and output?

How do you know if a user has pressed a button?

- By Polling
- By Interrupt

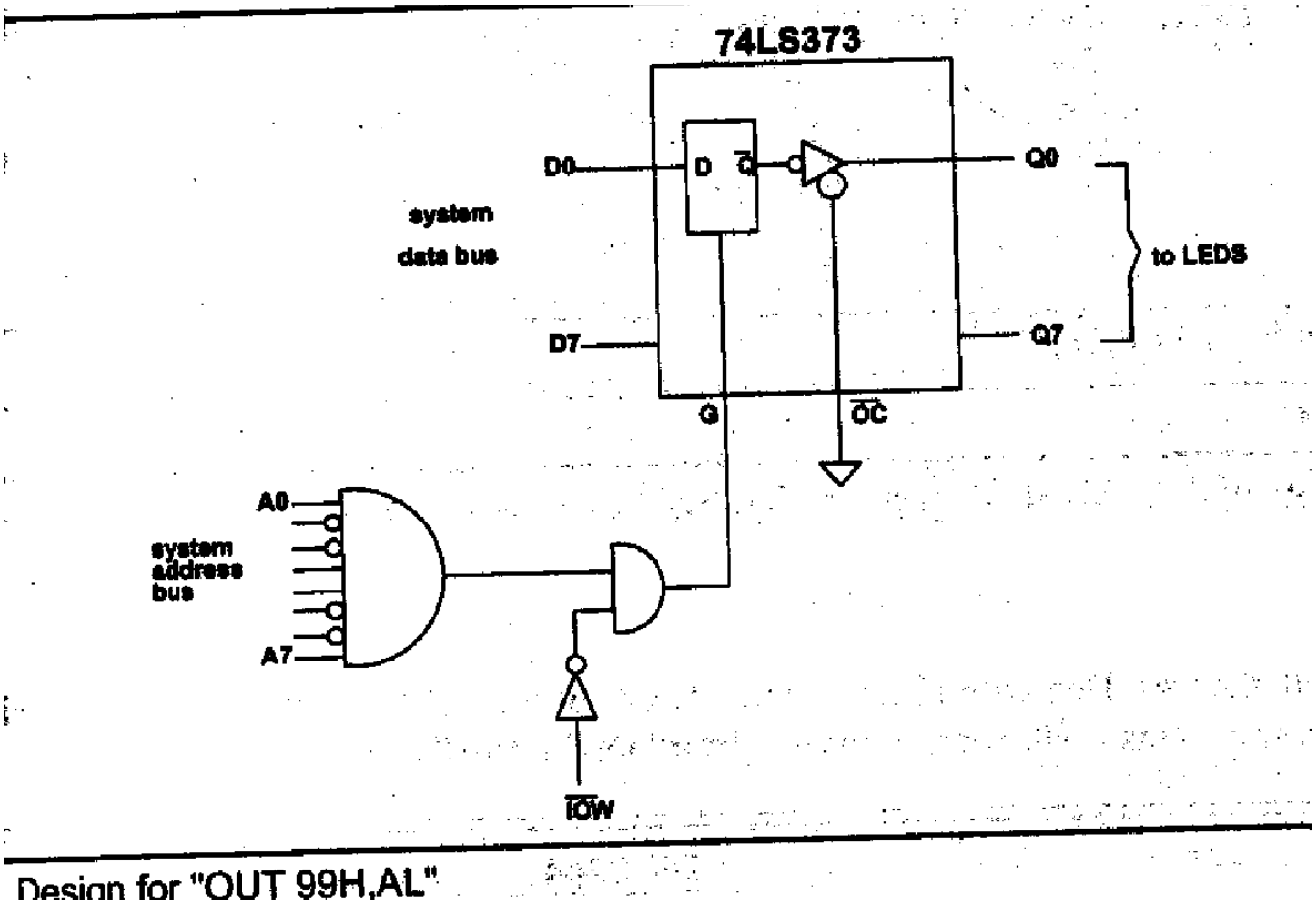


Polling




```
        mov dx, F000
L1:     in  al, dx
        cmp al, FF
        je  L1
            :
            :
```

Output Port Design



Design for "OUT 99H,AL"

T1 – T4 of OUT 99H, AL ?

Input Port Design

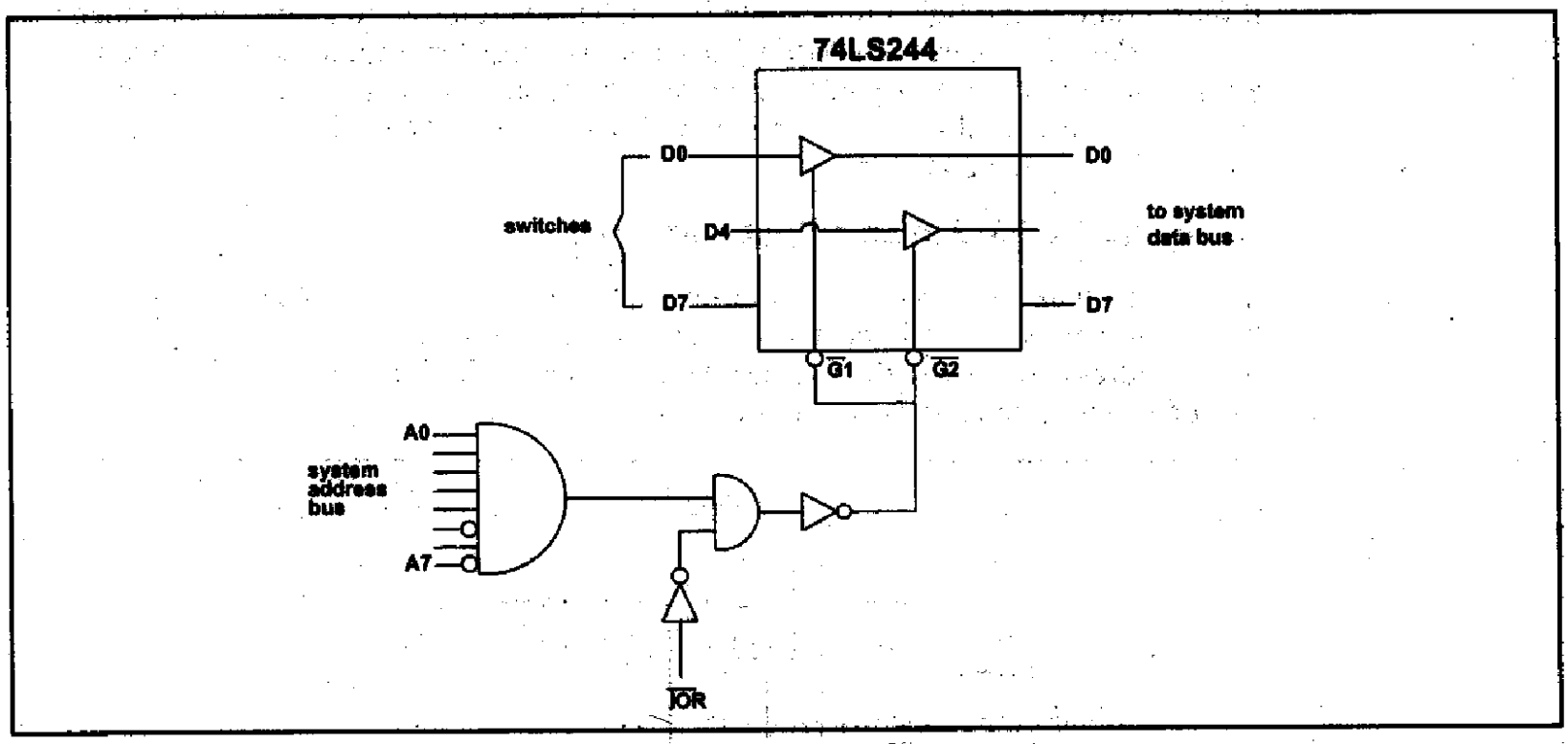


Figure 4-2. Design for "IN AL,5FH"

T1 – T4 of IN AL, 5FH ?