LAB 6: Introduction to I/O programming with STATUS PORT

Objective

1. Understand the I/O Types, Instructions and Interface Circuit.
2. Understand addressing of I/O ports.
In previous experiments, you studied the microprocessor, its registers, its language and interaction with memory. From now on, you will be using the microprocessor to control other devices (Interfacing). Devices are connected via ports, where these ports are connected to microprocessor with an I/O interface circuit. An example of I/O interface circuit is shown in figure 1.

Ports are defined to be a fixed ports or variable ports. Fixed ports have fixed addresses and fixed device connected to them, while variable ports can take different addresses with an open connection. In this lab, you will practice control the blinking of Status Port lights. The actual function of status port is to indicate the current status of keyboard.

To control a device, we need to 'tell' the microprocessor where it is located (the address of the device), which can be obtained from the I/O interface circuit. We also need to 'tell' the microprocessor 'what' to do with the 'location'.

**Types of I/O**

I/O are categorized based on the I/O address are mapped with memory addresses. If the I/O addresses are treated separately from Memory addresses, then its isolated/peripheral I/O else its Memory mapped I/O. 8088 implements the early type of I/O with address ranging from 0000 (h) – FFFF (h) meaning its supports 16bit wide address.

**I/O instructions.**

When you write to or read from memory, you use the following statements.

\[
\text{MOV AX, [address]} \\
\text{MOV [address], AX}
\]

When using an address that is stored in another register, for example, BX, you use the following statements.

\[
\text{MOV AX,[BX]} \\
\text{MOV [BX],AX}
\]

When interacting with devices, you need to use new statements. Normally, the address of the device is stored in DX. To write to a device (output):

\[
\text{OUT DX,AL}
\]

To read from a device (input)

\[
\text{IN AL,DX}
\]

**Status Port**

In this lab, we will practice a very simple way of interfacing, that's to control simple LED. For this purpose we will utilize the status port, which the actual function of the status port is to indicate the current state of keyboard.
**Step 1: Determine address of Status Port**

From the Figure 1, determine the range of possible addresses for the device.


**Step 2: Determine data words**

From the Figure 2, determine the data word that will switch on each of the LEDs. D0 corresponds to bit 0, D1 corresponds to bit 1, and so on. These words will be our output from AL.

D0 – Caps Lock  
D1 – Insert  
D2 – Print  
D3 – Speaker

What data word will switch them all on?

Caps Lock: ____________________________________________________________

Insert: ______________________________________________________________

Print: ________________________________________________________________

Speaker: _____________________________________________________________

**Step 3: A simple output program**

Execute the following program.

```
MOV DX,FF70  
MOV AL,FE  
OUT DX,AL
```

**Question:** In the above program, what is FF70? What is FE? Why do we use AL instead of AX? Which light is switched on?
Repeat the program for by replacing FE with FD, FB and F7. Verify the results for each.

Step 4: Delay

When you are controlling mechanical devices such as speakers, you need to be aware that mechanical devices cannot move at the same speed as electronic devices. So, to allow the device to respond, you should create a delay between two outputs. A simple way to cause a delay is to execute meaningless statements.

Another reason to delay the output is so that you can slow down the output in order to observe it. Execute the following program:

```
** MOV DX,FF70
    MOV AL,FE
    OUT DX,AL
    MOV CX,1F40
## LOOP ##
    MOV AL,FF
    OUT DX,AL
    MOV CX,FFFF
@ LOOP @
    JMP **
```

**Question:** Using the program, explain what happens. Repeat with F7 instead of FE. Which line(s) is/are the delay(s) in the above program? How can you increase the delay? How can you reduce the delay? Test your answers. What other line can you put ** to give the same result? Why?
Examine the program below. Do not execute it.

```
MOV DX FF70
MOV AL,00
* OUT DX,AL
MOV BX,N1
** MOV CX,N2
*** DEC CX
  JNZ ***
  DEC BX
  JNZ **
  NOT AL
  AND AL,04
  JMP *
```

Which statements represent the delay? What is the approximate time the delay takes (hint: letting \( k \) be the time taken to execute DEC CX; JNZ ***, find the length of the delay in terms of \( k \), \( N1 \) and \( N2 \)).

---

**Question:** Write a program of a ‘racing light” (that switches the LEDs and speaker one at a time)
Question:

1. Explain: isolated Input/Output and Memory-Mapped Input/Output

2. Write a program to read in contents of ports 1 and 2 in circuit Figure 10.1 \(^1\) (new text book) and save them at memory addresses F0 (h) and F1 (h).

3. By referring to Figure no.1 (Manual), why input of logic 0 will on the lights?

4. For Figure 10.1, determine the device address if Port 1 is to be selected?

5. If address in Figure 10.12 to be 800A (H), what is the input and which ports?

6. If line A1 and A3 which logic 1, is to be connected into 74ls138 in Figure 1 (Manual), suggest an appropriate logic gate and where they will be connected to?

7. Examine the figure above. Its designed for instruction "OUT 99, AL". How do we obtain address 99H for the above figure.

\(^1\) Figure 8.13 in old textbook