Short Notes Chapter 9 (Lecture 24-27):

INT pin:

There is a single pin outside the processor called the INT pin that is used by external hardware to generate interrupts.

PIC:

Since there are many devices generating interrupts and there is only one pin going inside the processor and one pin cannot be technically derived by more than one source a controller is used in between called the Programmable Interrupt Controller (PIC).

- It has eight input signals and one output signal.
- It assigns priorities to its eight input pins from 0 to 7
- The eight input signals to the PIC are called Interrupt Requests (IRQ).
- The eight lines are called IRQ 0 to IRQ 7. These are the input lines of the 8451.

For example:

- **IRQ 0** is derived by a timer device. The timer device keeps generating interrupts with a specified frequency.
- **IRQ 1** is derived by the keyboard when generates an interrupt when a key is pressed or released.
- **IRQ 2** is the cascading interrupt connected to the output of the second 8451 in the machine.
- **IRQ 3** is connected to serial port COM 2.
- **IRQ 4** is connected to serial port COM 1.
- **IRQ 5** is used by the sound card or the network card or the modem.
- **IRQ 6** is used by the floppy disk drive.
- **IRQ 7** is used by the parallel port.

IRQ conflict:

An IRQ conflict means that two devices in the system want to use the same IRQ line.

Mapping:

Each IRQ is mapped to a specific interrupt in the system. This is called the IRQ to INT mapping. IRQ 0 to IRQ 7 are consecutively mapped on interrupts 8 to F. This mapping is done by the PIC and not the processor.

End of Interrupt:

At the end of servicing the interrupt the handler should inform the PIC that it is completed so that lower priority interrupts can be sent from the PIC. This signal is called an End of Interrupt (EOI) signal and is sent through the I/O ports of the interrupt controller.

Every interrupt handler invoked because of an IRQ must signal an EOI otherwise lower priority interrupts will remain disabled.

I/O PORTS

For communicating with peripheral devices, the processor uses I/O ports. There are only two operations with the external world possible, read or write.

We have the IN and OUT instructions to read or write from the peripheral address space.

When **MOV** is given the processor selects the memory address space, when **IN** is given the processor selects the peripheral address space.

IN and OUT instructions
The IN and OUT instructions have a byte form and a word form but the byte form is almost always used. The source register in OUT and destination register in IN is AL or AX depending on which form is used.

The port number can be directly given in the instruction if it fits in a byte otherwise it has to be given in the DX register. Port numbers for specific devices are fixed by the IBM standard.

For example: 20 and 21 are for PIC, 60 to 64 for Keyboard, 378 for the parallel port etc.

A few examples of IN and OUT are below:

```
in al, 0x21
mov dx, 0x378
in al, dx
out 0x21, al
mov dx, 0x378
out dx, al
```

PIC Ports

Programmable interrupt controller has two ports 20 and 21. Port 20 is the control port while port 21 is the interrupt mask register which can be used for selectively enabling or disabling interrupts.

How the keyboard controller communicates with the processor?

Keyboard is a collection of labeled buttons and every button is designated a number (not the ASCII code). This number is sent to the processor whenever the key is pressed. From this number called the scan code the processor understands which key was pressed.

For each key the scan code comes twice, once for the key press and once for the key release. Both are scan codes and differ in one bit only. The lower seven bits contain the key number while the most significant bit is clear in the press code and set in the release code.

Parallel port views:

The parallel port has two views:

- The connector that the external world sees and
- The parallel port controller ports through which the processor communicates with the device connected to the parallel port.

Parallel port connector:

The parallel port connector is a 25pin connector called DB-25. Different pins of this connector have different meanings. Some are meaningful only with the printer**. This is a bidirectional port so there are some pins to take data from the processor to the parallel port and others to take data from the parallel port to the processor.

Important pins for our use are

- The data pins from pin 2 to pin 9 that take data from the processor to the device.
- Pin 10, the ACK pin, is normally used by the printer to acknowledge the receipt of data and show the willingness to receive more data. Signaling this pin generates IRQ 7 if enabled in the PIC and in the parallel port controller.
- Pin 18-25 are ground and must be connected to the external circuit ground to provide the common reference point otherwise they won't understand each other voltage levels.

This is the external view of the parallel port. The processor cannot see these pins. The processor uses the I/O ports of the parallel port controller.

The first parallel port LPT1 has ports designated from 378 to 37A. The first port 378 is the data port. If we use the OUT instruction on this port, 1 bit result in a 5V signal on the corresponding pin and a 0 bits result in a 0V signal on the corresponding pin.

Port 37A is the control port. Our interest is with bit 4 of this port which enables the IRQ 7 triggering by the ACK pin.