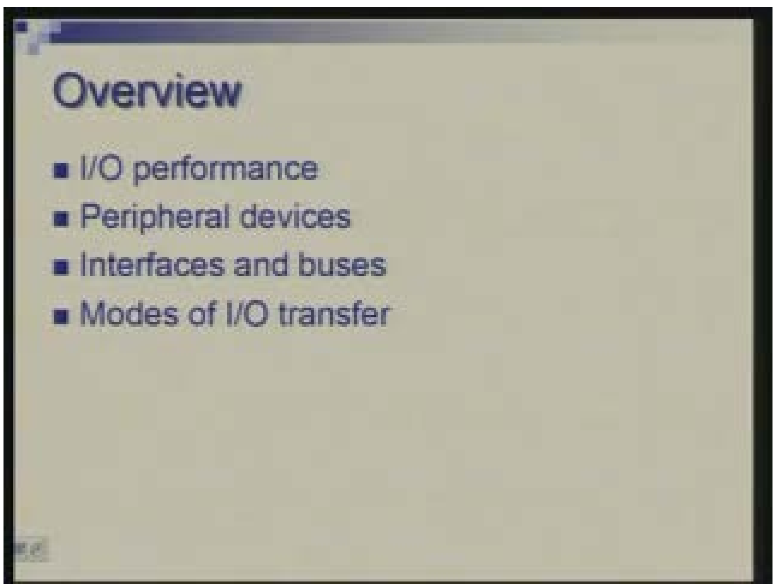
Computer Architecture
Prof. Anshul Kumar
Department of Computer Science and Engineering
Indian Institute of Technology, Delhi
Lecture - 33
Input/Output Subsystem: Introduction

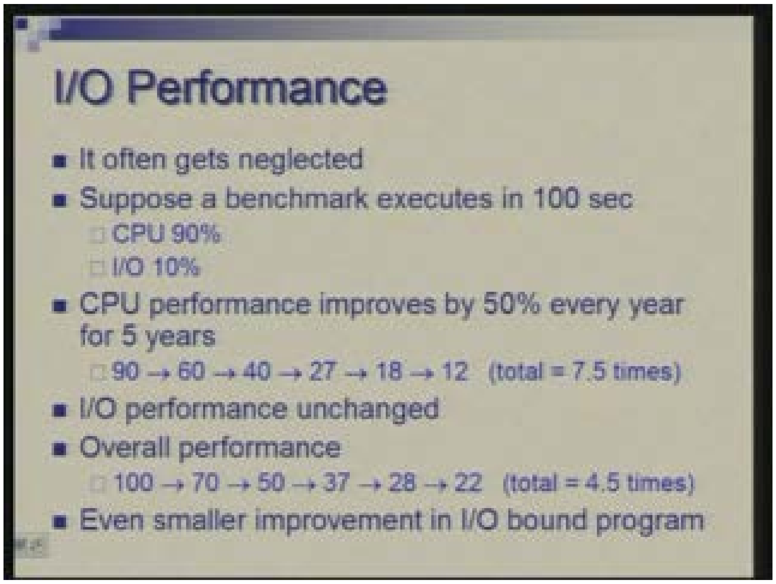
Today we are going to begin with the last topic of this course namely input/output subsystem. We have so far talked about processor and memory. Processor is the central component of the entire system where the programs are executed. Memory is essential in order to hold the program and data on which some instructions operate and input output is important to connect the computer to the rest of the world. If you do not have any means of input and output no information can enter the memory or the processor and no information can come out. So this is essential to make it useful because after all computation involves that you have some data, you process and produce the results. So ultimately you need input and output. We are going to look at various aspects of input/output subsystem.

(Refer Slide Time: 1:45)



We will talk about performance considerations particularly of I/O system. We will talk of peripheral devices which are the extremities of a computer system where the data gets transformed into a form which is understandable by others; it could be humans, it could be the environment or the other information is brought in the form which is understood outside into the computer in form of 1s and 0s. Then we will talk about interfaces and the buses. These are the means to connect the input output devices or peripherals to rest of the system and finally we will look at the operational aspect how I/O or input or output is actually carried out. So our focus today would be on the first two aspects.

(Refer Slide Time: 03:03)



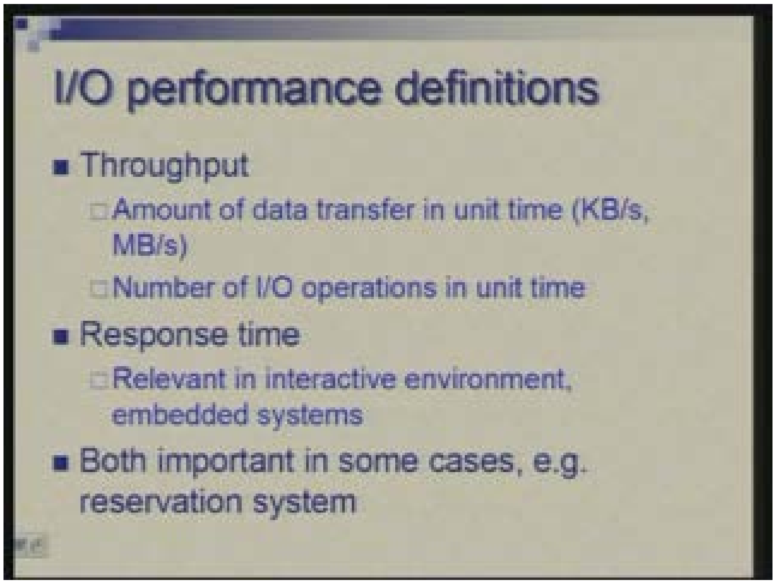
I/O performance is contributed to the overall performance but it often gets neglected. One would talk of speeding up the processor, executing more and more instruction anytime, making access to memory fast. But if you neglect I/O your benefit of performance improvements may not really be fully effective. So consider for example that you have some benchmark program which takes hundred seconds and all out of which the CPU consume 90 percent of the time and 10 percent of the time is taken for I/O that means the data is brought in, results are sent out and we know that by technological innovations the speed of processor is continuously improving and if we have 50 percent improvement every year so over a period of five years we could see that this 90 seconds which the CPU consume would get reduced to 60 40 27 18 and 12 leading to an overall improvement by factor of seven and a half times.

Now, on the other hand, suppose I/O performance remains unchanged then what is the effect of this on the overall performance. So the total time is what we for the CPU and 10 seconds of the I/O remain unchanged so the total figure would be 100, next state goes to 70, next state goes to 50 and so on and finally we have 22 after five years. So now if you look at the ratio of 122 it is only a 4.5 times improvement. So, not improving I/O performance could bring down your expected performance improvements. This was the case where the program is computation bound. Computation bound means that majority of the time is taken in performing computations: arithmetic, logic, decision and so on and there is a comparatively much small amount of I/O.

On the other hand, there are situations which are I/O bound where processing is very little, it is essentially input and output. If you if you look at scientific computation, let us say weather prediction or nuclear modeling and so on, here I/O is very little and most of the time it is number crunching or heavy computation. But in business data processing or database oriented application it is often most of the time gets spent in input and output.

So if the let us say ratios were reversed and it is only 10 percent part which you are improving then obviously there is not much improvement which gets ((ramped)). So it is important to take care of I/O performance improvement also in order to improve the overall performance.

(Refer Slide Time: 05:55)



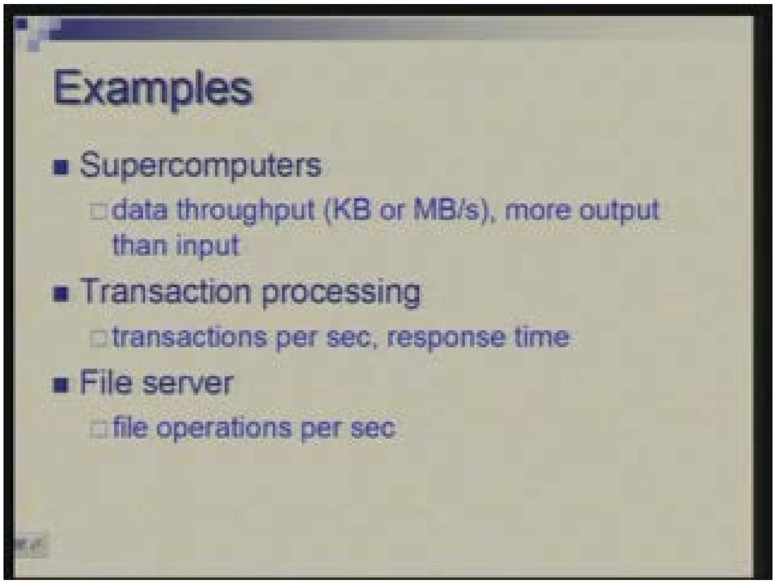
The next question is how do you define I/O performance; what do you mean by I/O performance?
As we saw in case of CPU, depending upon your perspective your requirement there may be different definition of what is I/O performance. It could be throughput oriented measure that means how much of input output gets done, how much of data transfer gets done per unit time. One could talk of amount of data transfer in unit time like kilobytes per second or megabyte per second or you could talk of number of I/O operations per unit time. Each I/O operation may involve some transfer: it could be small, could be few bytes, could be few kilobytes and we want to let us say maximize how many such transfers take place.

On the other hand, it may be the response time which is important. So, from the time you take some action, you give some command and it takes to respond to that so there may be environments where that may be more important. There could also be cases were both are important. For example, if you take reservation system for railways or airlines as a as a user as an individual user you want your job to be done as fast as possible. Once your requirements are fed in you want the system to respond immediately. So, response of course would involve computation as well as input/output.

On the other hand, if you look at the whole system, then there is a central database somewhere where requests are being sent in and it is responding. So at that point, at the point of the central database it is a throughput of I/O which might become bottleneck and you may be trying to improve that. So, at different in different parts of the system or with different perspectives the consideration for I/O performance may be different.

So if you are interested in let us say total time the program takes to execute as we had seen for processor then clearly you can divide the total time between the time which is taken for execution for computation and the time for taking in the data or sending out the data. Now, based on these considerations suppose you look at a supercomputing application or typically scientific application there it may be typically a computation bound application but there also may be at times lots of results or simulation results which may have to be printed out. So here it is not the response time but again the data throughput which is important which may be measured in kilobytes or megabytes per second.

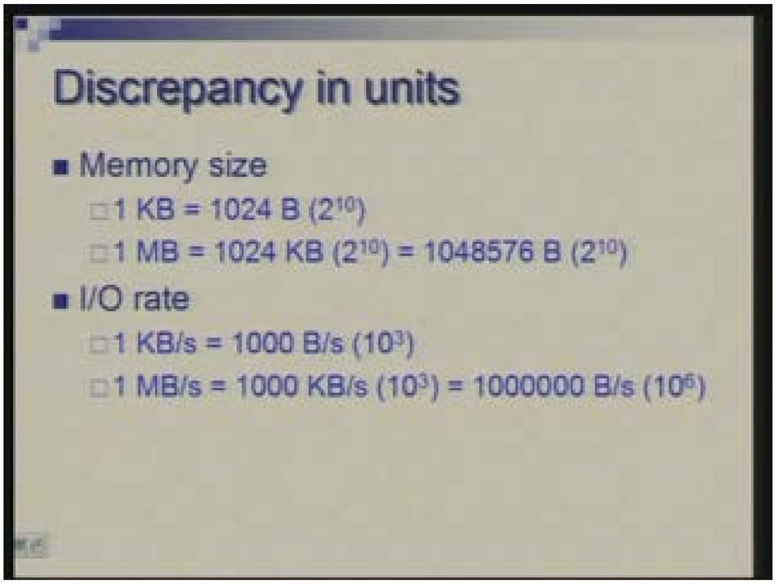
(Refer Slide Time: 08:32)



Transaction processing, for example, let us say ATM, bank ATM. So here transaction per second is important from the overall system point of view and again for a person, when you feed in a request for drawing some money you want it to respond immediately so response time and transactions per second both are important.

Take example of a file server. Suppose you have a cluster of computers as we have in our department and there is one of the machines which is acting as a file server maybe in your homes are there so this file server would receive request for creating files, opening files, reading, writing and so on each such request is essentially an I/O operation. so your concern would be how many I/O operations the system is able to sustain and that could be one thing you would like to look at. so depending upon the situation you may describe your I/O consideration, I/O performance in terms of some operations or some data transfer per second or in terms of time it takes to respond to you or some combination of both.

(Refer Slide Time: 10:18)

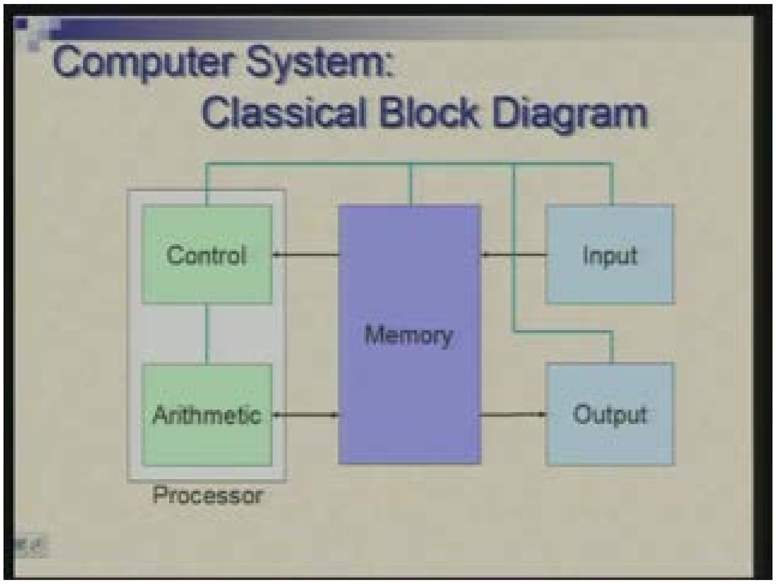


While talking of I/O performance one must keep in mind that often there could be a discrepancy in terms of units. When we have been talking of memory we talk of KB, MB and so on. Once again here we are talking of kilobytes per second, megabytes per second. But typically when you talk of memory size you mean 1kilobyte as 1024 bytes or 2 raised to the power 10. Similarly, 1 megabyte is 1024 kilobytes or 1048576 bytes which are basically which is a power of 2/10th power of 2 and this should have been 20th, it is 20th power of 2 whereas in I/O you are talking of thousands and millions in the real sense so 1 KB per second here is thousand bytes per second or 10 is to the power 3 and similarly 1 megabyte per second is 1000 kilobytes per second or 1 million bytes per second.

Approximately they are same but somewhere if you want to be little more accurate one must keep in mind that there may be a difference; you know the language one is talking in memory domain or in processor domain may be different from what you are talking of in I/O domain so this is just a word of caution.

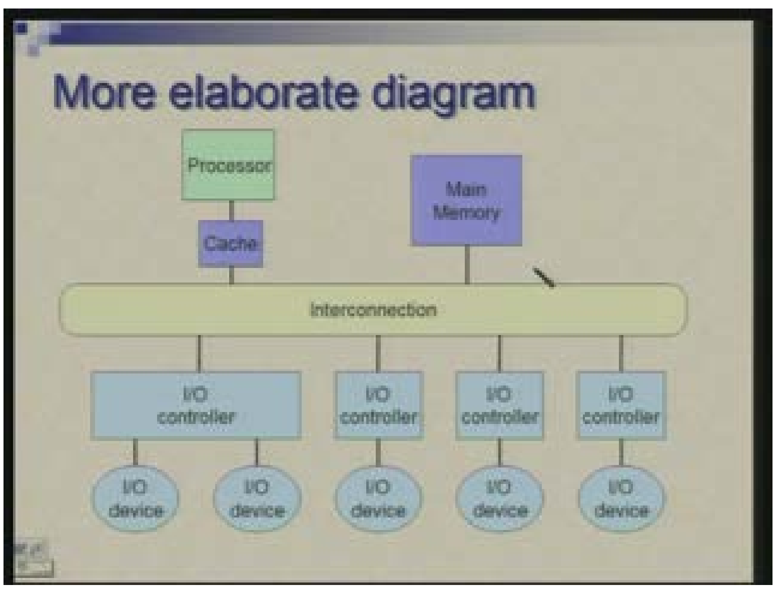
Now let us move towards peripheral devices. Before we do that let us look at the overall system, how we conceive a stored program computed typically.

(Refer Slide Time: 12:05)



This is a classical block diagram showing five units of a computer which probably you would have seen in your school days also. You have control, arithmetic, memory, input and output and they are connected to each other so control and arithmetic are..... the data path has been mentioned here, these form the central processing unit or the central processor. So we are talking of this part which is input and output (Refer Slide Time: 12:31).

(Refer Slide Time: 12:35)



Elaborating it further this is how the picture looks like. I am also bringing in two levels of memory hierarchy here. There is a cache memory and main memory and the interconnection I am not showing point to point here unlike in the previous diagram. Here we put it as something in between the processor, memory and the I/O. what is shown at the bottom is the overall I/O subsystem and somehow they are interconnected. We will elaborate on this in subsequent lectures. But notice here that there are two things involved here in I/O subsystem; they are I/O devices or peripheral devices and there are I/O controllers which which sit between these devices and the rest of the system.

So a controller may be taking care of multiple devices also, that possibility is also shown or a controller may take care of a single device. So we need to understand what these devices and controllers are and we will have some discussion on this today. And in subsequent lectures, we will talk of how these get connected. So we will talk of buses and other means of connection and then finally we talk of how information goes across between processor and these devices or memory and these devices. We will focus now, our attention on devices and controller.

You may often not see these as separate things (Refer Slide Time: 14:13). Physically the controller and device may be all packaged together. So roughly speaking, the devices, the actual transducers so as to say which converts information from one form to another form; let us say printer, you send in some bits and bytes, at some point it is getting converted to a printed page but there is something between the processor and buses on one hand and this printing mechanism, so there is a controller which takes care of movement of data, sometimes even conversion of data in some form.

(Refer Slide Time: 14:57)

So there is a wide variety of peripheral devices and you can try to classify them in many different ways. You could, for example, talk about their behavior, in terms of whether say input that means bring information into processor in memory combination or take information out or play both the roles so that is one aspect. So, for example, keyboard is an input device, printer is an output device.

The second issue is who is the partner, with whom are these devices trying to communicate or with whom do these devices make the processor talk to. So it could be, at the other end it could be a human programmer, human operator or there could be another computer, another machine or the devices could let a computer talk to the environment in general, neither a machine nor a human being or they could be simply some kind of internal devices in the sense that they only store information, disk drive for example. So let me give examples of each of these.

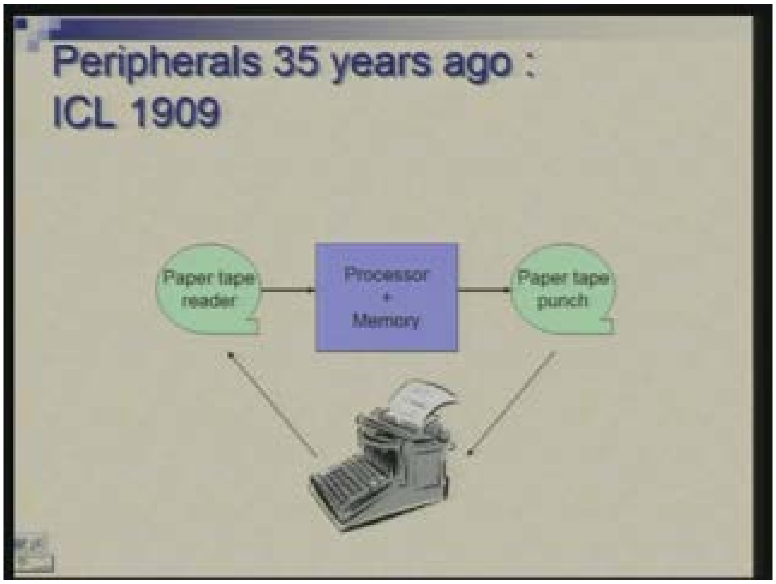
Keyboard is for example for human operators. Network controller will connect one machine to another machine. Environment that is typically in an embedded domain, it could have a sensor for example sensing temperature or computer trying to turn a motor. Storage example is a disk drive. then thirdly they could be characterized in terms of the kind of data which is moving across; in terms of either its speed, some devices are very slow, some are very fast, some could be medium; of course this is a very coarse classification, one could specify the rate of transfer; how many bytes or kilobytes or megabytes getting the device is capable of transferring. So, depending upon how you organize the whole system that capability may not be met, may be met or may not be met. So suppose there is a disk which is capable at the rate of 10 megabytes per second, you have to organize the rest of your system so that this capacity gets utilized; if you do not the disk may be ready to transfer data but you may not be willing to take it.

How much of data gets transferred in in one chunk one quantum?
It could be serial data which means information goes bit by bit as it goes on the network for example or it could be parallel data which may be 8 bits, 16 bits or it could be a larger size.

How it is encoded?
You have 8 bits going, what does one group of 8 bits mean; so different devices may have different kind of encoding and a whole variety exists there.

Now, before we talk of the current situation let me recall how a set of peripheral devices looked like years ago. So, this is the story of this is the picture of IIT Delhi which I am trying to paint when I was exposed to the computer for the first time.

(Refer Slide Time: 18:41)

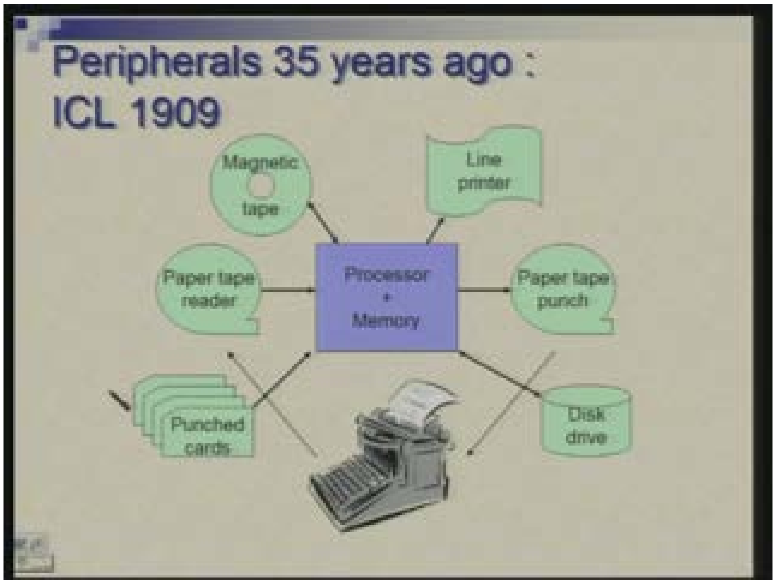


We had not in the present position there was a computer called ICL 1909, it was somewhere on E or F floor, and the only device, the only peripheral device it had was a paper tape reader and paper tape punch. The paper tape basically was just a strip of paper with holes punched on it and group of holes form one character. So the way one could input information was to use a device called........... a typewriter like device, a typewriter like device called flexolator. Basically you could think of a typewriter with a sort of spool of tape attached at the end at one end. So as you type, information could be punched on that. And similarly, if you run a punch tape it could also print out on a sheet of paper.

So, as far as computer is concerned that directly connected peripherals are paper tape reader and paper tape punch. So everything everything was in the form of paper tape; the operating system, the compiler, the user program, the library so everything had to be fed one by one and finally you will your results will be punched on paper tape so there was a larger room roughly of this size and a smaller room. In the smaller room there were a couple of these, typewriter like devices, where you could prepare your programs and when results come back you print them out here.

So, of course, couple of years later the additional devices were introduced, magnetic tape where program and data could be stored for repeated use and this was of course much faster. The paper tape was just.... the reader was a little faster; something like a 300 character per second; punch was only some 10 or 20 character per second and a line printer. Line printer was something like I think 300 lines per minute so it could print one line at a time so that makes life a little easier. And subsequently tape got replaced by punched cards and at that time the system also moved to the current location.

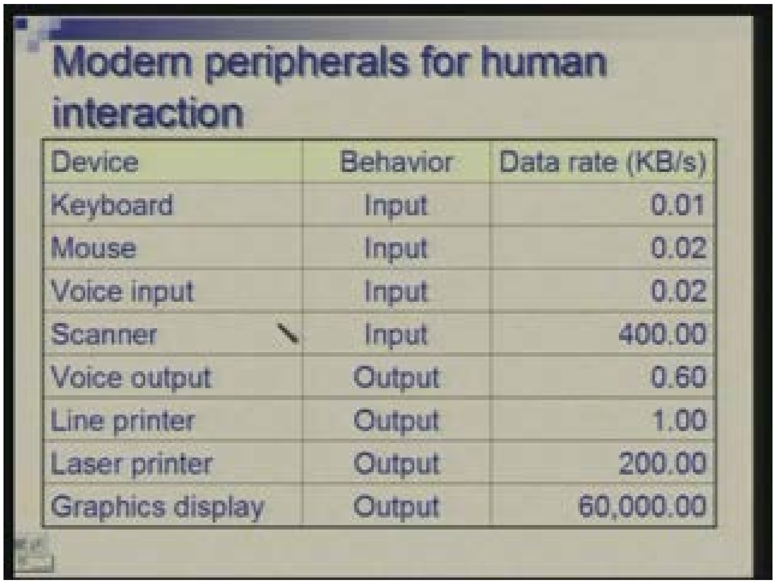
(Refer Slide Time: 21:17)



The punch card's added advantage is it has like a deck of cards; each card basically is one line of your program, one statement. So if you want to add it, you could take out a few cards and you put back new cards; you could you could add it to your program like that. You cannot change your card, but in the deck you can take out a few cards and you can insert a few cards. Of course, editing in paper tape environment was much much more painful, you will have to cut the tape and patch it with something, another piece. So obviously it is not surprising that people some people could look at the tape and read it, look at the punched holes.

Disk drive has again added convenience which was of course much faster than tape; the tape goes absolutely serially whereas in disk you can reach a particular track and then of course within a track you are sequential but you can you do not have to go from one track to another track by going through one track completely; tape is entirely sequential. But the disk capacity of those days were something like 20 megabytes was there, was a big figure obviously because memory was 32 kilobytes or 64 kilobytes of that order.

(Refer Slide Time: 22:48)



So now we fast forward into the modern era, we have whole variety of peripherals and let me list some of these with the kind of classification we did earlier. Let us look at the peripherals you have for introduction with human beings, human programmer or operator. So the important thing here is to.... I mean you are...... I am sure you are familiar with all the names which I am putting here but it is important at this stage to get a feel of what kind of data rate or throughput rate these peripherals are dealing with. Because when you design a complete I/O system you should have a feel of these things.

So devices like keyboard and mouse are limited by the rate at which you can type so it is only a fraction of kilobytes or basically a few bytes per second is the rate at which you can feed information and that is what you require. I mean these are designs to take care of let us say the fastest typist; we do not need anything faster then.

You see here voice input and voice output so the trend is towards developing these type of capabilities; voice input voice output, pictorial input pictorial output which are which are more natural to human being and instead of typing if you can talk to computer it is definitely much nicer. But it is not just an issue of peripheral device, it is also a matter of the recognition of what is being spoken or generating speech outputs so we are not worrying about we are not concentrating on that part but the voice input voice would be digitized and brought to the computer in the form of a sequence of samples.

So, of course the figures given here these are from your textbooks; they are 0.02 bytes per second or 20 bytes per second this is a this is the rate of information if it is compressed. But typically the speech inputs are sampled at least at a rate of 10 kilo ten samples per second and if each sample is 1 byte so it has to be at least 10 kilobytes per second and not 20 bytes per second.

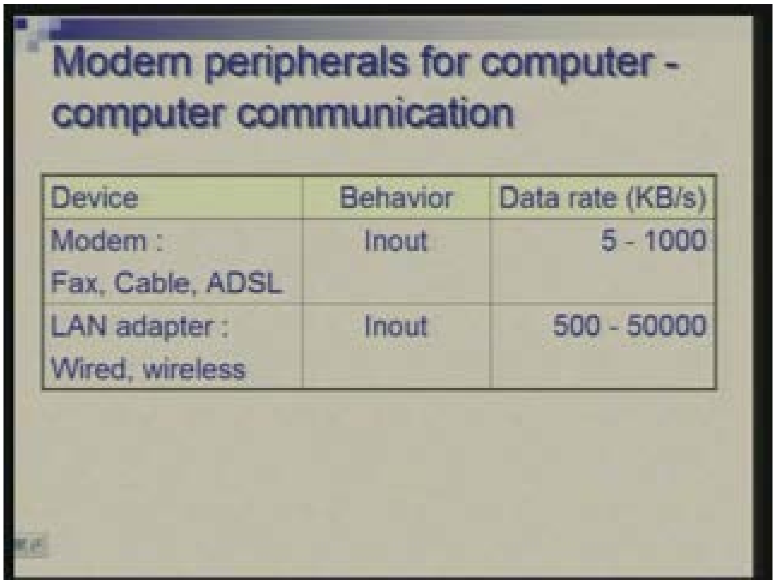
Similarly, a plain speech would be again typically sampled 8 to10 kilobytes per second.

If you are talking of music output, that is sampled at a much higher rate, typically in the range of 40 50 1000 samples per second or so many kilobytes per second. Then another input device is scanner which can basically scan text or image whatever is there in a sheet of paper and depending upon the size and the resolution you may have different data rates. So the resolution or the accuracy here is measured in terms of Dots Per Inch or DPI. You have scanners which are 600 DPI or 1200 DPI and so on, that means how closely or how finely the information can be resolved.
So if this DPI value is lower, then you will get a very sort of coarse picture, if you have higher DPI then you get a finer, shaper picture.

We have variety of output devices, line printers, laser printers, graphic display. As you would notice graphic display is a highly demanding peripheral. So it is shown as 60000 kilobytes per second or 60 megabytes per second. Why is it so high is because it again depends upon the resolution. On a screen suppose roughly speaking suppose you have 1000 by 1000 pixels, it could be more accurately 1024 by 760 or whatever, so as an approximate calculation suppose you have 1 million pixels on the screen then the question is how much how many bits are required to represent each pixel so you would have seen color settings, you could use 1 byte to represent one pixel or you have 24 bits if you are going for a good quality that means 3 bytes per pixel and 1 million pixel means 3 million bytes is the information present on screen and for a persistent display this needs to be repeated refreshed so that you get a persistent picture. So the rate of repetition could be something like twenty-five frames per second or fifty frames per second or sixty frames per second. So if you if you take that into account you realize that it is a really very highly demanding peripheral device.

Then you have a variety of peripheral devices which connect a computer to a network of computers. You have various kinds of modems; it could be fax modem, cable modem; cable modem connects to a TV, a video cable and fax modem connects to a telephone line. ADSL modem also connects to a telephone line which is capable of carrying digital information.

(Refer Slide Time: 29:04)

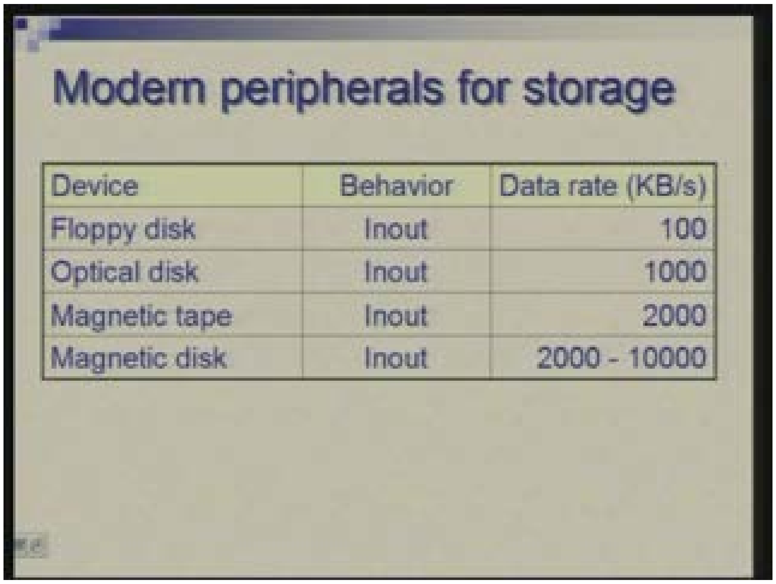


On the other hand, you could have LAN adapters Local Area Network adapters wired or wireless. So these could be, these are typically input and output both and the data rates for LANs are higher depending upon whether it is a 10 megabytes per second LAN or 100 megabytes per second LAN or 1 gigabytes per second LAN so these are the three common standards. So many bits per second can be easily translated into bytes per second by reducing one order, for example. Similarly, you have modems in a wide variety of speeds. The slowest one would be the fax modem which works on telephone lines. So typically they are 56 kilobytes 56 kilobits per second. You might have heard the term baud b a u d so baud is same thing as bits per second; it is a unit used in terms of communication specifying the rates in terms of bits per second. So it is 56 kilo bauds or 56 kilobits per second.

Example of peripheral devices for storage: floppy disk drive, optical disk or CD ROM and DVDs, their storage is in the form of optical so on a disk you have portions which are opaque which are portions which are transparent and depending upon how finely you can resolve, how closely you can space the opaque region and transparent regions you will get more bits per unit area and that will also determine how much of data can be read out as the medium moves.

Migrated tape: As I mentioned earlier they are sequential devices, so information is stored from one end to the other end in a serial form and they are for archival purpose, for long time storage where you are not reading and writing very often. Among these disks are the fastest.

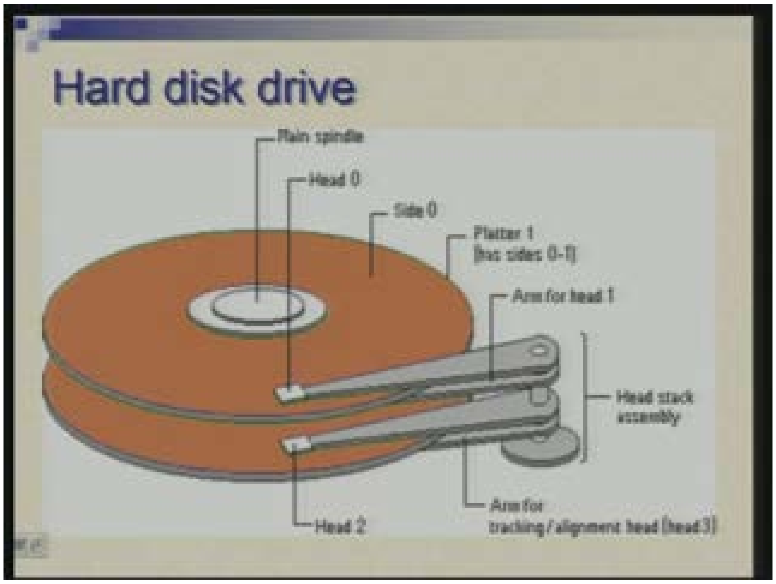
(Refer Slide Time: 31:33)



As you can see that the data rates could be varying from something like 100 kilobytes per second of floppy drive to as much as 1000 kilobytes per second or 10 megabytes per second of hard disk drive.

Another memory you can add to this list is these flash memories or the pen drive USB drive, the kind of which I am carrying in my lecture and plugged into the USB port. So these are...... there is no moving part in that, it is basically a semiconductor memory which is a non-volatile type called flash memory and it is comparatively..................... what would be the transfer rate, I think it will be, it is slightly slower than the migrated disk, I do not have the exact figure. Let us look little bit deeper into some of the peripherals.

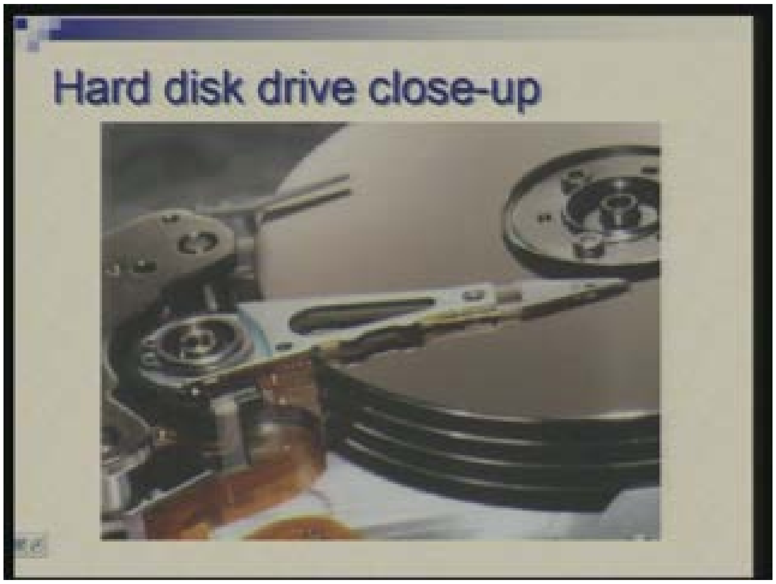
(Refer Slide Time: 32:59)



This is a schematic of the hard disk drive. So basically it is a group of rotating media or platters as they are called. This picture shows two platters. So these are rotating disks where on the surface, information is stored. Each platter typically is capable of storing information on both sides so you can there is some coating which is magnetically sensitive and it is on that 110 gets stored in the form of polarity or magnetization. So there is a there is a set of read write heads which are mounted on an arm, here it is pivoted on this (Refer Slide Time: 33:47) and it rotates around this. By that rotation this end which is the head can move closer to the center or move towards the circumference towards the peripheral. You have circular tracks on the disk, very closely spaced circular tracks and these heads get positioned over some tracks. So all these heads move together. so at one point of time let us say all the heads are positioned on track one of that particular platter then you move them they could be positioned on and all of them could be positioned on their track number 10 and so on so there is a movement of head from track to track and the movement of disk is rotary.

So, if you need to reach some point in the some some information which is stored somewhere here, then this part has to rotate and come in contact with the head, and head also effectively this moves radially and it has to get positioned at that point so once that is done by information can be recorded or read out.

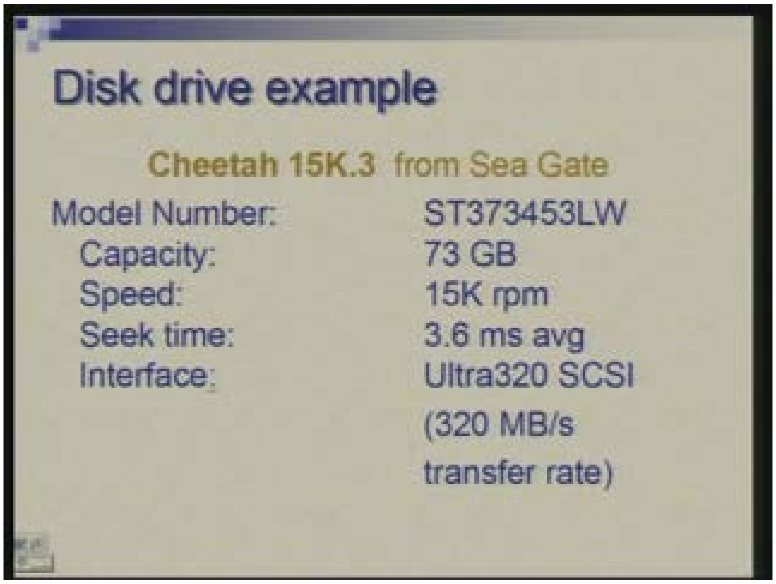
(Refer Slide Time: 35:15)



This is a real picture of a hard disk drive. Here you can see four platters, so there will be eight surfaces eight recording surfaces and this is the head, this is the head assembly and this is sorry this is the arm (Refer Slide Time: 35:37) and the head is actually close to the tip. So it is pivoted around this point so this head will move outside outwards or inwards, move from track to track.

The rotational speeds are typically....... they follow certain standards. so standards, these keep changing, 3600 revolutions per minute or from that it changes to 5400 revolutions per minute or 7200 revolutions per minute, this example is of 15.3000 or 15300 revolutions per second. So, example of disk from Sea Gate.

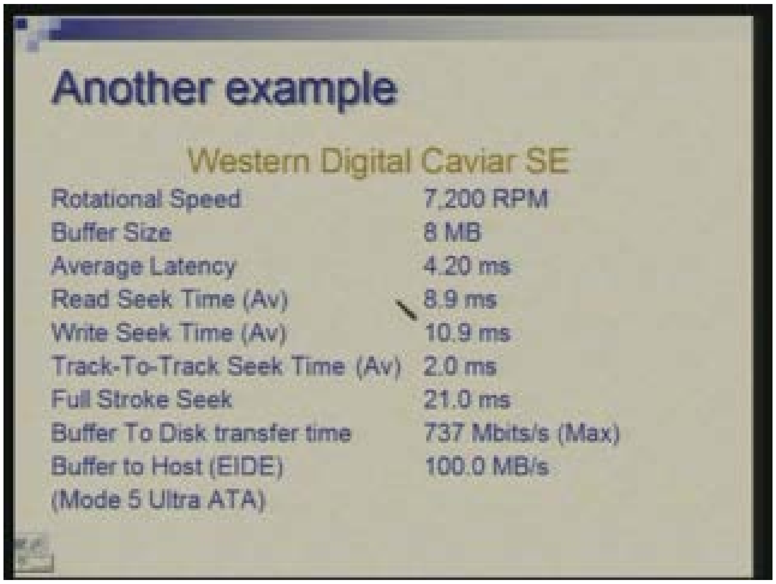
(Refer Slide Time: 36:15)



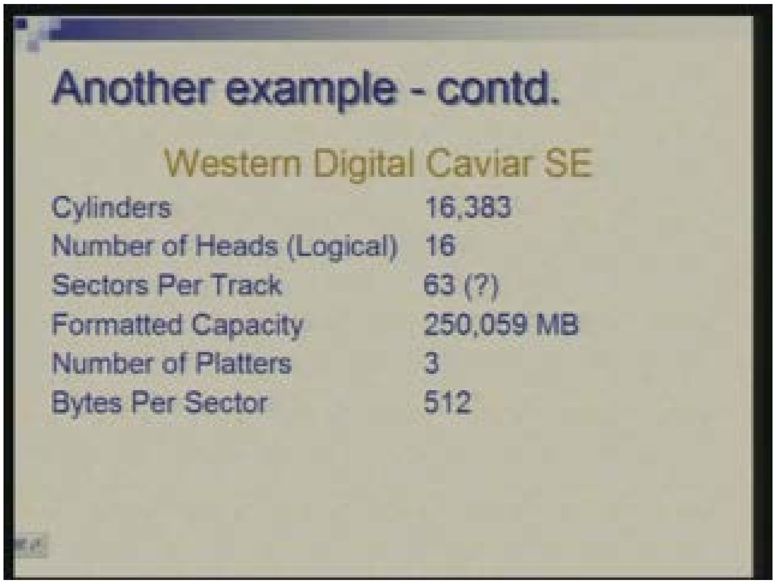
So 15 point 15K is the highest speed you find these days commercially and that decides how fast you can get the data. It influences both the time it takes to reach the initial point and the time to transfer. When the disk is rotating once you have positioned the head at the right point the rate at which it rotates determines the rate at which you are encountering the data and that is the rate of transfer. This has total capacity of 73 GB, speed is 15K rpm, seek time seek time is the time to reach the desired point which could be more or which could be small. So if you are already close to that it may take less time; if you are sitting far away then it might take more time, so 3.6 is the average. The interface also follows certain standards and this standard is called ultra 320 SCSI. We will talk of these little more a little more later and this is characterized by the rate of transfer 320 megabytes per second. So this is the rate sustained by the interface that interface, on that interface depending upon what disk you connect and what speed it is rotating at you will have actual data generated at a different rate. So this is not an example of a very high capacity disk but a fast disk.

The next one is somewhat slower, about half the speed but capacity is larger, so capacity I think it will come on the next part.

(Refer Slide Time: 38:00)

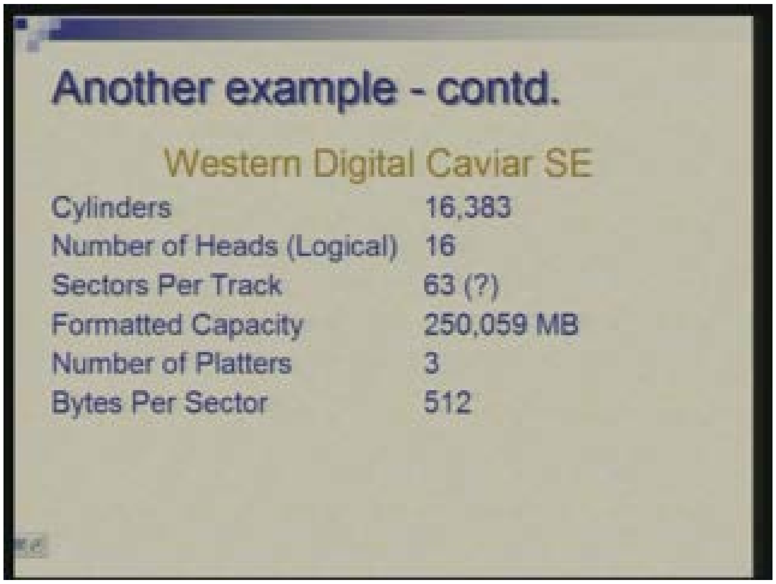


(Refer Slide Time: 38:12)



This is the 250 GB disk. 8 MB buffer size which means the data does not go to the processor or memory directly, it is first brought into a local memory which is of size 8 megabytes and then from there it is transferred. Average latency: this is a rotational latency so the time to reach a particular point is determined by two things: One is the time taken for rotation and second is time for the head to go from one track to other track. So the rotational latency in this case 4.2 millisecond and this can be determined if you take the RPM figure how many revolutions per minute, you can find the time for one revolution and if you take half of that, that is typically taken as average rotational latency. So, if you do that arithmetic you will get 4.2 milliseconds.

(Refer Slide Time: 39:23)

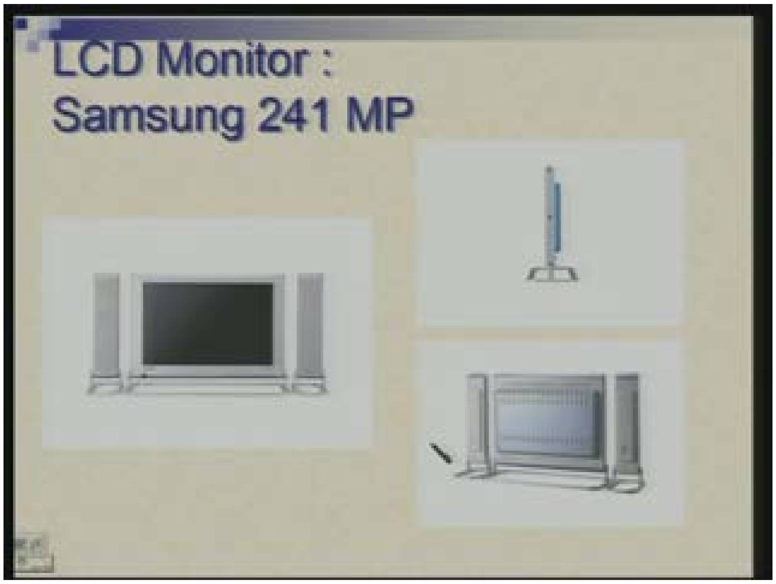


The seek time is..... we can look at this here; track to track seek time is moving from one track to other track, is 2 milliseconds and this is if you were to move one track only. But on the average there is some average figure given, it is for reading 8.9 milliseconds and writing 10.9 milliseconds. So basically the time involved to reach the first piece of data is of the order of several milliseconds and the time to transfer the data is much faster; once you reach that point then things are faster so this is the buffer to disk transfer time, between the disk and the buffer; this is 737 megabits per second maximum and this is dependent upon the rotational speed. And buffer to host here it is a different interface EIDE. In the previous stage there was a different interface, this takes 100 megabyte per second.

So here the peak of the transfer between disk and buffer is much higher. But what happens is that the disk is not always busy in transferring, it is at times busy in seeking also. Therefore this is this figure is okay. Initially one might think that disk is faster but you have arranged for a slower interface but disk cannot be continuously seeking, it has to....... disk is not continuously transferring, in-between it spends time in moving to the desired point.

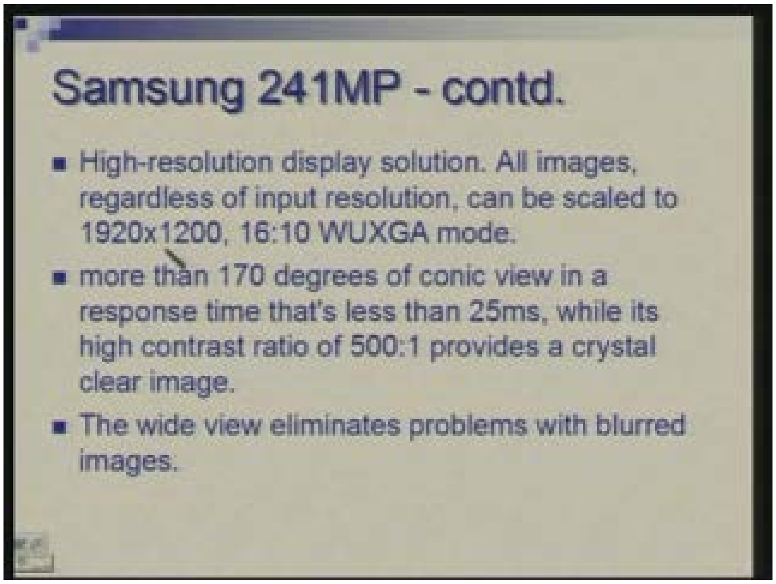
So, continuing with that there are some parameters which define the storage layout. Cylinder is actually collection of all the tracks which are sort of coinciding on different platters. So suppose you have four platters, you talk of tenth track on each of these so all this together will form what is called a cylinder. often you do not write in terms of number of tracks, you talk of total number of cylinders, so this is one figure (Refer Slide Time: 41:18), the total number of heads each surface is accessed by different head, then number of each track is divided into number of sectors. Well, this is from the western digital internet site but I think there is a there is something wrong in this figure because it does not tell you the rest. Number of sectors per track typically is few hundreds, total capacity is 250 GB, bytes per sector is 512 so half kilobyte in each sector.

(Refer Slide Time: 42:25)



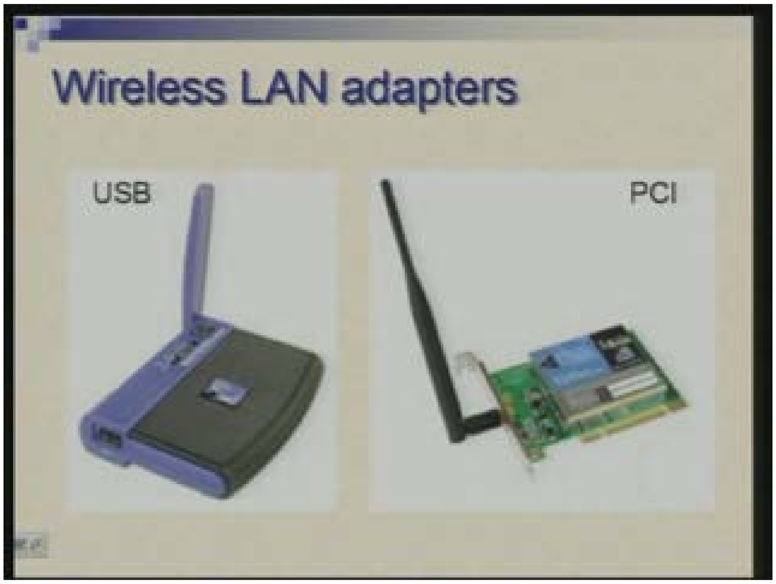
So this was one of the very important peripheral devices that disk......... well, sometimes it is not considered as an I/O device, some people try to characterize this as a storage device, okay but all the same. The way a processor memory have to communicate with disk is something like an I/O device. This is an example of an LCD monitor from Samsung, different views are shown. Interesting thing is this view which shows how thin it is unlike, well, this is also a LCD monitor. You have CRT monitors which are very bulky. So you can get some idea of the resolution here. The one shown in the picture is of a really very high resolution.

(Refer Slide Time: 43:24)



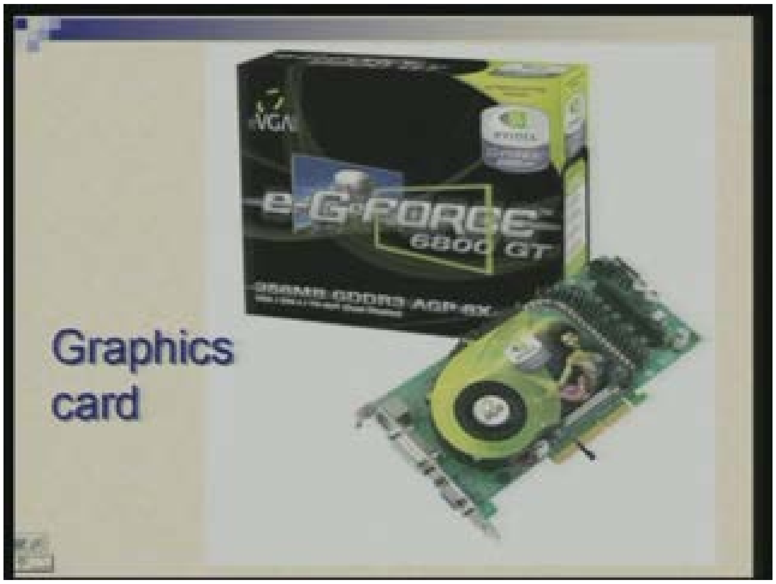
1920 into 1200 and the viewing angle......... so, often LCD monitors may have limited viewing angle. If you look from side you may have problem in seeing, but this one is 170 degrees so you can say almost 180.

(Refer Slide Time: 43: 47)



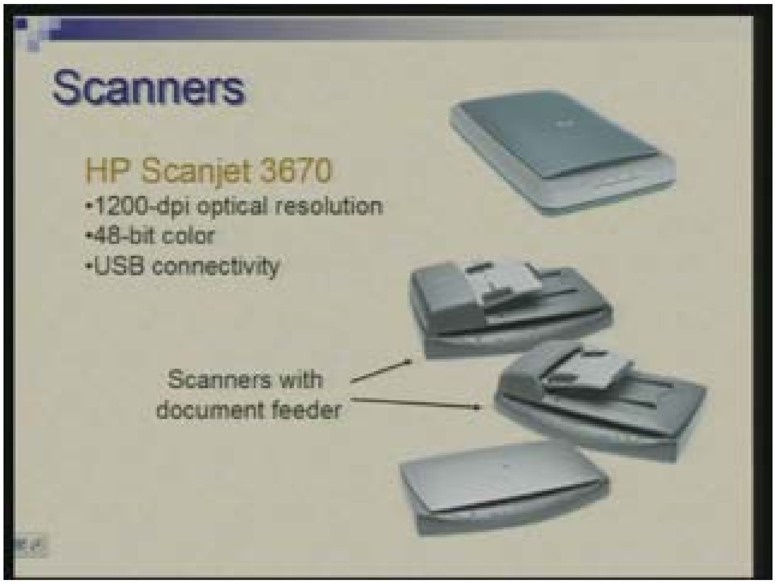
Then these are wireless LAN adapters. So this one is the one which connects to a USB. USB is a serial bus. We will talk of these terms little more later. So this connects externally to a computer.
This goes inside the computer (Refer Slide Time: 44: 17) this connects to a PCI bus. Again we will talk of this term little later.

(Refer Slide Time: 44:21)



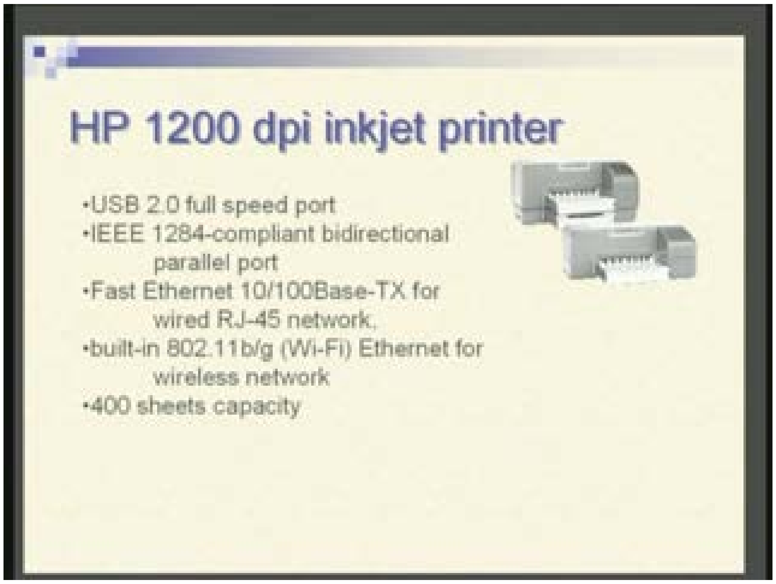
Graphics card: so these are the cards which drive really high resolution monitors and they are able to show you changing scenes very conveniently.

(Refer Slide Time: 44:50)



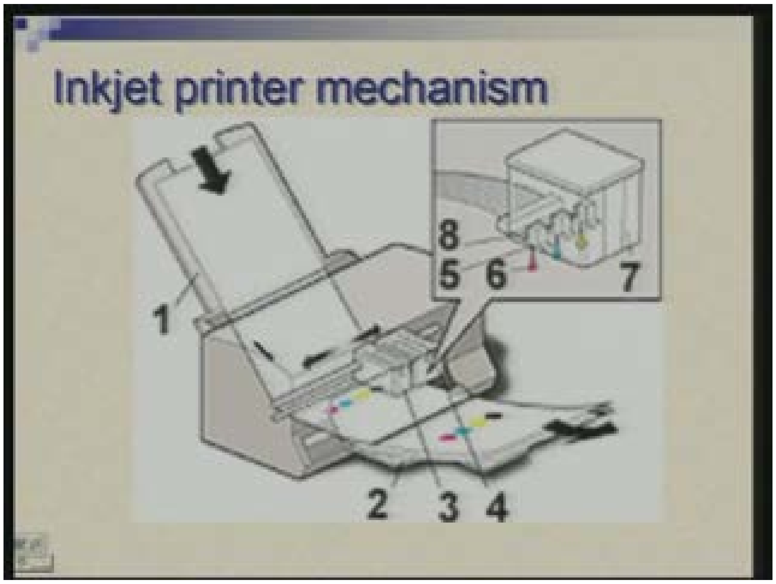
Scanners: Scanners are characterized by their resolution in DPI as I mentioned, so 1200 DPI. The number of colors they can resolve so 48-bit color, 6 bytes. These scanners have a document feeder attached. That means you can put a bunch of papers and they can automatically feed like a photocopy machine if you have seen one with a feeder.

(Refer Slide Time: 45:15)



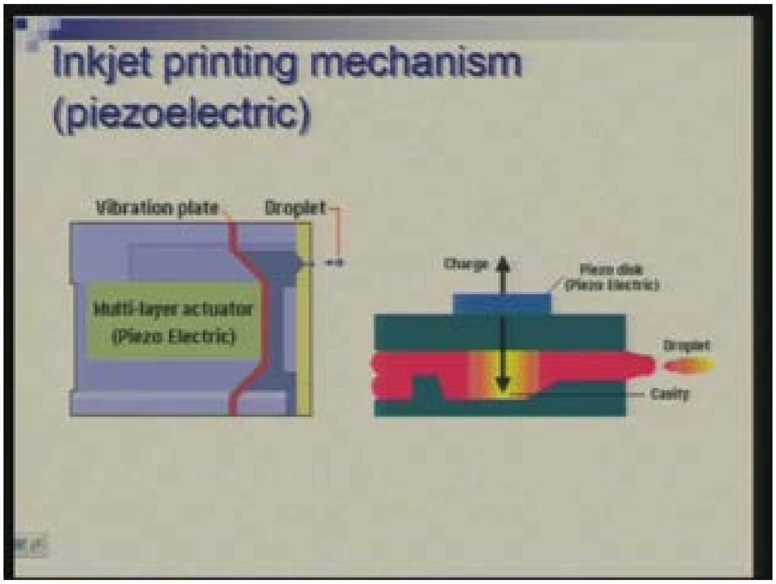
Inkjet printers: They are very cost effective and quality is continuously improving. They are also characterized by the number of dots per inch that is the printing quality and the number of colors they can resolve. I do not have those figures here. These have direct network interface so that it can be shared over in a network.

(Refer Slide Time: 45:55)



Well, I am sorry for this picture. Idea was to show you a little close-up of an inkjet printing mech- printer mechanism. If you can see here in this one, the paper gets set from the top and goes through this area where printing gets done (Refer Slide Time: 46:11) and then moves out here. There are ink cartridges here and the key thing here is this ink cartridge, the crucial mechanism is sitting here, rest of it is essentially paper feed and movement of this and interface. But the new technological breakthrough which has happened over last couple of years are in this. So how very fine drops of ink can be thrown on paper to form high quality pictures. So they are typically....... there are two techniques: one is piezoelectric mechanism and other is thermal system. So, to understand this, we will just quickly take a minute here.

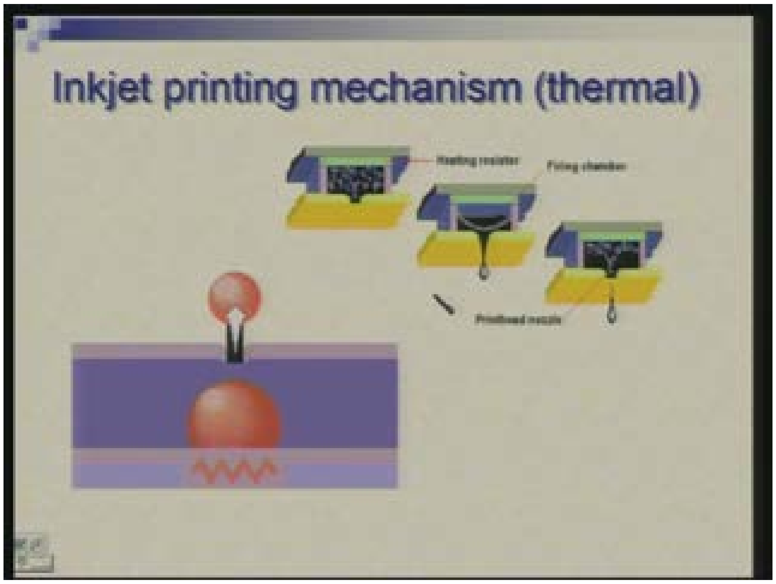
(Refer Slide Time: 47:10)



So this is ink cartridge where in this chamber there is ink (Refer Slide Time: 47:11) and here is a plate quartz plate which is piezoelectric and it can vibrate and by that vibration it can throw out a very tiny very fine finely controlled droplet of ink. So this vibration would be controlled by the electrical signal which it receives depending upon the information which is to be printed.

Another view is that again the same thing. This is the chamber (Refer Slide Time: 47:41) where ink and a droplet gets thrown by this vibration of piezoelectric disk here.

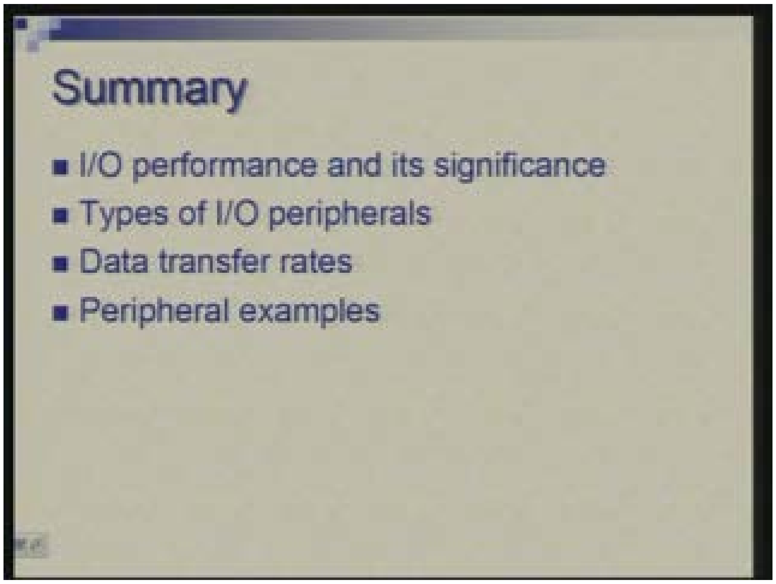
(Refer Slide Time: 47:52)



This is a thermal mechanism where the ink droplet is thrown out by a small, by very localized heating process. So here there is a heating register which gets which gets heated by electrical current and it is a very controlled heating, so by heating there is an expansion and the droplet goes out.

So, you remember that we are talking of resolution of nature of 1200 dots per inch so you can imagine the amount of the level of control which you have to have on this ink droplet to make it printed. So just contrast it with the technology of peripheral devices which existed couple of decades back and where we are now. Okay let me close at this point.

(Refer Slide Time: 48:49)



Just to summarize, we talked of I/O performance consideration, their importance. Briefly we talked about throughput type of I/O, definition I/O performance definition or response time type of definition. We looked at variety of peripherals, tried to classify from different counts, tried to get a feel of their data transfer rates and we looked at a few examples of peripherals in little more detail. In the next few lectures we will talk of how they get interfaced to the rest of the system. Thank you.