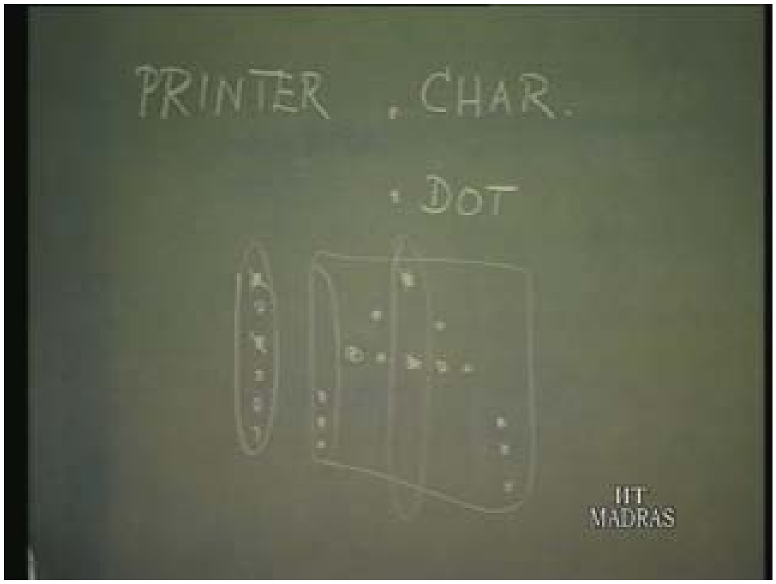
Computer Organization
Part – III
Prof. S. Raman
Department of Computer Science & Engineering
Indian Institute of Technology, Madras
Input/output
Lecture – 30
I/O Devices (contd...)

In this lecture we will continue our discussion on I/O devices. In the previous one, we took a look at keyboard, display, and to some extent, the printer. In fact, looking at something like a line printer, which will print at the character level and a dot matrix printer which is at the point level, or at dot level, or specifically in the representation, 1 bit will correspond to 1 dot of the CPU or some controller.

(Refer Slide Time: 01:44)

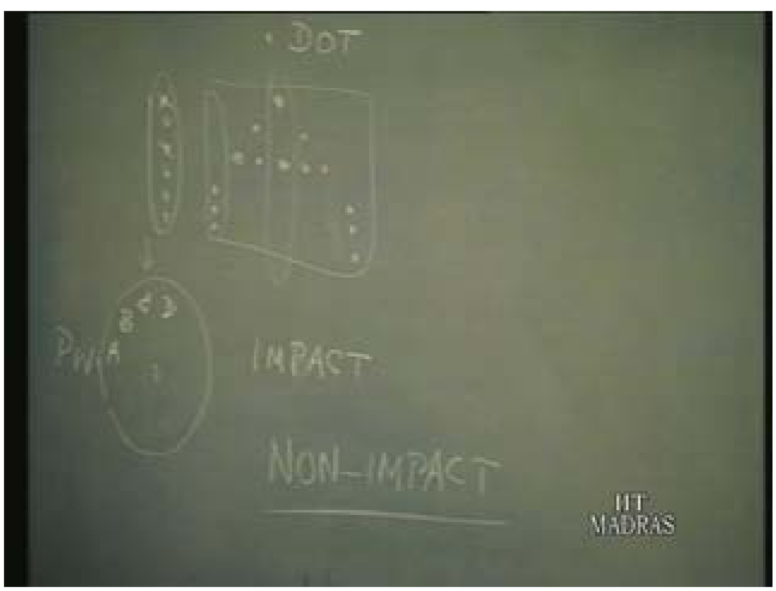
The CPU is not directly involved; it will send a code to the printer. That code will identify either the character or the pattern for the character because, in the case of a dot, what you have is in fact something like a column of pins. If, for instance, you have 7 × 5, there will be 7 pins and these 7 pins will be moved in adjacent positions and then an appropriate pattern will be formed. For instance, if A is to be printed, then, in the first column, only this will be activated; may be this or may be this. Let us say these 3 are activated.

(Refer Slide Time: 03:42)



In the second column only this is activated; this pin is activated, and then in the third column only this is activated; in the fourth column only this is activated – it is not 7 into 5, in fact it is more than that and so on and so forth. So now what is needed is this particular pattern – more than 7 into 5. You just take one specific column. When you take a specific column, you find two dots printer; actually it means from the printer this pin and possibly this pin these two pins will be activated and that is why these dots have been created.

(Refer Slide Time: 06:12)

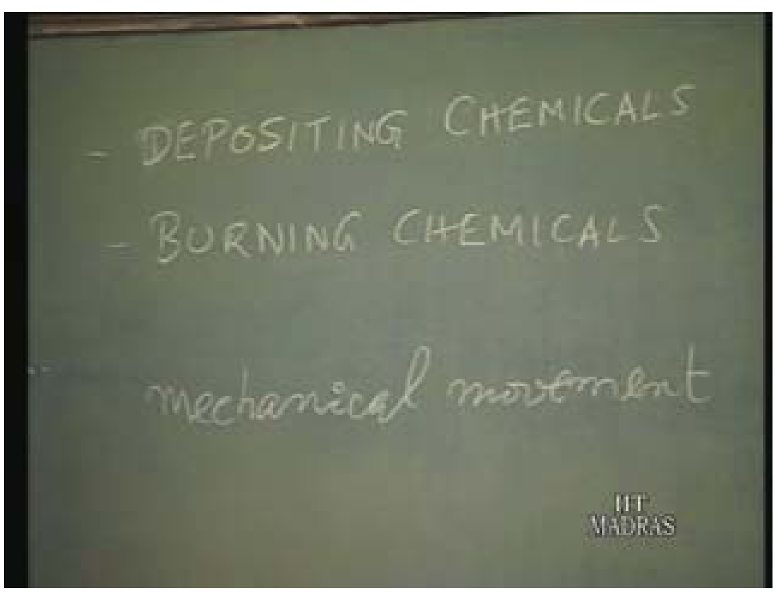


In another words, with a column of pins and by moving the column over the paper, we can print it out. So a coder comes from the CPU, or a controller will have to identify this pattern in this time sequence, whereas in the character-oriented printing, possibly the Daisy wheel there is a print head, the block for printing will already be there. In case D will have to the printed, then D will have to be positioned. Say this is the print wheel, which has all the things – all these have been shown just so as to see clearly. For instance while printing A or B or C or D, actually this will have to be reverse form or mirror image form of it. Now if A has to be printed, a command must be given so that A will be brought under some position and then a hammer will go and hit. Now if D must be printed, the wheel must be rotated in the other direction; D must be brought here. Both these versions – one working at character level and the other working at dot level – in fact correspond to the impact type printers.

That is, by making impression through impact by using a hammer on the wheel or just the pin itself getting activated or on the print media, that is the paper, the impression is created. You also have non-impact type printers; non-impact process will be somewhat like what you have in photography. That is, possibly spraying of some chemicals or let us says depositing some chemicals on appropriate paper; the paper may have to be coated. So, it is depositing some chemicals or let us says burning the surface that has already been coated or something like that. In all these things, there is no mechanical impact but you create a pulse to burn; you create a pulse so that the some deposition can be made. Similarly, you also have the non-impact printers. Let us not worry too much about the mechanism by which the printing is done. As far as the interface is concerned, a code comes and then, depending on whether it is a character or a dot matrix or impact type or non-impact type, the appropriate signals must be generated to create a pattern such as this.

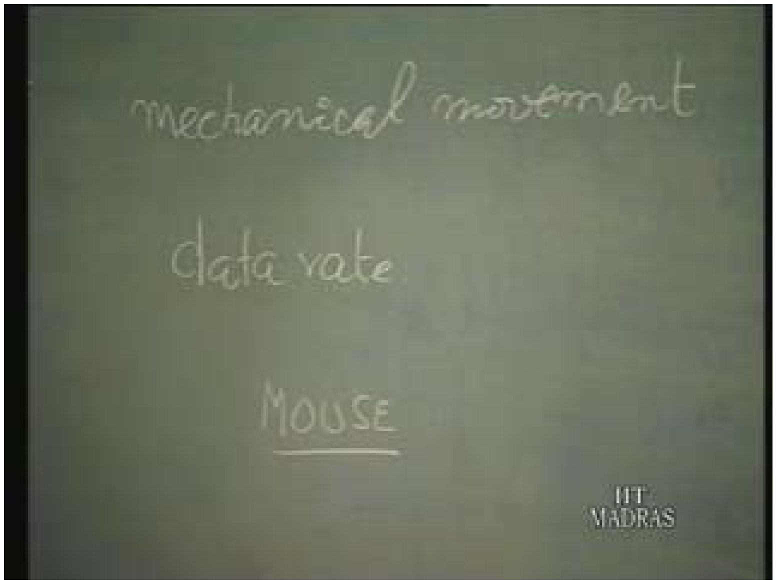
In the case of non-impact, as you can see, except for the movement of the paper itself, suppose the printing is on the paper, except for the movement of the paper there is no other movement, whereas in the impact type, the wheel will have to be positioned; the print head must be moved; and so on. So whenever there is a mechanical movement involved, it is going to cause some delay and this is going to create some problem; it is characteristic of that; it is going to cause some delay.

(Refer Slide Time: 08:34)



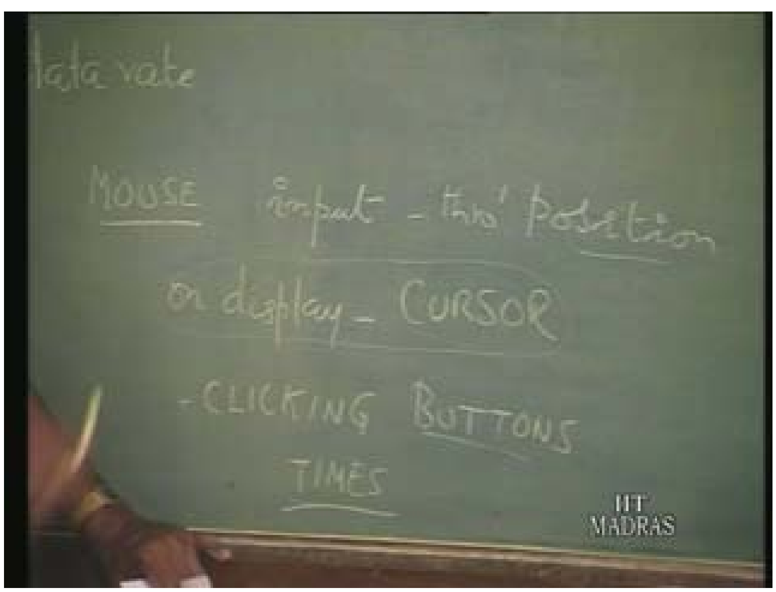
In other words, we will get low-speed printing; wherever mechanical movement is less, then we can hope for high-speed printing. Now depending on that, the data rate will vary.

(Refer Slide Time: 09:14)



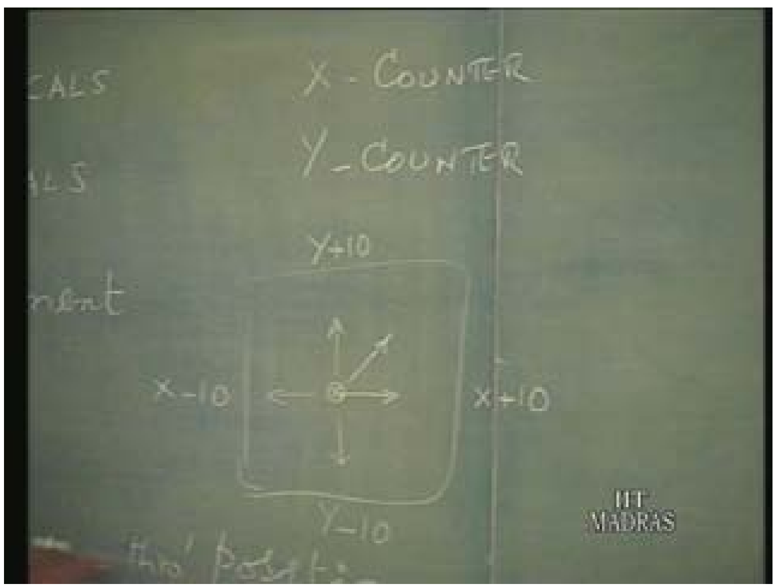
Now let us look at another device, which is commonly used these days, namely, a mouse. It is a common device that is used – we put on the pad and move it around. What exactly does it do? What type of device is it? First of all it is inputting; the mouse itself will be mechanically moving over the pad and within that particular mouse arrangement you also have buttons. You click the appropriate buttons and although the mouse is input, the corresponding cursor is on the display, you have the corresponding cursor.

(Refer Slide Time: 11:50)



So, one really looks at the cursor. Though it is input device, apart from the program needed for processing the location of the mouse, there is a mechanical position of the mouse on the pad; there must also be a display routine associated with the cursor. It is, in fact, some kind of an indirect thing; meaning, here the display is in fact the output, which is what we are seeing. We use it for monitoring the movement. There are many ways in which this particular mouse position can be sensed – where exactly it is. We are moving a cursor, say, on icons and clicking the button – this is another aspect of it – one is we are using the mouse and positioning it. Actually we are inputting through the mechanical position. Another thing is clicking the appropriate buttons. Now you may have more than one button, in which case, we talk about which button to click and then some times the same button – how many times do you click? So the number of times the buttons are clicked – that also is information. So using the mouse and noting the position and using which button and how many times the buttons are clicked, we are inputting the information – all this is done by looking on the display at the cursor, which is supposed to correspond to the position of the mouse.

(Refer Slide Time: 13:20)

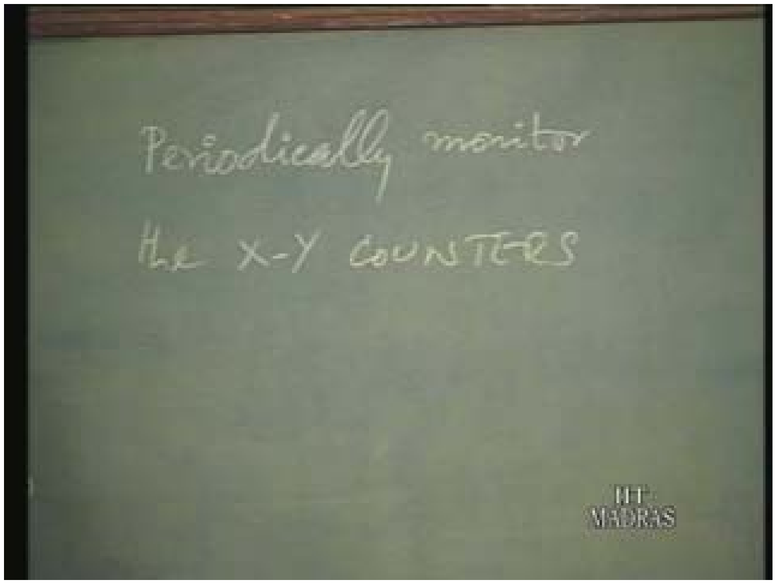


One implementation of the mouse would involve somewhat like this: have a pair of counters; call it x counter and y counter; and whatever may be the position of the mouse on the pad, let us say, this is the initial position and if the mouse is moved in this direction, arbitrarily let us say the x counter position will be incremented by say 10. On the other hand, if the mouse is moved in this direction, the x counter contents are to be decremented. Similarly if the mouse is moved in that direction, the y counter contents will be incremented. If it is moved in this direction – I am assuming – equal amount of movement in all the directions is decremented by 10. On the other hand if the movement is in this direction both x and y will be incremented. If you do it at any other angle, it is not going to be equal increment; it will be unequal increment in x and y.

So actually that is why we said the position was important; when you move the cursor, by monitoring how the x and y counter contents incremented or decremented and by how much, one can actually find out or locate the coordinate; that is, in fact, for understanding where exactly the mouse is.

Now corresponding to this, the cursor must be moved on the display. In other words, apart from registering these, the processor or the controller must periodically monitor these counters.

(Refer Slide Time: 14:36)

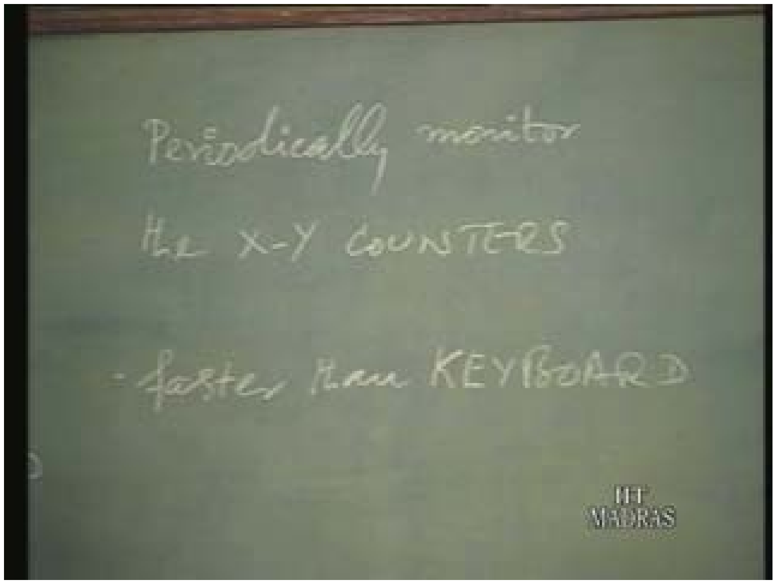


I just assumed x and y counters. So while monitoring this particular counter and knowing what exactly is this correspondingly, the cursor will have to be moved on the display. I already said you can have different buttons and clicking on the buttons certain number of times to indicate different operations. So one click means one thing, two click means something else. You can see that compared with keyboard, the mouse is certainly faster because in the keyboard for indicating one particular item, let us say, close the file, at least something like C and L will have to be typed, whereas with the mouse we are going to move on the display and click on some icon which is indicative of closing the file.

So the number of entries or the time taken to indicate that particular command is going to be less because it is going to support the graphical interface. So we can say that it is slightly faster than keyboard. Certainly it is quite useful in the case of interfacing because we can make use of the icons on the display. That is about the I/O devices. Now we move on to the next, quite important subsystem, namely the disk subsystem.

We may call the disk system I/O or, in general (Refer the slide time 17:03), put it under the category of storage because we are generally going to have the files in the disk and you can make use of the files as input and similarly create the files for recording the output; that way they service as I/O. But essentially the user is not going to directly put something in the disk because the disk is quite fast if I compare with other I/O things. In fact the disk will be totally under the control of the processor directly or indirectly through the disk controller. So you cannot say for instance that like a mouse or keyboard, disk is not going to be directly used by the user.

(Refer Slide Time: 16:21)

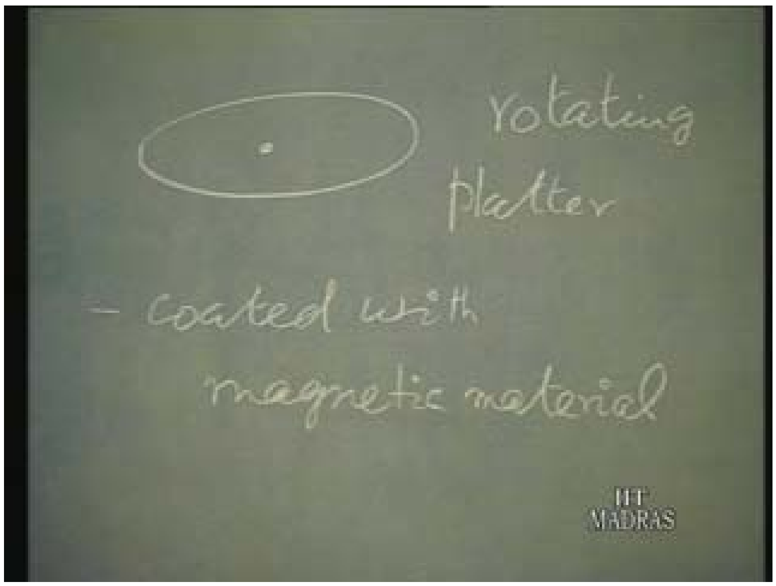


So essentially it is for storage because the system is going to be totally under the control of the operating system also. Now the main thing about this particular disk is that this particular type of thing is non-volatile, meaning (Refer the slide time 18:29) even if power goes the information or the data that is stored there is not a waste; it is available all the time. In case there is a power failure which may be sensed immediately all that are available in semiconductor memory will be available in usually a few milliseconds because when the AC power fails, the DC power does not fail immediately – that is going to take a few milliseconds.

(Refer Slide Time: 20:06)

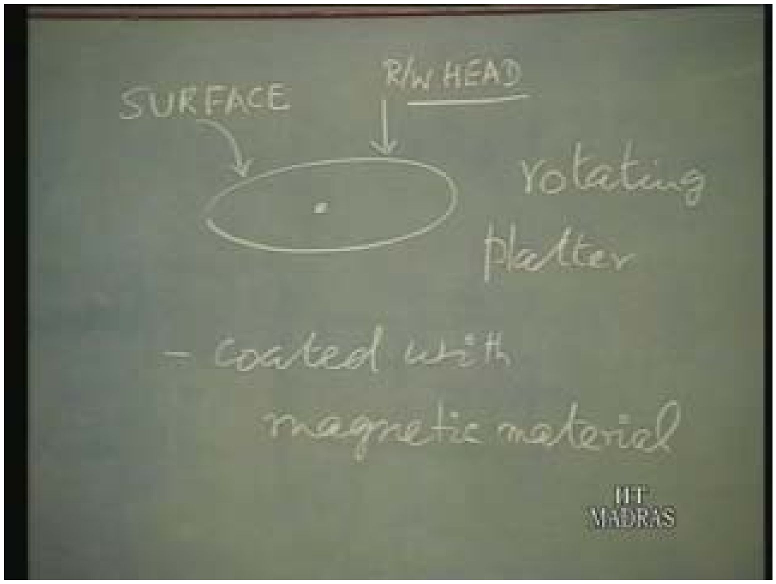
That is plenty of time for storing; whatever was stored in the semiconductor memory can all be stored in the disk. That, in fact, is the main function of the power failure trap routine. Essentially disks are of two types: the first one is very flexible and is called floppy because the size is small. Generally we talk about diskettes. The medium is a diskette and a system is the floppy subsystem. The second one is quite flexible as you would have noticed; the second one is the hard disk system. Now whether it is a floppy or the hard disk, essentially what you have is a rotating platter; let us say a rotating platter, which will be rotated about the center and this particular rotating platter will be coated with some magnetic material.

(Refer Slide Time: 20:59)



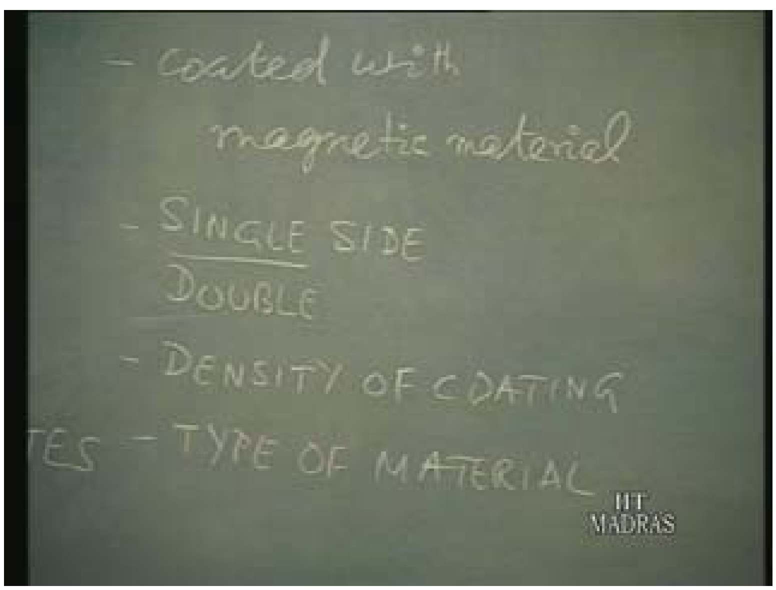
In fact it is because of this magnetic material that there is non-volatility – that is why we do not lose the information with power failure. The material in the case of hard disk is a metal or a glass. In case of hard disk, the base platter material may be glass or metal. In the case of floppy disk, it is invariably some plastic stuff, some synthetic one, because it will be very flexible. In either case, it is a coated material. In fact it is this particular magnetic material, which is going to be what you say as polarized in one or the other direction; depending on that particular magnetic polarity we talk about saving 1 or 0, logical 1 or 0. The system will essentially consist of a constantly rotating platter, and this coating may be on either side. For instance, both the top surface and the bottom surface have been coated and, depending on the type of material, we talk about the density also; we will see that little later. Over the rotating platter, there is a read/write head usually positioned very close to that particular surface.

(Refer Slide Time: 23:01)



In some it will even be touching. In the case of hard disk, we can talk of more than one platter stacked, whereas in the case of floppy, we always talk about only one specific platter. In this one there is a single side depending on the surface that is coated. Suppose both sides have coating and both sides are used for storing the information by appropriately magnetizing the material that would correspond to writing.

(Refer Slide Time: 24:36)



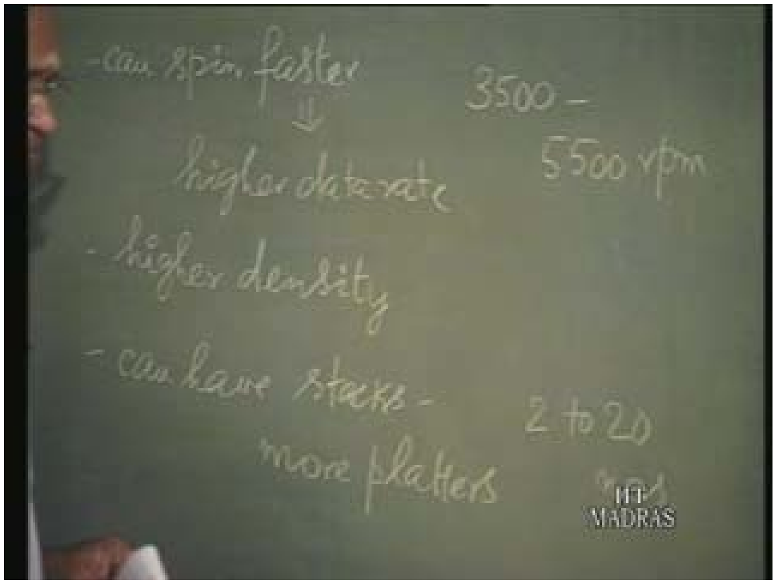
The density of coating, which will appropriately support the density of storage, will all be linked by the type of material. Depending on this, the disk has been coming in different capacities. So we talk about disk capacity because if we take the hard disk, this hard disk will certainly be very rigid. Whenever something is very rigid, it can rotate very fast.

When it can rotate very fast, then obviously while reading or writing, we can also access the data faster. On the other hand if it is a very flexible one as in the case of floppy, it cannot rotate very fast and hence the data rate also will be low. Similarly, depending on whether you have single or both side coating and depending upon the density of coating which supports the density of using storage and also the material, the capacity will be varying. I will say something more about the system. We will give some more figures also about this. We were talking about the rotating platter. Now if the platter were rigid I said there are some advantages.

(Refer Slide Time: 27:28)

Let us see; suppose it is rigid obviously the particular platter can be larger. Now depending on this particular one, the size being large or small, data handling is possible. Typically this particular size is in the range of 1 inch to 10 inches – that is the normal size. If it is rigid, it also can spin faster. If it is flexible, then that would be problem. When it can spin faster, it would mean the data rate is higher. What is the usual speed of rotation? It is in the range of something like 3500 to 5500 rpm, that is, revolutions or rotations per minute. If it can be rigid, then obviously it will be of higher density and more data can be accommodated.

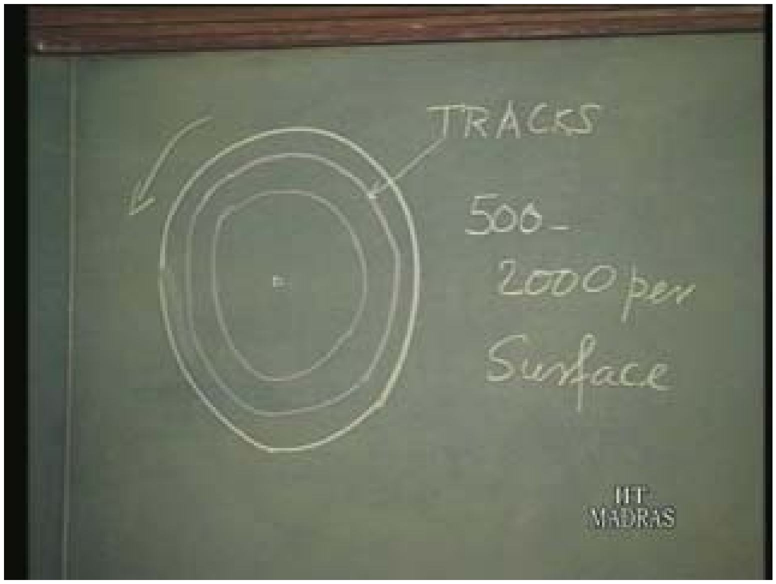
(Refer Slide Time: 28:28)



We also said that we can have stacks of them, that is, you can have platters stacked one above the other. It need not be a single one; but if it flexible, we cannot accommodate them neatly. When you talk about the stacks usually it is in the range of 2 to 20 numbers of this platter, and generally it is called a pack also. How is the data organized? This is somewhat mechanical information. Typically, if we just take one platter, then the read or write head will be over the surface and it will be moving appropriately – that is, the center of it, around which the whole thing rotates. So this is just one platter; read or write head will be moving. Obviously the read or write head is a cell; let us say read head is going to send the magnetization; whether it is of positive or negative sense, it is going to interrupt it as 1 or 0. Read or write head is going to of this or even smaller size, which means basically the whole pattern is not being occupied. So it is going to be confined to one small area, and we also said the whole thing is rotating in one direction on the other.

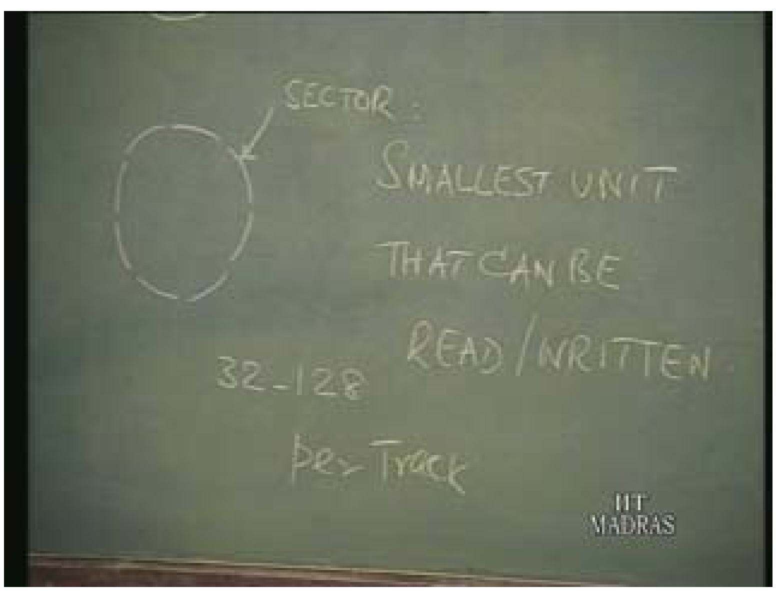
So when a read head is placed in one position and when it is rotating, it is obvious that the data can be formed along the circumference, whatever may be the radius. That is why we would say that the data is organized so that the data will be on what may be called tracks. The tracks have been shown here. Just two tracks on the data may be stored as shown here. For instance the read head may be here, in which case, the information or data on this track is going to be sensed or the data is going to be written on this track. Now when the read head is moved then it is going to be concerning this particular track and so on and so forth. Generally the number of tracks will be of the order of something like 500 to 2000 per surface because you can have 500 to 2000 per surface on both the sides. The data is going to be organized around on a track.

(Refer Slide Time: 31:19)



Here we have shown just one track. You can see that this particular track has been split into what is known as a sector – we have this particular arrangement because the sector is the smallest unit that can be read or written; meaning, the data is going to be stored in a sector and that is a smallest unit. If we have 64 bytes per sector, even if you have just 2 bytes to be stored, it is going to be stored in that sector; only 2 bytes will be stored in that. If you want to store 128 bytes, you would be using two sectors. So sector is essentially the smallest unit that can be read or written.

(Refer Slide Time: 33:35)



You can see that a platter consists of surfaces, and given a surface, it is going to be split into tracks and given a track, it is going to be split into sectors.

The number of sectors will be typically of this order: 32 to128 sectors per track. Now given a surface, it will be done during formatting of the disk. Whatever may be the track, the number of sectors will be the same. So although you can see that the smaller the track the denser it is going to be. So if we say 128 sectors per track, it is true whether it is the track which is outermost or the innermost. That is, these things will be done during formatting. Now take one specific sector; the data would be stored in the magnetic form. The format of the data is somewhat like this: there will be a number to identify the sector.

(Refer Slide Time: 35:55)

Because there are one 28 sectors, first the sector number must be stored. That is, basically we are looking at what is stored, given a sector. Then there will be a gap. These gaps are necessary because this is continuously rotating. If we do not give a gap sometimes these things – the general information and then the data that is stored – will all get mixed up. After a gap the data is stored using some format or other; apart from the data we also have some error correcting code because it is always possible that the data that is stored gets corrupted and some error is introduced for some reason.

So the error correcting code will either just indicate that there is an error or it will help you understand what the error is, that is, detect the error and correct it also? Sometimes we may be able to correct; at least detection is good enough. For instance let us see an odd parity error correcting code – odd parity indicates that the total number of data bits including this will be odd. Even parity means the total number of 1s, including the 1s in this code, will be even. This is simple. After this, there is a gap and then the other sector starts; that is, sector number and a gap, then the data with the error correcting code, and then there is a gap. That is typically what you have in a sector, that is, the format in which the data is stored.

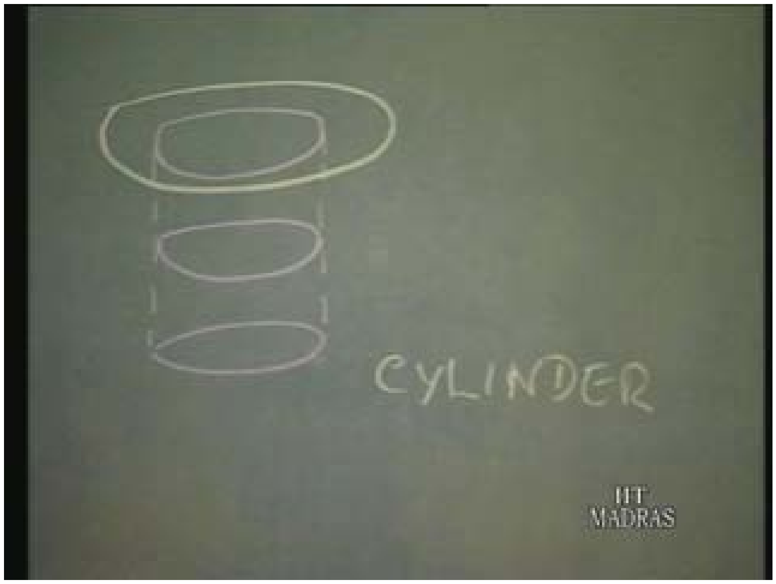
So as I said suppose the data format is such that the data that is stored in a given sector is 40 bytes, even if you stored less than 40 bytes one full sector will be used. The rest of them will not be useful for that purpose. So now remember as I said that all sectors or rather all tracks have the same number of sectors. I also said sometimes during formatting we will be able to adjust that. Now take one platter and one track and as shown here; assume that there are three platters.

(Refer Slide Time: 38:05)

So what I have done is I have just taken those three platters in the same track. Suppose this particular track is track number 5; then I have taken the same track number 5 on the other two platters also. You can see here and I also said a read head is always over the surface. In fact read is also both/write head; generally that is how it is. Read/write head will be fixed on an arm; in fact my arm that is like that. So you can see that over the rotating platter there will be a moving arm holding the read/write head, so that by moving the arm over the different track, we can read the information from the respective track. Now when you have multiple platters, obviously it means multiple arms.

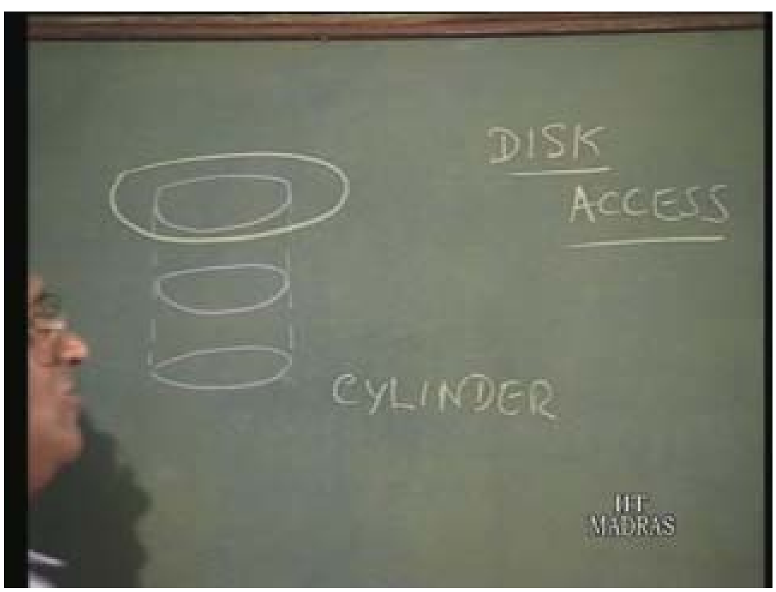
Certainly it has multiple read/write heads and there can be a single arm, in which case, the same read/write head will not be used. What will happen if you use only one read/write head? It will have to be used here or here or here having the number of other multiple platters is not going to be meaningful. So if we have n platters, it is meaningful to have n read/write heads. But then, all of them will be moving in the same way; that is, all the read/write heads will be synchronized.

(Refer Slide Time: 40:51)



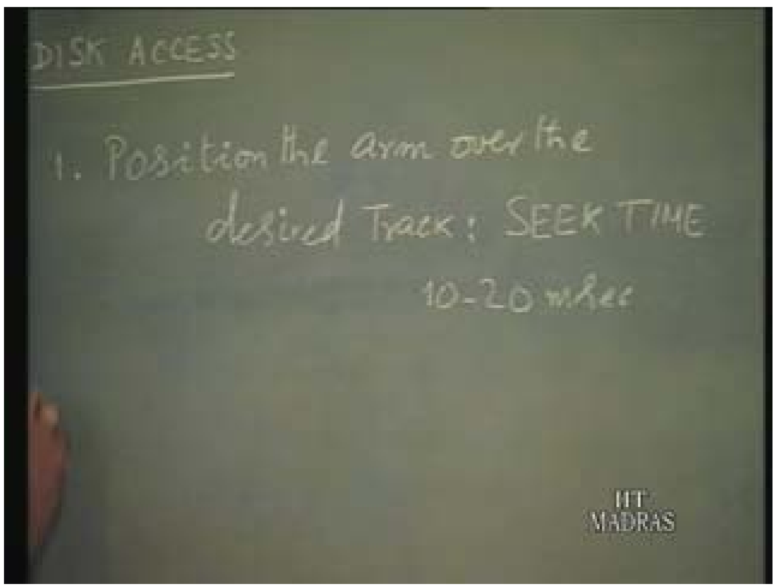
So you can say that if track 5 has been chosen on one platter or one surface, similarly track 5 in the other things will also be selected. Actually for an arrangement shown like this, there will be three read/write heads. That means basically data can be sensed simultaneously from three tracks on the three platters. Then we talk of a cylinder; as you can see a cylinder is nothing but an arrangement like this and a cylinder consists of tracks on multiple platters. It is not always necessary that there must be only one read/write head; you can have one read/write head here; another one here; another one here; and another one here.

(Refer Slide Time: 42:01)



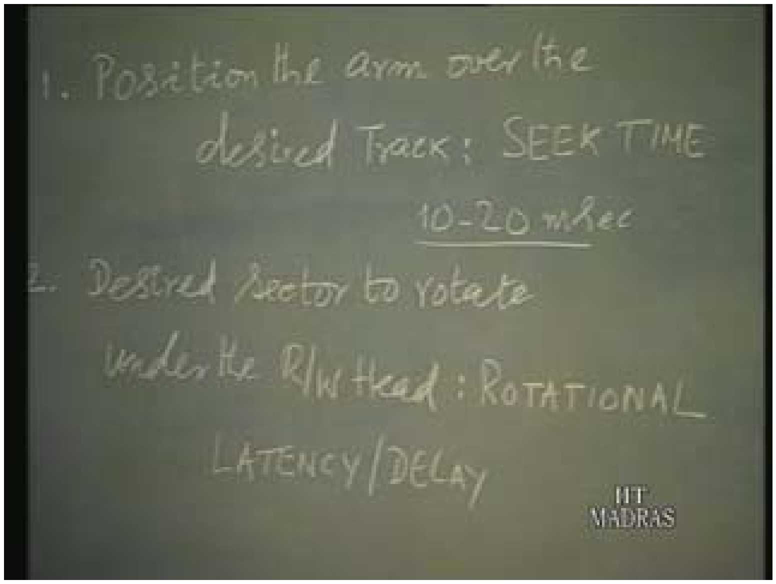
One can have it; so multiple read/write heads are also possible, so that the reading can be very fast. But even one will do mainly because this is rotating any way. So under the particular thing, it will any way come. Now let us just see how the data will be accessed. This generally comes under what we put in general as a disk access. We were talking about the disk access earlier. Now we know that the disk access will consist of reading data from a sector, which is a part of a track like this, and the tracks may be spread over different platters. The read/write head itself is positioned over an arm, which will move over the appropriate track. So given this picture, we can easily list all that is involved in reading from a disk. While talking about the I/O throughput – I was mentioning about the disk access earlier remember – why is it that we have to pay some attention to how many disk accesses per second? This is a very important figure that is needed, because different times are involved. First as we have seen earlier the arm must be positioned over the track.

(Refer Slide Time: 43:52)



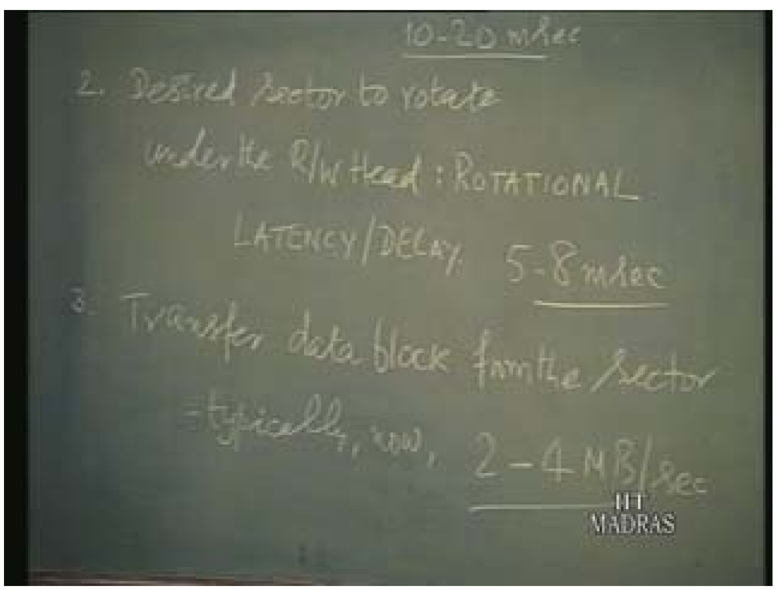
So position the arm over the desired track; this is the first step. This particular one is called the seek time. On an average, the seek time is something like 10 to 20 milliseconds; then average takes 20 milliseconds. So the arm must be moved. We do not know where the earlier position was. Suppose the position is here; it has to move only this much. On the other hand, here it has to move more. That is why we talk about and average seeks time. The desired sectors must come under the read head because we only talked about the track. Suppose we may read from here or from here or from here. So the desired sector must be rotated and brought under the read/write head. The desired sector has to rotate under the read/write head.

(Refer Slide Time: 45:26)



That must come there. The time involved in this is called rotational latency or delay. Typically the figures for this particular one are a few – only one read/write head. You can see that it is the time taken for half the rotation. That is the time on an average; we are talking about the average; half the rotation at that would depend on the speed, that is, the rpm. So typically from that particular thing, we can work out, and this will come to something like 5 to 8 milliseconds. Now this is the second time that it is involved. The third one in this note is that smaller the disk, faster it will be – that is another thing.

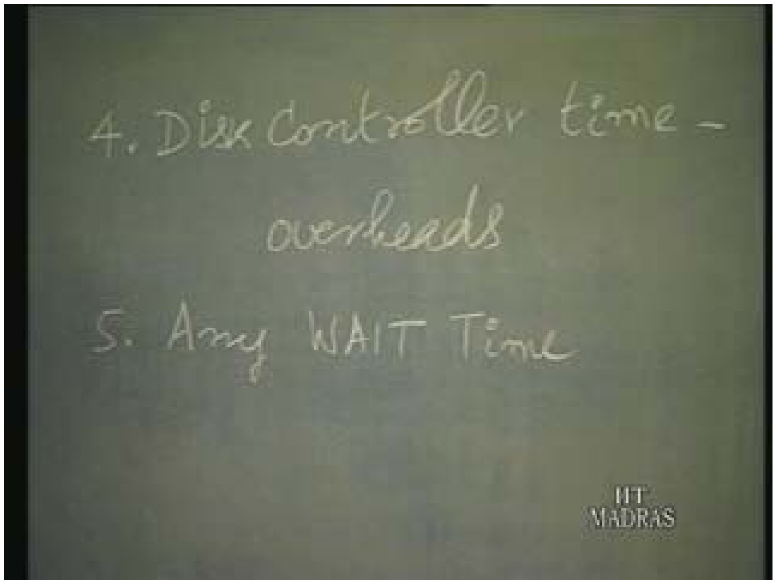
(Refer Slide Time: 46:19)



The size of the disk, the rpm at which it rotates – all these things matter. We have to transfer this data block. We are talking about the block of data from the sector; so it is the time taken for transfer.

So transfer data block from the sector – the actual reading from the medium, from the sector – typically depends on the technology. We may say now, in present days, it is something like 2 to 4 MB per second. This is the rate at which the data is transferred.

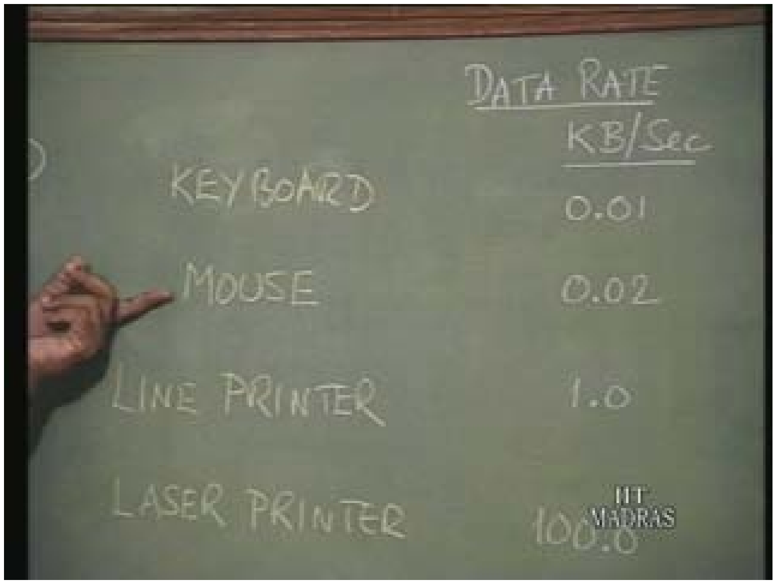
(Refer Slide Time: 48:16)



Obviously the size of the data, the rotation speed, the recording density, all these things will play some role in this particular one. Then fourth, all these things are controlled by the disk controller. The disk controller will add its own overheads. We generally call it the time taken by the disk controller because the disk controller will initiate; it will bring in the overheads introduced by the controller, and in this process, any other wait time.

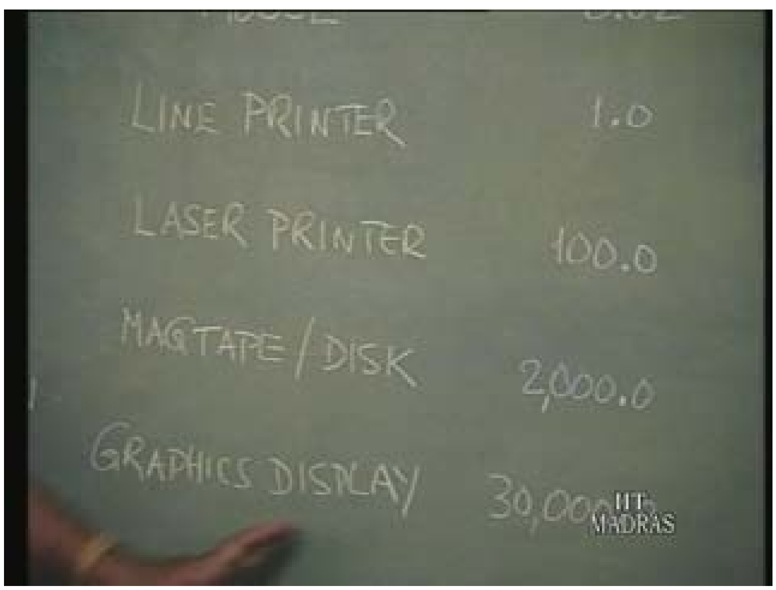
It is possible that the processor has indicated that it wants the disk to be accessed, but then the particular disk pack may be busy with some other process in the case of multiple processor, or it is half way through doing something else or there is any other wait time. So in fact one has to take all these things into account and only then talk about an average disk access time; that will give us an idea. Now having talked about these different types of devices, it is time that we take a look at the data rate, which is quite useful for us.

(Refer Slide Time: 48:50)



So we took a look at keyboard, mouse and so on. The keyboard transfers data at 0.01, typically 0.01 KB per second; the mouse is slightly better at 0.02; the line printer is character-oriented, that is, 1 KB per second; laser printer is quite fast because it is non-impact.

(Refer Slide Time: 49:26)



Generally it is 100 KB per second. If we take a disk or a Mac tape, typically it is 2000 KB per second and the display, specifically graphics display, is going to be 30000 KB per second. Of course these things will depend on the resolution and what not. So you can see the wide range from 0.01 KB per second through to 30000 KB per second – just see the diversity.

That is why we talk about a keyboard controller, which will be different; a mouse controller will be different; a printer controller will be different because the mechanisms in which these accept are different. From the CPU point of view, it is just one, throwing one code. In the case of keyboard, it will accept a code; in the case of graphic display it will throw out a code. The job is over; but that specific controller, keyboard controller or display controller will have to take care of not only the data rate but also the mechanism by which it has to either accept the code or display it. So we talk about the diversity. In the series so far we have taken a look at the CPU, memory and I/O. Now, we will consider the one interconnecting these, that is, the bus, and a few other things in the next part. We just call it bus, miscellaneous, etc.