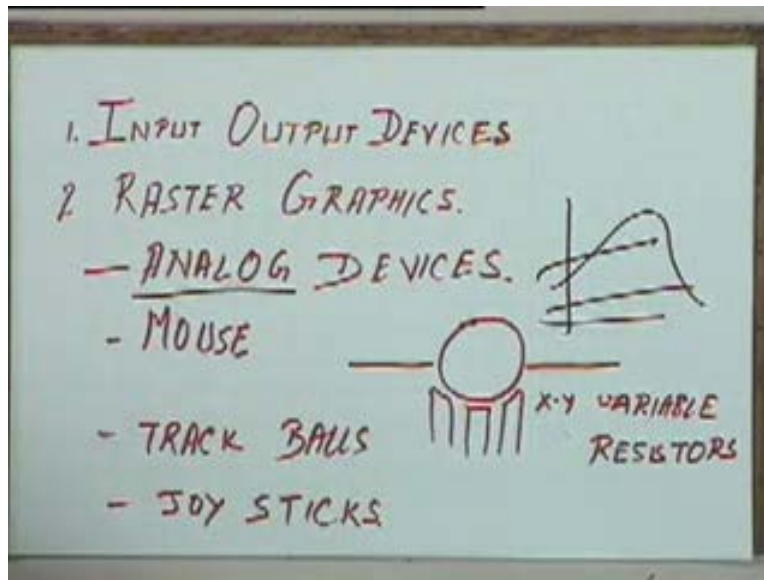


**Computer Aided Design**  
**Prof. Anoop Chalwa**  
**Department of Mechanical Engineering**  
**Indian Institute of Technology, Delhi**  
**Lecture No. # 02**  
**Input Output Devices, Raster Graphics**

Today we will be talking about input output devices which are used in CAD and then if time permits we will go on to some basic principles of raster graphics.

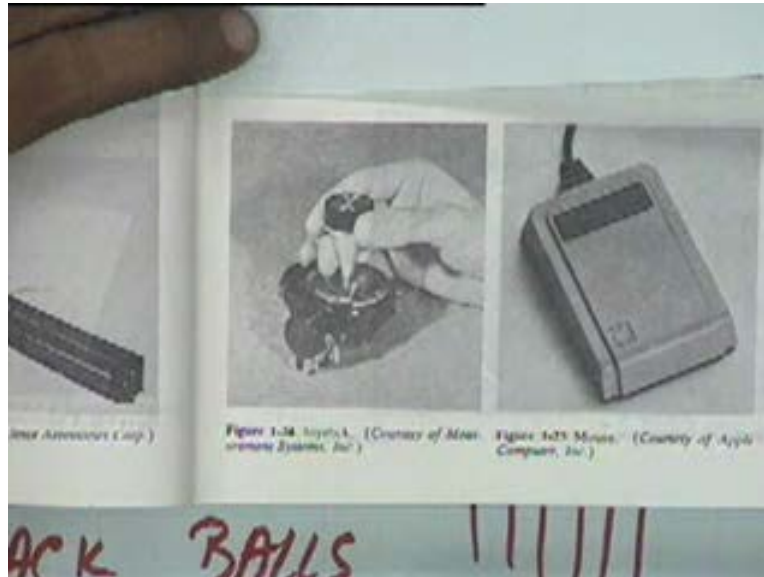
(Refer Slide Time: 00:01:23 min)



Let's say if we start with the input devices, we will first talk of what are analog devices and one of the most common analog devices that are used in CAD system is a mouse. The basic principle on which a mouse works is that you have a ball and under the ball, we have xy transducers which are fixed. So these are basically let's say x and y variable resistors. As a result of this, if you move the mouse in the x direction the resistance of the x variable resistor will change. If you move it in the y direction, the resistance of the y variable resistor will change and this resistance is then taken as a measure of the position of the mouse. As you move the mouse on the board, there is a set of resistors inside whose resistance is changing. This is what is called the mechanical type of mouse. We also have light type and other types which essentially work on similar principles.

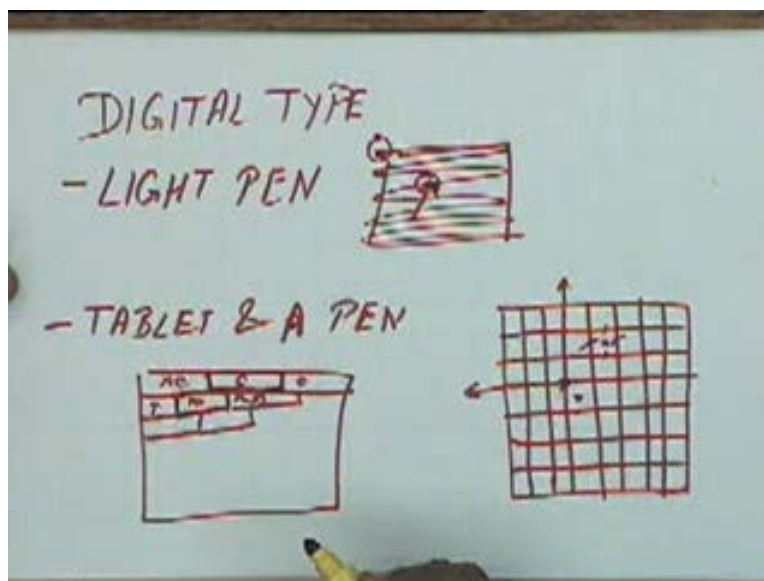
Similar to the mouse we also have trackballs and joysticks both of them work on similar principles. The trackball is a ball which is there almost next to the keyboard and you can change the orientation of the ball and that changes the position of the cursor on the screen so that is a trackball and joysticks I presume you are all familiar with joysticks. Have you seen joysticks? No, in this book we have a figure of a joystick. Can you see that?

(Refer Slide Time: 00:03:39 min)



This one, this is a figure of a joystick. I move this stick in either direction, the resistors fixed at the bottom, the resistance changes and their position can be seen on the screen. This figure here is of a mouse, mouse you are all familiar with it. You have a single button mouse or three button mouse or a four button mouse, so this is a mouse. All these are analog devices; by analog device we mean that the cursor location that is output from this device is an analog position. You know the difference between a analog signal and a digital signal. Digital signal will be in discrete numbers or discrete quantized numbers, analog signal is a continuous signal because position given by these devices is a continuous position, position would be a continuous curve like this. So these are all analog devices. In contrast to these we also have digital devices.

(Refer Slide Time: 00:04:56 min)



Digital type input systems one common variety, one common type is what is called as a light pen. In a light pen you have a light sensitive diode which is used to point on to the screen. Let's say this is your screen, you take a diode and point it on to the screen like this. I think I might have a figure here.

(Refer Slide Time: 00:05:43 min)



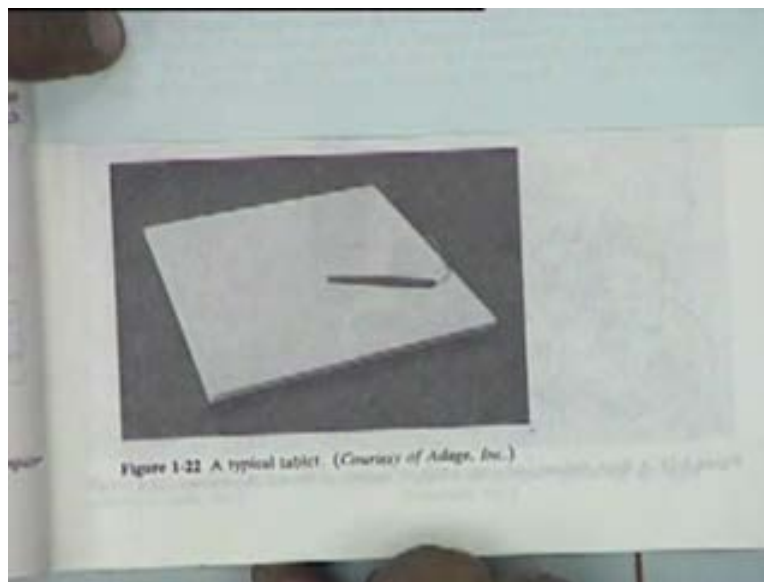
Are you able to see this figure clearly? See this is the screen and on the screen you see the hand of a person which is holding a small pen. From this pen he is picking up items from the menu, he can directly point your item on the menu and click that. Normally if you have a mouse you are moving the mouse on the pad or on a board. In this you don't have to do that, in this you just take the pen and point it on to the screen and click it there. This basically works on a principle of a sensing the light signal over there and the position of the screen is then sent back to the computer or to the CPU.

The way it works is let's say we will be talking of raster graphics devices. In raster graphics devices there is an electron gun that electron gun starts from one corner and spans the full screen like this at a rate of about 40 times a second or 50 times a second, it will be spanning the full screen. that means the guns starts from one corner and goes to the right, comes back to the left corner again goes to the right and so on. So the moment the light pen senses the light over here, it knows how much time lag is there between starting point and this point. And that time lag is used to compute the total distance traveled and hence the position of the light pen at that point. This is how the position of the light pen is sensed and fed back to the machine. This is the basic principle of a light pen.

In another similar thing, not similar in another device that are used is what is called a tablet and a pen. Sir, all the other machine devices are equally sensitive. Please again. All the machine devices are equally sensitive with respect to time. With respect to time? What do you mean by sensitive with respect to time? Like equally responsive in a same period of time. This light pen is like setting up some time.

See more or less they are roughly the same because what happens is the response with which a person gives the input that is an orders of magnitude more than the time response of these devices. **Sir practically for the users.** So practically they are all the same but depending on the monitor that is being used, depending on the requirements we will see that let's say tablet can satisfy a different kind of requirements while a mouse cannot. So we will come to those things but as far as the response time is concerned just practically the same, response time is a fraction of seconds and people can't respond faster than that. Now a tablet is a rectangular pad, of this rectangular pad has got let's say a grids of wires under it and on the top of that we take a pen. I think I have a figure here.

(Refer Slide Time: 00:09:14 min)



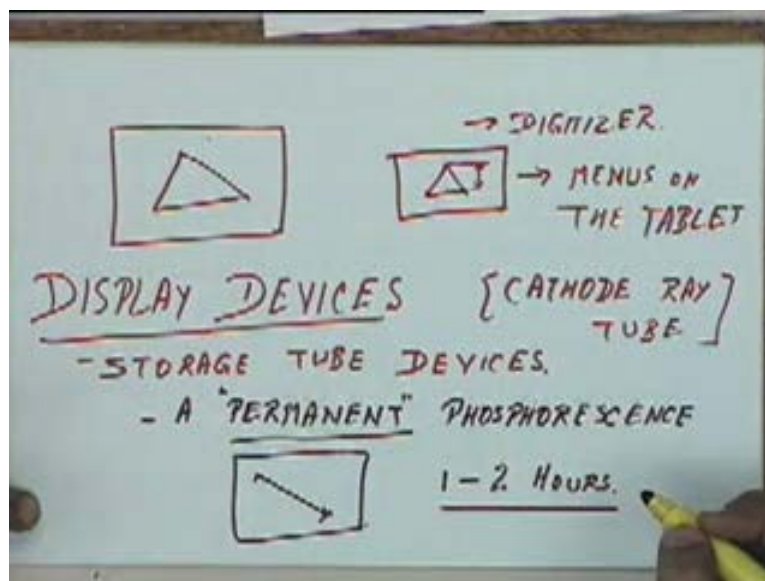
Now you are seeing is a rectangular pad and this pen. Now this pad is not like the ordinary pad of a mouse. The pad of a mouse is just a friction pad so that the ball can move easily on that pad but in a case of a light pen, this pad has got a grid of wires under it and the way it works you have this grid of wires under the pad and the pen emits a radio frequency signal. So the pen is located at this point, its emitting a signal at a radio frequency that signal is picked up by these wires. So the intensity of the signal picked up by each wire will be different. Let's say the pen is at this location, the intensity will be highest for this wire and for this wire. As a result of which the intensity of the signal that is available at each of these wires that can be used to calculate the exact position of the cursor of the pen. So if I take the pen and click it here, I will get some intensity here and some intensity on this wire.

Similarly I will get some intensity on this wire and some intensity on this wire. Using these intensity values I can find out the exact location of the pen inside this grid and the intensity that is picked up by any of the other wires will be much smaller than this intensity. So this tablet and a pen give us an additional advantage that on this tablet, if I am clicking my pen anywhere I am getting the exact coordinate of that pen on this tablet. If I have a mouse and when I move the mouse on the pad, I do not get the exact position on the pad. At different positions, location of the pad in the case of a mouse I will still get the same cursor location. You can try it out, you

move the mouse, lift it up and move it again. If you lift the mouse and keep moving it again and again your cursor will keep moving in the same direction that will not happen in the case of a tablet. In the case of a tablet even if you lift up the pen and take it back to the same location, you will still get the same location on the screen. The reason is that we are sensing the intensity signal at each of these wires and the advantage we get from this let's say if we take a menu and paste it on this tablet. On the menu I can have a set of commands. So if my menu looks like this, may be some command here, some command here, some command here and may be I have a some symbols over here. So if I take my pen and take it to any of these locations and click there, I can click that command because my tablet is sensing the exact location on that grid which means that if I click my pen on this, I mean **in a set** giving this command.

Similarly if I click my pen on this location I mean if I giving this command, so I can use this tablet and a pen to make a menu on the tablet itself, this can be in addition to the menu on the screen. This cannot be done by most of other devices.

(Refer Slide Time: 00:13:40 min)

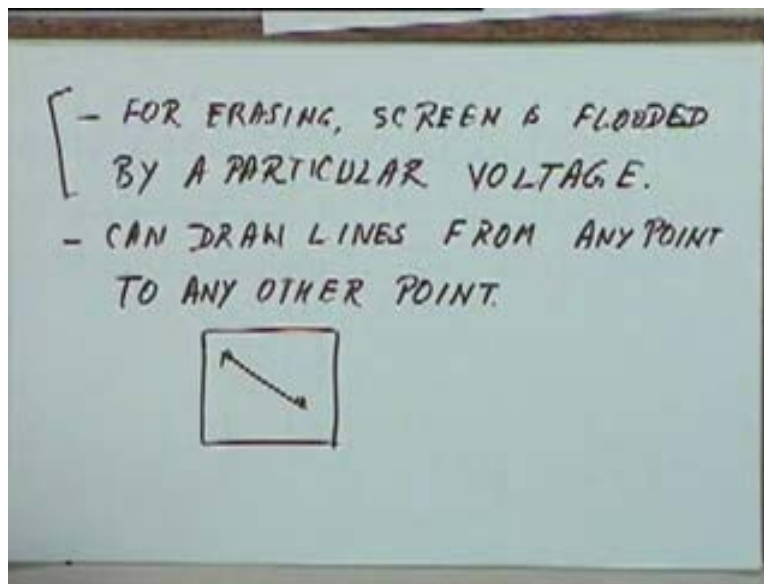


Similarly let's say if this is the screen and this is my tablet, if on this tablet I keep a drawing let's say I keep a drawing like this. I take my pen and move it exactly along these corners; I will be able to get a similar figure on the screen because this point corresponds exactly to the point that is being picked. Similarly this point will correspond exactly to the point that I am picking by the pen. So I can digitize this drawing and get a figure on the screen so that is why sometimes this is also referred to as a digitizer. So the tablet has an additional advantage that it can be used as a digitizer plus we can have menus on the tablet. So these are some of the input devices which are very frequently used in CAD systems.

Now if we look at the display devices, any question with respect to the input devices? If we look at the display devices the first about display device that we will consider is what is referred to as a storage tube device or storage tube devices. Now most of these devices are variations of what is called a CRT or cathode ray tube. Most of the display devices used these days are variations of

cathode ray tubes. Are you aware of what is a cathode ray tube? Are you aware of its functioning how it works? Now in storage tube devices the essential, the important features that it has what is called a permanent phosphorescence on the screen. A Permanent phosphorescence what we mean is that on the screen if we make a particular figure or on the screen if the electron gun moves across this screen, let's say this is a screen and the electron gun is moving from this point to this point. The screen is having a permanent phosphorescence so this line will be visible, so to say permanently. Permanently means the visibility of this is of the order of about 1 to 2 hours. Let me, if you make any figure here, if you write any text on that it will visible for about an hour or two. So these days I presume you have not seen any of any such devices. Now if that figure or if that text is visible for 1 or 2 hours then how do you change the text? For changing the text what is done is a particular voltage is fed to the system, so the whole screen gets flooded so that affects, that amounts to erasing the complete screen.

(Refer Slide Time: 00:18:11 min)



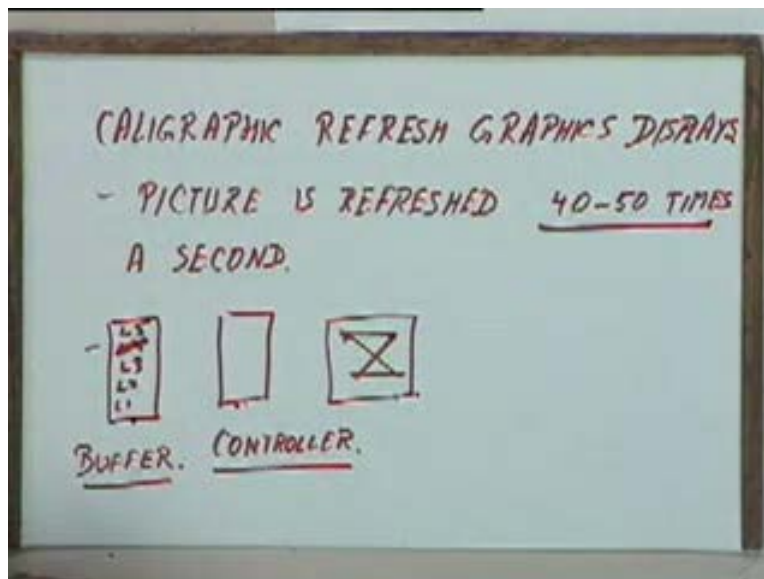
So for erasing, screen is flooded by a particular voltage. So the basic disadvantage of this type of screen is that every time you want to make any change, the whole screen has to be erased and then redrawn from beginning. We cannot erase hidden paths. If you write anything on that, that is going to be visible on that for an hour or two and if you want to erase it before that, erase the full screen and rewrite it again. That is the basic disadvantage of this. And this type of device, it can draw lines from any point to any point that means this is the screen and you ask it to draw a line from one point to a second, you can draw a straight line. This property is not shared by a number of other devices, we will see that very soon. The basic disadvantage of this is erasing is very difficult.



The other type of display device is what is referred to as calligraphic refresh graphic displays. The basic feature of this type of display is that here the picture is refreshed or redisplayed about 40 to 50 times a second. In the previous one we saw that the picture was made once and the picture was visible for 1 to 2 hours. Now the picture is redisplayed about 40 to 50 times a second. Since it is redisplayed 40 to 50 times a second, if you want to make any change is quite easy. And since it is redisplayed so frequently, it appears as a continuous figure to the eye because the eye is not so sensitive to be able to make out changes at this frequency. The way it is done is that if this is your let's say the display in the screen and you have some figures on it, this figure has to be redisplayed 40 to 50 times. So all these entities are stored in a buffer. This is a buffer, this buffer is nothing but it's a certain memory location.

All the entities which have to be displayed or description of each of these entities will be available in the buffer and then we have what is called a controller. What the controller does is it looks at the buffer whatever the list of entities in the buffer, it will draw those entities on the screen and it will keep doing that 40 to 50 times a second. So at the rate of about 40 times a second it will read the buffer, display it on the screen, again it will read the buffer and display it on the screen it will keep doing that. So the job of the controller is just to do this thing in a cycle. Whatever is there in the buffer read it, display it, read it, display it, read it, display it and so on that is all that a controller is supposed to do.

(Refer Slide Time: 00:20:00 min)

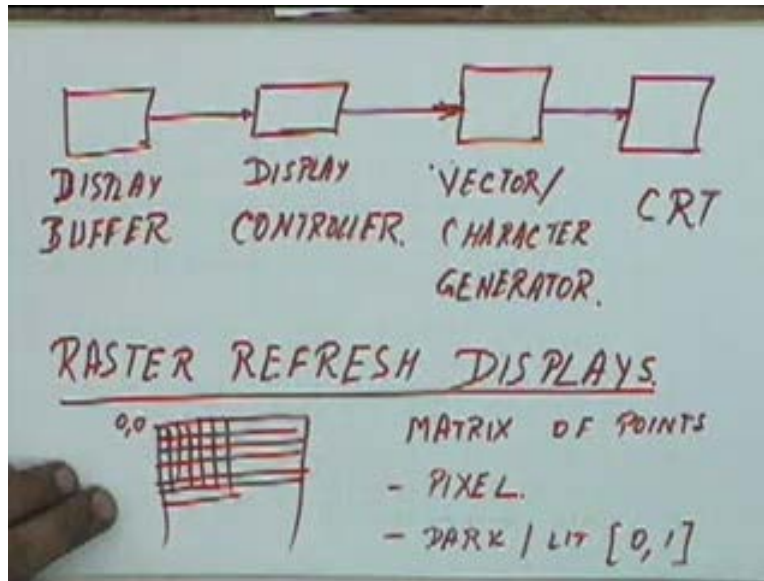


Now the programmer if he wants to make any changes in this drawing let's say if he has line one, line two, line three he has put all these entities in the buffer, we have 4 lines here so line one, line two, line three and line four. The description of four lines is there in the buffer. Now he suddenly decides that is line four is not required, he just deletes this line four from the buffer and this buffer is just a memory location. So in that memory location he deletes this line four. As a result when the controller looks at the buffer the next time, it does not find the line four which means that till the previous iteration it was drawing all the four lines, in the next iteration it will draw only three lines.

Since the controller is working at a very high frequency, at one point of time will suddenly find that this line four is not visible, that amounts to deleting this line four from the screen. So in this type of device, making changes in an existing drawing is much easier. In the previous type of storage tube type of devices, making any changes was quite difficult. We have to redraw the complete screen. Here the screen is being redrawn 40 times a second anyway. So for making any change we only have to make changes in the buffer.

Similarly if you want to add any entities, we will only add entities over here. The programmer will now only see the buffer, all that he has to do is go and make changes in the buffer, he will go and make some entries in the buffer or delete some entries from the buffer. He is not going to do any, he is not going to give any command on the screen directly, so these are called calligraphic refresh graphic displays.

(Refer Slide Time: 00:25:20 min)



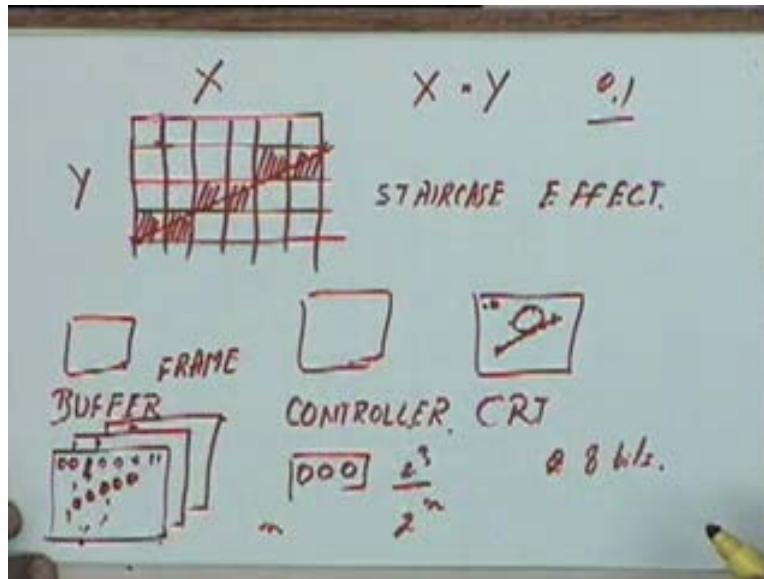
Essentially the way they are working is we have a display buffer followed by a display controller and this is followed by what is called a vector or character generator and this is finally followed by the CRT, a cathode ray tube. A vector or a character generator is just a device which will generate a particular character or which will move the, which will give instructions to the cathode ray tube to move in a particular manner so that a particular character is generated and so on or a particular vector generated.

Controller reaches the buffer and invokes the vector or character generator at a particular rate 40 times a second, 50 times second or whatever. So this is the basic working of a calligraphic refresh graphics display. The next type of graphics display is what we call as raster refresh displays. The raster refresh displays are the most commonly available displays these days. Probably all the displays that you have seen so far, they are all raster refresh displays and we will also be seeing how these displays can be programmed and how we draw lines and circles on these displays. In these displays the complete screen is divided into an array of points, so full screen is divided into an array of points, the top left corner is normally treated as a 0 0.



This matrix of points, in this matrix each of these squares can be addressed as a particular dot and this is also referred to as a pixel. If we are talking of a black and white or monochrome monitor each pixel can either be dark or it can be lit. If it is dark let's say we will call it a 0, if it is lit we will call it may be a 1, so each pixel can be made either dark or it can be lit.

(Refer Slide Time: 00:28:54 min)



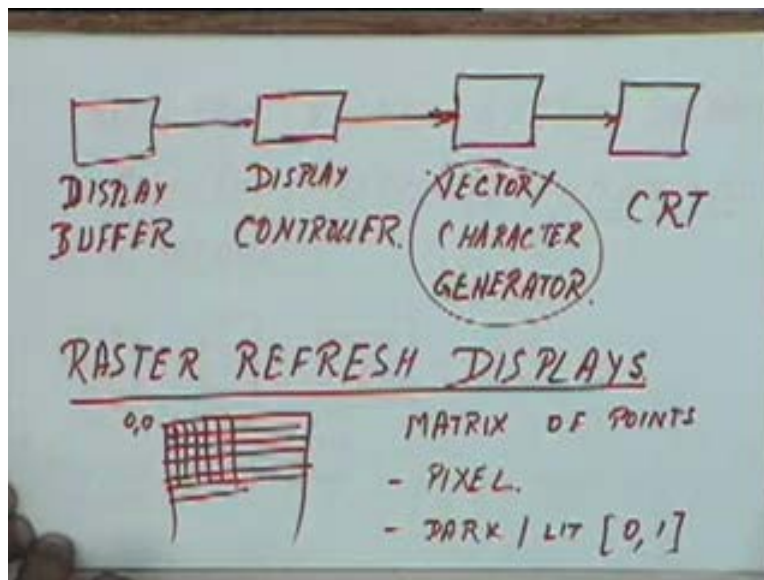
On this screen let's say I will make it on a larger scale. If I want to make a line let's say from this point on to this point, my line should ideally be like this but now I can take each of these points and either make them dark or make them bright, I can either make them a 0 or a 1. So if I make this dark maybe I should make this dark, this dark and this dark, maybe I will make this dark and this dark I mean I will make them bright **sorry** I will make these pixels bright. If these pixels are bright my line would look like this, even though ideally my line should look like a straight line. So in these types of displays, lines and curves never look like line and curves. We always have the so called staircase effect because each pixel is addressed separately as a distinct point and again we are calling them raster refresh displays, so the complete picture on the screen has to be displayed about 40 to 50 times a second.

So for displaying it again and again, we follow a similar method that whatever is displayed on the screen that is first kept in a buffer and I think I have a figure here. Now the only difference is in this buffer or rather in the previous buffer, we were storing the description of entities. You should remember this. In a calligraphic refresh display, my buffer contains the description of these lines and my controller was reading this description and drawing the lines but now my buffer has to store... on each of these points. So if this grid of points as X by Y that means the resolution is X units in the X direction and Y units in the Y direction that means the total number of pixels is X multiplied by Y. Then to represent this screen we need a buffer equivalent to so many bits, each bit can be a 0 or a 1. Again we are talking of monochrome monitor. Monochrome monitor with no gray levels, either it is dark or it is light. So each pixel will be a 0 or a 1. So the complete screen will be stored as an array of zeros and ones, so each pixel will have a 0 0 1 0 0 I mean in this way.

Each 0 or 1 will correspond to one pixel on the screen. So this buffer is going to contain a set of zeros and ones or 1 0 or 1 for each pixel on the screen. So this is also referred to be called a frame buffer. This frame buffer is essentially a set of zeros and one such that each 0 or 1 corresponds to a particular location on the screen, it corresponds to a particular pixel on the screen. Now from this frame buffer we have a controller and then the DV monitor, this is CRT and this is a controller. All other controller will do is it will look at the frame buffer, at a particular pixel if it is 0 it will make it off over here, if it is 1 it will make it on and it will repeat that process 50 times a second or 40 times a second. It will keep reading this frame buffer, it will read the frame buffer, update the CRT, read the frame buffer and update the CRT, the whole cycle will be repeated 40 to 50 times a second and the programmer will now modify only this frame buffer.

So if I have to draw a line from this point to this point, I will have to decide which are the pixels which have to be made on, which are pixels which have to be bright corresponding to those pixel I will make ones in those locations. All the others I will keep them at 0. If I have to delete this line, I will change these ones to zeros.

(Refer Slide Time: 00:34:55 min)



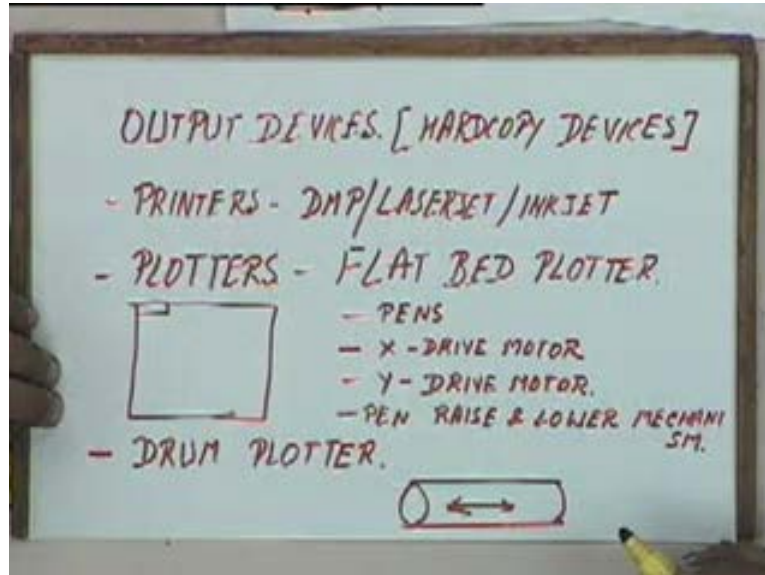
So in this the job of the programmer is slightly more difficult than the job in the previous case because in the previous case, in the case of a calligraphic refresh display all that the programmer had to do or they will go and put a line just the description of the entity  $L_1$   $L_2$   $L_3$  or  $L_4$  or whatever. You just write down the details of the line this is starting point, this is the end point. the rest of it will be taken care of by the vector generator. If I say that my line is starting from one point ending at a second point, my vector will give the instructions to the gun to move from that point to the second point. But in the case of a raster graphics display, the programmer has to make changes in the frame buffer. If he wants to make a line or if he wants to make a circle, he will have to find out which are the pixels which should be on and corresponding to that he will have to make ones over here. So programmer's job is more difficult.

But the hardware is slightly simpler because we don't have a vector generator, we don't need a vector generator anymore. The disadvantage in having a vector generator is that if you want have some complicated curve, it is very difficult to generate that. For the vector generator we will be able to generate only a particular set of vectors, particular types of vectors. If you want to make an arbitrary curve we cannot do that, while that can be done in the case of a frame buffer. I will go and make the changes in the frame buffer and the corresponding line will be generated. This was we said, we have a frame buffer but this is in the case of a black and white monitor. For each pixel is either on or off, it is either displayed or it is not displayed but if you have a colour monitor then each pixel can take a set of colors.

So what is typically done is that we will have a number of such frame buffers. In each of these buffers, there will be one location for a particular pixel. So for describing one pixel, we will have one bit here, one bit here and one bit here. Let's say if we have three such frames then for describing each pixel, we will have three bits available. This is the first bit, this is the second bit, this is the third bit. So if we have these three bits then I can have a maximum of 2 to the power 3 that is 8 possible colors. If I have n bits, I can have 2 to the power n colors. Essentially this combinational bit will correspond to a particular color or particular intensity. So whenever you hear that the so and so some monitor is capable of 256 colors, 256 will... [Conversation between Student and Professor - Not audible ((00:38:04 min))] Yeah, that will have 8 buffers. So each pixel will correspond to 8 bits. Any question up to this point?

We will be talking of raster refresh displays or raster graphics displays in a bit more detail when we talk of different algorithms for drawing lines and circles on these displays. So we will be seeing how we can decide whether this pixel should be on or not. In fact in this case you might have noticed at this particular pixel, I did not put that on even though this line was touching this pixel. Similarly here this pixel, I will not put that on. Why? We will be looking at those issues very soon. Now before we go into those issues that is details on how lines and circles are drawn, let's quickly see some of the output devices.

(Refer Slide Time: 00:39:22 min)

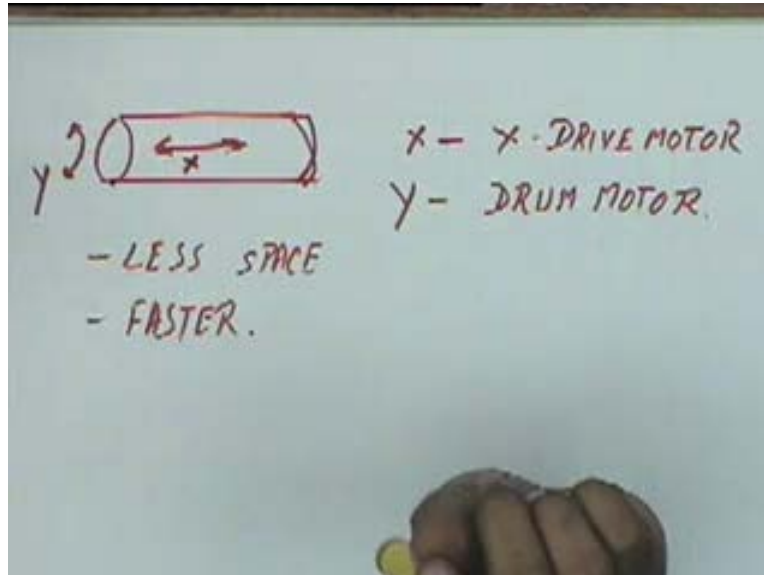


By output devices I mean hard copy devices. I think you are probably all familiar with different kind of printers, dot matrix printers or laser jet printers or inkjet printers and so on. Probably you are familiar with these different kinds of printers. They can all be used for taking graphic print outs, probably you are aware of that also. What we will see is briefly different type of plotters that can be used for taking graphics printouts. Have you seen any plotter, are you familiar with plotters? No, so plotters they are basically two types, first is what is called a flatbed plotter and the second is what we call as a drum plotter.

In a flatbed plotter **we have a** let's say we have a table, on the table we fix a sheet of paper. Now on this sheet of paper we will have a set of pens. You can see the pens are mounted on the corner, the pens can move in the x and y directions. So the pens will be mounted on a, may be a carriage or something like that and they can move in a x and y direction, the paper is fixed. So if you want to make a drawing of a A four size, we will a take a paper of A four size, fix it on this plotter and the pens will move in the x and y directions. So we have let's say a set of pens which will be there even in a drum plotter. For moving the pens in the x direction, we have a X drive motor and for moving the pens in the y direction, we will have a Y drive motor and as the pen is moving we need a mechanism to either lower the pen for drawing it or to raise it so that no line is drawn during the motion. so we have a pen raise and lower mechanism.

So in this case essentially a paper is fixed on to the table and the pen can move in a x and y direction. In the case of a drum plotter you have a drum, on this drum the paper is fixed. the paper can hang on both the sides and the paper is fixed on this. The drum can rotate, by the rotation of the drum we get the motion in the y direction and the pens can move in this direction, so x direction motion.

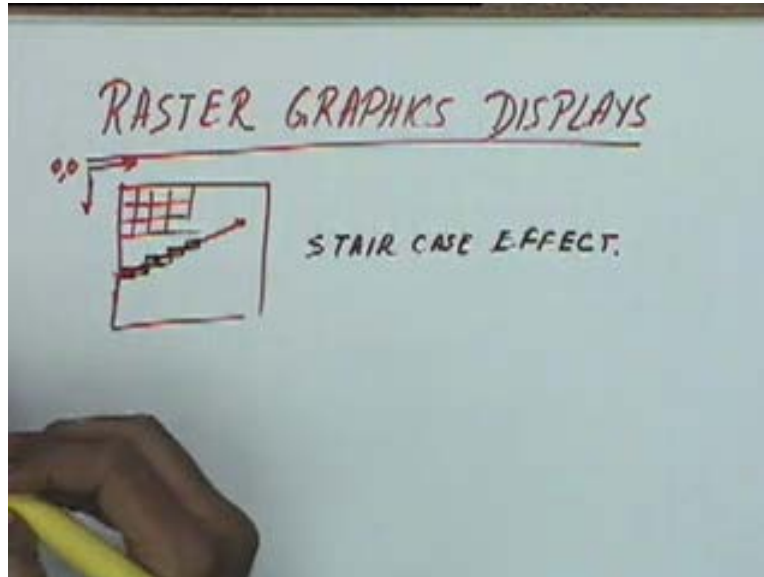
(Refer Slide Time: 00:43:31 min)



Pens move in this direction and the drum moves in this direction and this gives the x direction motion, this gives the y direction motion. So the x direction motion is given by let's say an X drive motor and the y direction motion is given by the drum motor and of course we have a mechanism to raise and lower the pen. The basic advantage of this is it takes less space and the total moment of the pen is much less and typically these plotters are faster. In fact some of these plotters you will be able to see them in our computer center. If you go there you will be able to see these drum type plotters and in fact the flat bed type plotter will be there in the departmental computer lab, you will be able to see this. So there are two types of plotters which are used for taking printouts of I mean graphics printouts. Essentially they are used as output devices in CAD software. Any questions with respect to input output devices?

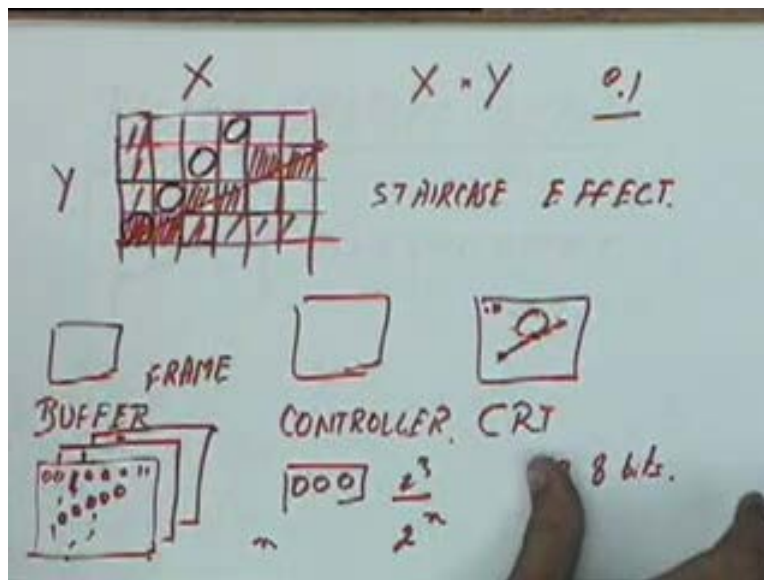


(Refer Slide Time: 00:45:30 min)



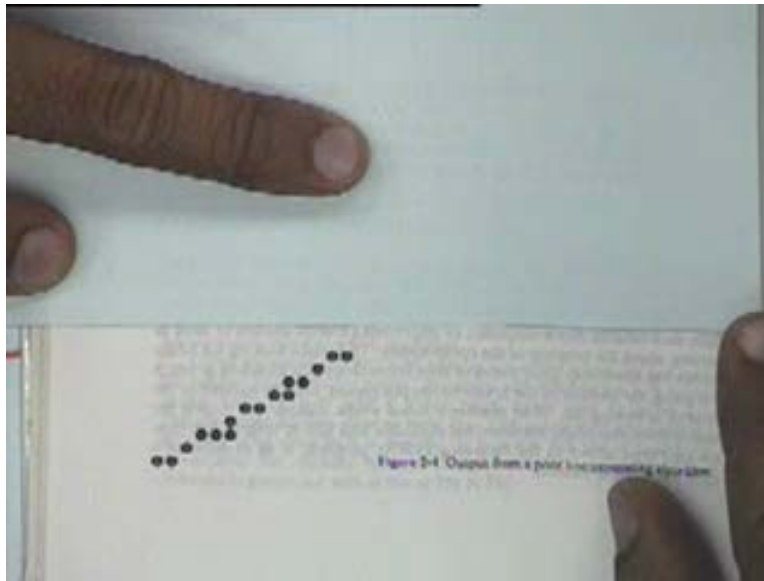
In that case I will briefly give, I will briefly start the topic of raster graphics. We are coming back to this topic and we will see how the raster graphics displays are programmed for drawing different kinds of entities. I have just mentioned in this type of displays, the screen is divided into an array of pixels with 0 0 being the top left corner, this is by convention and I also mentioned that if I want to draw a line from one point to another point then depending on the set of pixels that we choose the line will look staggered, this is referred to as staircase effect. If you want to draw this line instead I will get the previous figure.

(Refer Slide Time: 00:46:51 min)



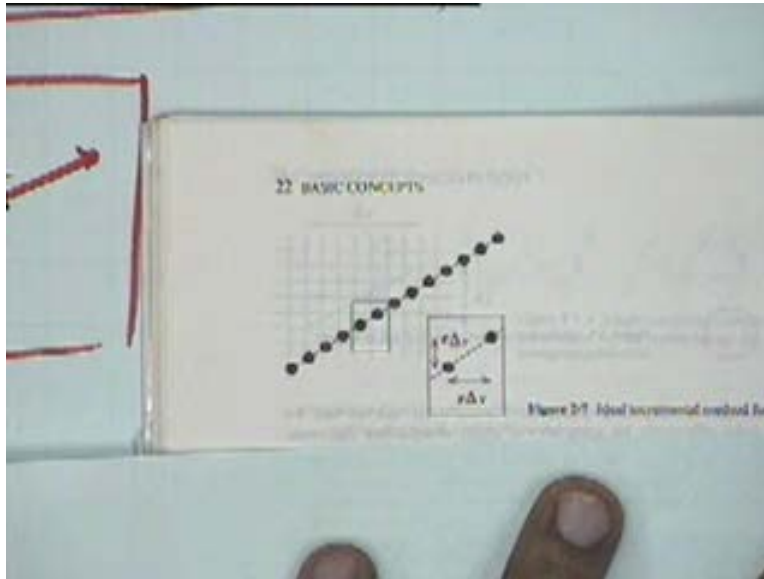
If this line was being drawn at an angle of zero degrees, then it would have been very easy to decide which pixels are to be displayed, I can just display this this this and so on. Similarly if the line was being drawn at 90 degrees, it would have been very easy to draw the line. Even at 45 degrees I can easily draw the line as in this pixel, this pixel, this pixel and so on. But at any other arbitrary position, it is not so easy to draw a line because I have to decide which pixels have to be on and which should be off. And if I do not decide that correctly, my line might not look uniform, it might not look good. In fact I think I have. If you look this line that has been drawn, are you able to see it?

(Refer Slide Time: 00:47:52 min)



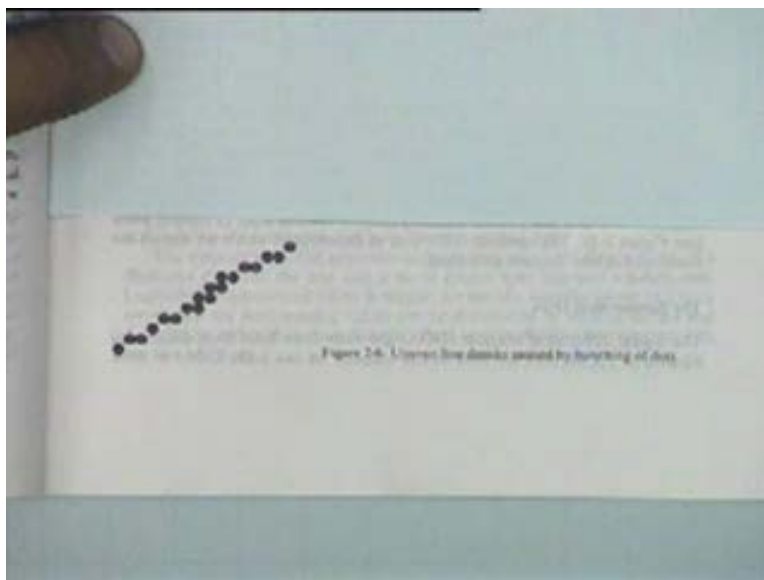
Now this line does not look uniform, it does not look good. In contrast to this if you look at this line, it looks much more uniform. Are you able to see this line? this lines looks much more uniform.

(Refer Slide Time: 00:48:20 min)

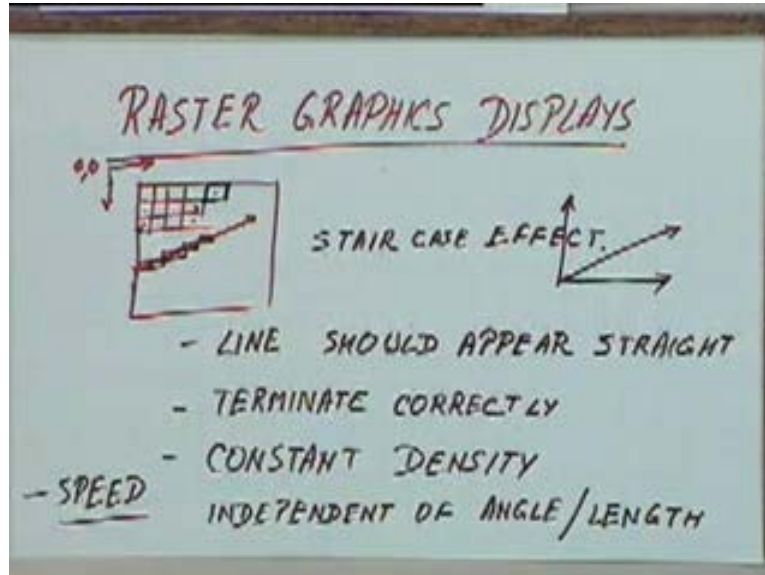


Similarly if you look at this line, again this line is not looking uniform. It's having a much greater intensity in this area and much lower intensity at other two corners. All this is happening because the algorithm used to decide which pixels are on, algorithm is not correct. We are not getting a line of uniform intensity in uniform brightness, so we will not go into details of the line algorithm today, we will just see what are the essential characteristics which a line algorithm should have in order to be suitable for CAD purposes.

(Refer Slide Time: 00:48:46 min)

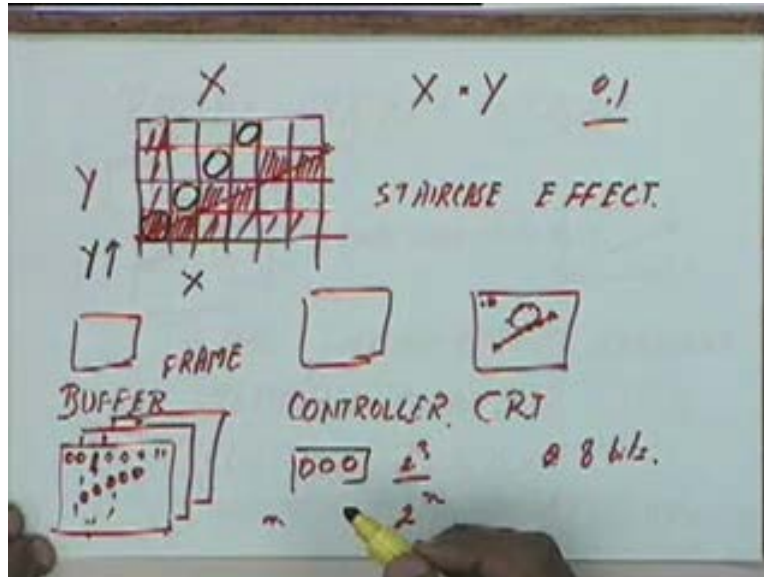


(Refer Slide Time: 00:49:41 min)



The first is that when we draw a line from this algorithm, it should appear straight. The second is it should terminate accurately or correctly that means if the line is to be drawn from this pixel to this pixel or let's say from this point to this point, it should start from this pixel and end at this pixel, it should not start at this pixel and end at this pixel. It looks quite obvious when I draw it at this magnification and when the pixels are very small, this becomes a very critical question. In fact in this case itself when I am saying this corner, this corner pixel would normally be this pixel. In such a case my line would be ending here which means effectively it would look like ending over here. So those kinds of pixels, one has to be careful about, it should terminate correctly. Then it should have, the line should have constant density. This density should be independent of angle of the line or length of the line. That means whether I am drawing a line on a horizontal line or a vertical line or a line at any arbitrary angle, it should look like having the same intensity.

(Refer Slide Time: 00:51:51 min)



Again this is not very easy because let's say if you look at this figure, when number of pixels in the X direction is X, in Y direction the number of pixels is Y. So if I am moving let's say at an angle of 30 degrees, the number of pixels that I will be crossing will still be X in the X direction. While if I am moving in this direction my number of pixels is still X. If I am drawing line at 45 degrees, you can see that these dots are spaced at a greater distance. While when I am drawing a line which is horizontal, these dots will be much closer. So same number of dots, 4 dots will appear on a much smaller distance, in 45 degrees we will have 4 dots in much larger distance. So it is not always possible to get lines in the uniformed intensity at different angles and of course my line drawing algorithm should be fast, the speed of the algorithm should be good.

The speed of the algorithm is low then the algorithm is not a good algorithm. So these are the some of the requirements of a good line drawing algorithm. In the next class we will take up some of the line drawing algorithms and see how they perform with respect to one another. That's all.