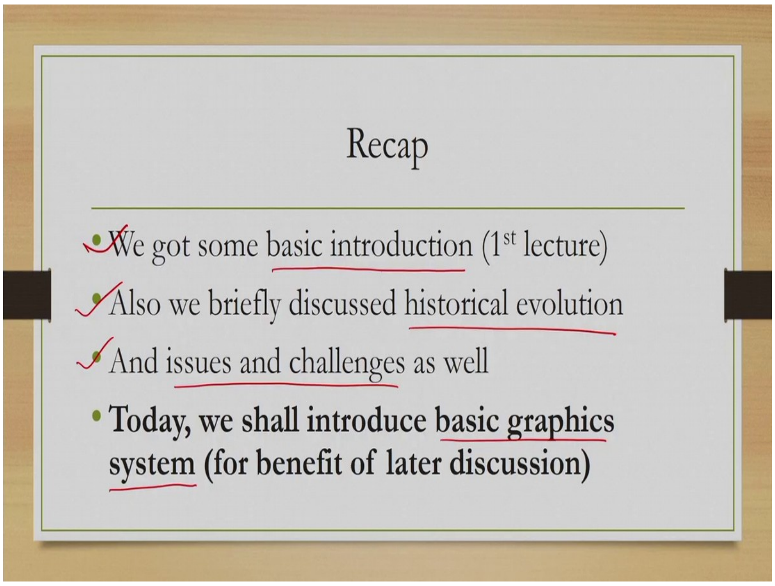
Computer Graphics
Professor Doctor Samit Bhattacharya
Department of Computer Science and Engineering
Indian Institute of Technology Guwahati
Lecture 3
Basics of a graphic system

Hello and welcome to lecture number 3, in the course Computer Graphics. Before we go into the topics of today's discussion, let me briefly recap what we have learnt in the previous lectures.

(Refer Slide Time: 0:45)

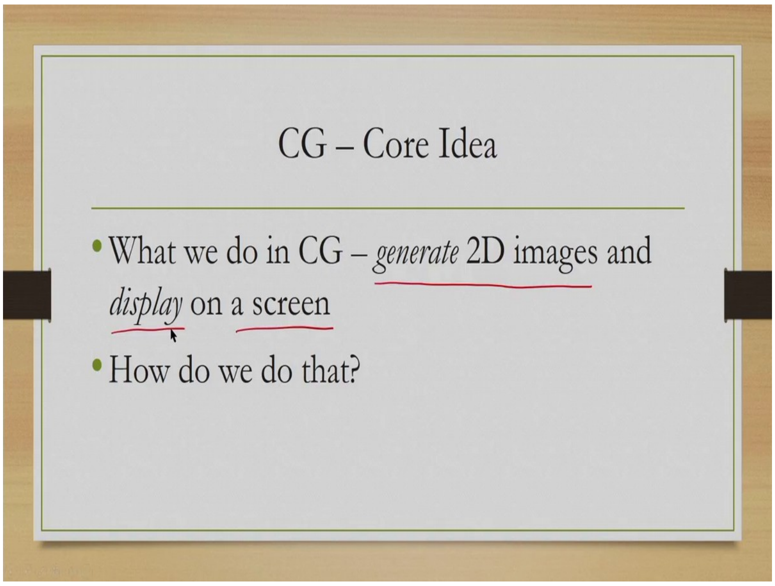


Recap

- We got some basic introduction (1st lecture)
- Also we briefly discussed historical evolution
- And issues and challenges as well
- **Today, we shall introduce basic graphics system (for benefit of later discussion)**

So in the first lecture we got some basic introduction to the field, what is graphics and what are the main characteristics of this field. This was followed by a brief discussion on the historical evolution as well as the issues and challenges that confronts the researchers and the workers in this area. So, these three topics we have covered in the previous lectures. Today, we shall introduce a basic graphics system so that in subsequent discussions it will be easier for us to understand the content.

(Refer Slide Time: 1:25)

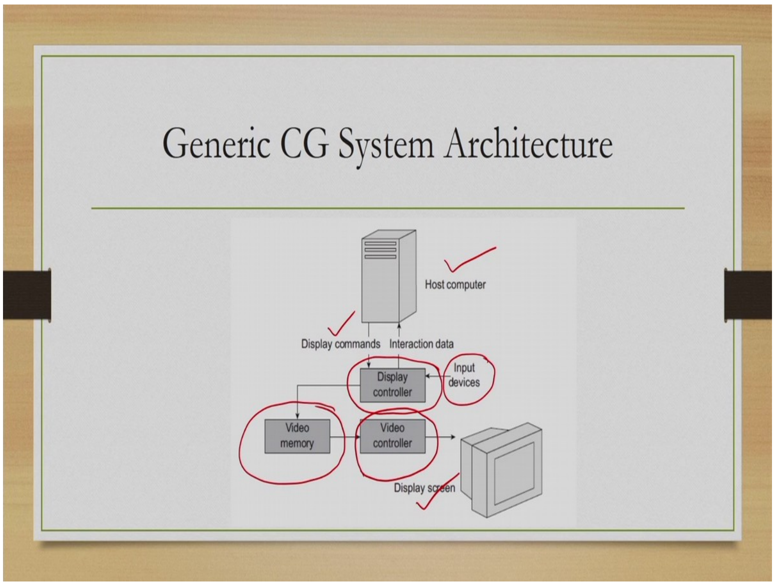


CG – Core Idea

- What we do in CG – *generate* 2D images and *display* on a screen
- How do we do that?

So, what we do in computer graphics? The answer is simple, we generate or synthesize a 2D image from some scene and we display it on a screen. So essentially generation of the images and display on the screen. Now, how do you do that? So in the previous lectures we went into some details of this questions, now let us try to understand the answer from the perspective of the graphics system.

(Refer Slide Time: 2:00)

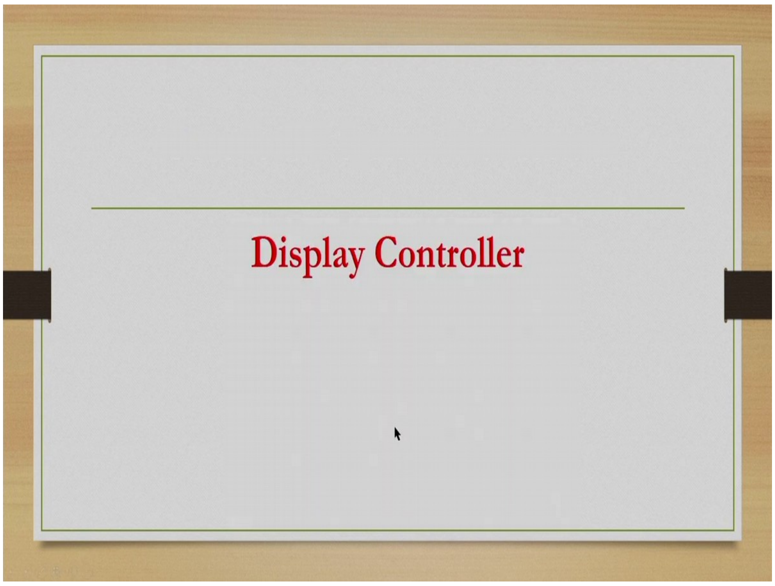


Generic CG System Architecture

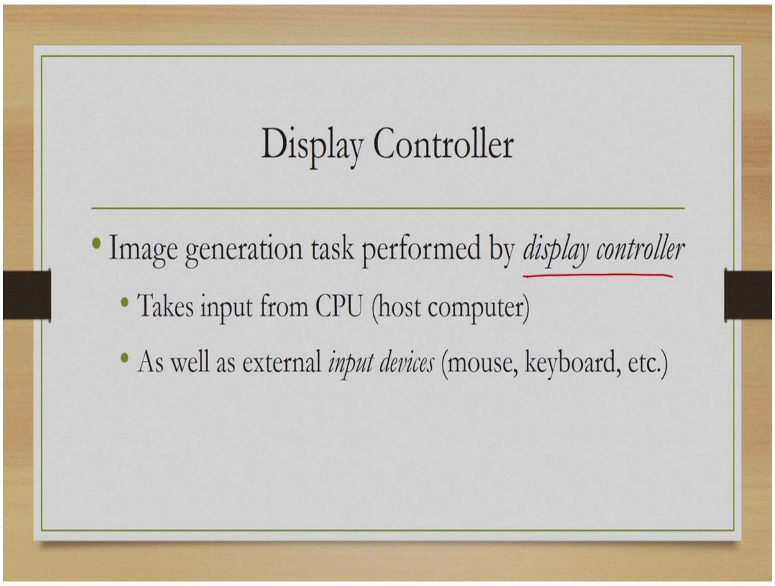
So if we look at a graphic system, the components that are likely to be there looks something like this. So we have a host computer, where all the processing takes place, then we have a display controller one component of the graphics system and this display controller takes input from the host computer in the form of display commands and also it takes input from input devices, various input devices we mentioned earlier for enabling us to interact with the screen content.

Now the output of the display controller goes to another component called video memory. Video memory content goes to third component called video controller which eventually displays or which eventually helps to display the image on the display screen. So, there are broadly three components that are unique to a graphic system; display controller, video memory and video controller. So we will have a discussion brief discussion on each of these components for better understanding.

(Refer Slide Time: 3:40)



Display Controller

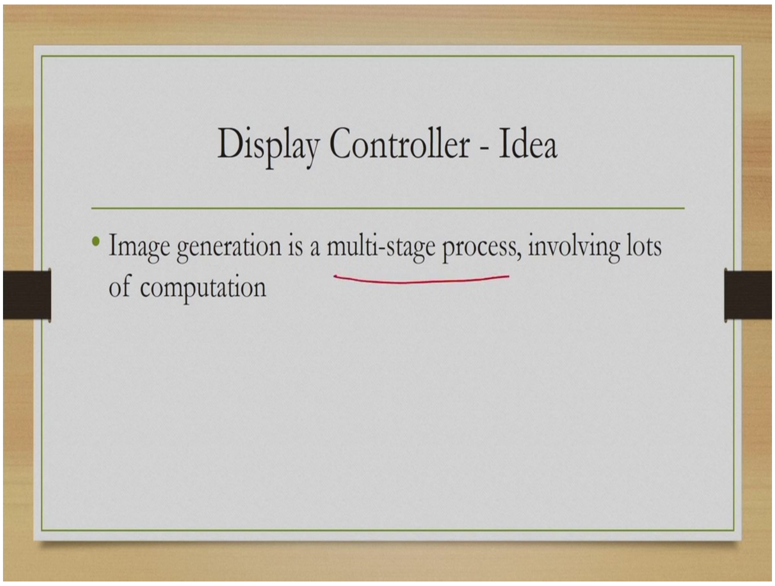


Display Controller

- Image generation task performed by *display controller*
  - Takes input from CPU (host computer)
  - As well as external *input devices* (mouse, keyboard, etc.)

Let us start with display controller. Now image generation task is performed by the display controller, so when you say that in computer graphics our primary objective is to generate an image, that generation task is performed by the display controller and it takes input from the CPU of the host computer as well as external input devices such as mouse, keyboard, joystick etc.

(Refer Slide Time: 4:12)

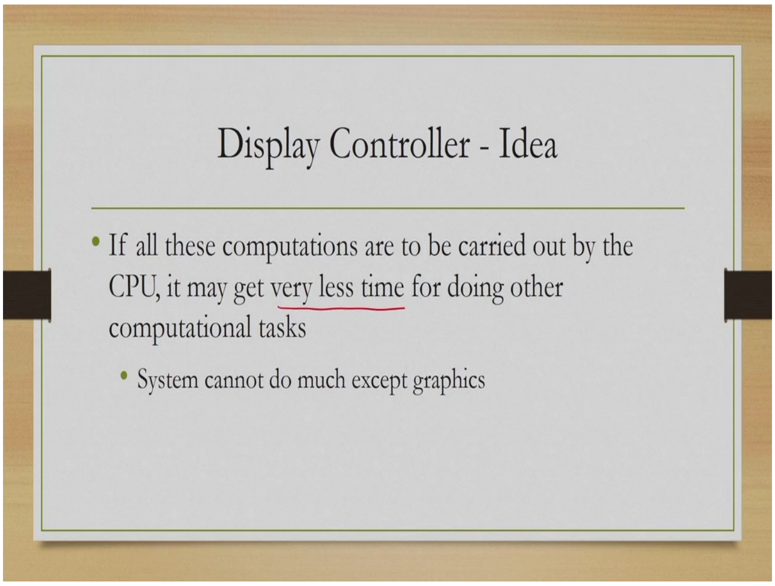


Display Controller - Idea

- Image generation is a multi-stage process, involving lots of computation

And based on these inputs it generates images, now these images are generated following a multistage process which involves lots of computation.

(Refer Slide Time: 4:30)

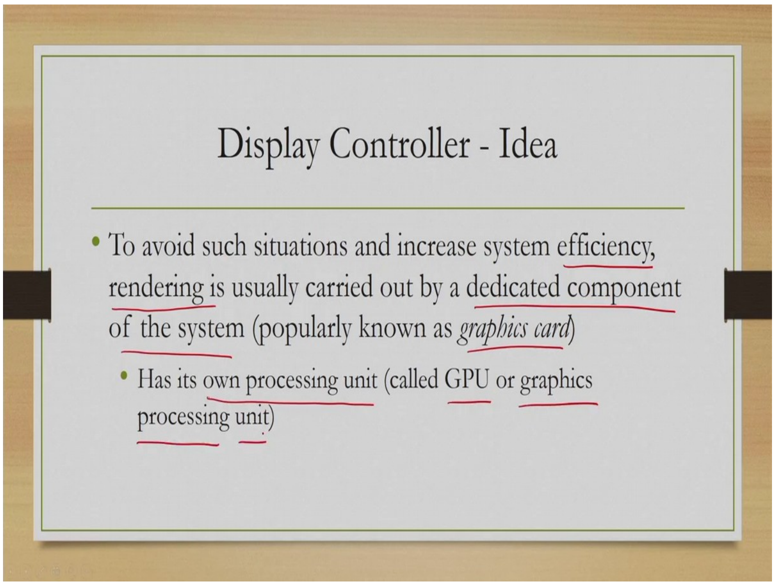


Display Controller - Idea

- If all these computations are to be carried out by the CPU, it may get very less time for doing other computational tasks
  - System cannot do much except graphics

One concern here is that if all these computations are to be carried out by the host CPU, then it may get very less time to perform other computations. So a computer is not meant only to display, it is supposed to perform some other activities as well. Now if the CPU is engaged with only the computations relevant for display, then it will not have time to perform other computations which in effect will affect the throughput of the system. So in such a situation the system or the host computer system would not be able to do much except graphics which definitely is not a designable situation.

(Refer Slide Time: 5:20)

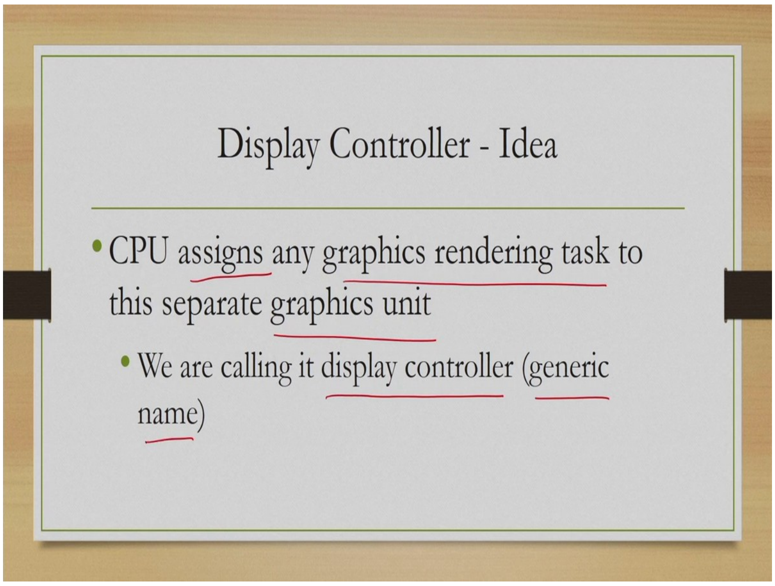


Display Controller - Idea

- To avoid such situations and increase system efficiency, rendering is usually carried out by a dedicated component of the system (popularly known as *graphics card*)
  - Has its own processing unit (called GPU or graphics processing unit)

To avoid such situations and increase efficiency of the system the job of rendering or displaying is usually carried out by a dedicated component of the system which probably some of us or all of us had heard of is called graphics card. Now in this card there is a dedicated processor like CPU we have a dedicated processing unit for graphics computing which is called GPU or Graphics Processing Unit. Later on will have one lecture on the basic idea of GPU, for the time being will just mention that there is a unit called GPU in the graphics card.

(Refer Slide Time: 6:24)



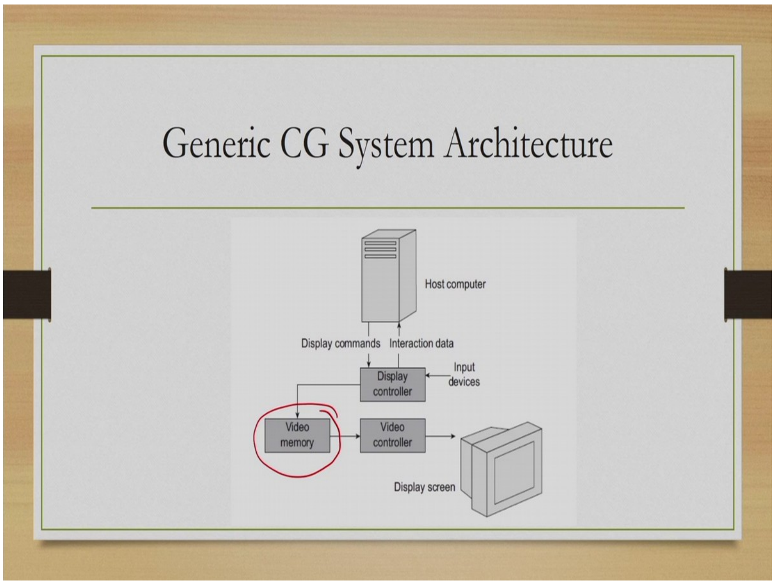
Display Controller - Idea

- CPU assigns any graphics rendering task to this separate graphics unit
  - We are calling it display controller (generic name)

And the CPU as science any graphics rendering task to this separate graphics unit and we call this graphic unit as the display controller which is a of course generic name and in different systems it is called in different ways. So essentially display controller deals with performing the multi-stage operations required to create or synthesize a 2D image.

(Refer Slide Time: 7:15)

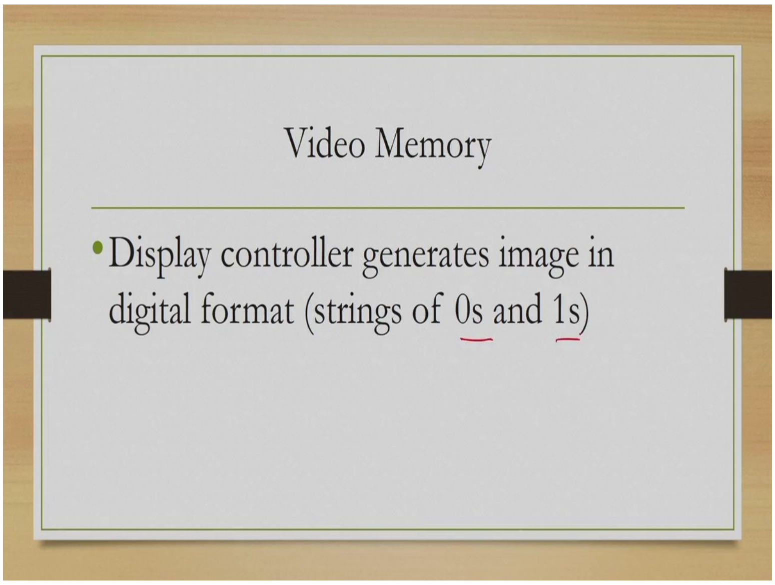
Video Memory



Generic CG System Architecture

Now the second component is video memory, so output of display controller is some representation of the 2D image and in video memory which if we recollect from this generic architecture which takes as input output of the display controller, it stores the representation.

(Refer Slide Time: 7:29)

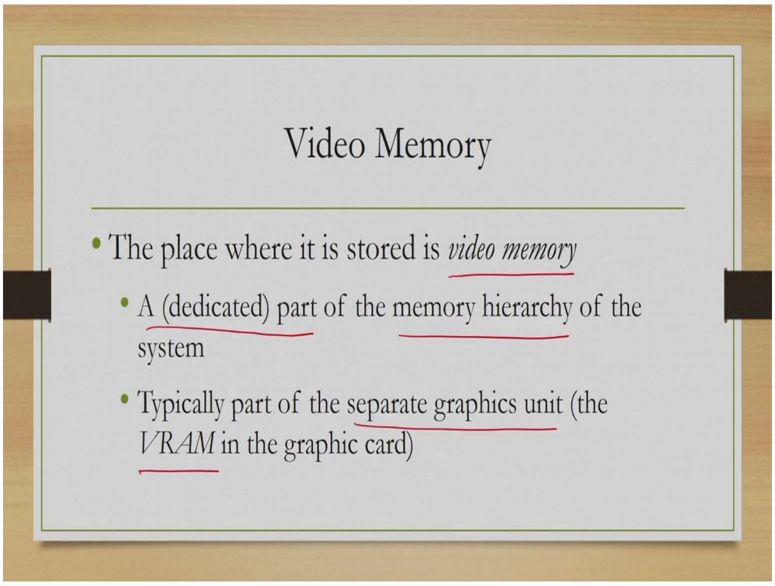


Video Memory

- Display controller generates image in digital format (strings of 0s and 1s)

Now display controller generates the images in the digital format strings of 0’s and 1’s which is expected because computer understands and processes information only in terms of 0’s and 1’s.

(Refer Slide Time: 7:45)



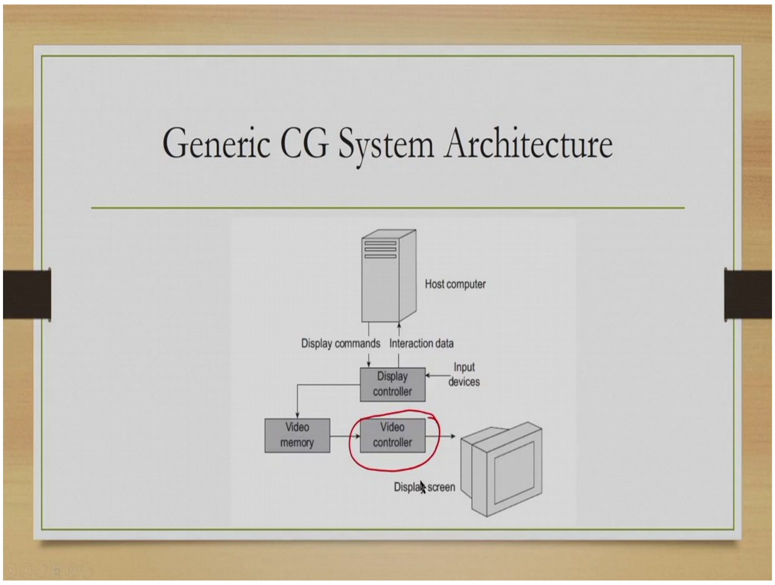
Video Memory

- The place where it is stored is *video memory*
  - A (dedicated) part of the memory hierarchy of the system
  - Typically part of the separate graphics unit (the *VRAM* in the graphic card)

The place where we store it is simply the video memory which is a dedicated path of the memory hierarchy. Now as we all know in the memory hierarchy of a computing system we have RAM, ROM, secondary storage, cache different levels video memory is also a part of those levels in the hierarchy and typically it is situated in the separate graphics unit or graphics card which is more popularly called VRAM or video RAM probably many of you or all of you have heard of this term. So display controller generates image representation and stores and that representation is stored in video memory.

(Refer Slide Time: 8:48)

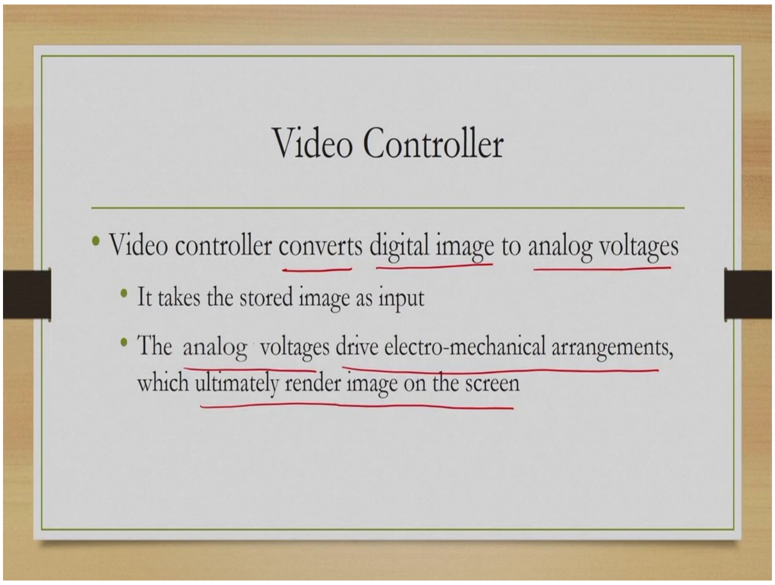
Video Controller



Generic CG System Architecture

Then comes video controller, again let us go back to that generic architecture here, video controller is situated here which takes as input the information stored in video memory and then it does something to display the image on the screen.

(Refer Slide Time: 9:13)

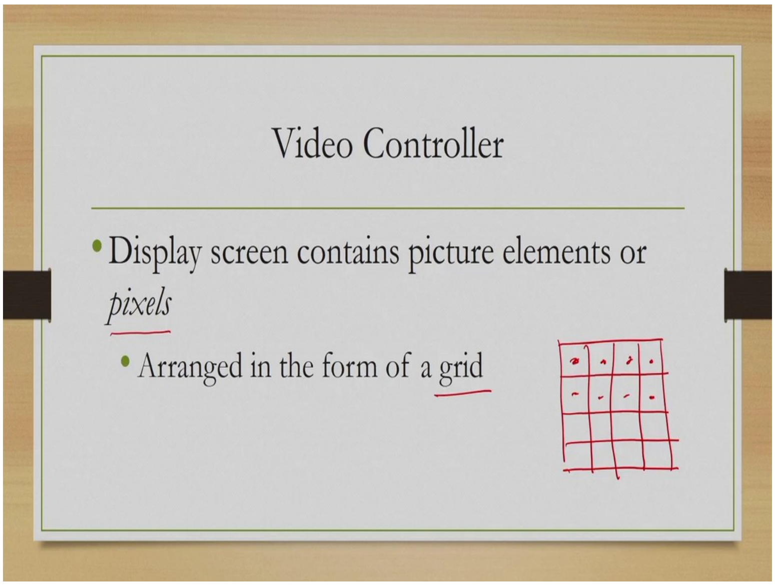


Video Controller

- Video controller converts digital image to analog voltages
  - It takes the stored image as input
  - The analog voltages drive electro-mechanical arrangements, which ultimately render image on the screen

So what it does? It essentially converts digital image that is represented in the form of 0's and 1's to analogue voltages, why? Because the voltages drive electromechanical arrangements which ultimately render image on the screen. So screen essentially is a electro mechanical mechanism and to run this mechanism we require voltage and this voltage is generated by the video controller based on the 0's and 1's stored to represent the image.

(Refer Slide Time: 10:05)

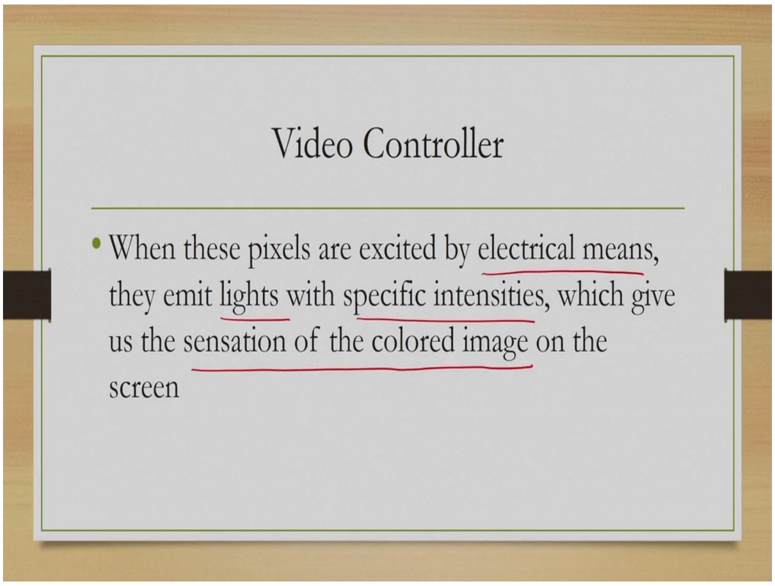


Video Controller

- Display screen contains picture elements or *pixels*
  - Arranged in the form of a grid

In each display screen we have a basic unit of display which is typically called pixels and typically it is arranged in the form of a grid or matrix like if I draw a screen like this so we will have pixel grid something like this, where each cell may represent a pixel essentially a matrix form of pixels.

(Refer Slide Time: 10:40)

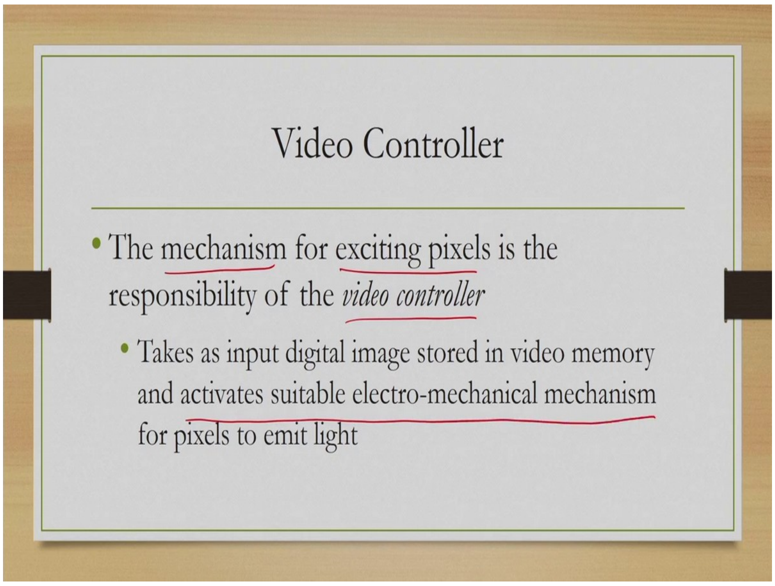


Video Controller

- When these pixels are excited by electrical means, they emit lights with specific intensities, which give us the sensation of the colored image on the screen

Now these pixels are essentially excited by electrical means and when they are excited they meet lights with specific intensities. Now these intensities give us the sensation of coloured images or give us the sensation of colours. So pixels are there on the screen pixels are excited by electrical means, so after excitation and they meet some light with the specified intensity which gives us a sensation of colour. So if some portion of an image is having the red colour, the corresponding pixels will emit light with intensity of red colour so that we get the red colour sensation.

(Refer Slide Time: 11:30)



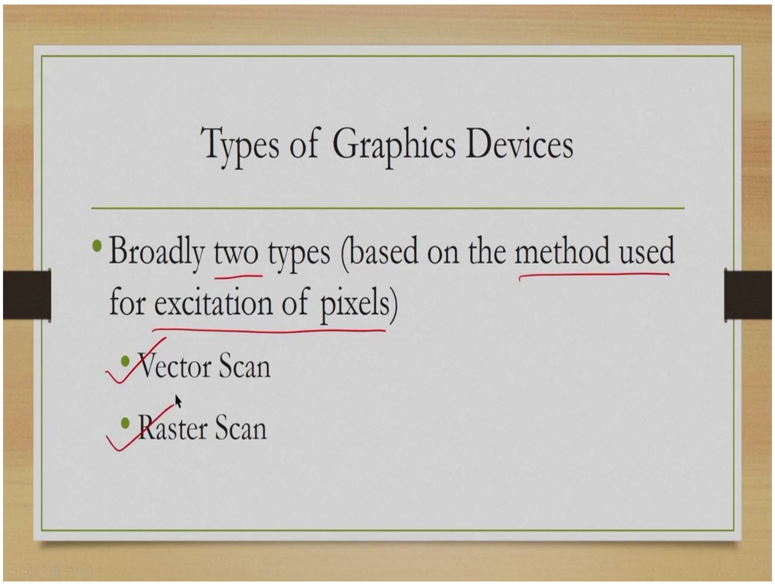
Video Controller

- The mechanism for exciting pixels is the responsibility of the *video controller*
    - Takes as input digital image stored in video memory and activates suitable electro-mechanical mechanism for pixels to emit light

Now the mechanism through which these pixels are excited is the job of the video controller, so video controller essentially is tasked to excite pixels through electrical means by converting the digital input signal 0's and 1's into some analogue voltage signals which in turns activates the suitable electromechanical mechanism which is part of the controller. So that is in a very broad sense how a graphics system look like, so it has three unique components, display controller, memory and video controller.

Display controller is responsible for creating a digital representation of the image to be displayed which is stored in the video memory and then this image information is used to basically excite pixels on the screen, to emit light of specific intensity, to give a sensation of coloured images. So this job of exciting pixels on the screen is done by video controller. Now, in light of this broad description of a graphic system, let us now move to our next topic of types of graphic systems or graphic devices.

(Refer Slide Time: 13:08)

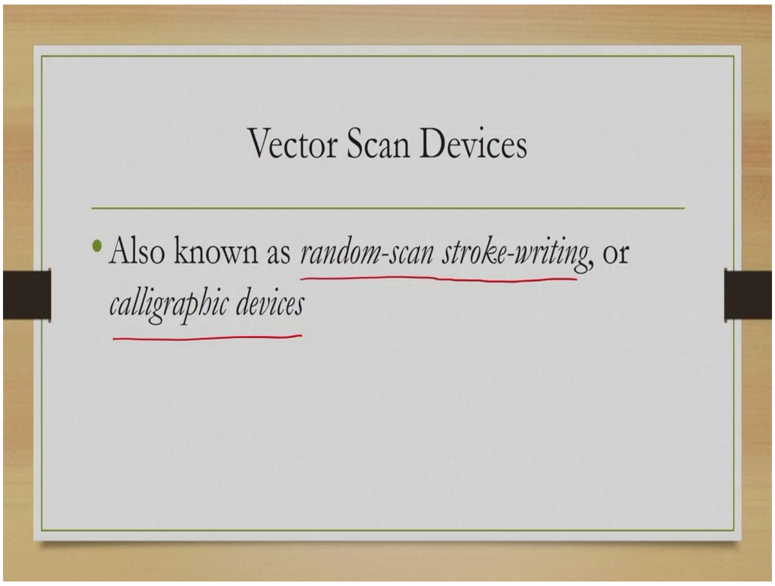


Types of Graphics Devices

- Broadly two types (based on the method used for excitation of pixels)
  - Vector Scan
  - Raster Scan

So there are broadly two types of graphic systems which is based on the method used to excite the pixels. Now what are these two types? One is the vector scan device other one is the raster scan device.

(Refer Slide Time: 13:33)

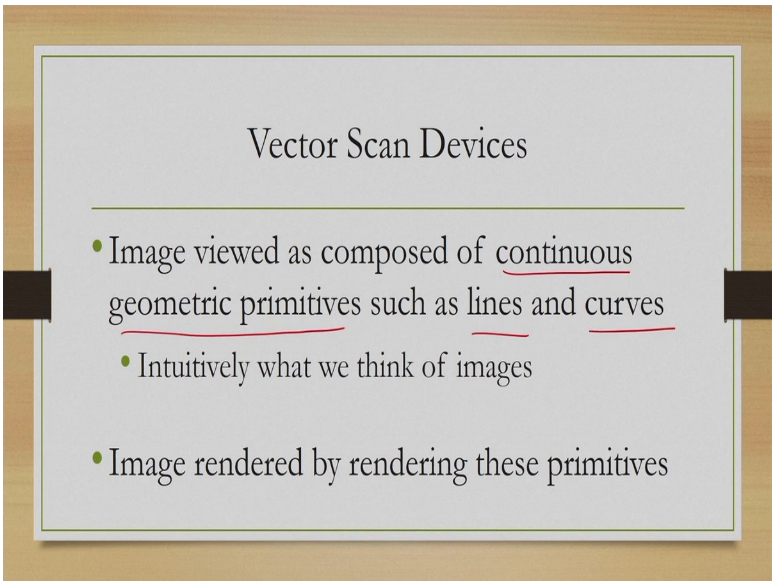


Vector Scan Devices

- Also known as *random-scan stroke-writing*, or *calligraphic devices*

Let us start with the vector scan device. This type of devices or graphic devices are also known as random scan stroke writing or calligraphic devices.

(Refer Slide Time: 13:47)

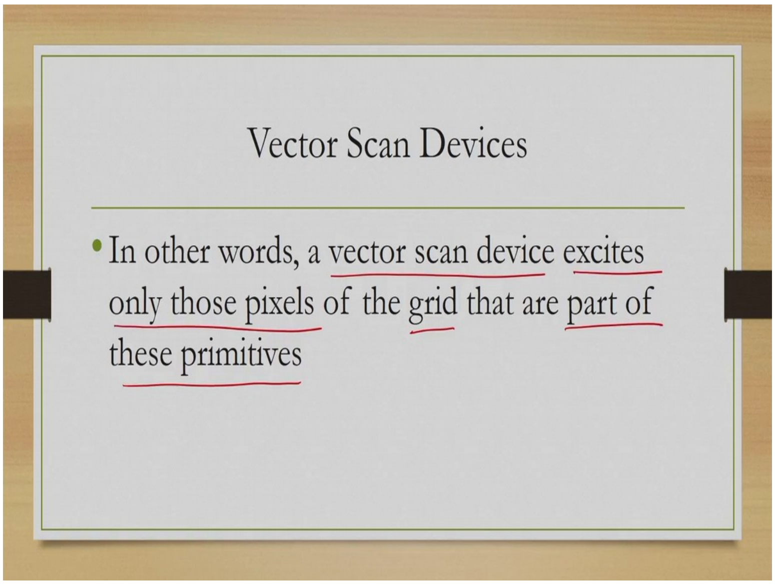


Vector Scan Devices

- Image viewed as composed of continuous geometric primitives such as lines and curves
  - Intuitively what we think of images
- Image rendered by rendering these primitives

In this type of devices when we are talking of an image that image is represented or assume to be represented as a composition of continuous geometric primitives such as lines and curves. So any image is assumed to be composed of lines and curves and when we render or display these images on the screen essentially we render these basic geometric shapes. So we no longer talk about the whole image instead we talked about the component lines and curves that define the image.

(Refer Slide Time: 14:37)

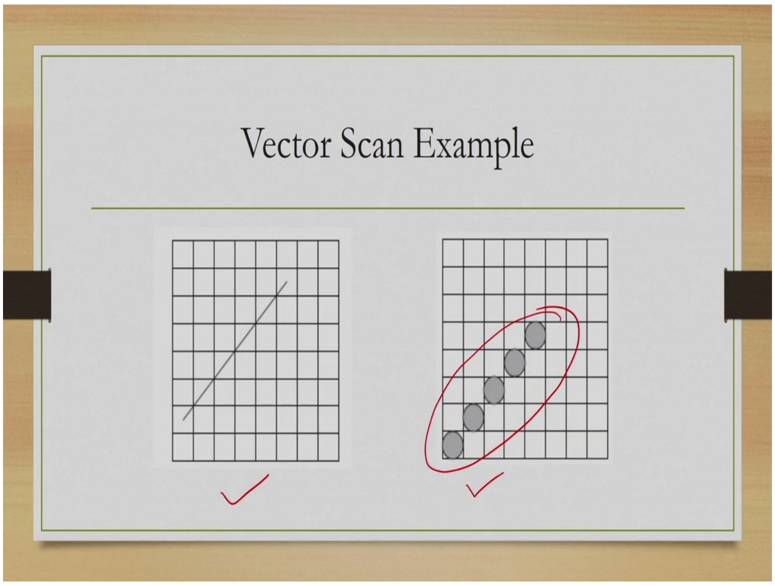


Vector Scan Devices

- In other words, a vector scan device excites only those pixels of the grid that are part of these primitives

In other words, a vector scan device excites only those pixels of the pixel grid that are part of these primitives, so to a vectors can device there is no such concepts as a full image, instead it only knows about constituent, geometric primitives and it excites the pixels that are part of those primitives.

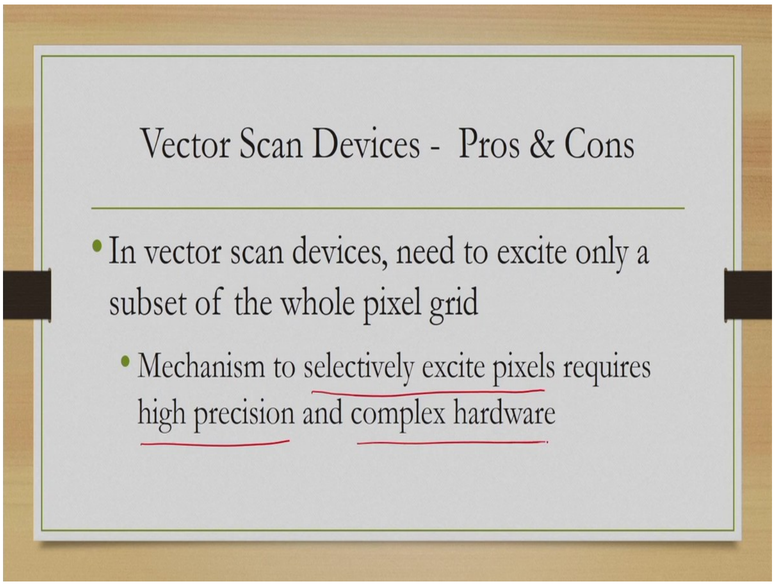
(Refer Slide Time: 15:10)



Vector Scan Example

An example is shown here, consider this line in this left figure and the corresponding pixels is a truncated part of the grid the corresponding pixels are highlighted in this right figure. So to a vector scan device the image is not the line but only the set of pixels. It knows only about these pixels instead of knowing about this line and these pixels are excited to generate the line image and only these pixels are excited other pixels are not excited, this is important that in case of a vector scan device we excite only the pixels that are part of the primitives, other pixels are not touched.

(Refer Slide Time: 16:04)

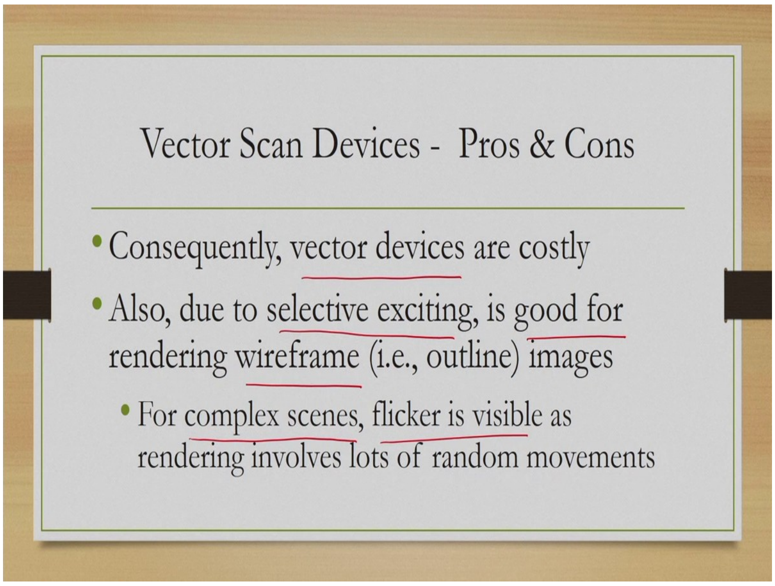


Vector Scan Devices - Pros & Cons

- In vector scan devices, need to excite only a subset of the whole pixel grid
  - Mechanism to selectively excite pixels requires high precision and complex hardware

As a result, what we need to do? We need to selectively excite pixels which is very tough job which requires high precision which is obvious and complex hardware.

(Refer Slide Time: 16:26)

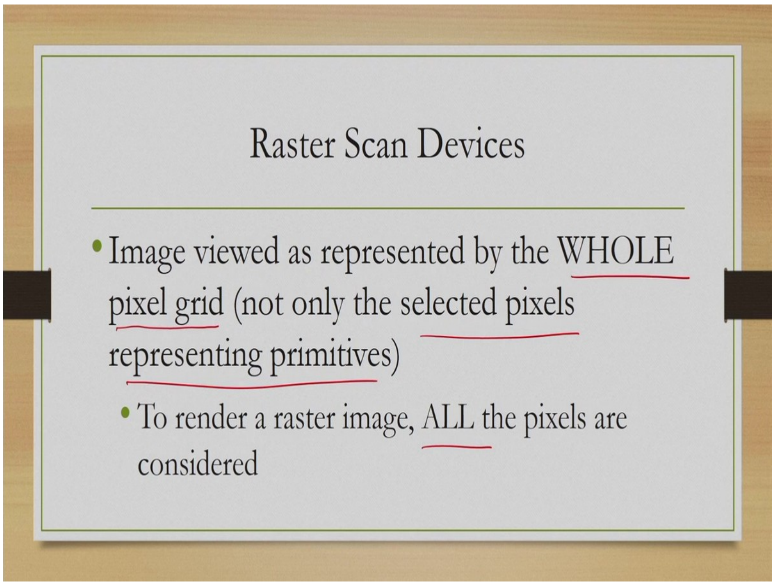


Vector Scan Devices - Pros & Cons

- Consequently, vector devices are costly
- Also, due to selective exciting, is good for rendering wireframe (i.e., outline) images
  - For complex scenes, flicker is visible as rendering involves lots of random movements

Which in turn makes these devices costly because it takes money to develop such hardware with high precision. Also due to the selective exciting such type of devices, vector scan devices are good for rendering wireframes which are basically outlined images. For complex scenes which involves lot of field of areas, flicker is visible because of this mechanism of selective exciting which is not a good thing.

(Refer Slide Time: 17:18)

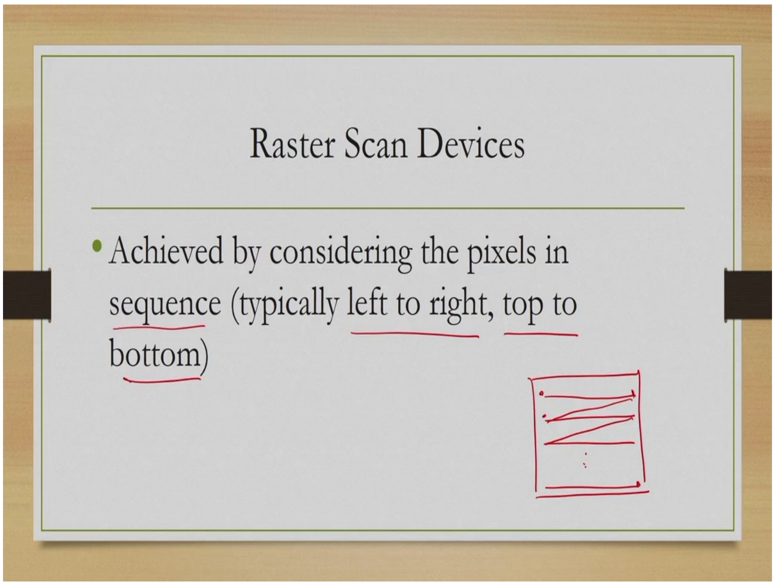


Raster Scan Devices

- Image viewed as represented by the WHOLE pixel grid (not only the selected pixels representing primitives)
  - To render a raster image, ALL the pixels are considered

The other type of graphic devices is raster scan device. Now in raster scan device an images is viewed as represented by the whole pixel grid, so earlier we considered an image to be represented by only a subset of the whole pixel grid but here we are considering the whole pixel grid and not only the selected pixels representing the primitives. So when we render an image on a raster scan device all the pixels are considered, in case of vectors can device be considered only a subset and other pixels were not touched but here all the pixels are considered. And how do we consider that?

(Refer Slide Time: 18:08)

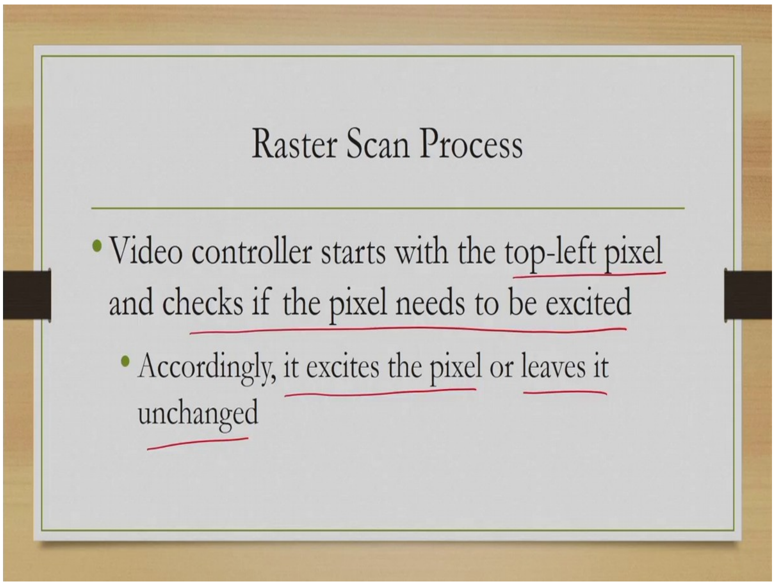


Raster Scan Devices

- Achieved by considering the pixels in sequence (typically left to right, top to bottom)

By considering the pixels in a sequence. What is the typical sequence? It is typically left to right top to bottom. So if we have a grid like this then typically we start from left move towards the right end then go to the next row move towards the right end and continue in this way so kind of this type of movement till we reach the lower right endpoint or end pixel.

(Refer Slide Time: 18:41)

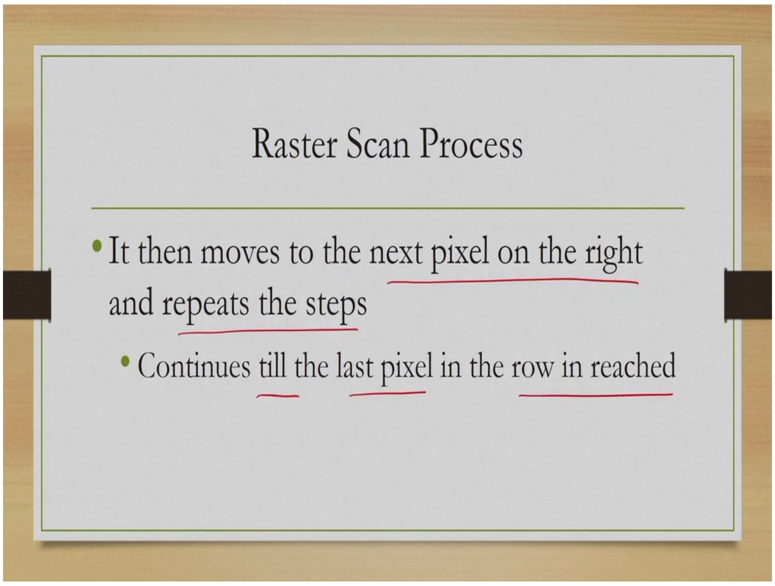


Raster Scan Process

- Video controller starts with the top-left pixel and checks if the pixel needs to be excited
  - Accordingly, it excites the pixel or leaves it unchanged

The same thing is mentioned here, so the controller starts with the top left pixel and checks if the pixel needs to be excited, that information will be stored in the memory. So if it needs to be excited it excites the pixel or leaves it unchanged but mind here that the pixel is considered for excitation and action is taken accordingly.

(Refer Slide Time: 19:16)

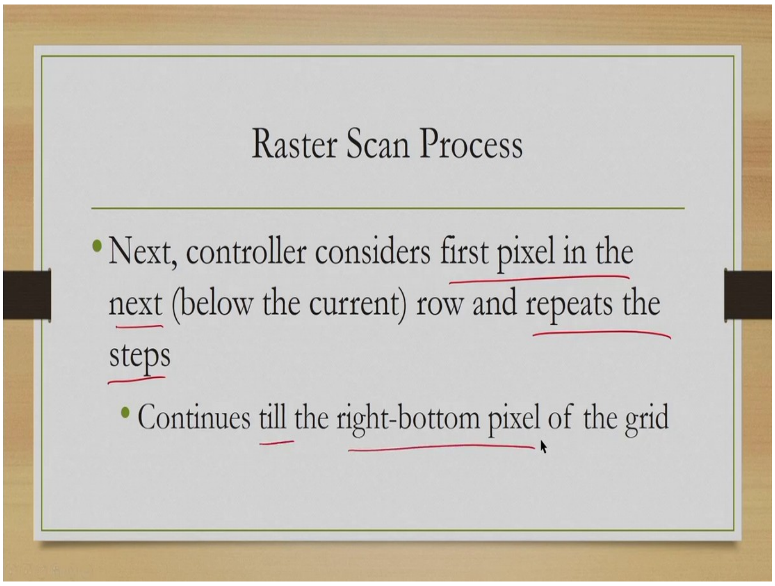


Raster Scan Process

- It then moves to the next pixel on the right and repeats the steps
  - Continues till the last pixel in the row in reached

It then moves to the next pixel on the right and repeat the steps till the last pixel in the row is reached.

(Refer Slide Time: 19:29)

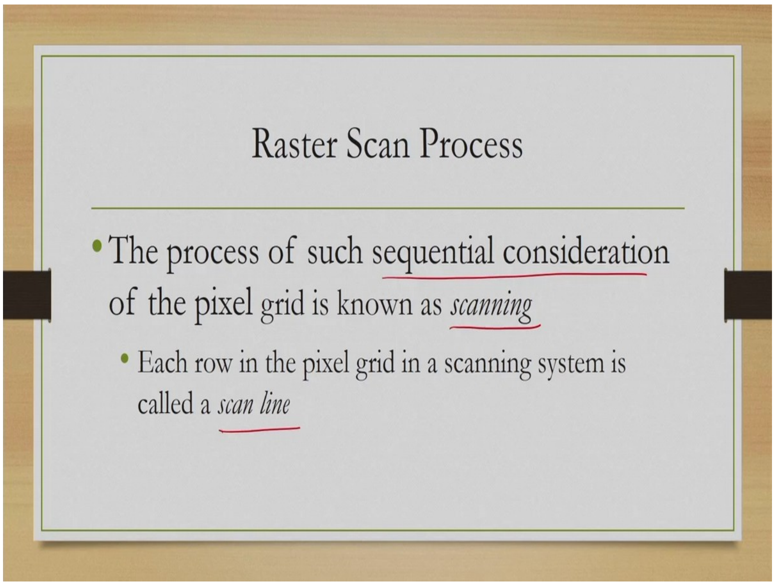


Raster Scan Process

- Next, controller considers first pixel in the next (below the current) row and repeats the steps
  - Continues till the right-bottom pixel of the grid

Then the controller considers the first pixel in the next row and repeats the steps and in this manner it continues till the right bottom pixel of the grid.

(Refer Slide Time: 19:43)

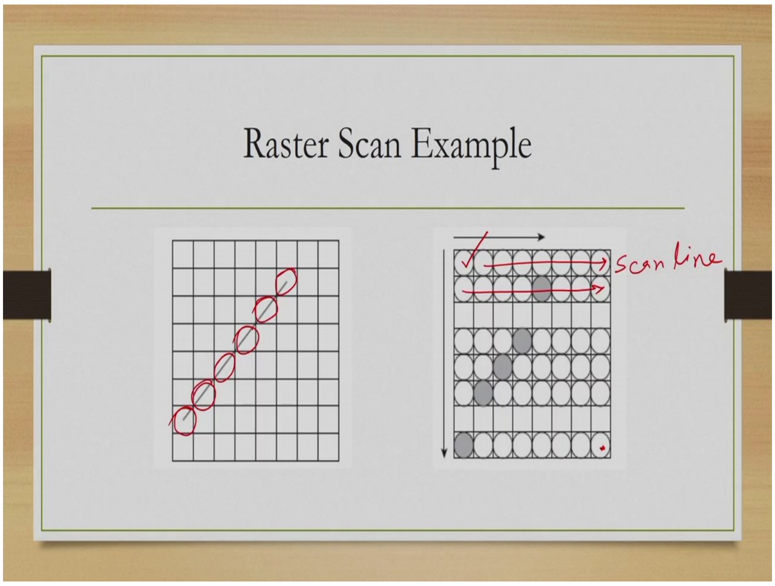


Raster Scan Process

- The process of such sequential consideration of the pixel grid is known as *scanning*
  - Each row in the pixel grid in a scanning system is called a *scan line*

Now this process of consideration of pixels in sequence or such sequential consideration of pixels is known as scanning this is a more generic term used that in raster scan devices, pixel scanning takes place each row of the grid is known as a scan line. So this sequential consideration is called scanning and each row in the pixel grid is known as scanline.

(Refer Slide Time: 20:23)

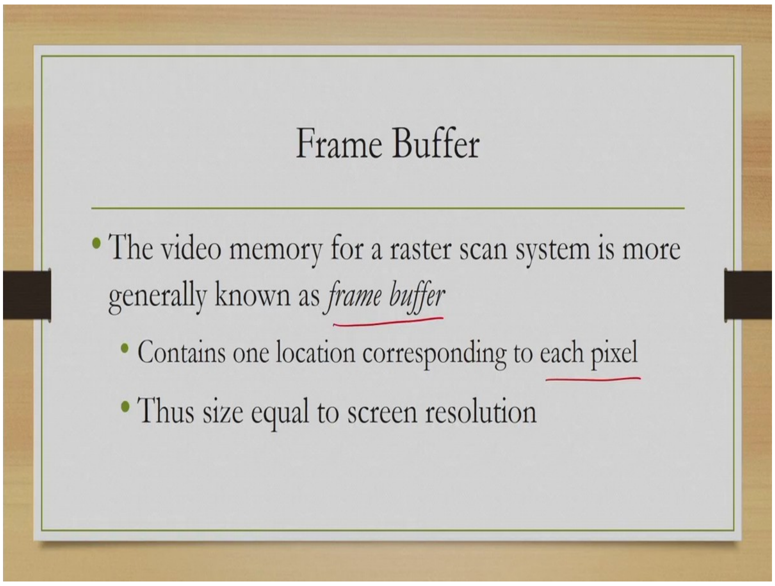


Raster Scan Example

Let us consider the same example here, earlier we considered only the pixels that are part of this line only these pixels, now we are considering all pixels starting from the top left corner moving in this direction then this row so on till this point. So each line is a scan line and as you can see in this figure, right hand figure, the white pixels means they need not be excited.

The system considered the pixel and found that they need not be excited so it move to the next pixel and the filled up circles indicate excited pixels which represents the line so that information was also there in the memory and the video controller found out that these pixels needed to be excited so it excited those pixels, in the process is it considered all pixels in the grid and excited only those which need to be excited.

(Refer Slide Time: 21:42)

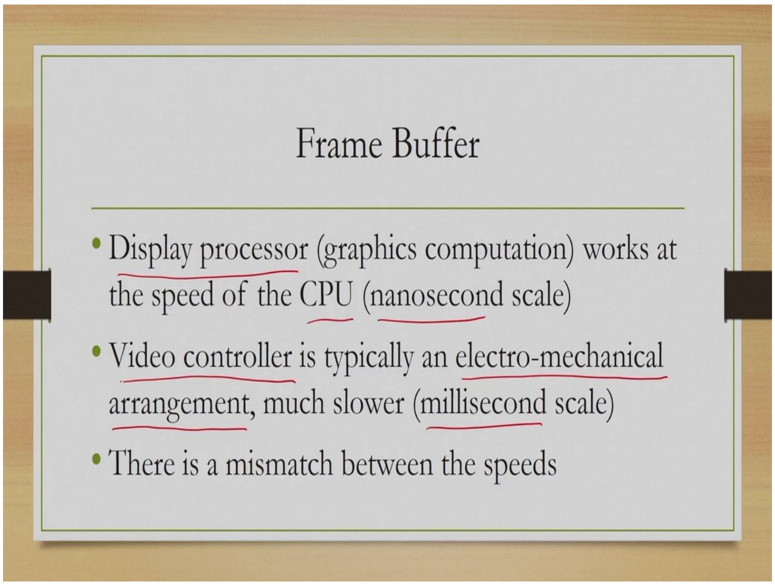


Frame Buffer

- The video memory for a raster scan system is more generally known as *frame buffer*
  - Contains one location corresponding to each pixel
  - Thus size equal to screen resolution

Now the video memory of a raster scan system is more generally known as frame buffer where each location corresponds to each pixel. So the size of a frame buffer is equal to the screen resolution the size of the pixel grid, which is very obvious of course.

(Refer Slide Time: 22:07)



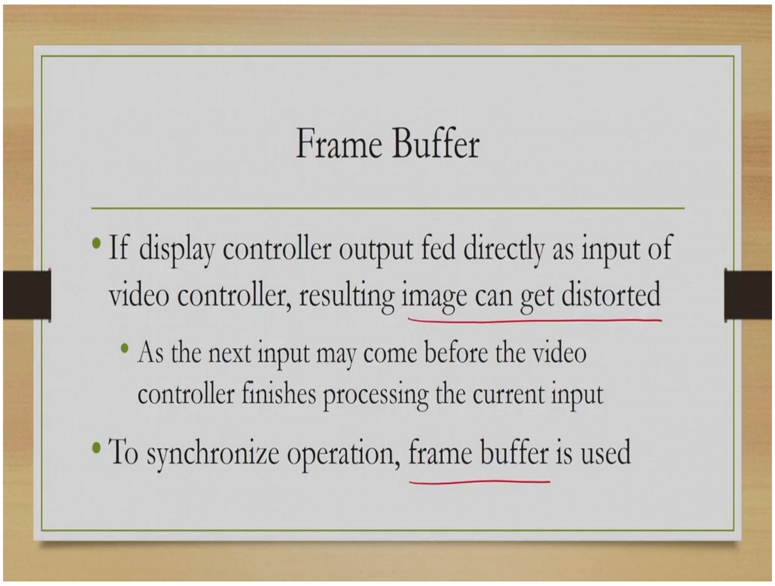
Frame Buffer

- Display processor (graphics computation) works at the speed of the CPU (nanosecond scale)
- Video controller is typically an electro-mechanical arrangement, much slower (millisecond scale)
- There is a mismatch between the speeds

Now there is one interesting fact you should be aware of it, display processors are typically very fast they work at the speed of CPU, that is nanosecond scale so any operation is done at a very less time nanosecond level. On the other hand, video controllers are typically slower, much, much slower compared to display controllers because they involve electromechanical arrangements which takes time to work.

So typical speed ranges in the millisecond level or millisecond scale. Clearly there is a mismatch between the way display processor produces output between the speed at which the display processor can produce output and the speed at which the video controller can take that output as input.

(Refer Slide Time: 23:15)

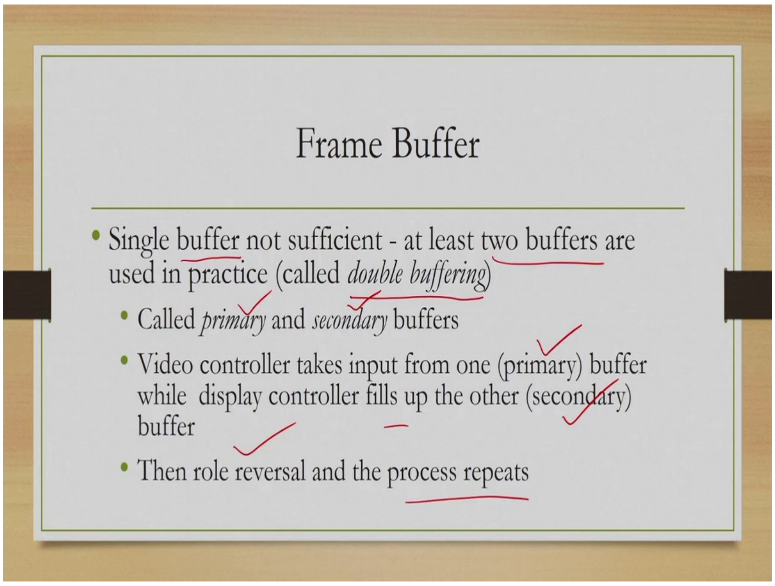


Frame Buffer

- If display controller output fed directly as input of video controller, resulting image can get distorted
  - As the next input may come before the video controller finishes processing the current input
- To synchronize operation, frame buffer is used

Now assume that there is only one video memory or frame buffer, if the display controller outputs are fed directly as input to the video controller through that framebuffer, now the output is being produced very fast but the input is being consumed at a much lower rate so the output may get overwritten before the entire output is taken by the video controller as input which in turn may result in the image getting distorted because before the current input is processed the next input is ready and overwrote the current input. So to address this concern, so we use the concept of frame buffers.

(Refer Slide Time: 24:14)

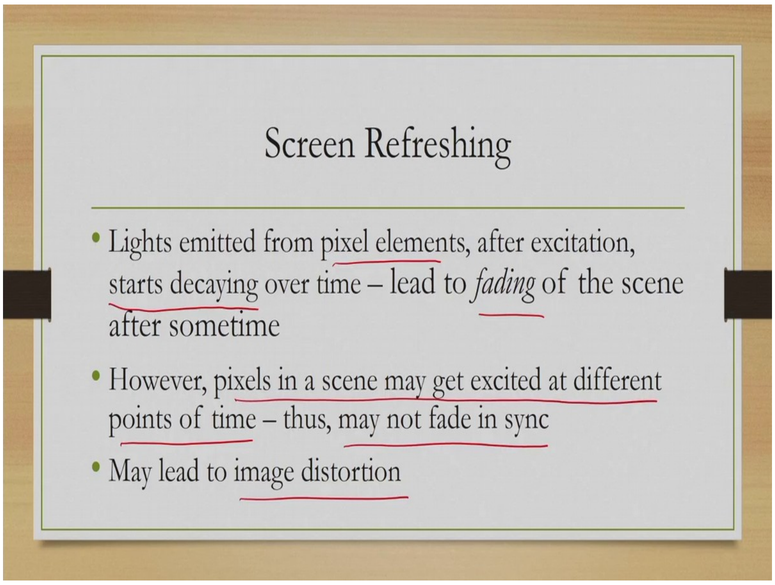


Frame Buffer

- Single buffer not sufficient - at least two buffers are used in practice (called *double buffering*)
  - Called *primary* and *secondary* buffers
  - Video controller takes input from one (primary) buffer while display controller fills up the other (secondary) buffer
  - Then role reversal and the process repeats

Where single buffer is not sufficient and will require at least 2 buffers and if two buffers are used it is called double buffering, of course there are cases with more than 2 buffers. Now in case of double buffering one buffer or one video memory is called primary and the other one is called secondary, so now video controller takes input from one of the buffers typically the primary buffer whereas the display controller fills up the other or the secondary buffer. Now when the video controller finishes reading input from the primary buffer, the primary now become secondary and the secondary becomes primary, so a role reversal takes place and the process repeats. So in this way the problem of overwriting the image information can be avoided.

(Refer Slide Time: 25:17)

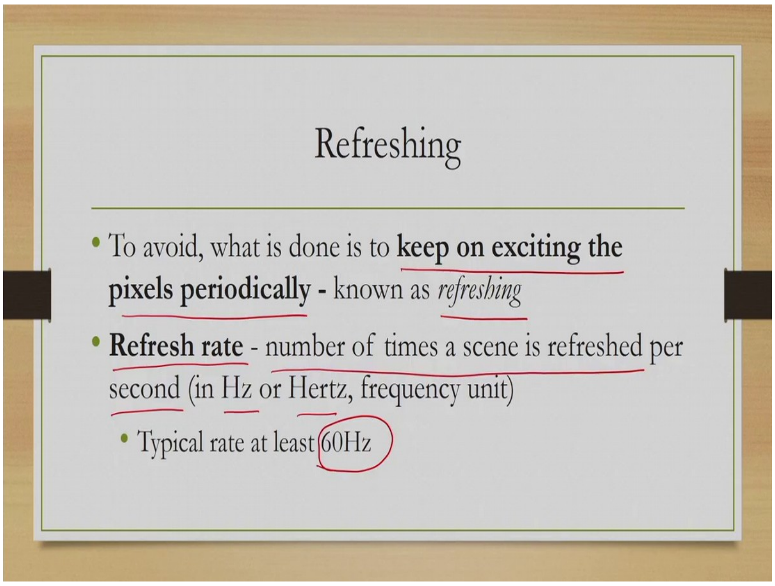


Screen Refreshing

- Lights emitted from pixel elements, after excitation, starts decaying over time – lead to *fading* of the scene after sometime
- However, pixels in a scene may get excited at different points of time – thus, may not fade in sync
- May lead to image distortion

Another interesting fact to note here is called refreshing, now lights emitted from pixel elements which gives us the sensation of colour starts decaying over time. So it is not the case that the intensity of the emitted light remains the same throughout the display session so over time it starts decaying so intensity changes which lead to fading of the scene after sometime. However, pixels in a scene may get excited at different points of time, thus the pixels may not fade in sync. So in an image it is not necessary that every pixels fade in sync so that it is not perceptible to the user so it may lead to image distortion.

(Refer Slide Time: 26:29)

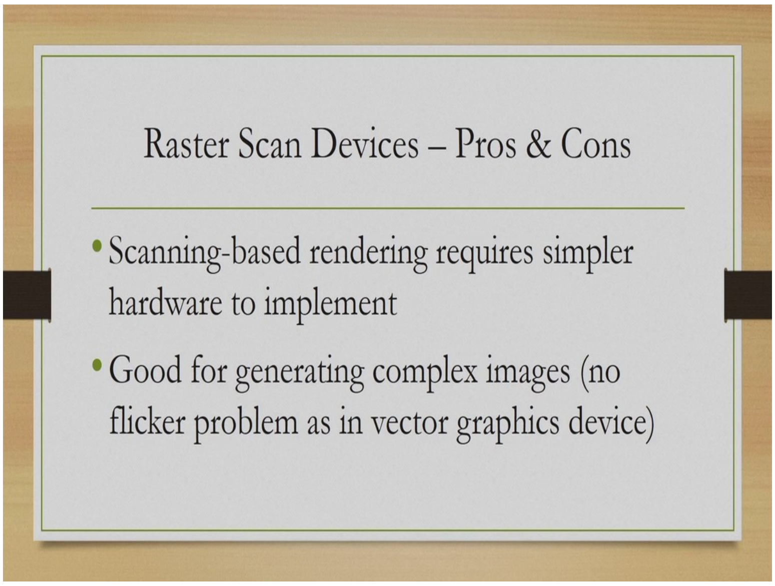


Refreshing

- To avoid, what is done is to **keep on exciting the pixels periodically -** known as *refreshing*
- **Refresh rate** - number of times a scene is refreshed per second (in Hz or Hertz, frequency unit)
  - Typical rate at least 60Hz

You know to avoid that situation, what is done is to keep on exciting the pixels periodically which is known as refreshing. So whatever is the excitation value with that value there is a periodic excitement of the whole pixel grid, so it is not an one time activity. One important consideration here is the refresh rate at which rate we should keep on refreshing the screen so that the changes are not perceptible to the human eye. So the number of times a scene is refreshed per second is known as the refresh rate which is represented in Hz or Hertz, it is typically the frequency unit. And in case of displays that is typically considered to be 60 Hertz or 60 time per second screen should be refreshed.

(Refer Slide Time: 27:33)

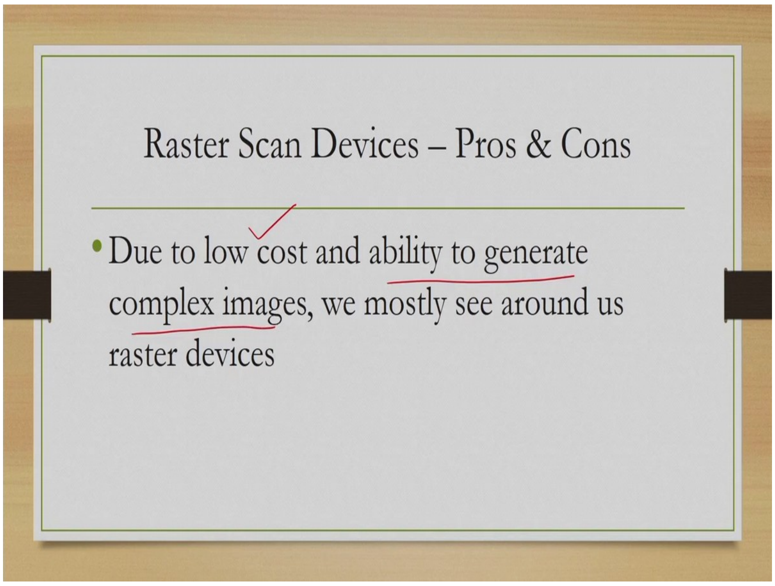


Raster Scan Devices – Pros & Cons

- Scanning-based rendering requires simpler hardware to implement
- Good for generating complex images (no flicker problem as in vector graphics device)

So what are the pros and cons of a raster scan device? Clearly here, since we are not trying to excite selectively, so we do not require a very high precision hardware. Scanning is a very straightforward job so a low precision hardware can do to the job. Also it is good for generating complex images since we are considering all pixels anyway, so it will not lead to flickers unlike in vector scan.

(Refer Slide Time: 28:10)

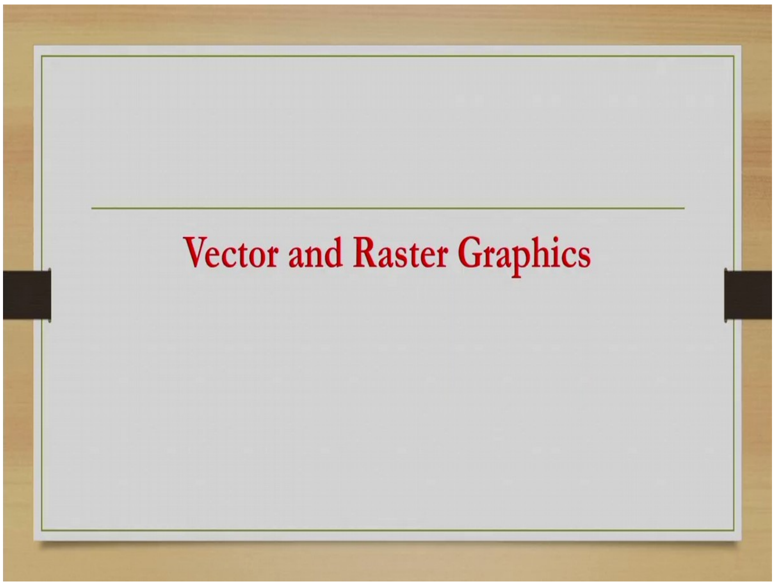


Raster Scan Devices – Pros & Cons

- Due to low cost and ability to generate complex images, we mostly see around us raster devices

Due to these benefits one is low cost the other one is ability to generate complex images most of the displays that we see around us are based on raster graphic concept, so you get to see only or mostly raster graphics devices around us because it is low cost and good at generating complex images.

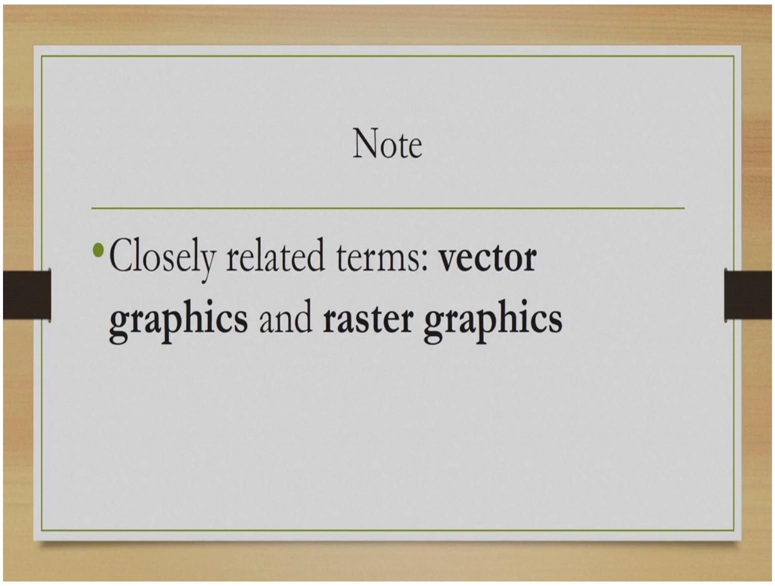
(Refer Slide Time: 28:43)



Vector and Raster Graphics

Now these two are from the point of view of hardware vector scan device and raster scan device, there is a closely related term which probably you may have heard of called vector graphics and raster graphics.

(Refer Slide Time: 28:58)

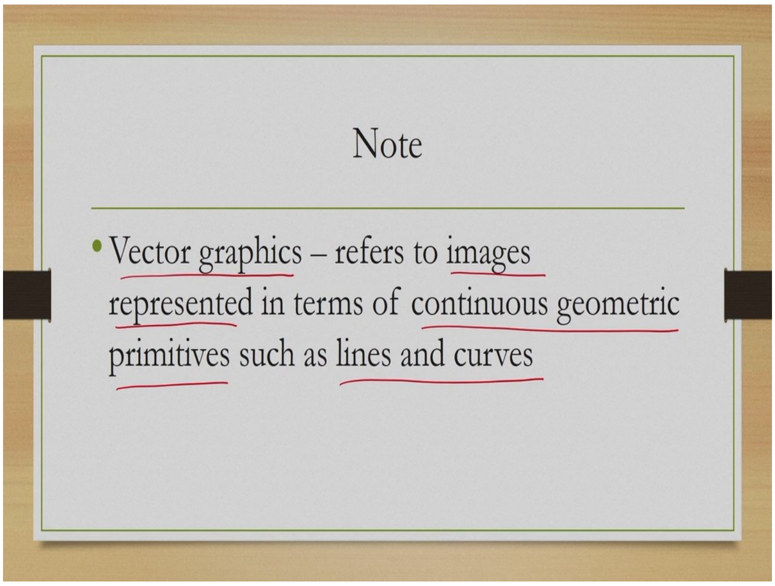


Note

- Closely related terms: **vector graphics** and **raster graphics**

Now these two are not related to any hardware characteristics unlike the previous terms vector scan and raster scan.

(Refer Slide Time: 29:10)

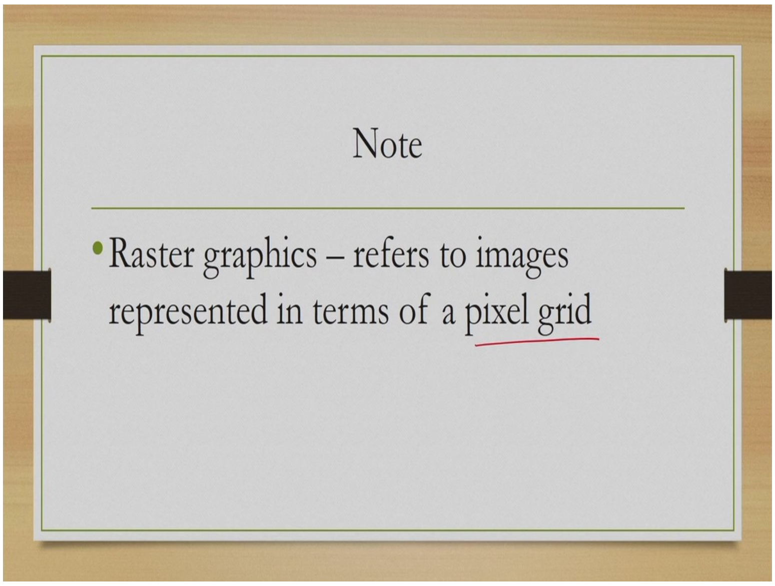


Note

- Vector graphics – refers to images represented in terms of continuous geometric primitives such as lines and curves

In case of vector graphics, what we actually refer to is a where the image is represented, so when we are talking of a vector graphics image we are talking of the representation in terms of continuous geometric primitives such as lines and curves, so if I say that particular image is a vector graphics image, that means I am representing that image in terms of its constituent geometric primitives, lines and curves.

(Refer Slide Time: 29:50)

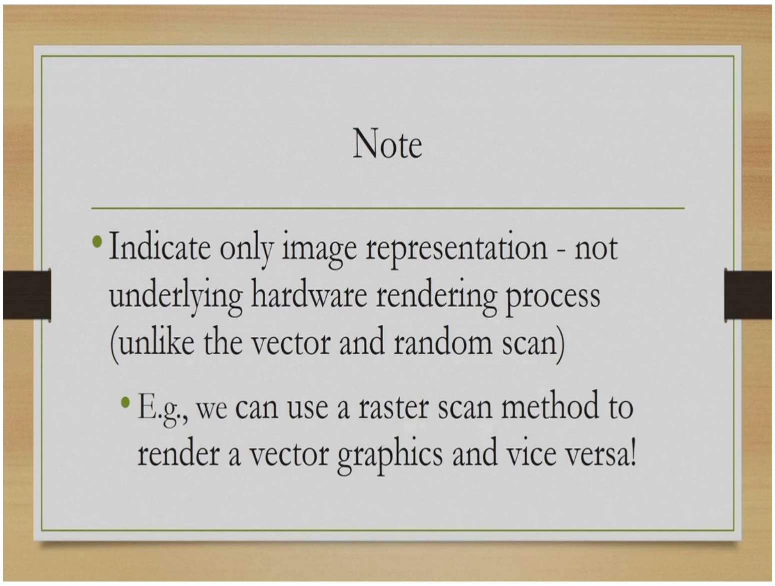


Note

- Raster graphics – refers to images represented in terms of a pixel grid

In case of raster graphics, the representation is different like in raster scan device in case of raster graphics what we refer to is essentially representing the image as the whole pixel grid with the pixels which are supposed to be excited in an on state and others in a off state. So if we are representing an image as a raster graphics image essentially the image is stored in a form of whole pixel grid where some pixels are in the excited or in the on state or at least it is indicated that these pixels should be in the on state.

(Refer Slide Time: 30:48)



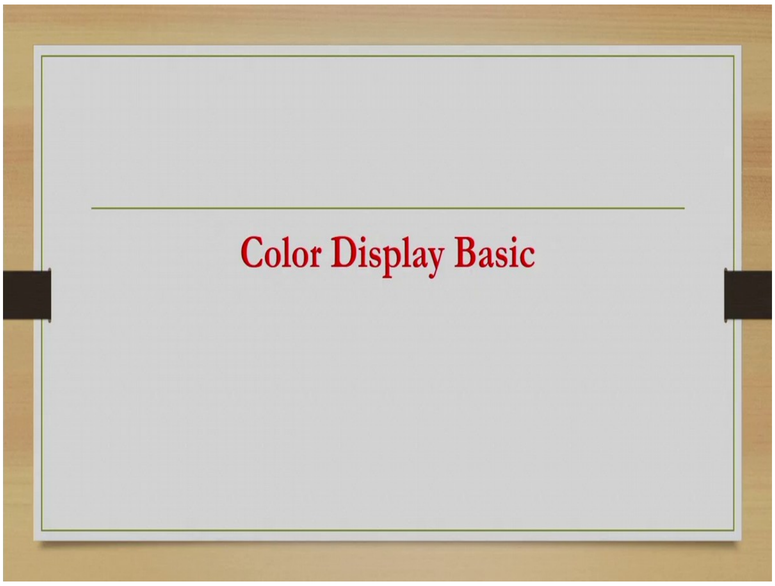
Note

- Indicate only image representation - not underlying hardware rendering process (unlike the vector and random scan)
  - E.g., we can use a raster scan method to render a vector graphics and vice versa!

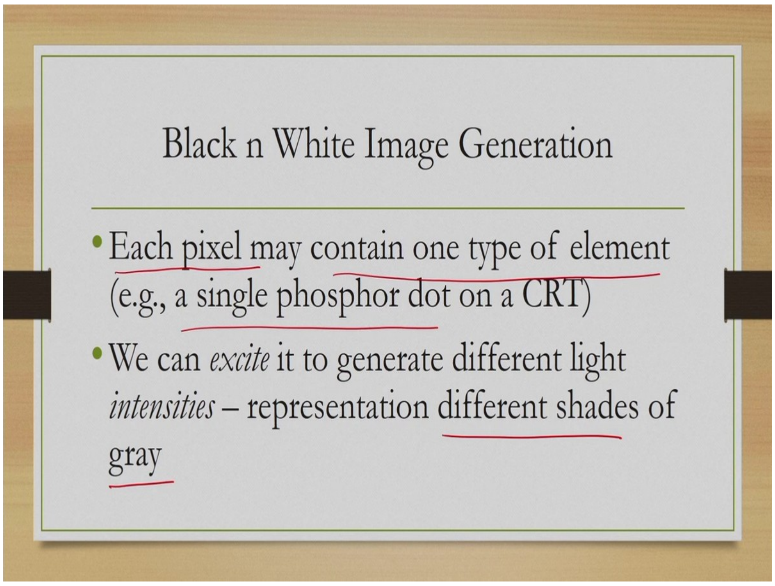
But again it should be noted that vector graphics or raster graphics are terms to indicate the way images are represented they have nothing to do with the underlying hardware. So even if I represent an image in the form of a vector graphics I can still use a raster scan device to display that image and vice versa if I represent an image as a raster graphics I can still use a vector scan device to render it.

So we should be always clear about the distinction between these terms, one term is vector scan device and raster scan device these are related to the way scanning takes place at the hardware level. Other terms are vector graphics and raster graphics these represent the way images are represented internally rather than how they are rendered through actual display hardware.

(Refer Slide Time: 32:00)



Color Display Basic

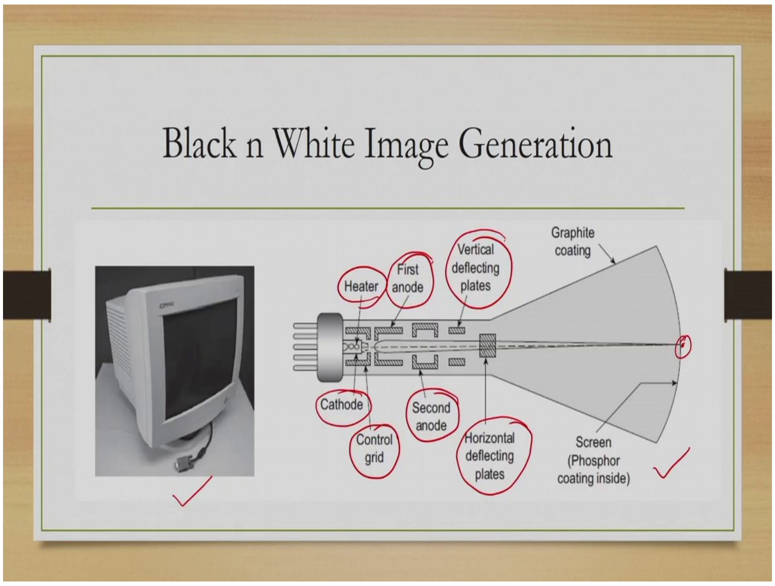


Black n White Image Generation

- Each pixel may contain one type of element (e.g., a single phosphor dot on a CRT)
- We can *excite* it to generate different light *intensities* – representation different shades of gray

Now let us come back to or let us discuss another important topic that is colour display. So far we are assuming that the pixels are monochromatic implicitly we are assuming that but in reality we get to see images that are having colours, so how they work. In a black and white display each pixel may contain one type of element, for example if you are aware of CRT or cathode ray tube displays and their internal mechanism then you may be knowing that each pixel on a CRT display is having a single phosphor dot. Now when we excite it to generate different light intensities, they result in different shades of grey because that is a single phosphor dot.

(Refer Slide Time: 33:05)



Black n White Image Generation

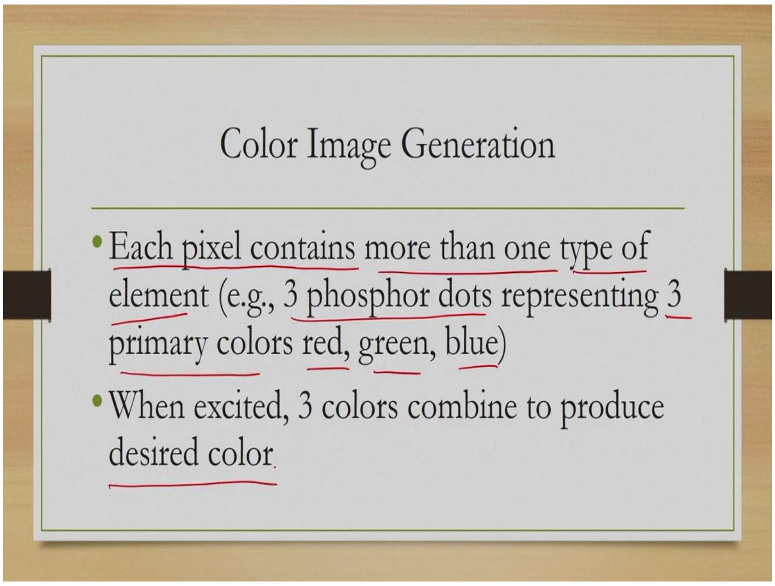
Heater
First anode
Vertical deflecting plates
Graphite coating
Cathode
Control grid
Second anode
Horizontal deflecting plates
Screen (Phosphor coating inside)

Like the illustration shown here this is for CRT or cathode ray tube, of course nowadays it is very rare to see such displays but it is good for pedagogical purpose to demonstrate in terms of a CRT, so left side shows a typical CRT display and on the right side we can see that how it works internally.

So it has a tube within which there are certain arrangements these arrangements together constitute the video controller component of a generic system that we have discussed earlier, so we have cathode, heater, anode arrangements, then a grid to control this electron flow, then deflection plates vertical and horizontal for deflecting the electron flow.

So essentially the arrangement generates a stream of electrons which hits a point on the screen a pixel, after hitting the pixel or the phosphor dot generates intensities which results in different shades of grey, that is in a very brief how CRT's work and in a similar way other displays also work in a similar way not in this exactly the same way.

(Refer Slide Time: 34:44)

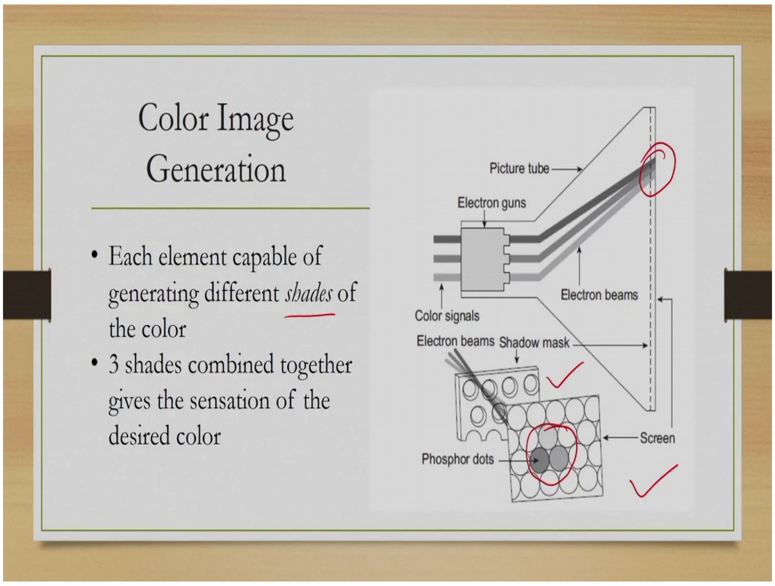


Color Image Generation

- Each pixel contains more than one type of element (e.g., 3 phosphor dots representing 3 primary colors red, green, blue)
- When excited, 3 colors combine to produce desired color

So what happens in case of a colour image? Now in that case each pixel contains more than one type of element, so like for CRT instead of having one phosphor dot we can have three types of phosphor dots representing three primary colours namely red, green and blue. So when excited each of these phosphor dots generates intensities related to this primary colours so the red dot generates red intensities, green dot generate green intensities and blue dot generates blue intensities. When this intensity is combined together, we get the sensation of desired colour.

(Refer Slide Time: 35:44)

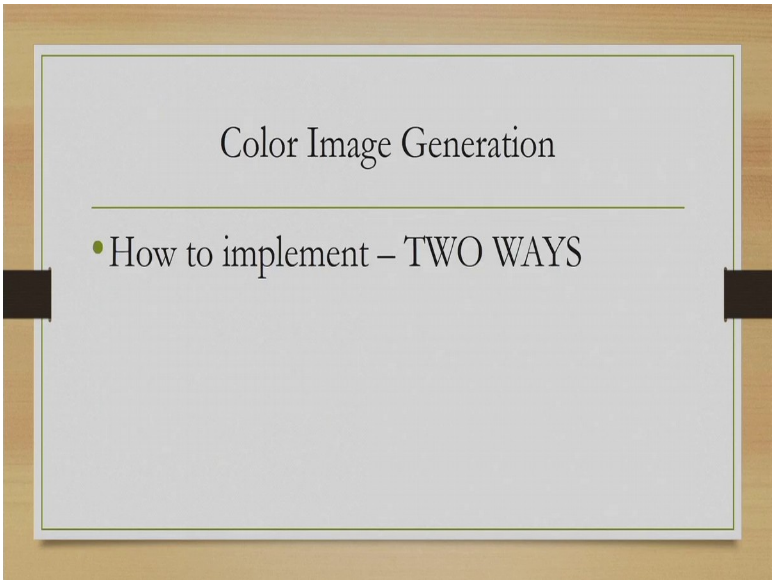


Color Image Generation

- Each element capable of generating different *shades* of the color
- 3 shades combined together gives the sensation of the desired color

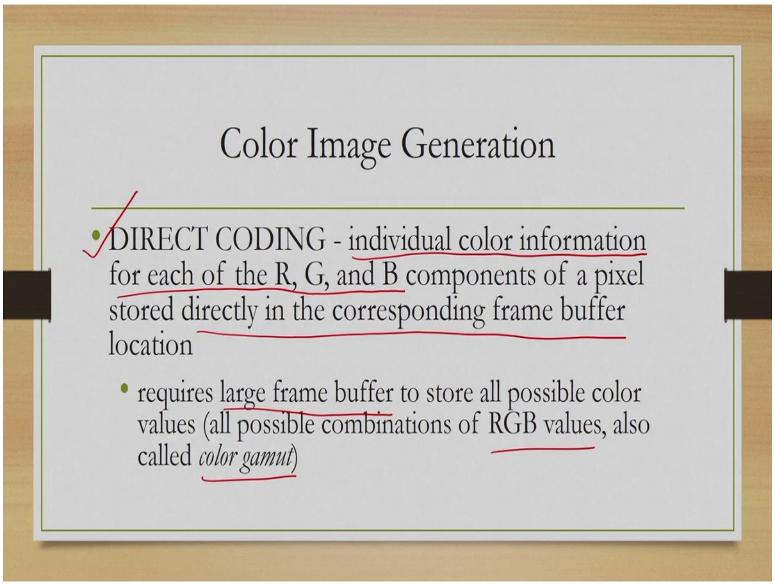
So as I said each element is capable of generating different shades of the colour and when this shades combine they give us the desired sensation of the colour, schematically it looks somewhat like this figure where we have three streams of electron beams hitting the three elements separately some special arrangements are there which are called masks to guide the electron beams to hit specific pixel group representing the three pixels like the three shown here and finally we get the combination of different shades as the desired colour.

(Refer Slide Time: 36:48)



Color Image Generation

- How to implement – TWO WAYS



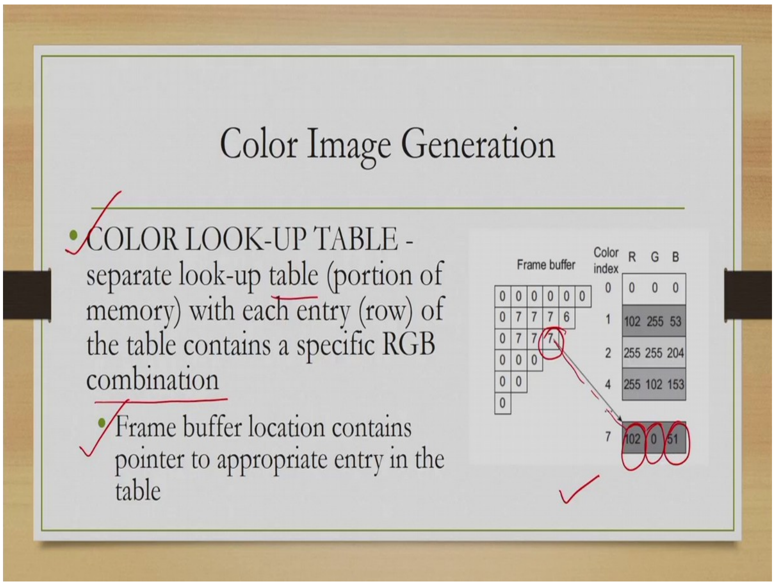
Color Image Generation

- DIRECT CODING - individual color information for each of the R, G, and B components of a pixel stored directly in the corresponding frame buffer location
  - requires large frame buffer to store all possible color values (all possible combinations of RGB values, also called *color gamut*)

Now there are two ways to generate this coloured images. Essentially what we want to do is we want to have some values to guide the exciting of the individual type of elements in a coloured display, so there are two ways to do that, one is direct coding in this case what we do individual colour information for each of the red, green and blue element of a pixel are stored directly in the corresponding frame buffer.

So in the frame buffer itself we are storing the information of what should be the intensities of this individual colours, clearly that requires larger frame buffer compared to black and white frame buffers because now in each location we are storing three values instead of one and this frame buffer should be capable of storing the enter combination of RGB values which is also called the colour gamut. So later on will learn more about this colour gamuts the idea but the point to be noted here is that if we are going for direct coding, then we require a large frame buffer.

(Refer Slide Time: 38:25)



Color Image Generation

- COLOR LOOK-UP TABLE - separate look-up table (portion of memory) with each entry (row) of the table contains a specific RGB combination
  - Frame buffer location contains pointer to appropriate entry in the table

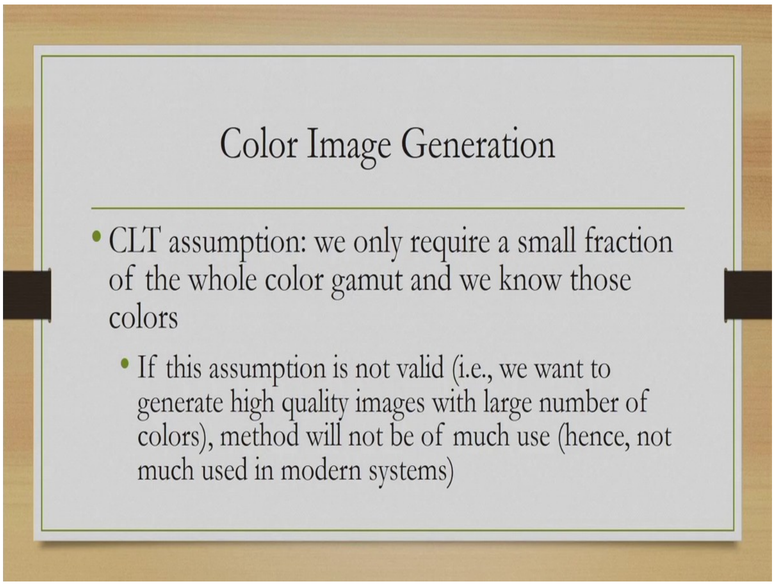
Frame buffer

0	0	0	0	0	0
0	7	7	7	6	
0	7	7	7		
0	0	0			
0	0				
0					

Color index	R	G	B
0	0	0	0
1	102	255	53
2	255	255	204
4	255	102	153
7	102	0	51

Another way is colour lookup tables where we use a separate table, lookup table which is of course a portion of the memory where each entry of the table contains a specific RGB combination and the frame buffer location contains pointer to the appropriate entry in the table. So frame buffer does not store the values directly instead it stores the location to the table which stores the actual values like illustrated in this figure as you can see this is a frame buffer location which stores the pointer to this particular table entry which stores the values of R G and B these are the values to excite the pixels accordingly.

(Refer Slide Time: 39:19)



Color Image Generation

- CLT assumption: we only require a small fraction of the whole color gamut and we know those colors
  - If this assumption is not valid (i.e., we want to generate high quality images with large number of colors), method will not be of much use (hence, not much used in modern systems)

Now if I want the CLT to work or the colour lookup tables scheme to work, then we have to know the subset of the colours that are going to be required in the generation of images. So the table cannot store all possible combinations of R G and B values, it stores only a subset of those combination so essentially a subset of the entire set or the colour gamut and we must know that subset in advance to make this scheme work. If it is not valid of course this method is not going to work but nowadays we do not have any problem with this frame buffer at the size of the frame buffer because memory is cheap.

So nowadays it is almost all graphic systems go for direct coding method but in the earlier generation of graphical systems when memory was a factor to determine the overall cost CLT was much in use. In that period of course the screens were not equipped to display all sorts of complex images and mostly wireframes were the images that were displayed. So that time CLT’s were much more useful but nowadays we do not need to bother about CLT much unless there is some specific application and we can directly go for direct coding method.

(Refer Slide Time: 40:56)

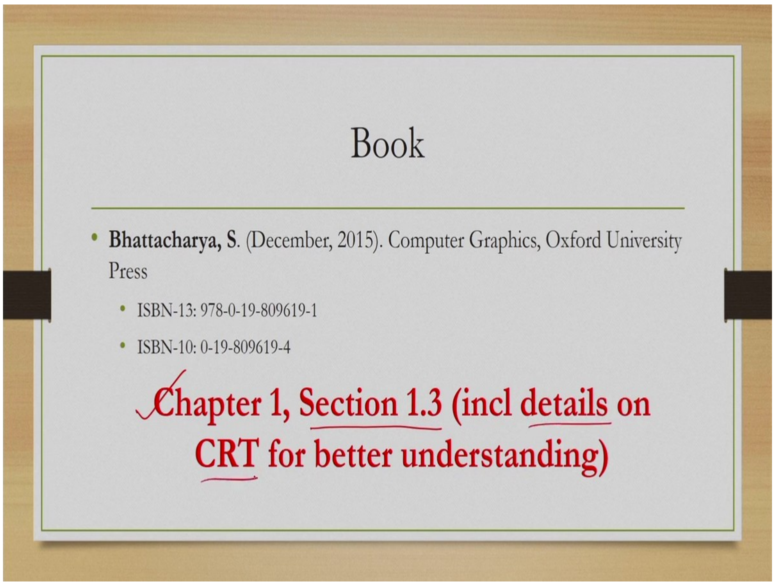
Summary

So let us summarise what we have learnt today, we have got introduced to a basic graphic system which consists of three unique components namely the display controller, the video memory and the video control. Display controller is tasked to generate the image which is stored in video memory and which is used by the video controller to render it on a computer screen.

We also learnt about different types of graphic systems namely the vector scan devices and the raster scan devices in brief and the associated concepts namely vector graphics, raster graphics, refreshing, frame buffers so on. Also we got some idea of how colour images are generated at the hardware level.

So these are basic concepts which will be useful in our subsequent discussions. In the next lecture we will get an introduction to the basic processing that is required to generate a 2D image that is the job of the display controller, now this processing is actually consisting of a set of stages which is collectively known as graphics pipeline, so in the next lecture we will have an introduction to the overall pipeline.

(Refer Slide Time: 42:41)



Book

- **Bhattacharya, S.** (December, 2015). Computer Graphics, Oxford University Press
    - ISBN-13: 978-0-19-809619-1
    - ISBN-10: 0-19-809619-4

Chapter 1, Section 1.3 (incl details on CRT for better understanding)

The topics that I have covered today can be found in this book chapter 1, section 1.3 and you are also advised to go through the details on the CRT or the cathode ray tube display that is mentioned in this section, although I have not covered it here, for better understanding of the topics. So we will meet again in the next lecture, thank you and goodbye.