COMPUTER GRAPHICS
SUBJECT CODE: BCA 303

UNIT-1

What is graphics?

Graphics are visual presentations on some surface such as a wall, canvas, computer screen to inform, illustrate and entertain. For example:- photographs, symbols, drawings, maps etc.

Computer Graphics

Computer Graphics is the use of computers to display and manipulate information in graphical or pictorial form, either on a visual-display unit or via a printer or plotter.

Or

Computer graphics are graphics created by computers and, more generally, the representation and manipulation of pictorial data by a computer.

The term computer graphics includes almost everything on computers that is not text or sound. Today nearly all computers use some graphics and users expect to control their computer through icons and pictures rather than just by typing. The term Computer Graphics has several meanings:

- the representation and manipulation of pictorial data by a computer
- the various technologies used to create and manipulate such pictorial data
- the images also produced

Today computers and computer-generated images touch many aspects of our daily life. Computer imagery is found on television, in newspapers, in weather reports, education, medicine, business, art and during surgical procedures. A well-constructed graph can present complex statistics in a form that is easier to understand and interpret. Such graphs are used to illustrate papers, reports, theses, and other presentation material. A range of tools and facilities are available to enable users to visualize their data, and computer graphics are used in many disciplines.

2D Computer Graphics

2D computer graphics are the computer-based generation of digital images mostly from two-dimensional models, such as 2D geometric models, text, and digital images, and by techniques specific to them. The word may stand for the branch of computer science that comprises such techniques, or for the models themselves. 2D computer graphics started in the 1950s.

2D computer graphics are mainly used in applications that were originally developed upon traditional printing and drawing technologies, such as typography, cartography, technical drawing, advertising, etc.. In those applications, the two-dimensional image is not just a representation of a real-world object, but an independent artifact with added semantic value; two-dimensional models are therefore preferred, because they give more direct control of the image than 3D computer graphics, whose approach is more akin to photography than to typography.

3D Computer Graphics

3D computer graphics in contrast to 2D computer graphics are graphics that use a three-dimensional representation of geometric data that is stored in the computer for the purposes of performing calculations and rendering 2D images. Such images may be for later display or for real-time viewing.3D computer graphics are often referred to as 3D models. However, there are differences.

Computer Animation

Computer animation is the art of creating moving images via the use of computers. It is a subfield of computer graphics and animation. Increasingly it is created by means of 3D computer graphics, though 2D computer graphics are still widely used for stylistic, low bandwidth, and faster real-time rendering needs. Sometimes the target of the animation is the computer itself, but sometimes the target is another medium, such as film. It is also referred to as CGI (Computer-generated imagery or computer-generated imaging), especially when used in films.

To create the illusion of movement, an image is displayed on the computer screen then quickly replaced by a new image that is similar to the previous image, but shifted slightly. This technique is identical to the illusion of movement in television and motion pictures.

Concepts and Principles

Image

In common usage, an image or picture is an artifact, usually two-dimensional, that has a similar appearance to some subject—usually a physical object or a person. Images may be two-dimensional, such as a photograph, screen display, and as well as a three-dimensional, such as a statue. They may be captured by optical devices—such as cameras, mirrors, lenses, telescopes, microscopes, etc. and natural objects and phenomena, such as the human eye or water surfaces.

Digital Image

A digital image is a representation of a two-dimensional image using ones and zeros (binary). Depending on whether or not the image resolution is fixed, it may be of vector or raster type. Without qualifications, the term "digital image" usually refers to raster images.

Pixel

In digital imaging, a pixel is the smallest piece of information in an image.[6] Pixels are normally arranged in a regular 2-dimensional grid, and are often represented using dots or squares. Each pixel is a sample of an original image, where more samples typically provide a more accurate representation of the original. The intensity of each pixel is variable; in color systems, each pixel has typically three or four components such as red, green, and blue, or cyan, magenta, yellow, and black.

Raster

Raster images have a finite set of digital values, called picture elements or pixels. The digital image contains a fixed number of rows and columns of pixels. Pixels are the smallest individual element in an image, holding quantized values that represent the brightness of a given color at any specific point.

Typically, the pixels are stored in computer memory as a raster image or raster map, a two-dimensional array of small integers. These values are often transmitted or stored in a compressed form.

Difference between text and graphics

Text(is what is written)	Graphics(images and pictures)
1.It is sequence of predefined symbols	1. Nothing is predefined.
2.Reading text is impossible without knowing its knowledge	2.Accessible for illiterate
3.Text requires careful deciphering (interpretation) of letters and words	3.Image is sufficient to grasp its idea
4.Text affect parts of mind but to much less degree as compare to graphics	4.images affect parts of mind to much higher degree as compare to text

Categories of computer graphics:-

Computer graphics can be broadly divided into 2 categories:-

Non- Interactive CG: - In non interactive computer graphics otherwise known as passive computer graphics, the observer has no control over the image. Familiar examples of this type of computer graphics include the titles shown on TV

Interactive CG: - Interactive Computer Graphics involves a two way communication between computer and user. Here the observer is given some control over the image by providing him with an input device for example the video game controller. This helps him to signal his request to the computer. The computer on receiving signals from the input device can modify the displayed picture appropriately. To the user it appears that the picture is changing instantaneously in response to his commands. He can give a series of commands, each one generating a graphical response from the computer. In this way he maintains a conversation, or dialogue, with the computer.

Interactive computer graphics affects our lives in a number of indirect ways. For example, it helps to train the pilots of our airplanes. We can create a flight simulator which may help the pilots to get trained not in a real aircraft but on the grounds at the control of the flight simulator.

The flight simulator is a mock up of an aircraft flight deck, containing all the usual controls and surrounded by screens on which we have the projected computer generated views of the terrain visible on takeoff and landing. Flight simulators have many advantages over the real aircrafts for training purposes, including fuel savings, safety, and the ability to familiarize the trainee with a large number of the world's airports.

Advantages of Computer Graphics

- Today, high quality graphics displays of personal computer provide one of the most natural means of communicating with a computer.
- It provides tools for producing pictures not only of concrete, "real world" objects but also of abstract, synthetic objects, such as mathematical surfaces in 4D and of data that have no inherent geometry, such as survey results.
- It has an ability to show moving pictures, and thus it is possible to produce animations with computer graphics.
- With computer graphics user can also control the animation by adjusting the seed, the portion of the total scene in view, the geometric relationship of the objects in the scene to one another, the amount of detail shown and so on.
- The computer graphics provides tool called motion graphics. With this tool user can move and tumble objects with respect to a stationary observer, or he can make objects stationary and the viewer moving around them. A typical example is walk throughs made by builder to show flat interior and building surroundings. In many case it is also possible to move both objects and viewers.
- The computer graphics also provides facility called update dynamics. With update dynamics it is possible to change the shape, colour or other properties of objects being viewed.
- With the recent development of Digital Signal Processing(DSP) and audio synthesis chip the interactive graphics now provide audio feedback along with the graphical feedbacks to make the simulated environment even more realistic.

Applications of Computer Graphics

Computers have become powerful tool for the rapid and economical production of pictures. Computer graphics are used in several areas such as

- Science
- Engineering

- Medicine
- Business
- Industry
- Government
- Art
- Entertainment
- Advertising
- Education and training

CAD (Computer Aided Designing):- A major use of computer graphics is in design process particularly for engineering and architectural systems, but almost all products are now compute designed. Computer aided design referred as CAD methods are now routinely used in the design of buildings, automobiles, aircraft, space craft and textiles etc.

- Useful for testing performance of a vehicle.
- System in virtual reality environments.
- Realistic renderings of design process.
- Realistic 3D renderings of building designs.

Name of some CAD tools:- Advance Design by GRAITEC, AllyCAD by Knowledge Bave, ArchiCAD by Graphisoft

Presentation Graphics:- Presentation graphics used to produce illustration for reports or generate 35mm slides etc., These graphics are commonly used to summarize and economic data for research reports, managerial reports, consumer information bulletins etc. for eg:- bar charts, line graphs and pie charts.

Name of Some Presentation S/w:- Database Reporting S/w-Zeho Creator, MATLAB report generator

Computer art: - Computer graphics methods are widely used in both fine art and commercial art applications. Artists uses variety of computer methods including special purpose hardware, paint brush programs etc.

Special Purpose S/w- Lumena, Pixel Paint, Supe-Paint.

Entertainment: - Computer graphics methods are now commonly used in making motion pictures, music videos and television shows. Sometimes graphics are displayed by themselves i.e., computer animation and sometimes graphics objects are combined with the actors and live scenes.

S/w Example: - Power point, movie maker

Education and training: - Computer generated models of physical, financial and economic systems are often used as educational aids. For some training applications special system are designed. For eg:- Flight simulators, automobile simulators etc.

What is image processing?

It is a technique to modify or interpret existing pictures, such as photographs. Two principal applications of image processing are:

1. Improving picture quality
2. Machine perception of visual information as used in robotics

Working of image processing: To apply image-processing methods, we first digitize a photograph or other picture into an image file. Then digital methods can be applied to rearrange picture parts, to enhance color separations, or to improve the quality of shading. An example of the application of image-processing methods is to enhance the quality of a picture. These techniques are used extensively in commercial art applications that involve the retouching and rearranging of sections of photographs and other artwork. Similar methods are used to analyze satellite photos of the earth and photos of galaxies.

Graphical User Interface

A major component of a graphical interface is a window manager that allows a user to display multiple-window areas. Each window can contain a different process that can contain graphical or non graphical displays. To make a particular window active, we simply click in that window using an interactive pointing device. Interfaces also display menus and icons for fast selection of processing options or parameter values.

An icon is a graphical symbol that is designed to look like the processing option it represents. The advantages of icons are that they take up less screen space than corresponding textual descriptions and they can be understood more quickly if well designed. Menus contain lists of textual descriptions and icons.

Origin of CG

Years of research and development were made to achieve the goals in the field of computer graphics. In 1950 the first computer driven display was used to generate only simple pictures. This display made use of a cathode ray tube similar to the one used in television sets. During 1950's interactive computer graphics made little progress because the computers of that period were so unsuited to interactive use. These computers were used to perform only lengthy calculations.

The single vent that did the most to promote interactive computer graphics as an important new field was the publication in 1962 of a brilliant thesis by Ivan E. Sutherland. His thesis, entitled 'Sketchpad: A Man- Machine Graphical Communication System proved to many readers that interactive computer graphics was a viable, useful, and exciting field of research. By the mid - 1960's large computer graphics research projects were under taken at MIT, Bell Telephone Labs and General Motors. Thus the golden age of computer graphics began. In 1970's thee researches began to bear fruit.

Important Terminologies Related to CRT (Cathode Ray Tube)

Persistence: - Time taken by the emitted light from the screen to decay to 1/10th of its original intensity is called persistence. Lower persistence phosphors require higher refresh rates to maintain a picture on the screen without flicker. A phosphor with low persistence is useful for animation; a high-persistence phosphor is useful for displaying highly complex, static pictures.

Resolution: - Number of pixels or points that can be displayed on the screen without overlap is called resolution. It can also be defined as number of pixels per square inch. The resolution of a CRT is depend on

- type of phosphor
- intensity to be displayed
- focusing and deflection system

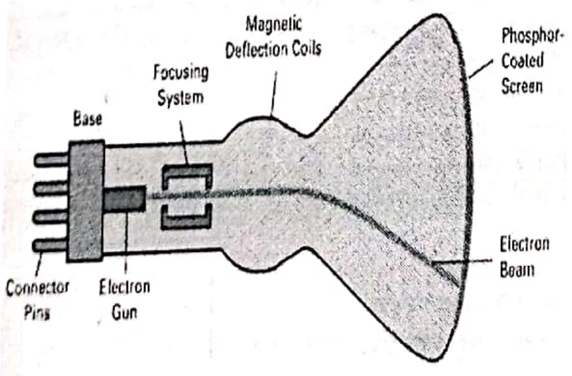
Aspect Ratio:-It is ratio of horizontal to vertical points necessary to produce equal length lines in both directions. Example: An aspect ratio of 3/4 means that a vertical line plotted with three points has same length as horizontal line plotted with four points.

Refresh Rate: - Number of times a screen is redrawn in one second. It is measured in hertz. Eg:- If refresh rate is 85 hertz that means screen is redrawn 85 times in 1 second.

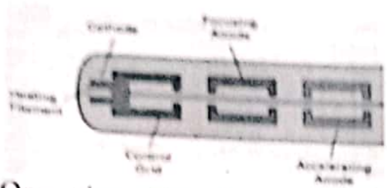
Addressability: - is a measure of the spacing between the centers of vertical and horizontal lines. The picture on a screen consists of intensified points. The smallest addressable point on the screen is called pixel or picture element. In graphics mode there are 800x600

CRT (Cathode Ray Tube)

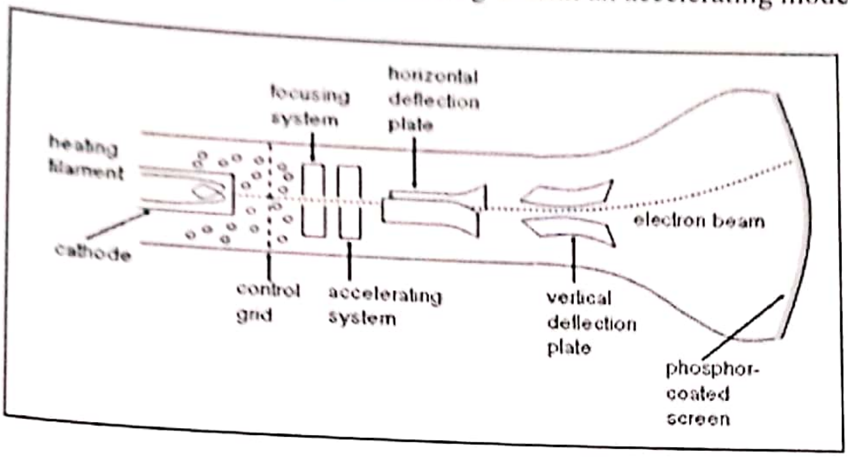
Refresh CRT:-A beam of electrons (cathode rays), emitted by an electron gun, passes through focusing and deflection systems that direct the beam towards specified position on the phosphor-coated screen. The phosphor then emits a small spot of light at each position contacted by the electron beam. Because the light emitted by the phosphor fades very rapidly, some method is needed for maintaining the screen picture. One way to keep the phosphor glowing is to redraw the picture repeatedly by quickly directing the electron beam back over the same points. This type of display is called a refresh CRT.



The basic operation of CRT



Operation of an electron gun with an accelerating mode



Electrostatic Deflection of the electron beam in CRT

Electron Gun: - The primary components of an electron gun in a CRT are the heated metal cathode and a control grid. The cathod is heated by an electric current passed through a coil of wire called the filament. This causes electrons to be boiled off the hot cathode surface. In the vacuum inside the CRT envelope, negatively charged electrons are then accelerated toward the phosphor coating by a high positive voltage. The accelerating voltage can be generated with a positively charged metal coating on the in side of the CRT envelope near the phosphor screen, or an accelerating anode can be used. Sometimes the electron gun is built to contain the accelerating anode and focusing system within the same unit.

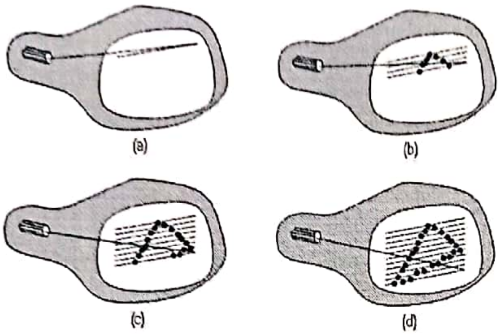
Focusing System:-The focusing system is used to create a clear picture by focusing the electrons into a narrow beam. Otherwise, electrons would repel each other and beam would spread out as it reaches the screen. Focusing is accomplished with either electric or magnetic fields.

Deflection System:-Deflection of the electron beam can be controlled by either electric fields or magnetic fields. In case of magnetic field, two pairs of coils are used, one for horizontal deflection and other for vertical deflection. In case of electric field, two pairs of parallel plates are used, one for horizontal deflection and second for vertical deflection as shown in figure above.

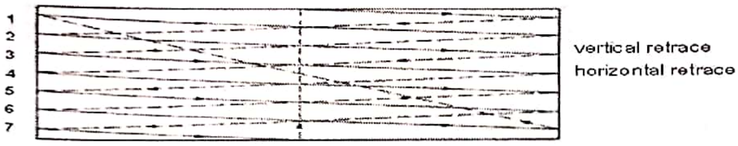
CRT Screen:-The inside of the large end of a CRT is coated with a fluorescent material that gives off light when struck by electrons. When the electrons in the beam is collides with phosphor coating screen, they stopped and their kinetic energy is absorbed by the phosphor. Then a part of beam energy is converted into heat energy and the remainder part causes the electrons in the phosphor atom to move up to higher energy levels. After a short time the excited electrons come back to their ground state. During this period, we see a glowing spot that quickly fades after all excited electrons are returned to their ground state.

Raster Scan Display

It is the most common type of graphics monitor based on television technology. In a raster scan system, the electron beam is swept across the screen, one row at a time from top to bottom. When electron beam moves across each row the beam intensity is turned ON and OFF to create a pattern of illuminated spots. Picture definition is stored in a memory called frame buffer which holds the set of intensity values, which are then retrieved from the frame buffer and pointed on the screen one row at a time as shown in figure below:



At the end of each line the beam must be turned off and redirect to the left hand side of the CRT, this is called Horizontal Retrace. At the end of each frame, the electron beam return to top left corner of the screen to begin the next frame called Vertical Retrace as shown in figure below:



Advantages

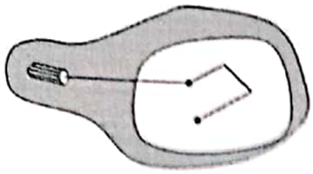
- produce realistic images
- also produced different colors
- and shadows scenes.

Disadvantages

- low resolution
- expensive
- electron beam directed to whole screen

Random Scan Display

In Random Scan System, an electron beam is directed to only those parts of the screen where a picture is to be drawn. The picture is drawn one line at a time, so also called vector displays or stroke writing displays. After drawing the picture the system cycles back to the first line and design all the lines of the picture 30 to 60 time each second. Picture definition is now stored as a set of line- drawing instructions in an area of memory called as display file(display list, display program).



Advantages

- Produced smooth line drawings
- High resolution

Disadvantages

- Designed only for line drawing applications.
- Can't display realistic images.

Difference between Raster Scan and Random Scan Display

Base of Difference	Raster Scan System	Random Scan System
Electron Beam	The electron beam is swept across the screen, one row at a time, from top to bottom.	The electron beam is directed only to the parts of screen where a picture is to be drawn.
Resolution	Its resolution is poor because raster system in contrast produces zig-zag lines that are plotted as discrete point sets.	Its resolution is good because this system produces smooth lines drawings because CRT beam directly follows the line path.
Picture Definition	Picture definition is stored as a set of intensity values for all screen points, called pixels in a refresh buffer area.	Picture definition is stored as a set of line drawing instructions in a display file.
Realistic Display	The capability of this system to store intensity values for pixel makes it well suited for the realistic display of scenes contain shadow and color pattern.	These systems are designed for line-drawing and can't display realistic shaded scenes.
Draw an Image	Screen points/pixels are used to draw an image.	Mathematical functions are used to draw an image.

Color CRT Monitors

A CRT monitor displays color pictures by using a combination of phosphors that emit different-colored light. By combining the emitted light from the different phosphors, a range of colors can be generated. The two basic techniques for producing color displays with a CRT are the beam-penetration method and the shadow-mask method.

Beam Penetration Method

Random scan monitors use the beam penetration method for displaying color picture. In this, the inside of CRT screen is coated two layers of phosphor namely red and green. A beam of slow electrons excites only the outer red layer, while a beam of fast electrons penetrates red layer and excites the inner green layer. At intermediate beam speeds, combination of red and green light are emitted to show two additional colors- orange and yellow.

Advantages

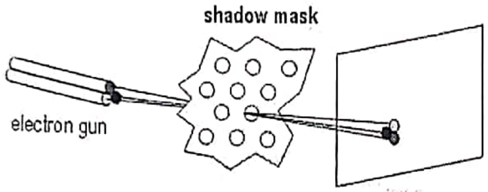
- Less expensive

Disadvantages

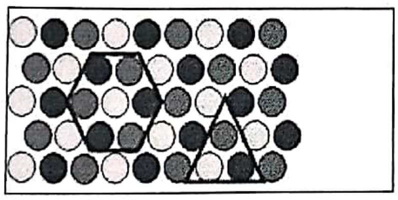
- Quality of images are not good as comparatable with other methods
- Four colors are allowed only

Shadow Mask Method

Raster scan system are use shadow mask methods to produced a much more range of colors than beam penetration method. In this, CRT has three phosphor color dots. One phosphor dot emits a red light, second emits a green light and third emits a blue light. This type of CRT has three electrons guns and a shadow mask grid as shown in figure below:



In this figure, three electrons beams are deflected and focused as a group onto the shadow mask which contains a series of holes. When three beams pass through a hole in shadow mask they activate dot triangle as shown in figure below:



phosphor dot pattern

The colors we can see depends on the amount of excitation of red, green and blue phosphor. A white area is a result of all three dots with equal intensity while yellow is produced with green and red dots and so on.

Advantages

- produce realistic images
- also produced different colors
- and shadows scenes.

Disadvantages

- low resolution
- expensive
- electron beam directed to whole screen

Full Color System

Color CRTs in graphics systems are designed as RGB monitors. These monitors use shadow mask method and take the intensity level for each gun. A RGB color system with 34 bits of storage per pixel is known as full color system or true color system.

Direct-View Storage Tubes

An alternative method for maintaining a screen image is to store the picture information inside the CRT instead of refreshing the screen. A direct-view storage tube (DVST) stores the picture information as a charge distribution just behind the phosphor-coated screen. Two electron guns are used in a DVST. One, the primary gun, is used to store the picture pattern; the second, the flood gun, maintains the picture display.
A DVST monitor has both disadvantages and advantages compared to the refresh CRT. Because no refreshing is needed, very complex pictures can be displayed at very high resolutions without flicker. Disadvantages of DVST systems are that they ordinarily do not display color and that selected parts of a picture cannot he erased. To eliminate a picture section, the entire screen must be erased and the modified picture redrawn. The erasing and redrawing process can take several seconds for a complex picture. For these reasons, storage displays have been largely replaced by raster systems.

Flat-Panel Displays

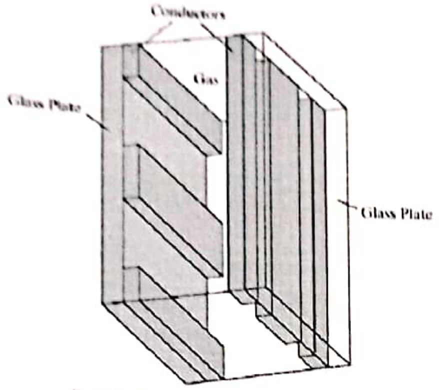
Although most graphics monitors are still constructed with CRTs, other technologies are emerging that may soon replace CRT monitors. The term flat-panel display refers to a class of video devices that have reduced volume, weight, and power requirements compared to a CRT. A significant feature of flat-panel displays is that they are thinner than CRTs, and we can hang them on walls or wear them on our wrists. Since we can even write on some flat-panel displays, they will soon be available as pocket notepads. Current uses for flat-panel displays include small TV monitors, calculators, pocket video games, laptop computers, armrest viewing of movies on airlines, as advertisement boards in elevators, and as graphics displays in applications requiring rugged, portable monitors.

We can separate flat-panel displays into two categories: emissive displays and non emissive displays.

The emissive displays (or emitters) are devices that convert electrical energy into light. Plasma panels, thin-film electroluminescent displays, and Light-emitting diodes are examples of emissive displays. Flat CRTs have also been devised, in which electron beams arts accelerated parallel to the screen, then deflected 90' to the screen. But flat CRTs have not proved to be as successful as other emissive devices.

Plasma Panel

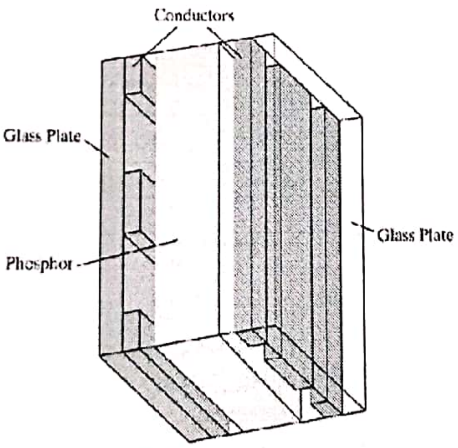
Plasma panels, also called gas-discharge displays, are constructed by filling the region between two glass plates with a mixture of gases that usually includes neon. A series of vertical conducting ribbons is placed on one glass panel, and a set of horizontal ribbons is built into the other glass panel. Firing voltages applied to a pair of horizontal and vertical conductors cause the gas at the intersection of the two conductors to break down into a glowing plasma of electrons and ions. Picture definition is stored in a refresh buffer, and the firing voltages are applied to refresh the pixel positions (at the intersections of the conductors) 60 times per second.



Basic design of a plasma-panel display device.

Thin-film electroluminescent

Thin-film electroluminescent displays are similar in construction to a plasma panel. The difference is that the region between the glass plates is filled with a phosphor, such as zinc sulfide doped with manganese, instead of a gas. When a sufficiently high voltage is applied to a pair of crossing electrodes, the phosphor becomes a conductor in the area of the intersection of the two electrodes. Electrical energy is then absorbed by the manganese atoms, which then release the energy as a spot of light similar to the glowing plasma effect in a plasma panel. Electroluminescent displays require more power than plasma panels, and good color and gray scale displays are hard to achieve.



Basic design of a thin-film electroluminescent display device.

LED (Light Emitting Diode)

A third type of emissive device is the light-emitting diode (LED). A matrix of diodes is arranged to form the pixel positions in the display, and picture definition is stored in a refresh buffer. As in scan-line refreshing of a CRT, information is read from the refresh buffer and converted to voltage levels that are applied to the diodes to produce the light patterns in the display.

Non emissive displays (or non emitters) use optical effects to convert sunlight or light from some other source into graphics patterns. The most important example of a non emissive flat-panel display is a liquid-crystal display.

LCD (Liquid Crystal Display)

Liquid Crystal Displays (LCDS)a re commonly used in small systems, such - as calculators and portable, laptop computers. These non emissive devices produce a picture by passing polarized light from the surroundings or from an internal light source through a liquid-crystal material that can be aligned to either block or transmit the light.

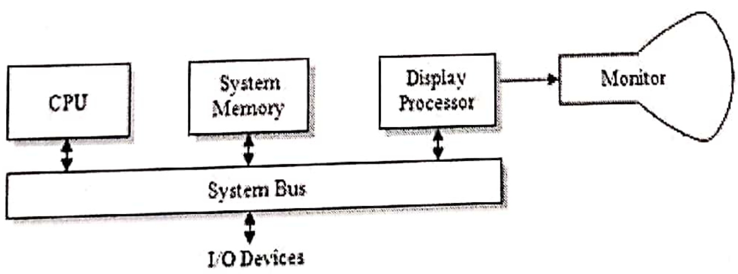
The term liquid crystal refers to the fact that these compounds have a crystalline arrangement of molecules, yet they flow like a liquid.

Color depth and Bit Depth

- The number of discrete intensities that the video card is capable of generating for each color determines the maximum number of colors that can be displayed.
- The number of memory bits required to store color information (intensity values for all three primary color components) about a pixel is called color depth or bit depth.

Raster Scan System

Interactive raster graphics systems typically employ several processing units. In addition to the central processing unit, or CPU, a special-purpose processor, called the video controller or display controller, is used to control the operation of the display device. Here, the frame buffer can be anywhere in the system memory, and the video controller accesses the frame buffer to refresh the screen.

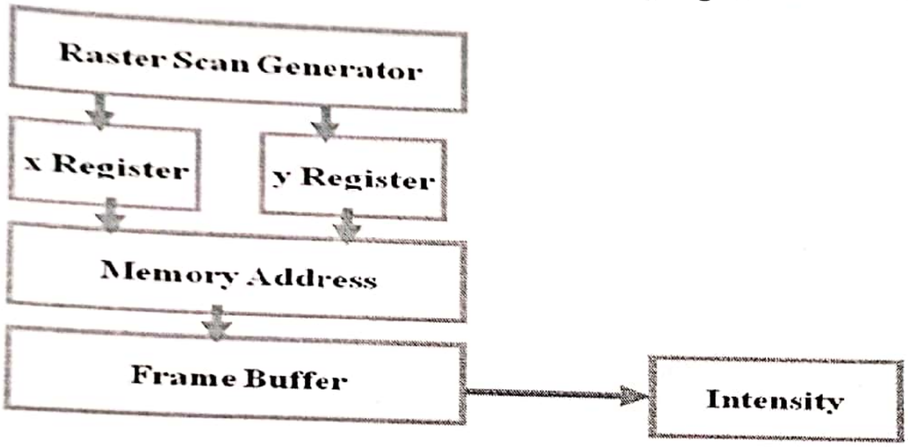


Architecture of Simple Raster Scan System

Video Controller

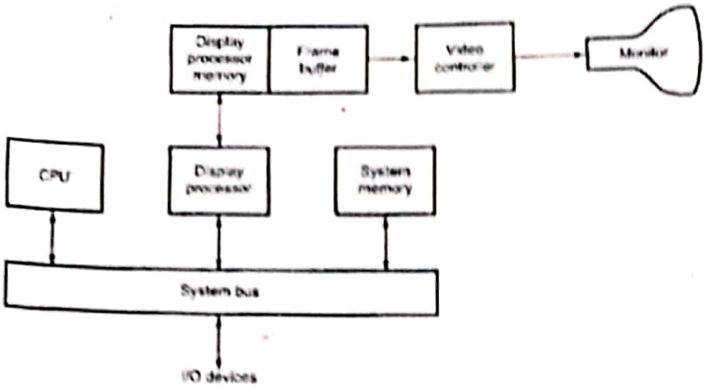
A fixed area of the system memory is reserved for the frame buffer, and the video controller is given direct access to the frame-buffer memory.(Refer Figure 2.26 from Hearn and Baker). For many graphics monitors, the coordinate origin is defined at the lower left screen comer. The screen surface is then represented as the first quadrant of a two-dimensional system, with positive x values increasing to the right and positive y values increasing from bottom to top. (On some personal computers, the coordinate origin is referenced at the upper left comer of the screen, so the y values are inverted.) Scan lines are then labeled from y, at the top of the screen to 0 at the bottom. Along each scan line, screen pixel positions are labeled from 0 to xmax

Two registers are used to store the coordinates of the screen pixels. Initially, the x register is set to 0 and the y register is, set to ymax. The value stored in the frame buffer for this pixel position is then retrieved and used to set the intensity of the CRT beam. Then the x register is incremented by 1, and the process repeated for the next pixel on the top scan line. This procedure is repeated for each pixel along the scan line. After the last pixel on the top scan line has been processed, the x register is reset to 0 and the y register is decremented by 1.



Raster Scan Display Processor

Figure below shows one way to set up the organization of a raster system containing a separate display processor, sometimes referred to as a graphics controller or a display coprocessor. The purpose of the display processor is to free the CPU from the graphics chores. In addition to the system memory, a separate display processor memory area can also be provided. A major task of the display processor is digitizing a picture definition given in an application program into a set of pixel-intensity values for storage in the frame buffer. This digitization process is called scan conversion. Graphics commands specifying straight lines and other geometric objects are scan converted a character defined as a into a set of discrete intensity points.

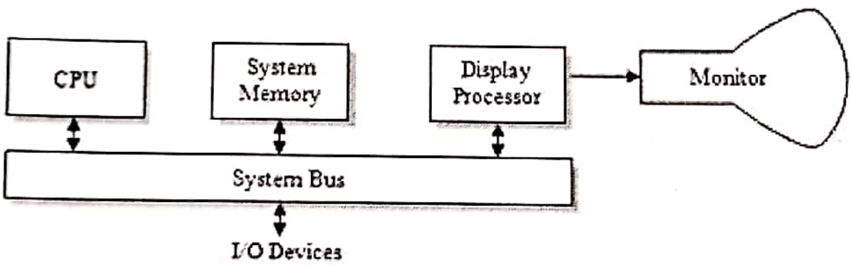


(Architecture of a raster graphics system with a display processor)

Display processors are also designed to perform a number of additional operations. These functions include generating various line styles (dashed, dotted, or solid), displaying color areas, and performing certain transformations and manipulations on displayed objects. Also, display processors are typically designed to interface with interactive input devices, such as a mouse.

Random Scan Systems

An application program is input and stored in the system memory along with a graphics package. Graphics commands in the application program are translated by the graphics package into a display file stored in the system memory. This display file is then accessed by the display processor to refresh the screen.



(Architecture of Simple Random Scan System)

Input Devices

Various devices are available for data input on graphics workstations. Most systems have a keyboard and one or more additional devices specially designed for interactive input. These include a mouse, trackball, spaceball, joystick, digitizers, dials, and button boxes. Some other input devices used In particular applications are data gloves, touch panels, image scanners, and voice systems.

Keyboards

An alphanumeric keyboard on a graphics system is used primarily as a device for entering text strings. The keyboard is an efficient device for inputting such non graphic data as picture labels associated with a graphics display. Keyboards can also be provided with features to facilitate entry of screen coordinates, menu selections, or graphics functions.

Cursor-control keys and function keys are common features on general purpose keyboards. Function keys allow users to enter frequently used operations in a single keystroke, and cursor-control keys can be used to select displayed objects or coordinate positions by positioning the screen cursor. Other types of cursor-positioning devices, such as a trackball or joystick, are included on some keyboards. Additionally, a numeric keypad is, often included on the keyboard for fast entry of numeric data.

Mouse

A mouse is small hand-held box used to position the screen cursor. Wheels or rollers on the bottom of the mouse can be used to record the amount and direction of movement. Another method for detecting mouse motion is with an optical sensor. For these systems, the mouse is moved over a special mouse pad that has a grid of horizontal and vertical lines. The optical sensor detects movement across the lines in the grid.

Since a mouse can be picked up and put down at another position without change in cursor movement, it is used for making relative changes in the position of the screen cursor. One, two, or three buttons are usually included on the top of the mouse for signaling the execution of some operation, such as recording cursor position or invoking a function. Mast general-purpose graphics systems now include a mouse and a keyboard as the major input devices.

Trackball

A trackball is a pointing device consisting of a ball held by a socket containing sensors to detect a rotation of the ball about two axes—like an upside-down mouse with an exposed protruding ball. The user rolls the ball with the thumb, fingers, or the palm of the hand to move a cursor. Large tracker balls are common on CAD workstations for easy precision. Before the advent of the touchpad, small trackballs were common on portable computers, where there may be no desk space on which to run a mouse. Some small thumb balls clip onto the side of the keyboard and have integral buttons with the same function as mouse buttons. The trackball was invented by Tom Cranston and Fred Longstaff as part of the Royal Canadian Navy's DATAR system in 1952, eleven years before the mouse was invented. This first trackball used a Canadian five-pin bowling ball.

(3-button trackball)

Joysticks

A joystick consists of a small, vertical lever (called the stick) mounted on a base that is used to steer the screen cursor around. Most joysticks select screen positions with actual stick movement; others respond to pressure on the stick. Some joysticks are mounted on a keyboard; others function as stand-alone units.

The distance that the stick is moved in any direction from its center position corresponds to screen-cursor movement in that direction. Potentiometers mounted at the base of the joystick measure the amount of movement, and springs return the stick to the center position when it is released. One or more buttons can be programmed to act as input switches to signal certain actions once a screen position has been selected.

Data Gloved

Data glove can be used to grasp a "virtual" object. The glove is constructed with a series of sensors that detect hand and finger motions. Electromagnetic coupling between transmitting antennas and receiving antennas is used to provide information about the position and orientation of the hand. The transmitting and receiving antennas can each be structured as a set of three mutually perpendicular coils, forming a three-dimensional Cartesian coordinate system. Input from the glove can be used to position or manipulate objects in a virtual scene. A two-dimensional projection of the scene can be viewed on a video monitor, or a three-dimensional projection can be viewed with a headset.

Digitizers

A common device for drawing, painting, or interactively selecting coordinate positions on an object is a digitizer. These devices can be used to input coordinate values in either a two-dimensional or a three-dimensional space. Typically, a digitizer is used to scan over a drawing or object and to input a set of discrete coordinate positions, which can be joined with straight-line segments to approximate the curve or surface shapes.

One type of digitizer is the graphics tablet (also referred to as a data tablet), which is used to input two-dimensional coordinates by activating a hand cursor or stylus at selected positions on a

Although Light pens are still with us, they are not as popular as they once were since they have several disadvantages compared to other input devices that have been developed. For one, when a light pen is pointed at the screen, part of the screen image is obscured by the hand and pen. And prolonged use of the light pen can cause arm fatigue. Also, light pens require special implementations for some applications because they cannot detect positions within black areas. To be able to select positions in any screen area with a light pen, we must have some nonzero intensity assigned to each screen pixel. In addition, light pens. Sometimes give false readings due to background lighting in a room.

Voice Systems

Speech recognizers are used in some graphics workstations as input devices to accept voice commands The voice-system input can be used to initiate graphics operations or to enter data. These systems operate by matching an input against a predefined dictionary of words and phrase.

A dictionary is set up for a particular operator by having, the operator speak the command words to be used into the system. Each word is spoken several times, and the system analyzes the word and establishes a frequency pattern for that word in the dictionary along with the corresponding function to be performed. Later, when a voice command is given, the system searches the dictionary for a frequency-pattern match. Voice input is typically spoken into a microphone mounted on a headset. The microphone is designed to minimize input of other background sounds. If a different operator is to use the system, the dictionary must be reestablished with that operator's voice patterns. Voice systems have some advantage over other input devices, since the attention of the operator does not have to be switched from one device to another to enter a command.

Hard Copy Devices

Hard copy device or output devices accept data from a computer and converted them into a form which is suitable for use by the user.

Basic output devices include the

- Monitors
- Printers
- Plotters

Printers

Printers are the most commonly used output devices for producing hard copy output.

The various types of printers in used today are

- Dot-Matrix Printers

DDA Algorithm

This is a scan conversion algorithm based on calculating either delta y and delta x. We sample the line at unit intervals in one coordinate and determine corresponding integer values nearest the line path for the other coordinate.

If the slope is less tan or equal to 1, we sample at unit x intervals(deltax=1) and compute each successive y value as :-

yk+1=yk+m

Subscirpt k takes integer values starting from1,for the first point, increases by 1 until the final endpoint is reached. Since m can be any real number between 0 and 1, the calculated y values must be rounded to the nearest integer.

For lines with positive slope we reverse the roles of y and x. That is we sample at unit intervals y(deltay =1) and calculated each succeeding x value as

xk+1=xk+1/m

```
dda(int x1,int y1,int x2,int y2)
{
float x,y,xinc,yinc,dx,dy;
int k;
int step;
dx=x2-x1;
dy=y2-y1;
if(abs(dx)>abs(dy))
step=abs(dx);
```

```
else
step=abs(dy);
xinc=dx/step;
yinc=dy/step;
x=x1;
y=y1;
putpixel(x,y,1);
for(k=1;k<=step;k++)
{
x=x+xinc;
y=y+yinc;
putpixel(x,y,2); }}
```

DDA algorithm is a faster method for calculating pixel positions than the direct use of eq:-

y=mx+b;

It eliminates the multiplication in above eq by making use of raster characteristics, so that appropriate increments are applied in the x and y direction to step to pixel positions along the line path.

Limitations of DDA:-

1. Round off errors:- The accumulation of round off errors in successive additions of the floating point increment, can cause the calculated pixel position to drift away from the true line path for the long lines.
2. Truncation error:- Because of division the digits after the decimal points are truncated as the value being stored in a variable of integer type.

Because off round off and truncation errors in DDA, we don't get a straight line instead get a step like line.

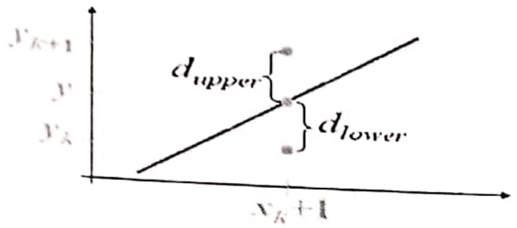
Bresenham Algorithm

The big advantage of this algorithm is that it uses only integer calculations.

Move across the x axis in unit intervals and at each step choose between two different y coordinates

For example, from position (2, 3) we have to choose between (3, 3) and (3, 4) We would like the point that is closer to the original line

At sample position xk+1 the vertical separations from the mathematical line are labelled dupper and dlower



The y coordinate on the mathematical line at xk+1 is:

$$y = m(x_k + 1) + b$$

So, dupper and dlower are given as follows:

$$d_{lower} = y - y_k$$

$$= m(x_k + 1) + b - y_k$$

and:

$$d_{upper} = (y_k + 1) - y$$

$$= y_k + 1 - m(x_k + 1) - b$$

We can use these to make a simple decision about which pixel is closer to the mathematical line

This simple decision is based on the difference between the two pixel positions:

$$d_{lower} - d_{upper} = 2m(x_k + 1) - 2y_k + 2b - 1$$

Let's substitute m with Δy/Δx where Δx and Δy are the differences between the end-points:

$$\Delta x(d_{lower} - d_{upper}) = \Delta x(2\frac{\Delta y}{\Delta x}(x_k + 1) - 2y_k + 2b - 1)$$

$$= 2\Delta y \cdot x_k - 2\Delta x \cdot y_k + 2\Delta y + \Delta x(2b-1)$$

$$= 2\Delta y \cdot x_k - 2\Delta x \cdot y_k + c$$

So, a decision parameter pk for the kth step along a line is given by:

$$p_k = \Delta x(d_{lower} - d_{upper})$$
$$= 2\Delta y \cdot x_k - 2\Delta x \cdot y_k + c$$

The sign of the decision parameter pk is the same as that of dlower – dupper

If pk is negative, then we choose the lower pixel, otherwise we choose the upper pixel

Remember coordinate changes occur along the x axis in unit steps so we can do everything with integer calculations

At step k+1 the decision parameter is given as:

$$p_{k+1} = 2\Delta y \cdot x_{k+1} - 2\Delta x \cdot y_{k+1} + c$$

Subtracting pk from this we get:

$$p_{k+1} - p_k = 2\Delta y(x_{k+1} - x_k) - 2\Delta x(y_{k+1} - y_k)$$

But, xk+1 is the same as xk+1 so:

$$p_{k+1} = p_k + 2\Delta y - 2\Delta x(y_{k+1} - y_k)$$

where yk+1 - yk is either 0 or 1 depending on the sign of pk

The first decision parameter p0 is evaluated at (x0, y0) is given as:

$$p_0 = 2\Delta y - \Delta x$$

Bresenham's Line-Drawing Algorithm for |m|< 1

1. Input the two line endpoints and store the left endpoint in (x0,y0)

2. Load (x0, y0) into the frame buffer; that is, plot the first point.

3. Calculate constants deltax, deltay, 2deltay, and 2deltay – 2deltax, and obtain the starting value for the decision parameter as

po = 2deltay - deltax

4. At each Xk along the line, starting at k = 0, perform the following test: If PK < 0, the next point to plot is xk + 1, yk and

Pk+1=Pk + 2deltay

Otherwise, the next point to plot is (xk+1, yk+1) and

Pk+1=Pk + 2deltay-2deltax

5. Repeat step 4 deltax times.

```
void linebres (int xa ,int y a , int xb, int yb)
(
int dx = a b s ( x a - x b ) , dy = abs (ya - yb):
int p = 2 * dy - d x ;
int twoDy = 2 * dy, twoDyDx = 2 * (dy - Ax);
int x , y, xEnd:
/' Determine which point to use a s start, which as end * /
if (xa > x b ) (
x = xb;
Y = yb;
xEnd = xa;
)
 Else{
x = xa;
Y = ya;
xEnd = xb;
}
putpixel (x, y,1);
while (x < xEnd) (
x++;
if (p < 0)
p += twoDy;
else {
y++;
p += twoDyDx;
)
putpixel ( x , y,1);
end
```

Antialiasing

In the line drawing algorithms, we have seen that all rasterized locations do not match with the true line and we have to select the optimum raster locations to represent a straight line. This problem is severe in low resolution screens. In such screens line appears like a stair step. This effect is known as aliasing. IT is dominant for lines having gentle and sharp slopes.

The aliasing effect can be reduced by adjusting intensities of the pixels along the line. The process of adjusting intensities of the pixels along the line to minimize the effect of aliasing is called antialiasing.

The aliasing effect can be minimized by increasing resolution of the raster display. By increasing resolution and making it twice the original one, the line passes through twice as many column of pixels and therefore has as many jags, but each jag as half as large in x and in y direction.

Clipping

Clipping may be described as the procedure that identifies the portions of a picture lie inside the region, and therefore, should be drawn or, outside the specified region, and hence, not to be drawn. The algorithms that perform the job of clipping are called clipping algorithms there are various types, such as:

- Point Clipping
- Line Clipping
- Polygon Clipping
- Text Clipping
- Curve Clipping

Further, there are a wide variety of algorithms that are designed to perform certain types of clipping operations, some of them which will be discussed in unit.

Line Clipping Algorithms:

- Cohen Sutherland Line Clippings
- Cyrus-Beck Line Clipping Algorithm

Polygon or Area Clipping Algorithm

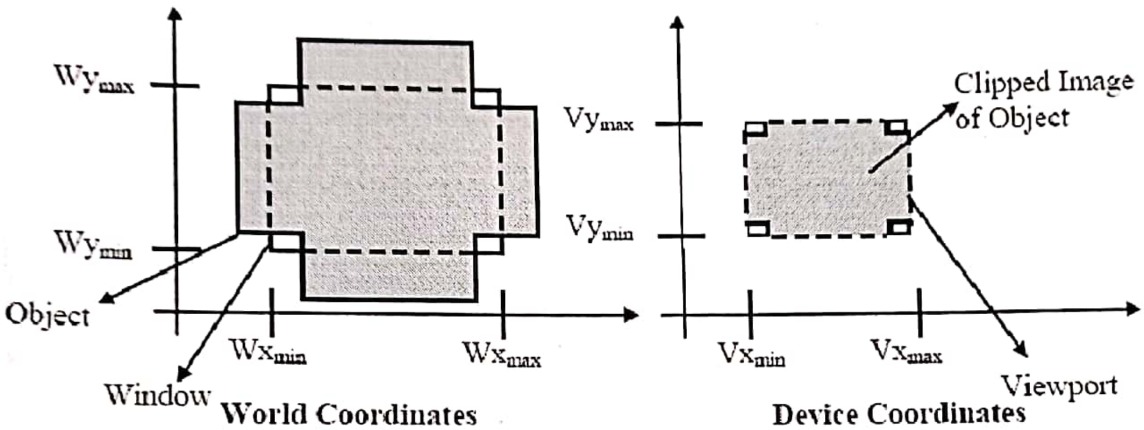
- Sutherland-Hodgman Algorithm

Window may be described as the world coordinate area selected for display. Viewport may be described as the area on a display device on which the window is mapped.

So, it is the window that specifies what is to be shown or displayed whereas viewport specifies where it is to be shown or displayed.
Specifying these two coordinates, i.e., window and viewport coordinates and then the transformation from window to viewport coordinates is very essential from the point of view of clipping.
Note:

- Assumption: That the window and viewport are rectangular. Then only, by specifying the maximum and the minimum coordinates i.e., (Xwmax, Ywmax) and (Xwmin, Ywmin) we can describe the size of the overall window or viewport.
- Window and viewport are not restricted to only rectangular shapes they could be of any other shape (Convex or Concave or both).

For better understanding of the clipping concept refer to Figure Below:



(Clipping alongwith Viewing Transformation through rectangular window and viewport)

Point Clipping

Point clipping is the technique related to proper display of points in the scene, although, this type of clipping is used less frequently in comparison to other types, i.e., line and polygon clipping. But, in some situations, e.g., the scenes which involve particle movements such as explosion, dust etc., it is quite useful. For the sake of simplicity, let us assume that the clip window is rectangular in shape. So, the minimum and maximum coordinate value, i.e., (Xwmax, Ywmax) and (Xwmin, Ywmin) are sufficient to specify window size, and any point (X,Y), which can be shown or displayed should satisfy the following inequalities. Otherwise, the point will not be visible. Thus, the point will be clipped or not can be decided on the basis of following inequalities.

Yw_{max}

$Xw_{min} \le X \le Xw_{max}$
$Yw_{min} \le Y \le Yw_{max}$

Yw_{min}

Xw_{min} Xw_{max}

It is to be noted that (Xwmax, Ywmax) and (Xwmin, Ywmin) can be either world coordinate window boundary or viewport boundary. Further, if any one of these four inequalities is not satisfied, then the point is clipped (not saved for display).

Line Clipping

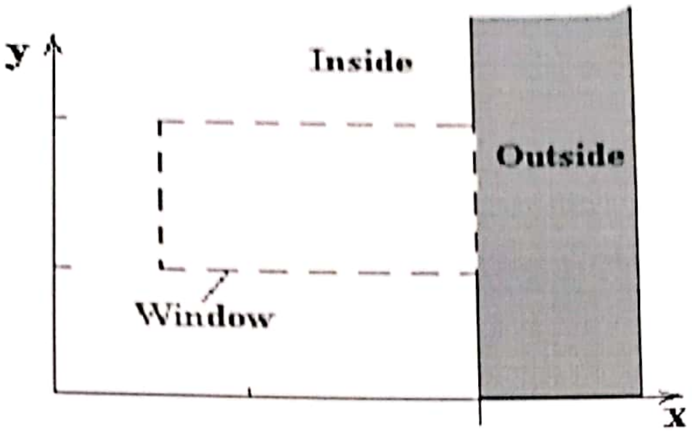
Line is a series of infinite number of points, where no two points have space in between them. So, the above said inequality also holds for every point on the line to be clipped. A variety of line clipping algorithms are available in the world of computer graphics, but we restrict our discussion to the following Line clipping algorithms, name after their respective developers:

1) Cohen Sutherland algorithm,
2) Cyrus-Beck of algorithm

Cohen-Sutherland Line Clipping

The Cohen-Sutherland line clipping algorithm quickly detects and dispenses with two common and trivial cases. To clip a line, we need to consider only its endpoints. If both endpoints of a line lie inside the window, the entire line lies inside the window. It is trivially accepted and needs no clipping. On the other hand, if both endpoints of a line lie entirely to one side of the window, the line must lie entirely outside of the window. It is trivially rejected and needs to be neither clipped nor displayed.

Inside-Outside Window Codes
To determine whether endpoints are inside or outside a window, the algorithm sets up a half-space code for each endpoint. Each edge of the window defines an infinite line that divides the whole space into two half-spaces, the inside half-space and the outside half-space, as shown below.



As you proceed around the window, extending each edge and defining an inside half-space and an outside half-space, nine regions are created - the eight "outside" regions and the one "inside"region. Each of the nine regions associated with the window is assigned a 4-bit code to identify the region. Each bit in the code is set to either a 1(true) or a 0(false). If the region is to the left of the window, the first bit of the code is set to 1. If the region is to the top of the window, the second bit of the code is set to 1. If to the right, the third bit is set, and if to the bottom, the fourth bit is set. The 4 bits in the code then identify each of the nine regions as shown below.

1001	**0001**	**0101**
1000	**0000** **Window**	**0100**
1010	**0010**	**0110**

For any endpoint (x , y) of a line, the code can be determined that identifies which region the endpoint lies. The code's bits are set according to the following conditions:

- First bit set **1** : Point lies to **left** of window $x < x_{min}$
- Second bit set **1** : Point lies to **right** of window $x > x_{max}$
- Third bit set **1** : Point lies below(**bottom**) window $y < y_{min}$
- fourth bit set **1** : Point lies above(**top**) window $y > y_{max}$

Algorithm

Before Clipping The Cohen-Sutherland algorithm uses a divide-and-conquer strategy. The line segment's endpoints are tested to see if the line can be trivally accepted or rejected. If the line cannot be trivally accepted or rejected, an intersection of the line with a window edge is determined and the trivial reject/accept test is repeated. This process is continued until the line is accepted.

To perform the trivial acceptance and rejection tests, we extend the edges of the window to divide the plane of the window into the nine regions. Each end point of the line segment is then assigned the code of the region in which it lies.

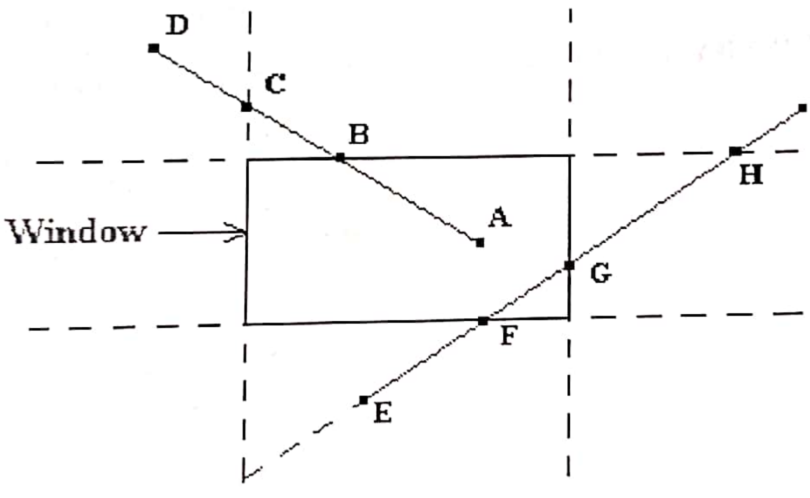
1. Given a line segment with endpoint $P_1 = (x_1, y_1)$ and $P_2 = (x_2, y_2)$
2. Compute the 4-bit codes for each endpoint.

If both codes are 0000,(bitwise OR of the codes yields 0000) line lies completely inside the window: pass the endpoints to the draw routine.

If both codes have a 1 in the same bit position (bitwise AND of the codes is not 0000), the line lies outside the window. It can be trivially rejected.

3. If a line cannot be trivially accepted or rejected, at least one of the two endpoints must lie outside the window and the line segment crosses a window edge. This line must be clipped at the window edge before being passed to the drawing routine.
4. Examine one of the endpoints, say $P_1 = (x_1, y_1)$. Read P_1 's 4-bit code in order: Left-to-Right, Bottom-to-Top.
5. When a set bit (1) is found, compute the intersection I of the corresponding window edge with the line from P_1 to P_2. Replace P_1 with I and repeat the algorithm.

Illustration of Line Clipping



1. Consider the line segment AD.

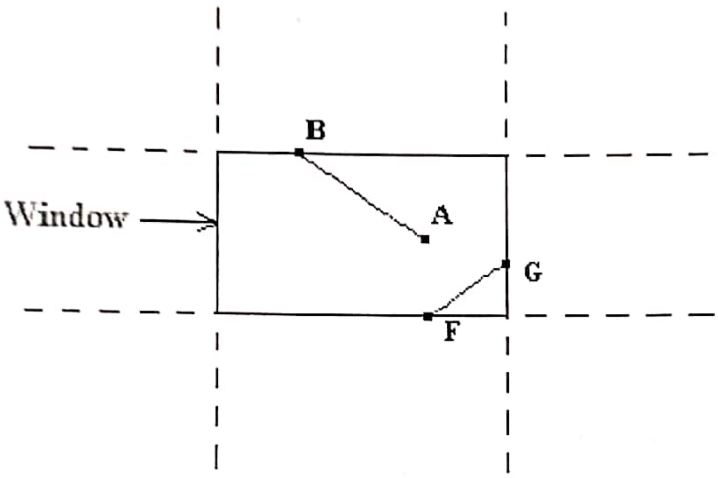
Point A has an outcode of 0000 and point D has an outcode of 1001. The logical AND of these outcodes is zero; therefore, the line cannot be trivally rejected. Also, the logical OR of the outcodes is not zero; therefore, the line cannot be trivally accepted. The algorithm then chooses D as the outside point (its outcode contains 1's). By our testing order, we first use the top edge to clip AD at B. The algorithm then recomputes B's outcode as 0000. With the next iteration of the algorithm, AB is tested and is trivially accepted and displayed.

2. Consider the line segment EI

Point E has an outcode of 0100, while point I's outcode is 1010. The results of the trivial tests show that the line can neither be trivally rejected or accepted. Point E is determined to be an outside point, so the algorithm clips the line against the bottom edge of the window. Now line EI has been clipped to be line FI. Line FI is tested and cannot be trivially accepted or rejected. Point F has an outcode of 0000, so the algorithm chooses point I as an outside point since its outcode is1010. The line FI is clipped against the window's top edge, yielding a new line FH. Line FH cannot be trivally accepted or rejected. Since H's outcode is 0010, the next iteration of the algorthm clips against the window's right edge, yielding line FG. The next iteration of the algorithm tests FG, and it is trivially accepted and display.

After Clipping

After clipping the segments AD and EI, the result is that only the line segment AB and FG can be seen in the window.



The sequence for reading the codes' bits is LRBT (Left, Right, Bottom, Top).

Once the codes for each endpoint of a line are determined, the logical AND operation of the codes determines if the line is completely outside of the window. If the logical AND of the endpoint codes is not zero, the line can be trivally rejected. For example, if an endpoint had a code of 1001 while the other endpoint had a code of 1010, the logical AND would be 1000 which indicates the line segment lies outside of the window. On the other hand, if the endpoints had codes of 1001 and 0110, the logical AND would be 0000, and the line could not be trivally rejected.

The logical OR of the endpoint codes determines if the line is completely inside the window. If the logical OR is zero, the line can be trivally accepted. For example, if the endpoint codes are 0000 and 0000, the logical OR is 0000 - the line can be trivally accepted. If the endpoint codes are 0000 and 0110, the logical OR is 0110 and the line can not be trivally accepted.

LINE CLIPPING: THE COHEN-SUTHERLAND LINE CLIPPING ALGORITHM

Given the endpoint coordinates (x1,y1) and (x2,y2) of a line, determine what part of that line lies inside the rectangular clip area whose lefthand border is x=xmin, righth and border x=xmax, bottom border y=ymin, and top border y=ymax.

Could do point test for all points on the line--i.e., for a point x,y:

if ((x<=xmax) && (x>=xmin) && (y<=ymax) && (y>=ymin)) the point x,y lies inside the clip area

We want a simple test involving the line's endpoint coordinates.

Observation-- All lines fall into one of three categories

1. Both endpoints inside rectangle (Trivially accept entire line)

2. Both endpoints outside rectangle on the same side of one of its borders (Trivially reject entire line)

3. Neither 1 or 2 ==> chop off the part of the line outside one of borders (which gives a new endpoint) and see if the resulting line has been reduced to Category 1 or Category 2.

A tool to use in assigning lines to Category 1 or 2: a 4-bit region code (RC) that can be assigned to an endpoint (x,y) . Any set bit means the endpoint is outside of one of the 4 borders of the clip rectangle--

RC = LRBA, L=left (if x<xmin, L=1, else L=0)
R=Right (if x>xmax, R=1, else R=0)
B=Bottom (if y<ymin, B=1, else B=0)
T=Top (if y>ymax, T=1, else B=0)

The entire x-y plane can be divided into 9 regions, depending on the value of the Region Code:

Points inside the clip region have a region code of 0000. Assume that the region codes for the two endpoints of a line are RC1 and RC2. A bitwise OR of RC1 with RC2 will give zero only if all bits of both are zero:

```
if (RC1 OR RC2 == 0)
  both RCs are 0000, and both endpoints are inside
  ==> Category 1 (trivial accept)
```

If both endpoints are outside the SAME border of the clip region (Category 2 line), then both endpoint region codes will have the SAME bit set in one of the four bit positions. A bitwise AND of RC1 with RC2 will give a nonzero result if that is the case:

```
if (RC1 AND RC2 != 0)
  Both endpoints are outside of the same border
  ==> Category 2 (trivial reject)
```

Clipped Line endpoints (x1,y1,x2,y2) [if AC == TRUE]

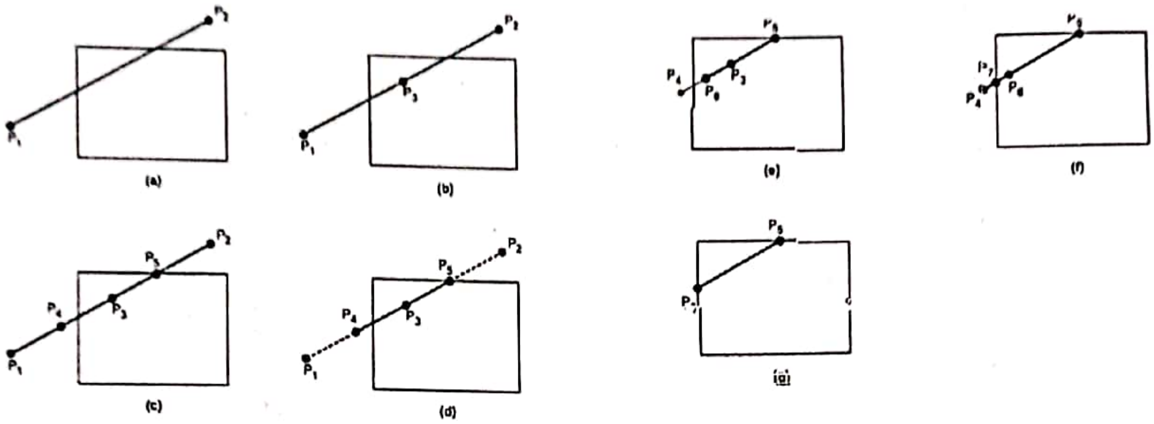
PSEUDO-CODE:

```
CS_LineClip(xmin,ymin,xmax,ymax,x1,y1,x2,y2,AC)
  Calculate m and b from x1,y1,x2,y2
  done = FALSE
  While (NOT done)
    Calculate endpoint codes rc1, rc2
    If ((rc1 OR rc2) == 0)      // Category 1, Trivial Accept
      done = TRUE
      AC = TRUE
    Else
      If ((rc1 AND rc2) != 0)  // Category 2, Trivial Reject
        done = TRUE
        AC = FALSE
      Else
        If (P1 is inside)
          Swap x1,x2, y1,y2, and rc1,rc2
        If (L-bit of rc1 is set)        // 1xxx
          x1 = xmin
          y1 = m*x1 + b
        Else
          If (R-bit of rc1 is set)      // x1xx
            x1 = xmax
            y1 = m*x1 + b
          Else
            If (B-bit of rc1 is set)    // xx1x
              y1 = ymin
              x1 = (y1-b)/m
            Else                        // xxx1
              y1 = ymax
              x1 = (y1-b)/m
```

Midpoint Subdivision Algorithm

Like Cohen Sutherland algorithm, initially the line is tested for visibility. If line is completely visible then it is drawn and if it is completely invisible it is rejected. If line is partially visible then it is subdivided into 2 equal parts. The visibility test is then applied to each half. This subdivision process is repeated until we get completely visible or completely invisible line segments.

As shown in figure line P1 P2 is partially visible. It is subdivided into 2 equal parts P1P3 and P3P2.Both the line segments are tested for visibility and found to be partially visible. Both the line segments is then subdivided in two equal parts to get mid points P4 and P5. It is observed that line segments P1P4 and P5P2 are completely invisible and hence rejected. However line segment P3P5 is completely visible and hence drawn. The remaining line segment P4P3 is still partially visible. It is then subdivided to get midpoint P6. It is observed that P6P3 is completely visible whereas P4P6 is partially visible. Thus P6P3 line segment is drawn and P4P6 line segment is further subdivided to get midpoint P7. Now, it is observed that line segment P4P7 is completely invisible and line segment P7P6 is completely visible and there is no further partially visible segment.



Algorithm

1. Read 2 end points of line say P1(x1,y1) and P2(x2,y2).
2. Read two corners (left top and right bottom) of the window say (Wx1,Wy1 and Wx2,Wy2).
3. Assign region codes for two end points using following steps:
   Initialize code with bits 0000
   Set Bit4- if(x<Wx1)
   Set Bit3- if(x>Wx2)
   Set Bit2- if(y<Wy1)
   Set Bit1- if(y>Wy2)
4. Check for visibility of line
   a) If region codes for both endpoints are zero then the line completely visible. Hence draw the line and goto step 6.
   b) If region codes for both endpoints are not zero and the logical ANDing of them is also non zero then the line is completely invisible, so reject the line and goto step6.
   c) If region codes for two endpoints do not satisfy the conditions in 4a) and 4b) the line us partially visible.
5. Divide the partially visible line segment in equal parts and repeat steps 3 through 5 for both subdivided line segments until you get completely visible and completely invisible line segments.
6. Stop.

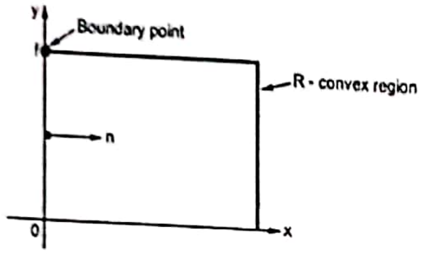
Generalized Clipping with Cyrus-Beck Algorithm

The algorithms explained above assume that clipping window is a regular rectangle. These algorithms are not applicable for non rectangular windows. Cyrus and beck have developed a generalized line clipping algorithm. This algorithm is applicable to arbitrary convex region. This algorithm uses a parametric equation of line segment to find the intersection points of a line with clipping edges. The parametric equation of line from P1 to P2 is :-

P(t)= P1+(P2-P1)t ; 0<=t<=1; **(eq1)**

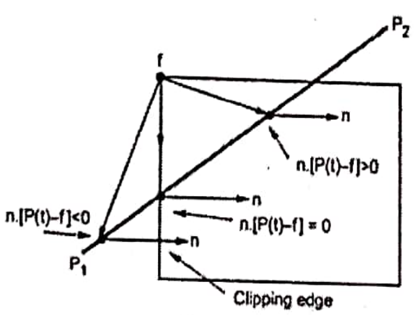
where 't' is a parameter , t=0 at P1 and t=1 at P2

Consider a convex clipping region R, f is a boundary point of the convex region R and n is a inner normal for one of is boundary as shown in the figure below:-



Then we can distinguish in which region a point lie by looking at the value of the dot product **n.[P(t)-f]** as shown in figure below:-

1. If dot product is negative i.e.
   n.[P(t)-f]<0 (eq2)
   Then the vector P(t)-f is pointed away from the interior of R.
2. If dot product is zero i.e.
   n.[P(t)-f]=0 (eq3)
   Then P(t)-f is pointed parallel to the plane containing f and perpendicular to the normal.
3. If dot product is positive
   n.[P(t)-f]>0 (eq4)
   Then the vector P(t)-f is pointed towards the interior of R.



As shown in the above figure, if the point f lies in the boundary plane or edge for which n is the inner normal, then that point t on the line P(t) which satisfies n-[P(t)-f]=0 condition is the intersection of the line with the boundary edge.

To get the formal statement of the cyrus beck algorithm we substitute value of P(t) in equation 3.

n.[P(t)-f]= n.[P1 +(P2-P1)t –f] =0 (eq5)

This relation should be applied for each boundary plane or edge of the window to get the intersection points. Thus in general form equation 5 can be written as

n_i.[P1-f_i]+n_i.[P2-P1]t=0 (eq6)

where i is the edge number

Solving eq6 we get

n_i.[P1-f_i]+n_i[P2-P1]t=0 (eq7)

Here, vector P2-P1 defines the direction of the line. The direction of the line is important to correctly identify the visibility of the line. The vector P1-f_i is proportional to the distance from the end point of the line to the boundary point.

Let us define

D= P2-P1 as the direction of the line
W_i=P1-f_i as a weighting factor.

Substituting newly defined variable D and W_i in equation 7 we get
n_i.W_i+(n_i.D)t=0 (eq8)

t= - n_i.W_i / n_i.D (eq9)

where D!=0 and i=1,2,3,...

The equation 9 is used to obtain the value of t for the intersection of the line with each edge of the clipping window. We must select the proper value for t using following tips:-

1. If t is outside the range 0<=t<=1
2. We know that the line can intersect the convex window in at most two points i.e. at two values of t. With equation 9, there can be several values of t in the range of 0<=t<=1 . We have to choose the largest lower limit and smallest upper limit.

If D_i.n_i>0 then equation 9 gives the lower limit value for t and if D_i.n_i<0 gives the upper limit value of t.

Algorithm

1. Read two end points of the line say P1, P2.

2. Read vertex coordinates of the clipping window.
3. Calculate D=P2-P1
4. Assign boundary point (f) with particular edge
5. Find inner vector normal for corresponding edge
6. Calculate D.n and W=P1-f
7. If D.n>0

    t_L= -(W.n)/(D.n)

    else

    t_U= -(W.n)/(D.n)
8. Repeat step 4 through 7 for each edge of the window.
9. Find maximum lower limit and minimum upper limit.
10. If maximum lower limit and minimum upper limit do not satisfy condition 0<=t<=1 then ignore the line.
11. Calculate the intersection points by substituting values of maximum lower limit and minimum upper limit in the parametric equation of the line P1P2.
12. Draw the line segment $P(t_L)$ and $P(t_u)$
13. Stop