Binghamton University

EngiNet™

State University of New York

Thomas J. Watson

School of Engineering
and Applied Science

EngiNet™

WARNING

All rights reserved. No Part of this video lecture series may be reproduced in any form or by any electronic or mechanical means, including the use of information storage and retrieval systems, without written approval from the copyright owner.

©2001 The Research Foundation of the State University of New York

CS 560

Computer Graphics

Professor Richard Eckert

Lecture # 4

January 31, 2001

Lecture 4

Wednesday, 1-31-01

- A. An Introduction to MFC Programming, continued
- B. Computer Graphics Hardware

SKETCH Application

- Example of Using AppWizard and ClassWizard
- Mouse used as a drawing pencil
- Left mouse button down
  - Line in window follows mouse motion (sketching)
- Left mouse button up
  - Sketching stops
- User clicks "Drawing Color" menu item
  - popup menu to choose drawing color
- User clicks "Clear" menu item
  - Window client area is erased

Last Class We:

- Used AppWizard to set up an SDI Skeleton
- Defined Variables in View Class:
  – m_butdn, m_ptold, m_pt, m_color, *pDC
- Used Menu Editor to add menu items:
  – IDM_RED, IDM_GREEN, IDM_BLUE, IDM_CLEAR
- Used ClassWizard to add handlers for:
  – WM_LBUTTONDOWN, WM_LBUTTONUP
  – WM_MOUSEMOVE

Code Requirements

- WM_LBUTTONDOWN:
  – Set m_butdn to TRUE & record point in m_ptold
- WM_LBUTTONUP:
  – Set m_butdn to FALSE

WM_MOUSEMOVE handler code:

- *if (m_butdn)*
  - *{pDC = GetDC();*
  - *m_pt = point;*
  - *CPen pen(PS_SOLID, 1, m_color);*
  - *CPen *pPenOld = pDC->SelectObject(&pen);*
  - *pDCMoveTo(m_ptold);*
  - *pDCLineTo(m_pt);*
  - *m_ptold = m_pt;*
  - *pDC->SelectObject(pPenOld);}*

Adding the WM_COMMAND menu item handlers

- Invoke ClassWizard and scroll "ObjectIDs" list (not "Messages") to "IDM_BLUE"
- Select it and choose "COMMAND" in "Messages" list box
- Click "Add Function" button
- Click "OK" in resulting "Add Member Function" dialog box

- Result:
  - ClassWizard generates *On_Blue()* handler to message map
  - Press "Edit Code" to go to skeleton handler and add following code after // TODO...

    m_Color = RGB(0,0,255);
- Do same to add *OnGreen()* and *OnRed()* handlers
- Add *OnClear()* handler in same way
- Code to be added:

  Invalidate(); //forces WM_PAINT msg

Member Variable Initialization

- All data members added to *CSketchView* class must be initialized
  - If not, they will contain garbage when application begins
- From Project Workspace window choose ClassView icon
  - Expand "sketch classes" icon and *CSketchView* Class icon

– Double click on *CSketchView()* constructor

– Add following initialization code under "// TODO..." comment:

```
m_pt = m_ptold = CPoint(0,0);
m_butdn = FALSE;
m_color = RGB(0,0,0);  // initial draw color black
```

Build project

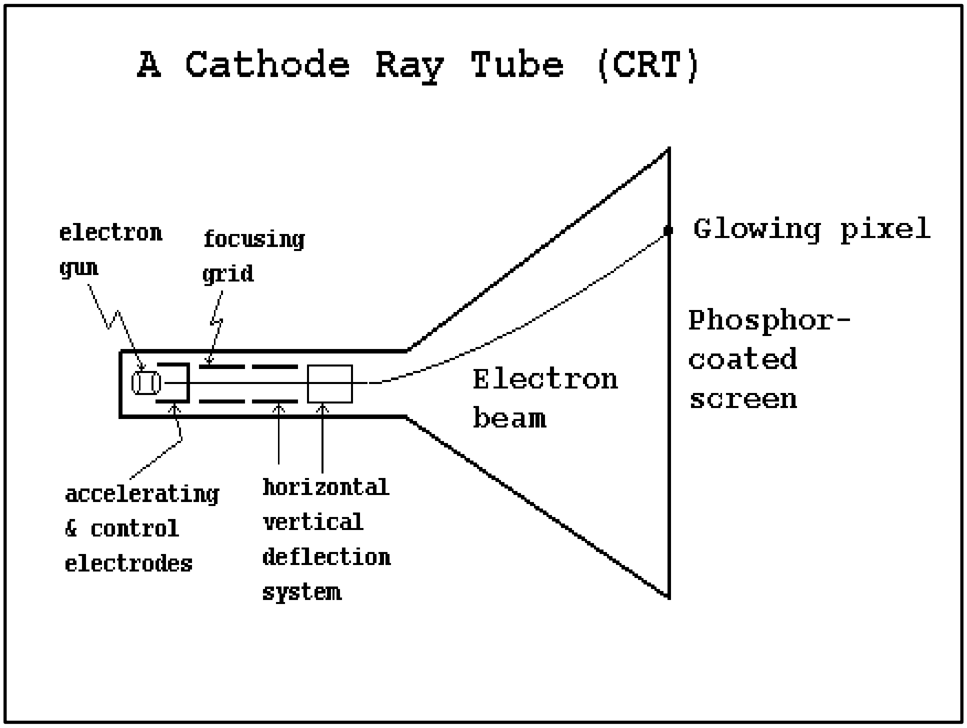
- “Build” from main menu
- If no errors, you get a functioning Document/View color sketching application

Lecture 4, Part B:

Introduction to Computer Graphics Hardware

GRAPHICS HARDWARE

- Display Devices
  - Vector Scan
  - Raster Scan
- Both based on CRT (TV)
  - Electron beam accelerated toward screen
    - focused
    - deflected
    - strikes phosphorescent material on screen
      -->pixel that glows



A Pixel

- Visible point where electron beam hits screen
- Screen phosphors glow & fade
- Have a finite size
- Not a mathematical point

Resolution

- Max number of pixels that can be plotted without overlap
- Expressed in # horizontal X # vertical pixels
- Depends on:
  - phosphor used
  - focusing system (how small a point)
  - speed of deflection system
  - video memory size (raster scan)--as we'll see

Aspect Ratio

- Ratio of # of pixel columns to # of pixel rows
- Examples:
  - SVGA VESA mode 100h: 640 X 400, A.R. = 1.6
  - Standard Windows: 640 X480. A.R. = 1.33
- **Pixel Ratio** (often called Aspect Ratio)
  - Ratio of pixel height to pixel width
  - Ratio of # of horizontal pixels to vertical pixels needed to produce equal length lines
  - For a square screen, A.R. = P.R.
  - If P.R. != 1, figures are distorted

Dot Pitch

- Minimum distance between centers of adjacent pixels of same color
- Should be less than 0.28 mm for sharp images
- For fixed sized screen
  - Decreasing distance between pixels ==> Increase Resolution
  - So dot pitch determines max resolution

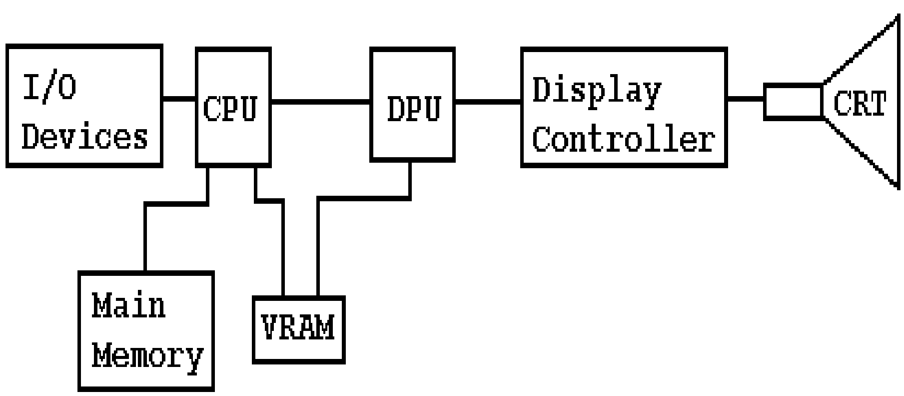
Persistence

- After beam leaves a phosphor, it fades
- Definition of persistence:
  - Time to reduce initial intensity by 10%
  - Value depends on type of phosphor (10 - 100 msec.)
- Finite persistence==>screen must be redrawn
  - Refresh rate determined by persistence
- Example: If persistence = 20 msec
  - 1st pixel on screen invisible after that time ==>
    - screen must be refreshed once every 20 msec
    - so refresh rate must be > 50 Hz.

A Graphics Hardware System

- CPU--Runs pgm specifying what is to be drawn
- CRT--does the actual display
- Display Controller--Provides analog voltages needed to move beam and vary its intensity
- DPU--generates signals that drive display controller
  - (offloads task of video control to separate processor)
- VRAM--Stores info needed to draw the picture
  - Dual-ported (written to by CPU, read from by DPU)
  - Fast (e.g., 640X480, 50 Hz ==> 65 nsec access time!)
  - Also called Refresh Buffer or Frame Buffer
- I/O devices--interface CPU with user

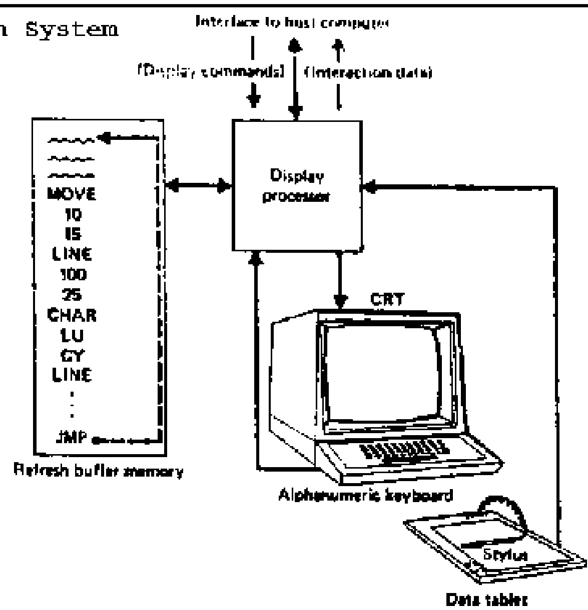
A Computer Graphics Hardware System (General)



Vector Scan systems

- Also called random, stroke, calligraphic displays
- Images drawn as line segments (vectors)
- Beam can be moved to any position on screen
- Refresh Buffer stores plotting commands
  - So Refresh Buffer often called "Display File”
  - provides DPU with needed endpoint coordinates
  - Pixel size independent of frame buffer
    - ==> very high resolution

Vector Scan System



Typical refresh display device. Display list in memory shows symbolic representation of plotting commands followed by values (e.g., *x*, *y* coordinates or characters).

Advantages of Vector Scan

- High resolution (good for detailed line drawings)
- Crisp lines (no "jaggies")
- High contrast (beam can dwell on a pixel==>very intense)
- Selective erase (remove commands from display file)
- Animation (change line endpoints slightly after each refresh)

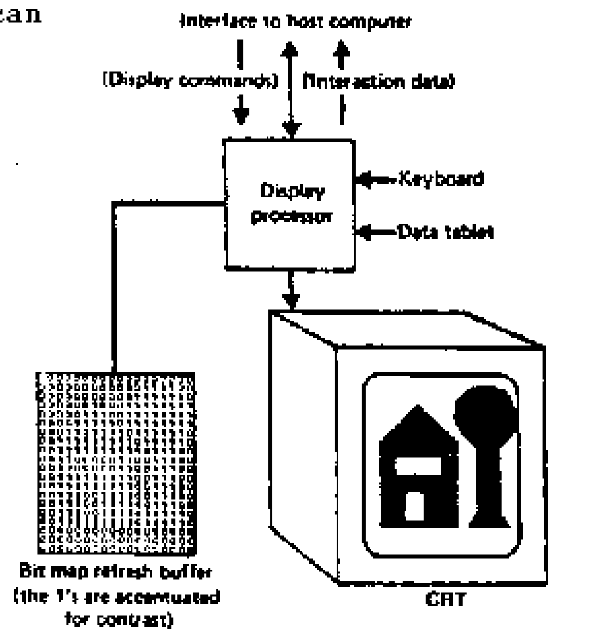
Disadvantages of Vector Scan

- Complex drawings can have flicker
  - Many lines
    - so if time to draw > refresh time ==> flicker
  - High cost--very fast deflection system needed
  - Hard to get colors
  - No area fill
    - so it’s difficult to use for realistic (shaded) images
  - 1960s Technology, only used for special purpose stuff today

Raster Scan Systems (TV Technology)

- Beam continually traces a raster pattern
- Intensity adjusted as raster scan takes place
  - Beam focuses on each pixel
  - Intensity value stored in refresh (frame) buffer
  - So resolution determined by size of frame buffer
- Each pixel on screen visited during each scan
  - Scan rate must be >= 30 Hz to avoid flicker

Raster Scan System



Typical raster graphics display showing house and tree.

Simplest system: one bit per pixel

- frame buffer called a bitmap

Gray Scale: N bits/pixel

- 2^N intensities possible
- memory intensive
  - Example:1000 X 1000 X 256 shades of gray ==> 8 Mbits

Scan Conversion

- Process of determining which pixels need to be turned on in the frame buffer to draw a given graphics primitive
- Need algorithms to efficiently scan convert primitives like lines, circles, etc.

Advantages of Raster Scan Systems

- Low cost (TV technology)
- Area fill (entire screen painted on each scan)
- Colors
- Selective erase (just change parts of bitmap)
- Bright display, good contrast
  - but not as good as vector scan:
  - can't make beam dwell on a pixel

Disadvantages

- Large memory requirement for high resolution
  - (but cost of VRAM has decreased!)
- Aliasing (due to finite size of pixels)
  - Jagged lines (staircase effect)
  - Moire patterns, scintillation, "creep" in animations
- Raster scan is the principal "now" technology for graphics displays!

Tektronix Direct View Storage Tube

- 1st "inexpensive" graphics display device
- Extension of vector scan technique
- Two electron guns
  - writing gun
  - flood gun

- Writing gun beam knocks electrons out
  leaves + charges behind (constitute image)
- Flood gun supplies continuous source of unfocused electrons
  - migrate toward the + charges on grid
  - pass through grid and strike screen phosphors
    --> lighted dots
  - electrons continue to hit + charges
  - continuous light (Up to an hour)

Erasure of DVST image

1. Plus charge applied to entire grid
   – Attracts electrons to entire grid
   – Entire screen flashes (Image gone)
2. Minus charge applied to entire grid
   – Provides electrons that can be knocked out by writing gun
   – Ready to draw next image with writing gun

Advantages to DVST

- No refresh needed
  – unlimited image complexity possible
- High resolution
- Crisp lines
- Low cost
  – no fast refresh circuitry needed

Disadvantages to DVST

- No selective erase
  - whole image or nothing
- No animation
- Low light output
  - poor contrast
  - must use in subdued light
- No color
- No area fill

Interlaced Displays

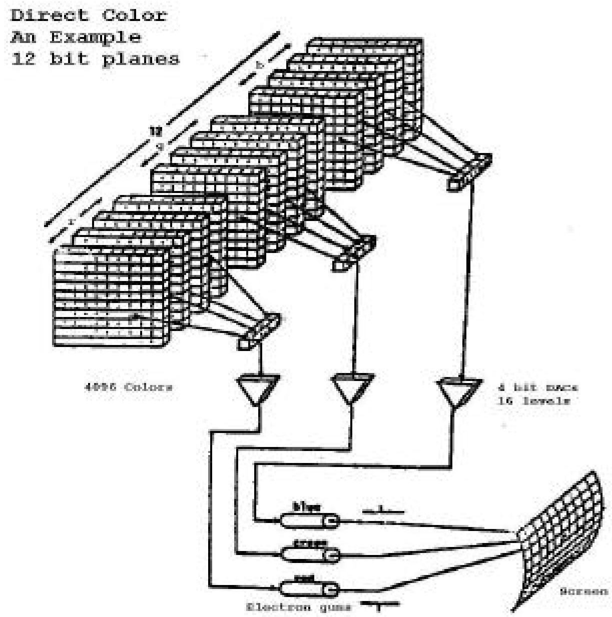
- All even then all odd screen lines scanned
- Typically 1/60 second each
  - Same image presented twice in 1/30 second
  - Image changed at 1/2 non-interlaced frequency
    - less demands on image generation system
    - can be less expensive
    - 30 Hz is borderline for flicker
    - lower quality image (seeing half the image at a time)

Color Display Hardware (raster)

- Each screen pixel composed of 3 phosphors
  - glow red, green, and blue
- 3 electron guns shoot their beams through a shadow mask
  - so beams hit the sensitive phosphors
- Intensity of 3 beams determines how bright each phosphor glows
- Human eye detects an additive color mix
  - e.g., max red, green, & blue perceived as white

Direct color systems

- Frame buffer divided into bit planes
- A bit plane contributes one bit to color of pixels
- If resolution of the screen is W x H pixels:
  - a bit plane is a W x H x 1 bit memory
- Bit planes can be organized into 3 sets
  - Each called a color channel: (R, G, B)
  - Bit planes of a color channel provide the intensity values fed to that channel's electron gun
- A system with N bit planes per color channel:
  - 2^N red, 2^N green, & 2^N blue shades
  - 2^3N different colors displayable simultaneously



True Color & High Color Systems

- True color: direct color system with:

  N=8

  so 2^24 = 16,777,216 different colors possible for each pixel on screen

  More colors than discernable by human eye

- High color: direct color system with:

  Nr=5, Ng=6, Nb=5

  2^16 = 65, 536 different colors possible