Macintosh: Calculating the Draw Rate

This article last reviewed: 14 October 1987

Apple is often asked about the "draw rate" of Macintosh computers. If the question is "how many pixels per second?" the answer is "it varies." This is explained below. If the question is "how many vectors per second?" the answer is "the Macintosh doesn't use vector processing."

This article explains how graphic images are produced on both the Macintosh and on vector graphics terminals, and shows why there's no constant "draw rate" for the Macintosh.

Vector Graphics vs Bit-mapping

Vector graphic (line drawing) systems evolved from the basic plotter design that was first used for displaying computer graphics. To conserve memory (remember when memory was expensive?), only two points are stored for each line. The display image is calculated, during a screen refresh, by image generating hardware. Entire images can be stored on a mainframe or mass storage device and off-loaded to a terminal for processing.

The Macintosh uses a different technology, bit-mapping, which deals with an array of pixels mapped into RAM. A new image is displayed as fast as the bit-map RAM is updated; there is no need to calculate vectors. Storing the pixel information requires a great deal of memory. A monochrome image requires one bit per pixel; a color pixel requires several bits to specify its color (pixel mapping). Because a display like that of the Macintosh II contains 307,200 pixels (640 x 480), a minimum of 37.5 Kbytes is needed for a monochrome display. The same display using 8 bits per color pixel would need about 300 Kbytes.

Bit-mapped images, once stored, are updated at the speed of the vertical refresh rate, which is 66.67Hz on the Macintosh II video card. This means a full screen of video information (307,200 pixels) is displayed 1/66.67 times per second 14.5ms and a horizontal line (480 pixels) takes only 1/35,000 of a second .0285714ms (35KHz scan rate). (The actual number of pixels is slightly more as there are black borders where the beam is off.)

Calculating the Draw Rate

To calculate the time it takes to draw a given line, you must define where and when on a system the drawing of the line occurs. On a vector graphic

system, the vectors are calculated (by a graphics generator) and sent to an image processor (video circuitry) for calculation of lines. Eventually they are converted by video circuitry to a display.

On a Macintosh, the "graphics generator" consists of the CPU and QuickDraw routines. The image is stored in video memory as a pixel map that is displayed by the video card circuitry. No further modeling is needed at the video card.

In both cases, to determine the time required to draw a line, a starting point and end point are needed. A number of other events can influence the timings and that will not accurately describe a working situation. This is not to imply that the two methods, pixel mapping and vector graphics, can or should be compared in this manner -- they're two different technologies. The benefit of pixel mapping lies in the finer control of each pixel for color or even video (analog) image processing at high speeds -- something not available through vector graphics processing.

Kinds of Graphics Generators

A graphics generator is a dedicated processor (or host) that calculates and stores graphics information in some format, not necessarily similar to what will be displayed. A video processor is the circuitry that examines and converts graphics data into signals for the actual display. The examined data on vector graphics systems needs an added function of the video processor for the purpose of calculating the points between the end points as well as outputting the display signals.

Some of the possible methods (there are endless permutations) of generating graphics on a CRT:

- A graphics generator (dedicated or computer system) mixed with live video sources and even stored video sources with output to a display monitor and possibly a transmission and/or recording device.

- A standalone graphics system connected directly to a monitor.

- A host computer connected to a monitor. Usually, update times limit the practicality of using graphics terminals to alphanumeric or vector displays.

- A host computer connected to a graphics terminal that has its own processor to handle the graphics calculations.

- A host computer connected to a graphics controller that is connected to a monitor. The display information is fed to the controller for channeling to the needed device.

One of the options being used with small non-graphics dedicated computers is to place a graphics generator on a controller card with a video processor. The purpose of this is to free up the system CPU from the graphics handling tasks that it shares with the various system tasks.

The Macintosh II uses a host computer to graphics controller configuration

with the storage of the RGB pixel information directly on the controller/video card. Other pixel maps can be created in the main memory of the Macintosh as well. This is useful for fast updates of the video card's pixel map. All logical maps or QuickDraw records are stored in main memory. This refers to the calculated pixels that are then transferred to the video card memory for updating of the display pixel map.

The Bottom Line

Now, to face the question of how long it takes to draw a given line. Because a Macintosh actually displays pixels, not lines, the question can't be answered with a simple number. The display times as noted above the display are themselves updated as fast as the Macintosh II CPU, running at 16Mhz, can calculate the line. This is still not display time, since it has to be combined with the time to transfer the calculated bits to the video card. The transfer time is not much of an issue when lines are the question, because very little information needs to be transferred. NuBus, the connection between the Macintosh II CPU board and the video card, can transfer 2.7 MBytes per second. Because a line in monochrome video mode might have 67 dots to the inch, only 67 bits need to be transferred. This doesn't take into account transfer protocol information and size of the data path on the NuBus.

The question then narrows down to "how long does it take QuickDraw to calculate a 1-inch line?" When you consider the other processes that must go on during an application:

- memory management
- vertical blanking interrupt processes
- interrupts from external devices
- the location of the line and its relation to other graphic entities

you begin to see why there is no single definition of the time it takes to draw a given line.

For More Information

The Raster Graphics Handbook, Conrac Division, Conrac Corporation
The AppleLink article "Macintosh II: Video Signals and Pinouts"
<None>

Keywords: <None>

==========

19960215 11:05:19.00
Tech Info Library Article Number: 836