Chapter 3

Display Devices

3.1 CRT's - Cathode Ray Tubes

Commonly called a *picture tube*, from the days when most people only encountered them in their television sets, the *cathode ray tube* or *CRT* is the primary device used to display digital images. Before the invention of transistors, chips and integrated circuits, electronic devices depended upon vacuum tubes as "electronic valves" and "switches". Since CRT's are simply highly specialized vacuum tubes, it will be useful to understand the basic concepts employed in vacuum tubes before studying studying CRT's.

3.1.1 Vacuum tube basics

A simple vacuum tube has four active elements enclosed in an evacuated tube, as shown in the schematic diagram of Figure 3.1. Wires extending to the base of the tube allow a voltage to be applied to the heater coil, and a second voltage across the cathode and plate. A control voltage can be applied to the grid. The device can be thought of as a "valve" or a "switch", since small changes in control voltage can produce a large change in current through the cathode/plate circuit. The tiny control voltage regulates like a valve, or can turn the current on or off like a switch. Here is how it works:

1. The cathode is negatively charged, giving it an excess of electrons
2. The heater heats the cathode, imparting energy that causes the release of some of the excess electrons
3. The free electrons are attracted to the positively charged plate, resulting in a current through the cathode/plate circuit

4. Since the electrical charge on the grid is nearer to the cathode than the plate is, it can prevent electrons from passing on to the plate or "encourage" them. A more positive grid voltage attracts electrons, increasing their acceleration towards the plate. A more negative grid voltage inhibits electrons.

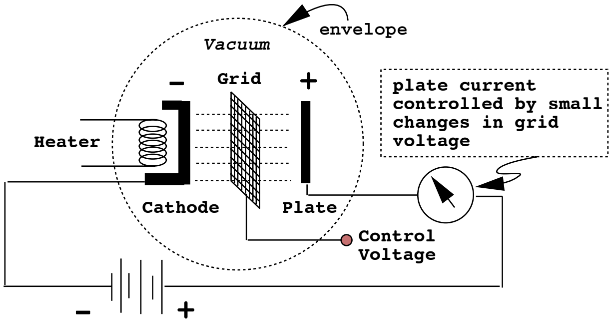

Figure 3.1: Schematic Diagram of Vacuum Tube

3.1.2 CRT's

A *CRT* or *Cathode Ray Tube* works on exactly the same principle as a simple vacuum tube but the internal organization is somewhat different. A schematic diagram showing the organization of a simple CRT is shown in Figure 3.2. As the electrons travel from cathode to plate they are focused into a beam and directed onto precise positions on the plate. In a CRT, the plate is a glass screen, coated with phosphor. The phosphor on the screen glows more or less brightly depending on the intensity of the beam impinging on it.

The flow of electrons from cathode to plate works like in a regular vacuum tube. However, focusing coils align the electrons into a beam, like a lens focuses light into a beam. Steering coils push the beam left/right and up/down so that it is directed to a particular spot on the screen. The grid control voltage adjusts the intensity of the beam, and thus the brightness of the glowing phosphor dot where the beam hits the screen.

A CRT can be used to display a picture in two different ways. The electron beam can be directed to "draw" a line-drawing on the screen – much like a high-speed electronic "Etch-a-Sketch". The picture is drawn over and over on the screen at very high speed, giving the illusion of a permanent image. This type of device is known as a *vector display*, and was

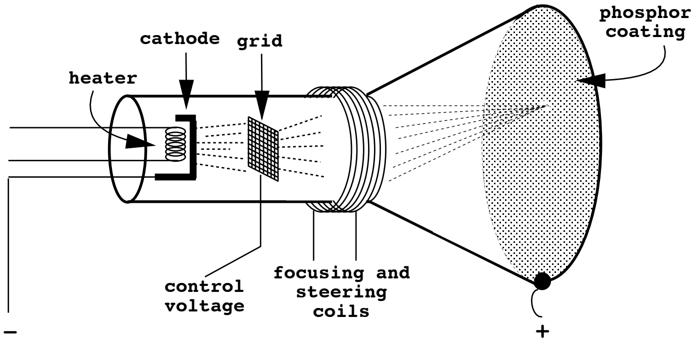

Figure 3.2: Schematic Diagram of CRT

quite popular for use in Computer Graphics and Computer Aided Design up until the early 1980's. By far the most popular type of CRT-based display device today is the *raster display.* They work by scanning the electron beam across the screen in a regular pattern of *scanlines* to "paint" out a picture, as shown in Figure 3.3. As a scanline is traced across the screen by the beam, the beam is modulated proportional to the intended brightness of the corresponding point on the picture. After a scanline is drawn, the beam is turned off, and brought back to the starting point of the next scanline.

The resulting pattern of scanlines is known as a *raster.* The NTSC broadcast TV standard that is used throughout most of America uses 585 scanlines with 486 of these in the visible raster. The extra scanlines are used to transmit additional information, like control signals and closed caption titling, along with the picture. The NTSC standard specifies a *framerate* of 30 frames per second, with each *frame* (single image) broadcast as two *interlaced fields.* The first of each pair of fields contains every even numbered scanline, and the second every odd numbered scanline. In this way the screen is *refreshed* 60 times every second, giving the illusion of a solid flicker-free image. In actuality, most of the screen is blank (or dark) most of the time!

Please see Foley, vanDam, Feiner & Hughes for many more details on CRT's.

A color CRT works like a monochrome CRT, but the tube has three separately controllable electron beams - we say it has three electron *guns.* The screen has dots of red, green and blue colored phosphors, and each of the three beams is calibrated to illuminate only one of the phosphor colors. Thus, even though beams of electrons have no color, we can think of the CRT as having red, green and blue electron guns. Colors are made using

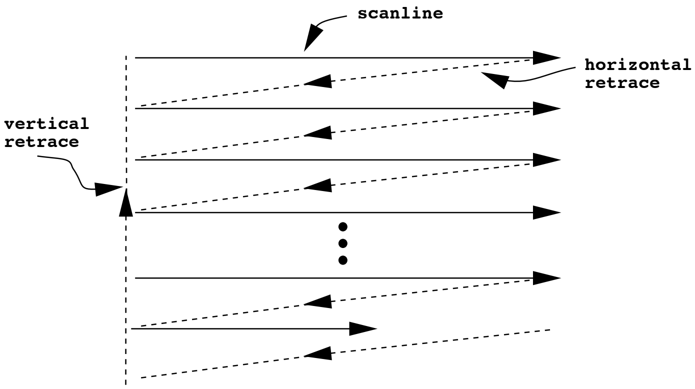

Figure 3.3: Raster Scan Pattern

the RGB system, as optical mixing of the colors of the adjacent tiny dots takes place in the eye.

Figure 3.4 shows the most typical triangular pattern or *triad* arrangement of phosphors on the back of the glass screen of color CRT screens. On the inside of the screen, an opaque *shadow mask* is placed between the three electron guns and the phosphors to assure that each gun excites only the phosphors of its appropriate color. High precision color CRT's, that require the ability to draw a fine horizontal scanline, use an *inline* rather than a triad phosphor arrangement, to keep the scanline confined to a more finely focused vertical area.

3.2 Framebuffers

A *framebuffer* is simply an array of computer memory, large enough to hold the color information for one frame (i.e., one screenful), and display hardware to convert the frame into control signals to drive a CRT. The simple framebuffer schematized in Figure 3.5 holds a monochrome (black & white) image in a bitmap. The circuitry that controls the electron gun on the CRT loops through each row of the image array, fetching each pixel value (1 or 0) in turn and using it to either turn on the electron gun for white or turn it off for black. Of course the timing has to be such that the memory fetches and conversion to grid voltages is synchronized exactly with the trace of the beam across the corresponding screen scanline.

Figure 3.6 shows a greyscale framebuffer with three bits per pixel, that uses a DAC (digital to analog converter) to convert numeric grey level to

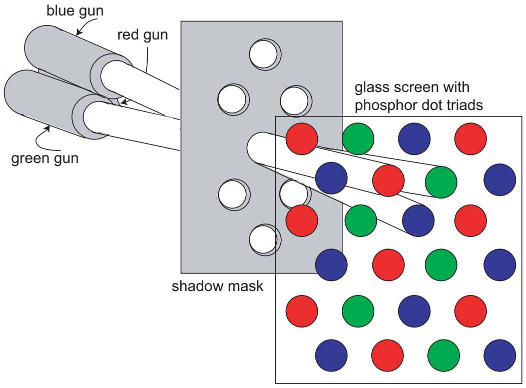

Figure 3.4: Phosphor Triad Arrangement in Color CRT's redrawn from Meko website, *The European Source for Display Data and Market Research*, http://www.meko.co.uk.

one of $2^3 = 8$ different voltages.

Figure 3.7 introduces the notion of a look-up table. Each pixel value from the framebuffer is used as an index into a table of 2^n entries (8 in the $n = 3$ example). Each table entry has a stored value whose precision is usually greater than the framebuffer resolution. This gives a palette of only 2^n colors, but the palette can be drawn from 2^m greylevels, where m is the number of bits per entry in the lookup table. One way this could be used in a greyscale framebuffer would be to correct for non-linearities in the display and in human perception so that each step in grey level would result in a uniform perceptual step in luminance level.

Figure 3.8 shows how a framebuffer can be arranged to drive a color display via three lookup tables. Virtually all 8-bit per pixel color displays – like the ones in older Macintoshes, PC's or workstations – utilize a 24 bit/pixel lookup table with 256 entries (2^8). This gives a palette of 256 colors per frame, but the colors are drawn from a selection of nearly 17 million ($2^{24} = 16,777,216$).

A full color resolution framebuffer, called a *truecolor* framebuffer, is shown in Figure 3.9. This type of device will have at least 24 bits per pixel (8 bits per color primary), either driving 3 color guns directly or (as shown in the figure) through a separate lookup table per color primary that can be used to correct nonlinearities or to obtain certain effects (like overlay planes). Very high end graphic displays may have more than 24 bits allocated per pixel, to handle such tasks as color compositing, depth-buffering for hidden surface resolution, double buffering for real-time animated dis-

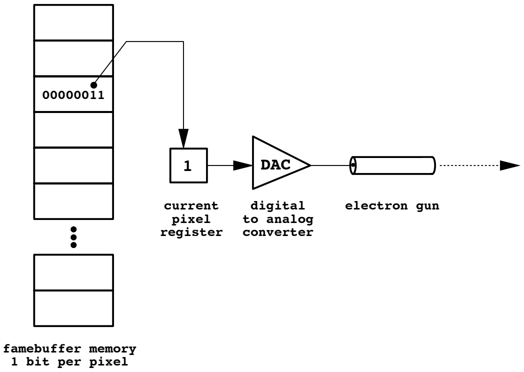

Figure 3.5: Monochrome Framebuffer

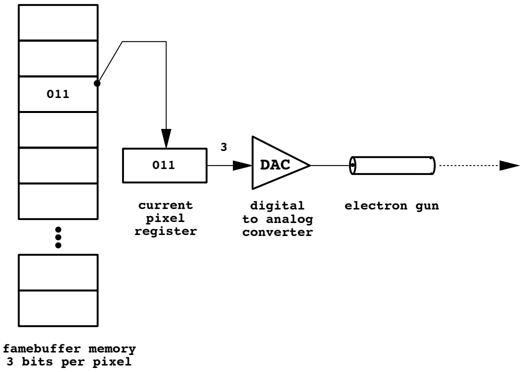

Figure 3.6: 3-Bit Per Pixel Greyscale Framebuffer

play, and overlays.

3.3 Gamma Correction

So far we have treated the transfer of information from the framebuffer to the CRT as if it were linear. For example, in an eight-bit grey-scale framebuffer without a lookup table, pixel values can range from 0, representing black, to 255, representing white. Thus, we would expect that a pixel value of 127 would represent a middle grey, and in that even increments of pixel values would represent even increments of grey. In fact, this is not the case.

There are two reasons why even increments of pixel values do not re-

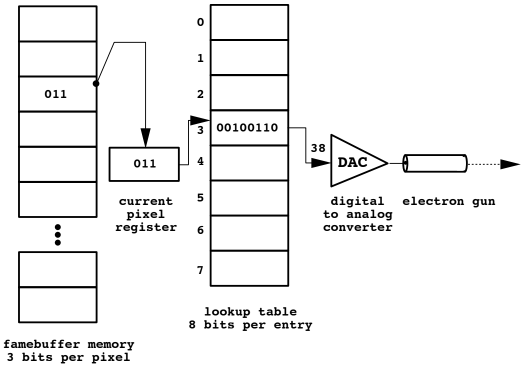

Figure 3.7: Greyscale Framebuffer with Lookup Table

sult in even increments of perceived grey. The first is that the human eye registers equal increments of color intensity, not as a function of the difference between intensities (luminance), but as a function of the ratio between intensities. In other words, if $I_1 < I_2 < I_3$ are three intensities, the step between I_2 and I_1 would look the same as the step between I_3 and I_2 if

$$I_2/I_1 = I_3/I_2,$$

but the steps would look unequal if

$$I_2 - I_1 = I_3 - I_2.$$

Because of the ratio law, in fact, the step between I_2 and I_1 would appear to be greater than the step between I_3 and I_2. This relationship is shown in Figure 3.10, which plots perceived intensity I_p in dimensionless units versus actual intensity I_a on a unit scale. The solid line indicates how perceived intensity will vary with actual intensity, and the dashed line shows how perceived intensity would vary if the relationship were linear.

In practice, the story is further complicated by the fact that the relationship between CRT grid voltage and phosphor luminance is also non-linear. The actual phosphor luminance I_a due to grid voltage I_v is given by a curve of the form

$$I_a = I_v^\gamma,$$

where γ is a positive constant greater than 1. This relationship is shown in Figure 3.11 for a CRT with $\gamma = 2$. For most CRTs in common usage, values of γ range between 1.6 and 2.4.

If we take both the nonlinearity of the CRT and the nonlinear perceptual response into account, the two effects interact to make a more complex relationship. This effect is shown in Figure 3.12.

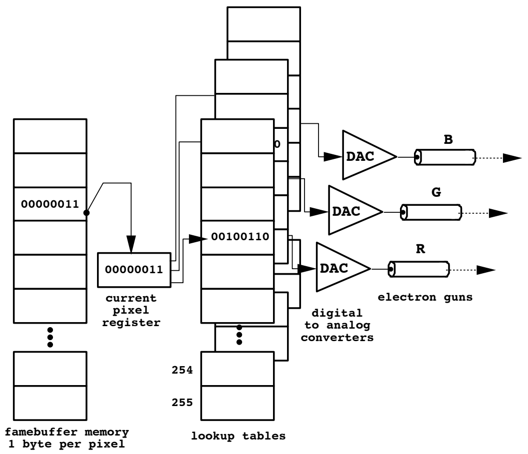

Figure 3.8: 8-Bit Color Framebuffer with 3 Lookup Tables

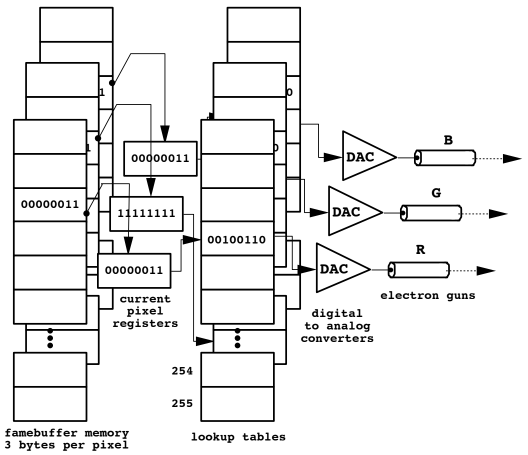

Figure 3.9: Truecolor Framebuffer with Three Lookup Tables

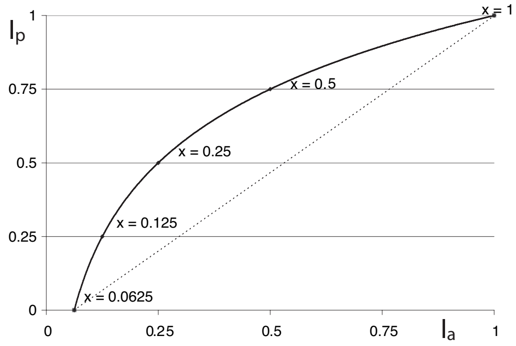

Figure 3.10: Perceived Intensity vs. Actual Intensity

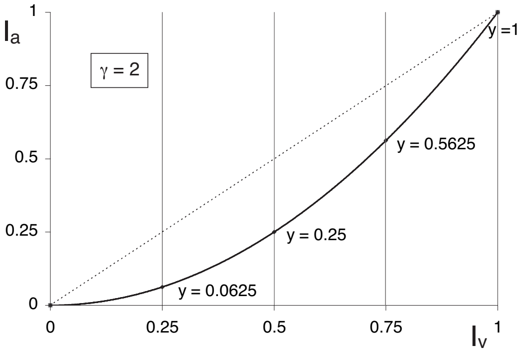

Figure 3.11: Actual Intensity vs. Stored Intensity (voltage), gamma = 2

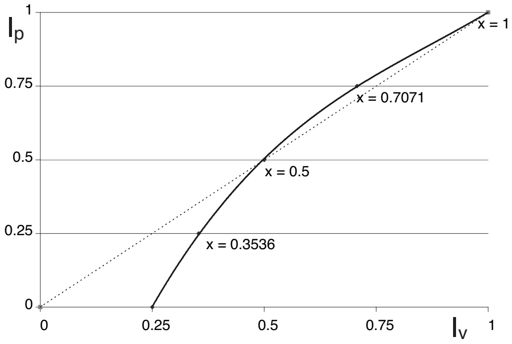

Figure 3.12: Perceived Intensity vs. Stored Intensity (voltage)

The process of correcting for the nonlinearities introduced by the nature of human perception and the physics of CRTs is called *gamma correction*. If we did not have to take the perceptual issue into account, gamma correction would simply consist of scaling each pixel value to the scale 0...1, raising it to the power $1/\gamma$, and then restoring it to its original scale before using it to determine grid voltage. Actually, the problem is complicated by the fact that perceived intensity varies as the ratio of actual intensity. In most cases a suitable value for γ can be found experimentally, that will give an even perceived gradation across the full range of greys. However, the γ value used for gamma correction and the γ of the CRT will differ somewhat.

Gamma correction can be done on a per-pixel basis at the time of display, replacing a stored image with a gamma corrected image. On a framebuffer with a lookup table, however, the lookup table can be loaded with gamma corrected intensity values, that are simply indexed by the colors in the image. This approach has the virtues that it need only be done once, even if we are displaying a sequence of images, that it requires much less computation even for a single image, and that it does not require modifying the image itself. When we study compositing, we will see that pre-gamma correcting images is a very bad idea if we plan on doing any work on the image that requires adding or averaging pixels. We will see that this includes compositing, spray painting, filtering and many other common operations.