



# Tech Info Library

## Grayscale Information

Article Created: 23 April 1989

Article Last Reviewed: 6 July 1992

Article Last Updated:

### TOPIC -----

A user has found (using very sophisticated photometric devices) that when he generates a monochrome palette from a color palette, some of the tones are identical. This article discusses this phenomenon.

### DISCUSSION -----

In regard to using the color palette, the Palette Manager does not place new colors into the display device's color look-up table (CLUT). It maps the requested colors in a palette to available colors within the current color table. The correct method for using specific color values is to use the Color Manager calls to create and load a programmed color table.

There are some other points that must taken into account after setting your colors with the Color Manager.

- It is possible to produce grays by setting the monitor to monochrome output and not setting grays, as the user in question probably does. When selecting monochrome from the monitor's Control Panel CDEV, luminance mapping of the colors in the color table is done to calculate and replace the RGB values for a monochrome pixel. The luminance mapping follows this equation:

$$\text{Luminance} = .3 * \text{red} + .59 * \text{green} + .11 * \text{blue}$$

The resulting value is placed into the display device's table for R, G, and B. The values are altered according to the above equation. This needs to be taken into account if the monochrome option is chosen for the Apple High-Resolution Color Monitor.

- If you place your own RGB "gray" values into a color table, you must also take into account gamma correction. Gamma correction is done by the display device's driver to correct the disproportionate light output of the CRT phosphors by comparison with the proportional input voltages of the video signal. To set a linear relationship between colors and output, the values set for the Macintosh II Video Card are changed to new values,

which then produce the corrected color output on the monitor. Test equipment detects a gamma-filtered set of values from the monitor.

The gamma correction is automatically handled for all values written to the device's color table. This is based on an empirically-derived gamma look-up table. If the selected values fall within the same look-up table reference, they can be mapped to the same gamma value. This would explain why some values are the same. The gamma table ID is located in the SCRN resource of the System file for the Macintosh II Video Card. A new GAMA Resource table can be defined and installed in the System file for use by the Macintosh II Video Card. To use the new gamma table, the GAMA resource ID must be changed in the SCRN resource to that of the new table.

The safest values for any testing will probably be a linear table that does no gamma correction. A quick and easy way to do this is to pass -1 to the SetGamma Control parameter block as the new table address. This causes a linear table to be used, guaranteeing that all of the possible gray levels of the card are available. At that point, the actual color output on the monitor depends on the phosphors themselves, which have a variance from monitor to monitor of +/- .02 for the red and green phosphors and +/- .015 for the blue. The CIE coordinates of the phosphors are:

Red: x=.625 y=.34 Green: x=.28 y=.595 Blue: x=.155 y=.070

To properly set the gamma table, see the information below about how the driver uses this table. You should also investigate how to control device drivers. This is described in "Inside Macintosh Volume II." Another source of information is the "Designing Cards and Drivers Manual for the Macintosh."

#### Gamma Information

-----  
In the current implementation of the video drivers, gamma correction is applied to requested, absolute colors, immediately before they are set in the color look-up table by the SetEntries control call. More specifically, some number of high-order bits are extracted from the red, green, and blue channels and used as an index into tables of corrected values. These values are then placed into the hardware, yielding corrected output. On the Macintosh II Video Card (the TFB card), the high eight bits of each channel are used to reference the gamma table.

There are a number of minor shortcomings in this implementation. First, there is not absolute symmetry between the SetEntries control call, which sets the CLUT, and the GetEntries status call, which reads the CLUT hardware, because the gamma correction took place as part of the SetEntries call. Also, the uncorrected values are generally unrecoverable (although a copy of the absolute colors are always available in the GDevice structure).

Finally, it is most desirable to extract more bits as an index to the gamma table than the number of bits of color information that will be set in the CLUT. This way, you avoid a loss of color resolution after correction. For example, the TFB card has an eight-bit-per-channel CLUT,

but only uses the most-significant eight bits of the (16-bit) channel information for gamma lookup. At lower intensities, the gamma correction increases the distance between adjacent values. As a result, on the TFB card, some dynamic range is lost at lower intensities. This could be corrected by extracting nine or ten bits of channel information rather than eight and using a larger, gamma-correction table, but this option was declined to reduce gamma table size.

#### The GammaTbl Data Structure

-----  
The structure itself has been a bit of a mystery, as it is not defined in either "Inside Macintosh Volume V" or the Designing Cards and Drivers Manual. This is the structure:

```
record GammaTable of
  gVersion: integer; {gtab version, currently 0}
  gType: integer; {drHwId value}
  gFormulaSize: integer; {size of formula data, below}
  gChanCnt: integer; {# of component channels}
  gDataCnt: integer; {# of values per channel}
  gDataWidth: integer; {size of data in tables}
  {gamma correction look-up tables}
  gFormulaData: array [0.. gFormulaSize] of byte;
  {data for gamma calculation formula}
  gData: array [0.. gDataCnt] of byte;
end;
```

In this structure, gVersion represents the gamma table format version, which is 0 for all current cards. The gType field holds the drHwId value for this board to identify the board that this table was measured for. Note that this means that a single gamma table can't directly be shared between two different cards, even if they both have the same CLUT response curve (which is usually linear). This lets the data in the gamma table be in an appropriate form for varying hardware (that is, a card could have four-bit/channel DACs and might prefer gamma data in the range \$0..\$F rather than \$0..\$FF).

gFormulaSize defines the number of bytes occupied by the gFormulaData field. On Apple's current video cards, gamma correction is done by modifying the values loaded into the CLUT by the SetEntries control call to approximate linear response on the display. On these systems, the gamma correction table acts as a final, look-up, data table, which translates the requested color into closest available linearized level. These gamma table values are determined empirically by measuring the output of a calibrated display.

More sophisticated systems may choose an alternative to this simple look-up mechanism. For instance, you can calculate gamma correction factors based on a mathematical response function. By default, the TFB card uses a single correction table for all three channels. No calculations are performed on the incoming color table other than simple lookup. Cards can remember the specific monitor configuration at the beginning of the gFormulaData field, allowing it to identify and use only

gamma tables developed for the currently connected monitor.

gChanCnt is the number of look-up tables in gData, below. If there is more than one channel of gamma correction data, then the R, G, and B tables follow each other, respectively, at the end of the structure.

gDataCnt is the number of discrete look-up values included in each of the channel's correction table. It is always equal to 2gDataWidth, but refers to number of bytes that this channel's data occupies.

gDataWidth describes the number of significant bits of information available in each entry in a channel's correction table. The data always appears as gDataWidth bits, right-justified in a field that is the next larger number of bytes than gDataWidth. Because it is rare to have devices with more than eight bits of CLUT resolution, virtually all devices pack their correction data into bytes.

gData is actual correction table data. If there is more than one channel's information, each table follows the next in R, G, B order. The standard tables included in Apple's driver have only one table, which is applied to all three output channels. Because Pascal cannot express variable size fields in record structures, the independent channels are not individually named.

#### "gama" Resource Format

-----  
In addition to the RAM data structure for gamma tables covered above, there is a standard resource format for gamma table resources. Like many other resource templates, the gamma structure is an image of the RAM form stored in resource format. There are no changes.

#### Using Gamma Correction

-----  
Gamma correction is always applied by the TFB video driver. At driver open time, the driver is usually initialized with a linear (noncorrecting) gamma table. When \_InitGraf is called, the "scrn" screen configuration resource is read from the System file. This resource (described in "Inside Macintosh Volume V") includes information about the size and orientation of the different monitors configured into the system, including their last video mode (pixelsize), color table, and gamma table.

If there is no "gama" resource ID specified, or the specified ID is not present, then a default gamma table, "gama" =0 is loaded from the System file and used (this is the table calculated for the TFB card). If the specified resource is found, then the appropriate resource is loaded, and a control call is issued to the driver to make this the current gamma table. Unfortunately, there is currently no tool to set the "gama" ID, short of modifying the "scrn" resource directly.

To facilitate the use of the gamma table, there are two calls in the standard, video-driver routines that set the gamma table (control call 4, SetGamma) and retrieve the pointer to the current gamma table (status call 6 on TFB rev 2 drivers and up). These calls simply take and return a

pointer to a GammaTbl structure.  
Copyright 1991 Apple Computer, Inc.

Keywords: <None>

=====

This information is from the Apple Technical Information Library.

19960215 11:05:19.00

Tech Info Library Article Number: 3867