More Colors for Your Apple

Allen Watson III
430 Lakeview Way
Redwood City CA 94062

The sales literature for the Apple II lists the specifications for the high resolution color graphics mode this way:

- 280 horizontal by 192 vertical resolution.
- Four colors: black, white, violet, and green.
- Displays 8 K bytes.

The specifications don't sound all that exciting. The resolution seems about right, but why are there so few colors? And why did they pick green and violet instead of, say, red and blue? Well, as it happens, the colors in the Apple II high resolution graphics can be red and green, or blue and yellow, or almost any two complementary colors you want. What's more, on many color television sets you can obtain as many as four colors along with black and white, as demonstrated by the accompanying photograph.

The classic approach to computer generated color is to generate separate signals for the red, green, and blue inputs of a color monitor. However, color monitors are expensive; it's more economical to use an ordinary color television set. Now instead of generating three simultaneous video signals, we have to generate a composite signal that resembles the standard broadcast signal the television set was designed to receive.

It's not merely that the signal has to be put onto a regular television channel by means of a radio frequency modulator; although that's certainly necessary, there's a lot more to it. Since all the fascinating features of the Apple II high resolution color graphics are the results of the way the Apple II designers solved this problem, let's take a look at just what they did.

The Color Signal

The standards for broadcast color television signals were established by NTSC (National Television Systems Committee) and approved by the Federal Communications Commission in 1953. In order to retain the existing system of black and white television broadcasting, the committee system adds color information to a signal which is practically identical to the black and white standard. The resulting composite signal includes a black and white component that amplitude modulates the television carrier frequency in the usual way, and a color component which rides on a 3.58 MHz subcarrier.

This superposition of color and black and white information is necessary in order to crowd a full color video signal into a channel whose high frequency response is limited to just over 4 MHz. The fact that human vision does not resolve image details in color allows us to limit the resolution of the color component of the signal to a maximum of 1.5 MHz. In fact, only part of the color signal gets even this much; the rest is limited to 0.5 MHz.

This narrow band color signal modulates a 3.58 MHz subcarrier which is then added to the black and white picture information. The color subcarrier modulation is a combination of amplitude and phase modulation: the amplitude of the subcarrier corresponds to the amount of color at each point on the screen, while the choice of color is determined by the phase of the color frequency relative to a 3.58 MHz reference signal. This reference signal is generated in the television set from a burst of 3.58 MHz transmitted in the interval between the lines of the picture.

A high subcarrier frequency reduces interference between the color and black and white components because the black and white signal contains less energy at high frequencies. Interference is further reduced by the fact that the subcarrier frequency is an odd multiple of half the picture scanning rates, both horizontal and vertical. This makes any color signal that gets into the black and white video reverse polarity on successive lines; the interference makes little dots in the picture, but the dots on one line will have "undots" above and below. These will tend to average out when viewed from a reasonable distance.

This is where the signal generated by the Apple II deviates radically from the standard signal. First of all, the Apple II signal omits a technique called interlacing, thus reducing the number of horizontal scanning lines by half and likewise the amount of information needed to fill the screen. Noninterlacing is common among low cost computer video displays. The significant deviation from the standard, however, is a slight change in the horizontal and vertical scanning rates such that the interference between the color and the black and white components is maximized, rather than minimized. This is not as strange as it sounds, because this is what en-

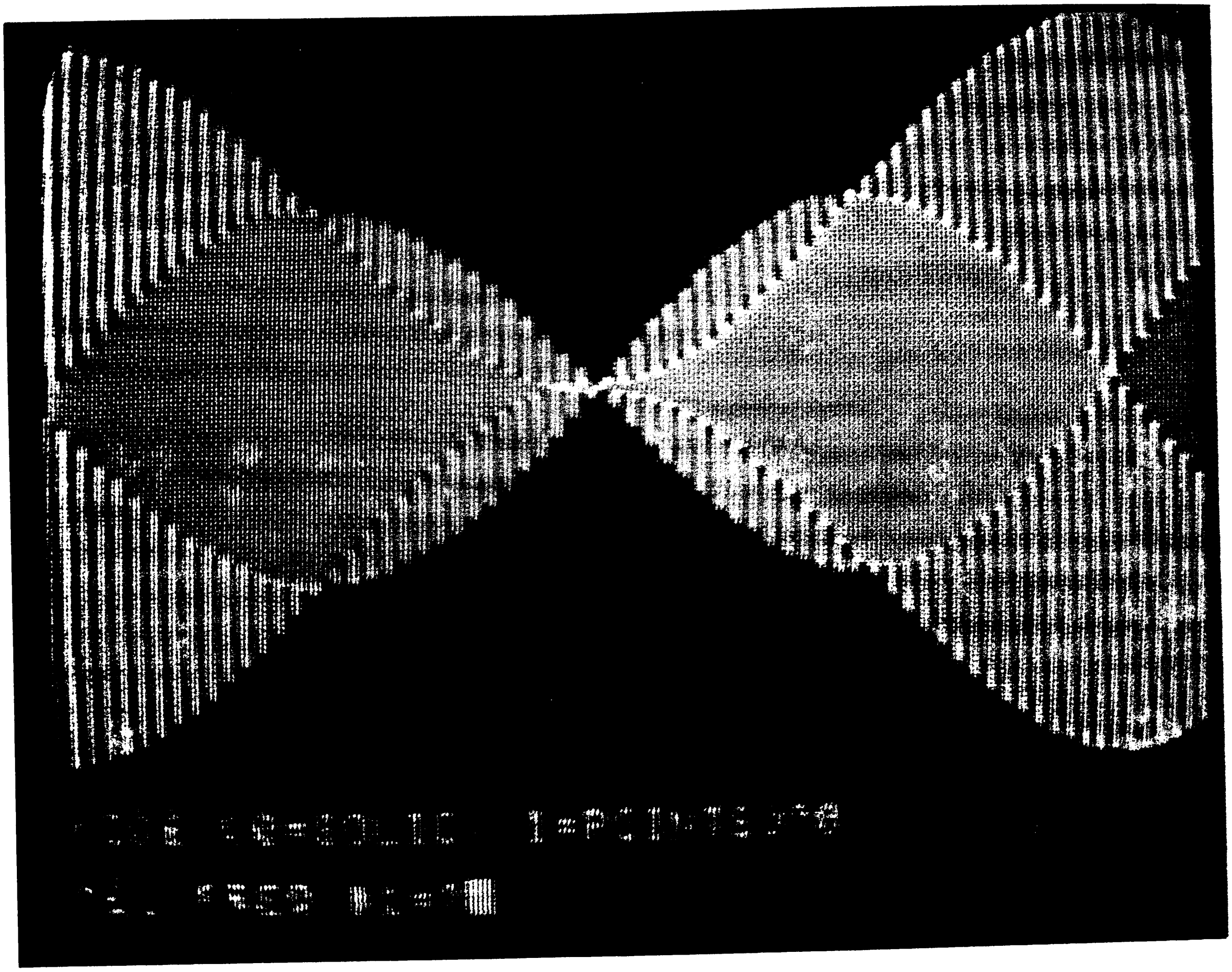

Photo 1: Apple II display showing four colors in high resolution mode.

ables the Apple II to generate color graphics with a signal made up only of ones and zeroes.

An Example

To see what this does to our display, suppose we try to display two small white dots side by side. The smaller the dots and the closer they are to each other, the higher the highest frequency picture signal going into the television set. But everything the Apple II puts out at the high end of the frequency range gets decoded as color, so that, even before our dots are made too small and too close together for a black and white set to be able to distinguish, something else has happened: they have merged into a single dot, and it isn't white, but color.

In other words, the resolution we can get using this method is somewhat limited compared with the separate red, green, blue approach, although it is produced with correspondingly less screen memory (ie: 8 K bytes compared with 24 K bytes). But even if the color interference were minimized, the television set's receiving circuits limit the horizontal resolution to about 300. Incidentally, this is why the Apple II displays only 40 characters in each line of text; the more popular 64 or 80 characters cannot be resolved by a standard color television.

Bits and Resolution

As we have seen, the Apple II produces color by simply putting its smallest dots at the right size and spacing: namely, the color subcarrier frequency. Each dot is really a half cycle, so the dot rate is twice the subcarrier frequency, or something over 7 MHz. Let's see how many of these dots will fit on one horizontal line. There is one horizontal scan every 63.5 μs, but part of this time is needed to get the electron beam into position to start the next line, and to keep the lines in synchronization. The picture signal is shut off, or blanked, during this time. That leaves about 45 μs, but just to play it

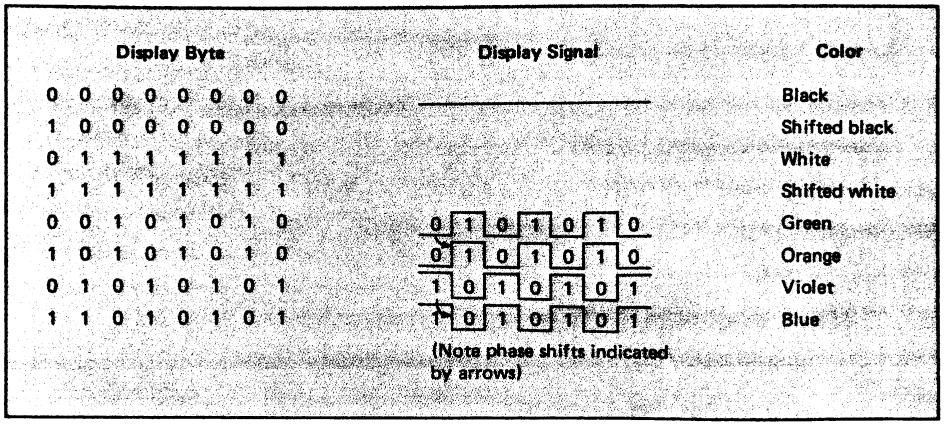

Figure 1: Colors produced by various bit patterns in relation to the color reference signal.

safe and to make sure that none of our valuable data gets cut off by the television set's normal overscan (the picture is set up to be bigger than the actual screen so there won't be any unsightly black borders), Apple II uses only about 40 μs of each line for data. This works out to 280 dots per line. In text mode, with 40 characters per line, this gives a character time of about 1 μs, which corresponds to the Apple II's system clock. Each character takes seven dot times, five for the character and two for spacing between characters.

The question is, exactly what does horizontal resolution of 280 refer to? Well, we can put a single dot at any of 280 different positions across the screen, but our dot will be colored, since it is a half cycle at the color subcarrier frequency. And if we put two dots too close together, they merge. Obviously, if the two dots are actually touching, no set could resolve them — this is really a single spot which happens to be two dots wide. But even if we put a black dot in between, we'll see only one dot, in color, because the dot spacing matches the color subcarrier frequency. Only by putting two or more black dots between our white dots will we be able to see a clear separation.

These relationships are diagramed in figure 1. The color reference signal is shown at the top. Any signal component at this frequency, even a single dot, will be displayed as colored. Theoretically, a double width dot contains no color frequency component, and hence will be displayed as white.

Apple II High Resolution Colors

Now we can see how the trade-off between color and resolution affects the way our computer bits are displayed by the television set. But let's look on the bright side: with the right bit patterns, we can put colors onto the screen. Let's ignore the resolution problem for a while and investigate the colors.

If we fill the screen memory with ones, the display will be all white; all zeroes paints it black. If we alternate ones and zeroes horizontally, we have a signal which is right at the color frequency, so it is displayed as a solid color. Now comes the interesting part — what color is it? As I mentioned earlier, the color is determined by the phase of the picture signal's color frequency component relative to the color reference signal, which is generated by the television set from the 3.58 MHz color burst which we transmit during the horizontal blanking interval. So our question becomes, "How can we control the relative phase of these two signals?"

First of all, our computer bits are output every half cycle of the color reference frequency. This means we can change the phase by 180 degrees by simply inverting the bit pattern so that alternating ones and zeroes become alternating zeroes and ones. Interestingly enough, since the color spectrum is allocated the 360 degrees of possible phase angles that we can have, complementing the bits also complements the color; that is, phase inversion amounts to 180 degrees of phase shift, and complementary colors are 180 degrees apart. The relation of color to phase angle is shown in figure 2. If the alternating bits are in phase with the color reference signal, the color will be yellow-green; out of phase bits will give us blue-violet. This determines the two colors Apple II specifies in addition to black and white. But there is another way to change the relative phase of our computer bits.

While we can't do this under computer control, we can manually adjust both the Apple II video circuit and the color television set so as to change the phase of the color reference signal itself. The Apple II control is labeled *color trim;* the television set's control for this is usually called *tint* or *hue.* The combined range of adjustment of these two controls is usually enough to go at least halfway around the color circle of figure 2, putting one or the other of our complementary pairs of colors at any point on the circle. Thus we can adjust for any pair of complementary colors we want: blue and yellow, green-blue and orange, cyan and red, green and magenta, or yellow-green and violet. So long as we don't require the ultimate in horizontal resolution, we can have any two complementary colors plus black and white for our high resolution graphics using only ones and zeroes as data. If the colors listed above and in figure 2 don't seem exactly complementary, it's largely because of the broad range of hues to which we carelessly apply the name *blue.* If we let the television picture-tube phosphors define our red, green, and blue, then the complementary colors are those of figure 2. The television set is adjusted such that red + green + blue = white. Since complementary pairs also add together to give white, it follows that the sum of any two of the three primaries gives the complement of the third: for example, the complement of red is actually green + blue, or cyan.

Extra Colors

Studying the Apple II specifications in the light of the National Television Systems Committee color standards led me to expect it to work this way, but that isn't quite the end of the story. After I got my Apple II, as I was casually watching the random sine wave program on the high resolution demonstration tape, there in living color was a display with *four* colors. After a bit of head scratching and experimenting with the adjustments on my portable color set, I think I have the explanation.

First of all, the single dot patterns give the two complementary colors, just like it says in the script. Alternating double dots, which ought to be displayed as black and white, actually show up as a weaker version of the same pair of complementary colors if the television set is adjusted normally, that is, with the fine tuning just backed off from the setting that first produces sound bars in the picture. But if I back the fine tuning farther away from this setting (any automatic fine tuning or tint controls should be switched off), just before the color signal drops out, the weak colors on the double dot patterns brighten and shift to another pair of complementary colors. The exact colors depend upon the setting of the tint control, but they are more than 30 degrees from the first pair, so if the single dot patterns give red and green, for example, the double dot patterns appear as orange and blue.

It's hard to figure out how the double dot patterns get displayed in color since they are square waves at half the color frequency and ought to contain a zero component at 3.58 MHz. Apparently the video detector circuit in the set produces enough second harmonic distortion to activate the color circuits. Mistuning puts this signal near the cutoff of the color bandpass filter where there is maximum phase distortion. I tried this out on the more expensive television set at the store where I bought my Apple II, and although it's more difficult to get the adjustments just right, the extra colors are there. Ironically enough, this trick seems to work better on cheap sets.

So there you have it. Whether you prefer colors or resolution, the Apple II high resolution graphics will put out all you can get through the antenna terminals of a color television set with just different patterns of ones and zeroes. To find out what your set will do, you need to display vertical lines with the single dot and double dot patterns. An easy way to do this is to load the Apple II high resolution demonstration tape and select the program that sums two sine waves. When the program asks for two frequencies, enter 63 and 64 to get the pattern shown in photo 1. Other numbers you may want to try are combinations of 31, 32, 33, 63, 64,

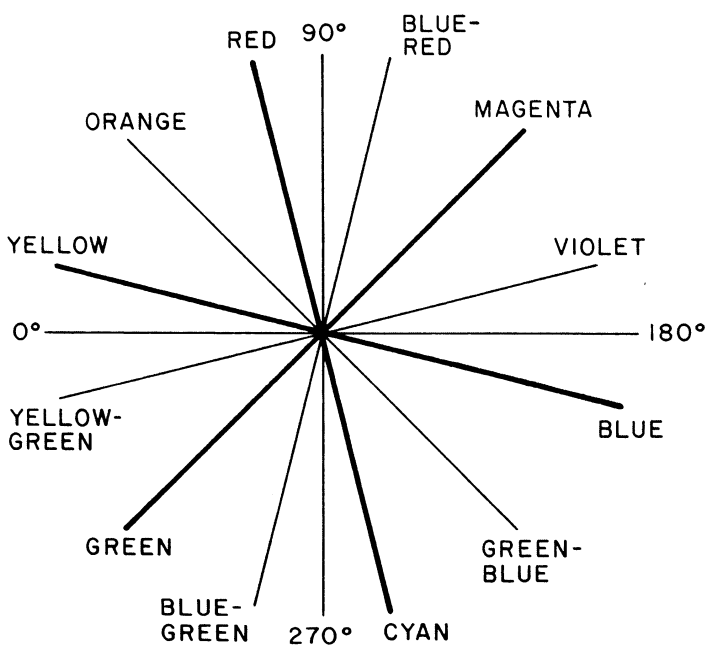

Figure 2: Arrangement of possible colors versus signal phase. The Apple II high resolution graphics outputs two complementary colors (colors that are separated by 180 degrees on the circle).

65, and 95. Apparently there is a lot of sampling error when the frequencies you select don't fit the table the program uses to generate the sine waves. If you experiment until you find the limits of your particular television set, you'll know how to make high resolution pictures on your Apple II in just about any colors you want.

BIBLIOGRAPHY

1. *Apple II Reference Manual,* Apple Computer Inc, Cupertino CA, 1978.
2. Herrick, Clyde N, *Color Television: Theory and Servicing,* Reston Publishing Co, Reston VA, 1973.
3. Schure, Alexander, *Basic Television,* revised second edition, volume 6, Hayden Book Co, Inc, Rochelle Park NJ, 1975.

Addendum

The following comments were received from Steve Wozniak of Apple Computers:

Thank you for passing along Allen Watson's article on the Apple II high resolution colors.

As Allen discussed, Apple II high resolution colors are the result of alternating zeroes and ones on the screen. The exact colors generated depend on the phase (or timing) relationship between the display signal and the color reference phase. By adjusting the television controls, any desired color pair may be displayed.

Oddly enough, only the seven least significant bits of the Apple II high resolution refresh memory bytes are used (examples are shown in figure 1). A simple modification allows the high order bit of each to specify one of two color sets by generating a 90 degree phase shift of displayed information. (Yet more colors may now be obtained by applying the technique suggested by Allen.)

Adding the High Order Bit Modification to the Apple II

1. Remove the Apple II printed circuit board from its enclosure.

(a) Remove the ten screws securing the plastic top piece to the metal bottom plate. Six of these are flat head screws around the perimeter of the bottom plate and four are round head screws located at the front lip of the computer. All are removed with a Phillips head screwdriver. Do not remove the screws securing the power supply or nylon insulating standoffs.

(b) Lift the plastic top piece from the bottom plate while taking care not to damage the ribbon cable connecting the keyboard to the printed circuit board. This cable will have to be disconnected from one or the other.

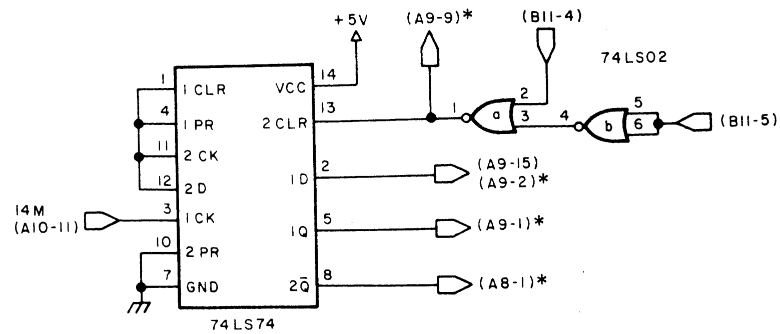

*Figure 3: Circuit which must be added to the Apple II to add more colors to the high resolution display. (***Caution**: *Adding this circuit voids the warranty.) A starred assignment (*) indicates that the connection is made to a pin which is out of its normal socket. Besides the connections which are shown, also connect pin (B8-14) to (A8-6) which is out of its socket, and (B8-7) to (A8-13) which has also been removed from its socket. The power connections to the 74LS02 are +5 V to pin 14 and ground connected to pin 7.*

(c) Disconnect the power supply from the printed circuit board.

(d) Remove the #8 nut and lockwasher securing the center of the printed circuit board. These will not be found on the earlier Apple II computers.

(e) Carefully disengage each of six nylon insulating standoffs from the printed circuit board (seven on earlier versions).

(f) Lift the printed circuit board from the bottom plate.

2. Above the board wiring method.

(a) Lift the following IC (integrated circuit) pins from their sockets.

IC	Pin Number
A8	1
A8	6
A8	13
A9	1
A9	2
A9	9

(b) Mount a 74LS74 (dual C-D flip-flop) and a 74LS02 (quad NOR gate) in the Apple II breadboard area (A11 to A14 region).

(c) Wire the circuit in figure 3.

3. Below the board wiring method.

(a) Desolder all pins of socket A8. Lift the socket and its 74LS257 integrated circuit off the printed circuit board taking care not to destroy it. Cut the trace between pins 6 and 13 of A8 on the top side of the board. Also cut the trace between pins 13 and 15 on the top. Reinsert socket A8 and the 74LS257. *Be careful.*

(b) Cut traces going to the following pins on the bottom of the Apple II board. Each pin should have a single trace going to it. *Be careful.*

IC	Pin Number	IC	Pin Number
A8	1	A9	1
A8	6	A9	2
A8	13	A9	9

(c) Connect pin 15 of ICA8 to ground (pin 8 of ICA7 on the keyboard socket is a nearby ground).

(d) Mount the 74LS74 and 74LS02 as per step (b) of the above the board wiring method.

(e) Wire the circuit of the above the board wiring method, step (c). All wires are on the bottom of the Apple II board and no pins need be removed from their sockets or soldered to.

4. Reassemble the Apple II and make sure it is operational. If not, check all wiring very carefully. Make sure that all integrated circuits are in their sockets and properly oriented.

5. The following color values are now applicable to the high resolution subroutines:

BLACK2	128
ORANGE	170
BLUE	213
WHITE2	255

For example, the program below draws an orange line from location (10, 20) to (200, 140). It is assumed that the high resolution routines are already in memory locations hexadecimal 800 thru BFF.

```
 0 X0 = Y0 = COLR
 5 INIT = 2048 : PLOT = 2830 : LINE = 2836
 7 ORANGE = 170 : CALL INIT
10 X0 = 10 : Y0 = 20 : COLR = ORANGE :
   CALL PLOT
20 X0 = 200 : Y0 = 140 : CALL LINE
30 END
```
■