Mr. Kaypro

By Charles B. Stafford

Regular Feature

Kaypro Support

Composite Video

84 Kaypro external video

Wherein we devise a magic spell to allow external video for K-10s and '84 K-2s & 4s. This will allow you to use a full size monitor instead of the rather small but usable built in screen.

Gather 'round all you "apprentices" and "neophytes" and ye shall hear of a project which can be as simple or as sophisticated as you wish to make it. It requires only 13 parts, all easily obtainable (even at Radio Shack) for somewhere around $10.00 total. The real "magic" is in the spell, for which we'll use Mbasic. You can use either a "prototype" printed circuit board, or build your own custom PCB, using the pattern included herein.

THEORY

Until recently the most easily obtainable external monitors have "composite" signal inputs, usually an RCA jack with one pin. Several signals are transmitted over this single conductor and the challenge is to create a device that will combine all these signals without affecting the original display. The signals involved are horizontal sync, vertical sync and video.

Inside the '84 K-2s,4s & 10s and the '83 K-10, the generation of the video signal is handled by a 6845 video controller chip. The sync signals come from an oscillator through a divider network. All three signals are routed through a 7406 buffer (U1) on the way to the video board. The video signal leaves U1 on pin 6, while the buffered vertical sync appears on pin 2 and the inverted horizontal sync is on pin 11. Since all the signals go through the same chip, we can remove that chip and plug in our circuit there. Fortunately power is available there as well (pin 14) so this will be the only connection we'll have to make to the Kaypro. By using a wire wrap socket on our PCB, we'll be able to plug the extended legs right into the U1 socket vacated by removing the 7406. We'll use a 74LS00 (see Fig 1) to buffer the signals to avoid affecting the original display, sections C & D of which will buffer the video signal. R1 & R2 form a voltage divider: when the video signal is high (producing white on the screen), the voltage at the base of Q1 goes close to 5 volts to turn on the transistor. When the video signal goes low, R1 & R2 divide the voltage, reducing it to about 2.5 volts, providing a black level on the screen. Sections A & B allow either the Vertical or Horizontal sync to reduce the base voltage to a diode drop below 0 via CR1, turning off Q1. At the same time CR2 speeds up the turn off time of Q1, and CR3 guarantees the base will be a diode drop below the voltage at the emitter, ensuring that Q1 is off. Using a 2N3904 transistor and the resistor values shown should produce a white level of about 2 volts, a black level of about 1 volt and sync levels in the neighborhood of 1/4 volt. These signal levels seem to satisfy most monitors.

CONSTRUCTION

I used a piece of prototype board for the first one I made. It was one of those that had two columns of 4 hole pads spaced just far enough apart that a socket would straddle the space between with one leg each in a 4 hole pad. I put the wire wrap socket in first, a regular socket following that and the transistor socket on the left side below the second socket. That made it easy to wire using jumpers and putting the diodes in from pad to pad. A friend of mine thought he could do better, and laid-out a single sided PCB, the pattern for which has been included, should you wish to etch your own. Where the schematic says "out", connect a two conductor piece of wire long enough to reach the video connector which will be mounted in one of the cooling slots just beneath the mother-board. I used a piece of small shielded coax out of the junk box, but the shielding isn't really necessary, since the distance is short.

THE MAGIC

After building the external adapter, and double checking the circuit against the schematic, plug the adapter into the host machine (U1), hook up your external monitor and turn the whole works on. The odds are that the monitor has been adjusted for standard video timing and the display will be rolling and torn.

Commercial television signals use a 60 Hertz vertical sync as do the 83 K-2s & 4s. The 84 K-2s & 4s and all the K-10s, however, use a vertical sync of 51 Hertz so we'll have to adjust the monitor's vertical hold.

Commercial television signals also provide a horizontal sync signal every 63.5 micro-seconds (corresponding to a sweep frequency of 15,750 Hertz) while the 84 Kaypro provides one every 47.56 micro-seconds (corresponding to a sweep frequency of 21,028 Hertz). Adjusting the monitor's horizontal hold will help solve the tearing and may, in fact, cure it completely. If so, you're finished with the project.

Many monitors, however, cannot handle the higher frequency, even with the horizontal hold turned as far as it will go, BUT since the '84 Kaypros use a 6845 video controller, the horizontal frequency can be adjusted using software.

HOW WE DO IT

The 6845 has several registers (storage locations) which can be loaded with data to control its operation. Register 0 controls the horizontal sweep rate and contains a value equal to the total number of characters on a horizontal line minus 1. The dot clock (which determines the rate at which dots are displayed on the screen) is 18 MHz. Kaypro uses 8 horizontal dots in each character cell. When the Kaypro is turned on, register 0 is set to 106, so (here comes the math !!) (106+1)*(8/18MHz)=47.56 microseconds or 21,028 Hertz. To increase the time for each line, (that is, to reduce the horizontal sweep frequency) to something your monitor can handle, we must increase this number stored in register 0. This is achieved by outputting to port 1C hex (28 decimal) the register we wish to address, followed by outputting to port 1D, the value we wish to put into the register. In Mbasic the statement is: OUT (28),0 : OUT (29),107

As the value (107) is increased, you will reach a point where the horizontal hold on your monitor will lock onto the signal and produce a stable display. If the number is increased too much, The internal screen will go out of sync and you'll lose the internal display. Increase the number gradually, so you don't go any farther than necessary. Usually the number will wind up some where between 108 & 113.

This process is a balancing act. As the time per horizontal line is increased, the vertical time per frame is also increased, reducing the frame rate, and the external monitor screen will begin to blink slowly enough to be seen. This will be most noticeable when the screen is full. The Kaypro's screen has a long enough persistence to minimize this effect, but the external monitor likely won't. There are two registers in the 6845 to control the vertical frame rate: register 5, a fine adjust (currently set to 10), and register 9, the number of scan lines per character. Setting register 5 to 0 will help considerably. In Mbasic: OUT (28),5 : OUT (29),0 To gain any more we'll have to reduce the number in register 9, but doing this will cause a change in the characters on display.

Each character cell is 16 lines high. If we reduce the number to 14 (the number stored is scan lines per character minus 1), we lose the underline capability. (In Mbasic: OUT (28),9 : OUT (29),14) This won't hurt the text much, but will upset the graphics which need all 16 lines.If we reduce the number to 13, we'll lose the lower line of the descenders, "yqgj", making the text pretty tough to read. If an even higher number is necessary, then the character ROM will have to be changed.

Once you have a readable display on your external monitor, the horizontal position of the displayed line can be adjusted by changing the value in register 2 (default 86) to move the line right or left. There are other registers in the 6845 that can be tinkered with, but not without consequences. Since the video driver software in the Monitor ROM expects the screen to be 25 lines of 80 characters, it too would have to be changed.

Now that you've twiddled the bits and figured out what parameters are right for your monitor, you can enter them interactively in Mbasic. You must do this each time you boot up, however. One alternative would be to write an Mbasic program to do all the dirty work for you and call it with an embedded initial command line, or with very little more work you could use the Sbasic compiler to produce a .com file to do the same job. There is no end to the possibilities. Whatever you come up with, drop me a line and let me know, and sometime in the future we'll publish a compendium of solutions.

PARTS LIST

1 14 pin wire wrap socket
1 14 pin socket
1 74LS00 integrated circuit
2 220 ohm resistors
1 100 ohm resistor
1 68 ohm resistor
3 1N914 diodes
1 2N3904 transistor
1 video connector (RCA type or F type for rear panel mounting)
1 prototype board

Note:

The PCB drawings are 2X actual size and printing may have distorted them as well. Reduce with copying machine and use transfer paper to iron on the PCB traces. Also note the backside is flipped end for end, check hole alignment for proper orientation.

The 6845 is the same chip used on IBM PC's for monichrome adapters. More information on setting the device up can be found in many of the PC hardware support books, and the chip is manufactured under license from Motorola to numberous vendors. BDK

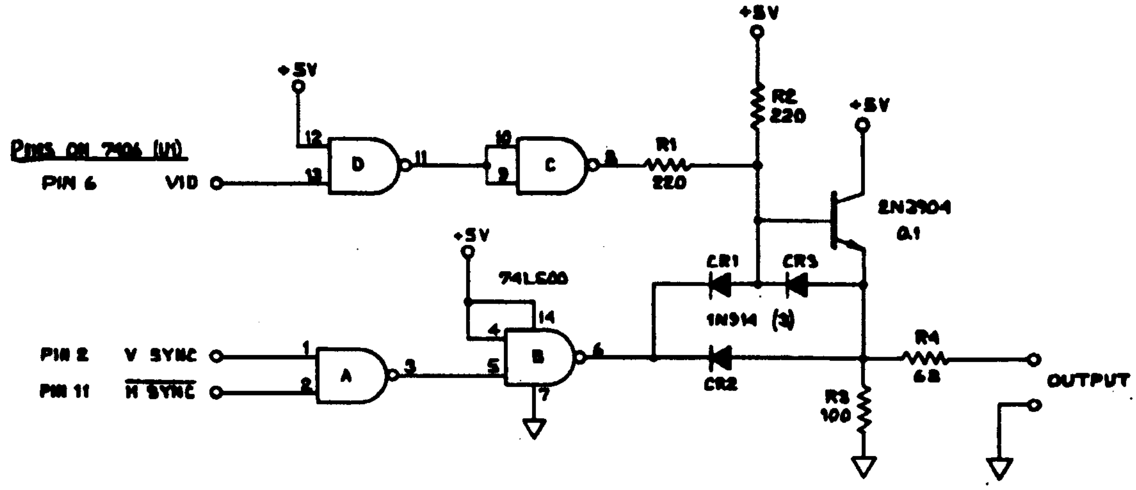

Figure 1. Schematic.

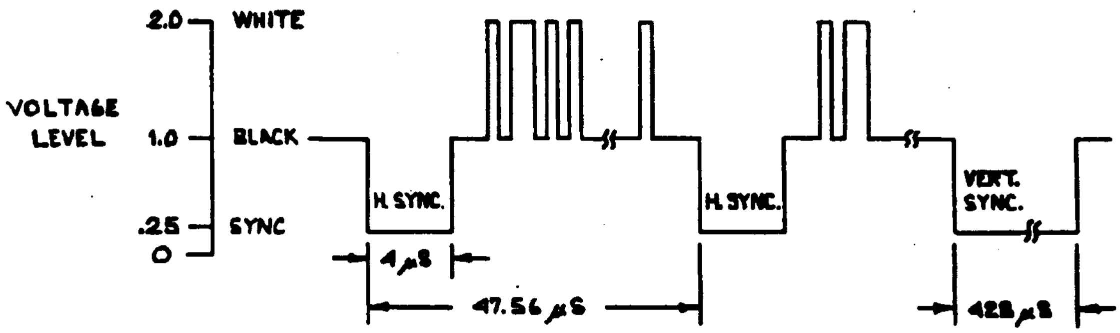

Figure 2. Video Signal.

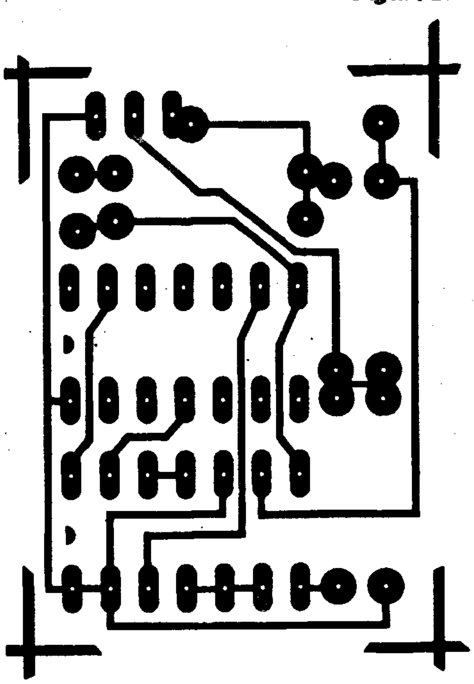

Figure 3. Printed Circuit Board—bottom side.

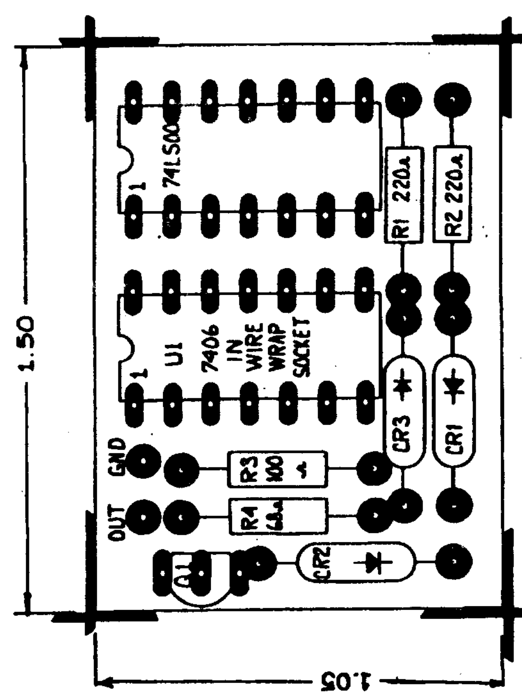

Figure 4. Printed Circuit Board—parts layout.