

Apple II Technical Notes



Developer Technical Support

Apple IIe

#8: Known Anomalies of Enhanced IIe ROMs

Revised by: Matt Deatherage
November 1988

Revised by: Cameron Birse
February 1986

This Technical Note describes three problems with the Enhanced IIe ROMs and some suggested solutions.

The following three anomalies are known to occur when the Enhanced IIe ROMs are present:

1. Some Apple II peripheral cards do not handle interrupts well since Apple II family members before the IIc and Enhanced IIe did not handle them very well either. If a card that cannot handle interrupts is used on the Enhanced Apple IIe, any interrupt is very likely to crash the system. A common example of this would be older, non-interruptible printer cards used with a Mouse card in the system. You can often work around this problem by disabling interrupts before printing to such a printer card.
2. There may be some problems when using the ROMs with communications packages. These problems are due to the way the 80-column firmware switches into 40-column mode. By sending a Control-Q through COUT, the firmware switches into

40-column mode. A simple solution to this would be to send an Escape-Control-D sequence, which disables the control functions. This solution will remain in effect until either the 80-column card is re-initialized by PR#3 or an Escape-Control-E sequence is sent through COUT. Another solution would be to simply not allow Control-Q sequences to get through to COUT by filtering them before they get there.

3. Many developers using double high-resolution graphics may wish to use 40-column text displays so the text can be read on a television set. There are a couple of possibilities:

A. You can define your own double high-resolution character set with any size characters you desire, then plot them on the double high-resolution screen.

B. You can print text to the Apple IIe text screen and toggle the screen on to display it.

Note: There is no way to display four lines of 40-column text at the bottom of the double high-resolution screen in mixed mode since the 80-column hardware must be active while double high-resolution graphics are being displayed.

To use the second method, however, does require some special considerations.

The Apple IIe scroll routine continues to use the window parameters when scrolling, but uses the `80COL` soft switch to determine if it should scroll the 80- or 40-column screen. Since the firmware has initialized a 40-column window, the scroll routines will move only the first 40 columns. But, the `80COL` flag has been turned on for double high-resolution, therefore, the scrolling routine takes every even column from auxiliary memory and every odd column from main memory. As a result, only the first 40 columns get scrolled, 20 columns from auxiliary memory and 20 columns from main memory.

One possible solution to the problem is to write your own scroll routines. Another might be to write to the screen so that scrolling will not occur. But there is yet another solution. Turn on the full 80-column mode with a `PR#3` or the equivalent. Now print your text to `COUT` in the normal manner, being careful not to exceed 40 characters per line. The 80 column firmware will scroll everything properly. When you are ready to display text, send a Control-Q sequence through `COUT` to switch to 40 columns. When you are ready to return to double high-resolution mode, send a Control-R sequence to `COUT`.

When switching modes, a momentary glitch may occur. If you send the Control-Q sequence to `COUT` while still in graphics mode, the screen will go to regular single high-resolution mode before finally going to text mode. If you switch to text mode first, the text will be in 80-column mode (with 40 columns displayed on the left half of the screen)

before ultimately going to 40-column mode. The same potential glitch may occur going back to double high-resolution mode. The glitch will be only momentary and may not present any problem for your application.

Further Reference

- *Apple IIe Technical Reference Manual*