

Appendix 2.

TECHNICAL PROBLEMS WITH FUNNY MODE.

The technical problems associated with using a modified DOS 3.3 and monitor ROM in Apple /// native mode are numerous and nontrivial. Some examples:

390 Disk Controller. The disk controller selection logic is different from the Apple II when running in native mode. In addition, native mode activates the disk switched detection logic, which DOS does not understand.

6522 System Control Registers. The addresses \$FFD0 through \$FFEF address the Environment, Zero page and Bank registers, when native mode is enabled. In the Apple II this memory space is occupied by the Apple II monitor ROM. Inadvertantly referencing these locations can drastically change the state of the machine in a fashion that Apple II software is not designed to handle.

Apple /// ROM Support. Since it is common practice in the Apple II to write software that is dependent upon the monitor ROM code, allowing applications to have full access to the ///'s native mode will tend to increase our dependence on current Apple /// ROM. This is in direct conflict with SOS's stated goal of providing hardware/firmware transparency to applications.

I/O Addresses. The disk controller, silentype port, smooth scroll and character download i/o locations are tightly intertwined. Use of these i/o facilities without very careful design and coding will cause applications which use one of these devices to disable, or worse yet, trash one or more of the others.

Scratchpad RAM. The set of RAM locations used by peripheral card firmware in the Apple II are now reserved for the character download hardware. As a consequence, Apple II software/firmware that use the "screen holes" will not work in conjunction with routines that download characters.

Graphics. Native mode hardware does not support mixed mode or Lo-res graphics.

Silentype. In order to use the onboard custom printer port to support the silentype, a special driver for the DOS environment would need to be written and supported.

RS232 Port. In order to use the full capabilities of the this port, simultaneous two-way transmission, buffering of unexpected input, XON/XOFF handshaking, etc., a special driver for the DOS environment would need to be written and supported.

Memory Addressing. One of the major reasons for running in native mode is that one can address a considerable amount of additional memory. Native mode allows two methods for gaining access to this memory, bank switching, and enhanced indirect addressing, both of which are foreign to standard Apple II software. One example of the difficulties of supporting feature in funny mode, is that the zero pages which support enhanced indirect addressing are in the range \$10 to \$1F. This memory region is normally occupied by Applesoft

A /// # 53 5/6 [218]