

When the IRQ' line on the Apple II's microprocessor is activated (pulled low), the microprocessor transfers control through the vector in locations \$FFFE-\$FFFF. This vector is the address of the Monitor's interrupt handler, which determines whether the request is due to an external IRQ or a BRK instruction and transfers control to the appropriate routine via the vectors stored in memory page 3. The BRK vector is in locations \$03F0-\$03F1 and ProDOS uses the IRQ vector in locations \$03FE-\$03FF. (See Table 4-11.) The Monitor normally stores the address of its reset routine in the IRQ vector; you should substitute the address of your program's interrupt-handling routine.

Apple Pascal doesn't use the BRK vector at \$03F0-\$03F1, but it does use the IRQ vector at \$03FE-\$03FF.

---

## Interrupts on Apple IIe series computers

The interrupt handler built into the enhanced Apple IIe's firmware saves the contents of the accumulator on the stack. (The original Apple IIe saves the contents of the accumulator at location \$45.) DOS 3.3, as well as the Monitor, rely on the integrity of location \$45, so this change lets both DOS 3.3 and the Monitor continue to work with active interrupts on the enhanced Apple IIe.

---

### Original IIe

Since the built-in interrupt handler on the original Apple IIe uses location \$45 to save the contents of the accumulator, the operating system fails when an interrupt occurs under DOS 3.3 on the original Apple IIe.

---

If you want to write programs that use interrupts while running on the original Apple IIe, Apple II Plus, or Apple II, you must use either ProDOS or Apple II Pascal 1.2 (or later versions). Both these operating systems give you full interrupt support, even though these versions of the Apple II don't include interrupt support in their firmware. (Versions of Pascal before 1.2 do not work with interrupts enabled on an original Apple IIe.)

Some other manufacturer's hardware, such as coprocessor cards, don't work properly in an interrupting environment. If you are trying to develop an application and encounter this problem, check with the manufacturer of the card to see if a later version of the hardware or its software will operate properly with interrupts active. You may not be able to use interrupts if an interrupt-tolerant version isn't available.