

Apple warns against using the Protocol Converter and states, "using the assembly-language protocol is fairly complicated." Nevertheless, a significant amount of the Apple firmware is used to implement the Protocol Converter features. With this in mind let us proceed.

#### Using the Protocol Converter

In order to use the Protocol Converter firmware, you need first to find it. The first step in finding it is to find which slot it is in. Cards with Protocol Converter firmware can be identified by four bytes: \$Cs01 = \$20, \$Cs03 = \$00, \$Cs05 = \$03, and \$Cs07 = \$00. The first three bytes in that list are the same for all disk drive controllers. The zero value at \$Cs07 distinguishes it as a disk controller PLUS a Protocol Converter.

The next step is to find the entry point in the firmware for Protocol Converter calls. The byte a \$CsFF is the key. That byte is the offset in the firmware page for ProDOS calls. If \$CsFF = \$45, for example, ProDOS device driver calls should take the form "JSR \$Cs45". To get the address of the Protocol Converter entry point, add 3 to the ProDOS entry point. In this example, "JSR \$Cs48" would enter the Protocol Converter firmware.

A program to find the slot and build the address of the Protocol Converter could look like this:

```
pcaddr .eq $01,$02
find.pc
        lda #0
        sta pcaddr
        ldx #$C7          slot = 7 to 1 step -1
.1      stx pcaddr+1
        ldy #7
.2      lda (pcaddr),y
        cmp pc.sig,y
        beq .3
        dex
        cpx             #$c1
        bcs .1          try next slot
        sec             signal could not find pc
        rts.3           dey
        dey
        bpl .2
```