CCB Sizes Information about the Command Control Block

The .XPP driver uses the memory provided at the end of the UserWrite, UserCommand and GetStatus functions parameter blocks as an internal command control block (CCB). Using the maximum block sizes specified in the call descriptions will provide adequate space for the call to execute successfully. However, this section is provided for developers who wish to minimize the amount of memory taken up by the CCB in the queue element.

Specifically, this memory is used for building data structures to be used in making calls to the ATP driver. This includes parameter blocks and buffer data structures (BDS). The size of this memory depends on the size of the response expected, and, in the case of UserWrite, on the size of data to be written.

In the UserCommand and GetStatus cases (along with all AFP calls which map to UserCommand), a BDS must be set up to hold the response information. The number of entries in this BDS is equal to the size of the response buffer divided by the maximum number of data bytes per ATP response packet (578), rounded up. As described in the ASP chapter in *Inside AppleTalk*, ASP must ask for an extra response in the case where the response buffer is an exact multiple of 578. Of course, no BDS can be larger than eight elements. XPP also needs bytes for the queue element to call ATP with, so the minimum size of a CCB, as a function of the response buffer size (rbSize) is

bdsSize = MIN ((rbSize DIV 578) + 10,8)*bdsEntrySz

ccbSize = ioQElSize + 4 +bdsSize

With UserWrite (and AFP calls mapping to userWrite), XPP must create an additional **BDS** and queue element to use in sending the write data to the server. Therefore the minimum size of a UserWrite CCB, as a function of the response buffer and write data sized (rbSize and wdSize) is:

wrBDSSize = MIN (((wdSize DIV 578) + 1,8)*bdsEntrySz

wrCCBSize = (2*ioQElSize) + 4 + bdsSize + wrBDSSize

Note: BDSEntrySz is equal to 12; ioQelSize is equal to 50.