GetVCBQHdr Obtain pointer to the volume control block queue header

#include <Files.h> **File Manager**

```
QHdrPtr     GetVCBQHdr( );
            returns       address of a 10-byte QHdr structure
```

GetVCBQHdr returns the standard Operating System queue's header address, which is used to maintain the linked-list of volume control blocks. There is one VCB entry for every online volume, whether it is currently mounted or not.

Returns: a 32-bit QHdrPtr; the address of the 10-byte QHdr structure whose qLink field points to the first volume control block in the queue.

Notes: C programmers may prefer to get this address from the global variable VCBQHdr (at 0x0356). The global variable DefVCBPtr points directly to the VCB of the current default volume.

It is probably a mistake to manipulate this queue directly, but if you need to, you can use the OS Utilities functions **Enqueue** and **Dequeue**, passing VCBQHdr as the queue you wish to modify.

Nearly everything you'll ever want to know about volumes can be obtained via **PBGetVInfo** (which lets you easily index through the list).

If you are running under HFS, the queue elements are 178-byte VCB structures. If you booted with a flat file system, the queue is made up of structures which include only the first 94-bytes of the VCB structure. The qLink field of either structure points to the next such structure in the chain or is 0 to indicate the end of the chain. The following example illustrates how to access the VCB queue.

Example

```
#include <Files.h>
#include <OSUtils.h>

QHdrPtr    qhp;
VCB        *vcbp;

qhp= GetVCBQHdr();         /*  address of queue header */

printf("First vcb is at = %lx\n", qhp->qHead );
vcbp = (VCB *)qhp->qHead;       /* address of a queue element */

printf("Volume's drive # is: %d\n", vcbp->vcbDrvNum );
printf("number of files in root = %d\n", vcbp->vcbNmFls );

printf("Next vcb is at = %lx\n", vcbp->qLink );    /* 0 means no more */
```