DCtlEntry structure

#include <Devices.h>

typedef struct **DCtlEntry** {

		Size	Offset	Description
Ptr	dCtlDriver;	4	0	pointer to ROM driver or handle to RAM driver
short	dCtlFlags;	2	4	flags
QHdr	dCtlQHdr;	4	6	driver I/O queue header
long	dCtlPosition;	4	10	byte position used by read and write calls
Handle	dCtlStorage;	4	14	handle to RAM driver's private storage
short	dCtlRefNum;	2	18	driver reference number
long	dCtlCurTicks;	4	20	used internally
WindowPtr	dCtlWindow;	4	24	pointer to driver's window
short	dCtlDelay;	2	26	number of ticks between periodic actions
short	dCtlEMask;	2	28	desk accessory event mask
short	dCtlMenu;	2	30	menu ID of menu associated with driver

} **DCtlEntry**; 32

typedef DCtlEntry ***DCtlPtr**;
typedef DCtlEntry ****DCtlHandle**;

Notes: When a driver serves a slot device the **Device Control Entry** has six additional fields added to the end and is known as an **AuxDCE**.

The low-order byte of the dCtlFlags word contains the following flags:

Bit Number	Meaning
5	Set if driver is open
6	Set if driver is RAM-based
7	Set if driver is currently executing

The high-order byte of the dCtlFlags word contains flags copied from the drvrFlags word of the driver.

DCtlQHdr contains the header of the driver's I/O queue. DCtlPosition is used only by drivers of block devices, and indicates the current source or destination position of a read or write call. The position is given as a number of bytes beyond the physical beginning of the medium used by the device. For example, if one logical block of data has just been read from a 3 1/2" disk via the Disk Driver, dCtlPosition will be 512.