OpenSlot Open slot device

#include <Devices.h> **Device Manager**

```
OSErr       OpenSlot(pb, async);
ParmBlkPtr  pb ;        address of a 50-byte IOParam structure
Boolean     async ;     0=await completion; 1=immediate return
            returns     Error Code; 0=no error
```

OpenSlot is the same as **PBOpen** except that you use it when opening bus slot devices. It sets the IMMED bit to signal an extended parameter block.

pb is the address of a parameter block. The following fields are relevant:

Out-In	Name	Type	Size	Offset	Description
->	ioCompletion	ProcPtr	4	12	Completion routine address (if *async* =TRUE)
<-	ioResult	OSErr	2	16	Error Code (0=no error, 1=not done yet)
->	ioNamePtr	StringPtr	4	18	Address of device driver name
<-	ioRefNum	short	2	24	Receives driver reference number
->	ioPermssn	SignedByte	1	27	Rd/Wrt permission (1=read, 2=write, et.al.)

Additionally, if the slot resource serves a single device, there is an extension that includes:

->	ioMix	ProcPtr	4	28	Reserved for the driver open routine
->	ioFlags	short	2	32	Cleared to indicate single slot device
->	ioSlot	SignedByte	1	34	Slot number for device being opened
->	ioID	SignedByte	1	35	Slot resource ID

If the slot resource serves more than one device, there is an extension that includes:

->	ioMix	ProcPtr	4	28	Reserved for the driver open routine
->	ioFlags	short	2	32	Cleared to indicate single slot device
->	ioSEBlkptr	ProcPtr	4	34	Address of external parameter block

async is a Boolean value. Use FALSE for normal (synchronous) operation or TRUE to enqueue the request and resume control immediately. See Async I/O.

Returns: an operating system Error Code. It will be one of:

noErr	(0)	No error
badUnitErr	(-21)	*refNum* doesn't match unit table
unitEmptyErr	(-22)	*refNum* specifies NIL handle in unit table
openErr	(-23)	Requested Read/Write permission and the driver's Open permissions don't match
dInstErr	(-26)	Couldn't find driver in resource file

Notes: **OpenSlot** opens a bus-based slot device driver when used by the **Device Manager**. Other than that, it is the equivalent of **PBOpen**