NAME

physio - initiate I/O on raw devices

SYNOPSIS

#include <sys/param.h>
#include <sys/systm.h>
#include <sys/bio.h>
#include <sys/buf.h>

int
physio(*struct cdev *dev*, *struct uio *uio*, *int ioflag*);

DESCRIPTION

The **physio**() is a helper function typically called from character device **read**() and **write**() routines to start I/O on a user process buffer. The maximum amount of data to transfer with each call is determined by *dev->si_iosize_max*. The **physio**() call converts the I/O request into a **strategy**() request and passes the new request to the driver's **strategy**() routine for processing.

Since *uio* normally describes user space addresses, **physio**() needs to lock those pages into memory. This is done by calling **vmapbuf**() for the appropriate pages. **physio**() always awaits the completion of the entire requested transfer before returning, unless an error condition is detected earlier.

A break-down of the arguments follows:

dev	The device number identifying the device to interact with.
uio	The description of the entire transfer as requested by the user process. Currently, the results of passing a *uio* structure with the *uio_segflg* set to anything other than UIO_USERSPACE are undefined.
ioflag	The ioflag argument from the **read**() or **write**() function calling **physio**().

RETURN VALUES

If successful **physio**() returns 0. EFAULT is returned if the address range described by *uio* is not accessible by the requesting process. **physio**() will return any error resulting from calls to the device strategy routine, by examining the B_ERROR buffer flag and the *b_error* field. Note that the actual transfer size may be less than requested by *uio* if the device signals an "end of file" condition.

SEE ALSO

read(2), write(2)

HISTORY

The **physio** manual page is originally from NetBSD with minor changes for applicability with FreeBSD.

The **physio** call has been completely re-written for providing higher I/O and paging performance.