NAME

io_setup − create an asynchronous I/O context

SYNOPSIS

#include <libaio.h>

int io_setup(unsigned *nr_events***, aio_context_t ****ctxp***);**

Link with −*laio*.

DESCRIPTION

io_setup() creates an asynchronous I/O context capable of receiving at least *nr_events*. *ctxp* must not point to an AIO context that already exists, and must be initialized to 0 prior to the call. On successful creation of the AIO context, **ctxp* is filled in with the resulting handle.

RETURN VALUE

On success, **io_setup**() returns 0. For the failure return, see NOTES.

ERRORS

EAGAIN
The specified *nr_events* exceeds the user's limit of available events.

EFAULT
An invalid pointer is passed for *ctxp*.

EINVAL
ctxp is not initialized, or the specified *nr_events* exceeds internal limits. *nr_events* should be greater than 0.

ENOMEM
Insufficient kernel resources are available.

ENOSYS
io_setup() is not implemented on this architecture.

VERSIONS

The asynchronous I/O system calls first appeared in Linux 2.5, August 2002.

CONFORMING TO

io_setup() is Linux-specific and should not be used in programs that are intended to be portable.

NOTES

Glibc does not provide a wrapper function for this system call.

The wrapper provided in *libaio* for **io_setup**() does not follow the usual C library conventions for indicating error: on error it returns a negated error number (the negative of one of the values listed in ERRORS). If the system call is invoked via **syscall**(2), then the return value follows the usual conventions for indicating an error: −1, with *errno* set to a (positive) value that indicates the error.

SEE ALSO

io_cancel(2), **io_destroy**(2), **io_getevents**(2), **io_submit**(2)

COLOPHON

This page is part of release 3.22 of the Linux *man-pages* project. A description of the project, and information about reporting bugs, can be found at http://www.kernel.org/doc/man-pages/.