



linuxlib.ru

[# Главная](#)

[# 0 библиотеке](#)

[# Выбор дистрибутива](#)

[преимущества Linux/UNIX](#) | [основные дистрибутивы](#) | [серверный Linux](#) | [BSD](#) | [LiveCDs](#) | [прочее](#)

[# Установка и удаление программ](#)

[общие вопросы](#) | [каталоги софта](#) | [специальные случаи](#)

[# Настройка и работа](#)

[установка, загрузчики](#) | [настройка Linux](#) | [консоль](#) | [файловые системы](#) | [процессы](#) | [шеллы, русификация, команды](#) | [виртуальные машины, эмуляторы](#)

[# X Window и оконные менеджеры](#)

[настройка X Window](#) | [GNOME](#) | [KDE](#) | [IceWM и др.](#)

[# Работа с текстами](#)

[редакторы](#) | [офис](#) | [шрифты, кодировки и русификация](#) | [преобразования текстовых файлов](#) | [LaTeX, SGML и др.](#) | [словари](#)

[# Графика](#)

[GIMP](#) | [фото](#) | [обработка изображений](#) | [форматы графических файлов](#)

FCNTL(2)

НАЗВАНИЕ

fcntl – управление файлами

СИНТАКСИС

```
#include <fcntl.h>
```

```
int fcntl (fildes, cmd, arg)
```

```
int fildes, cmd, arg;
```

ОПИСАНИЕ

Системный вызов fcntl выполняет управляющие операции над открытыми файлами. Аргумент fildes – это дескриптор открытого файла, полученный после выполнения системных вызовов creat, open, dup, fcntl и pipe.

Аргумент cmd может принимать следующие значения, определяющие выполняемую операцию:

F_DUPFD

Создать новый дескриптор файла с такими свойствами:

1. Его номер – есть минимальный из доступных номеров, не меньших arg.

Сети, администрирование

[общие вопросы](#) | [Dialup & PPP](#) | [брандмауэры](#) |
[маршрутизация](#) | [работа в Windows-сетях](#) |
[веб-серверы](#) | [Apache](#) | [прокси-серверы](#) |
[сетевая печать](#) | [прочее](#)

Программирование

[GCC & GNU make](#) | [программирование в UNIX](#) |
[графические библиотеки](#) | [Tcl](#) | [Perl](#) | [PHP](#) |
[Java & C#](#) | [СУБД](#) | [CVS](#) | [прочее](#)

Ядро

Мультимедиа

Интернет

Почта

Безопасность

Железо

Разное

Linux HowTo (как сделать)

Книги и руководства

Материалы на английском языке

Реклама:

[Купить биткоин у нас](#)

Самая подробная информация купить биткоин у нас.

[yobit.net](#)

2. Он ассоциирован с тем же открытым файлом (или каналом), что и исходный дескриптор `fildes`.
3. У него тот же указатель текущей позиции в файле, что и у исходного (то есть они разделяют общий указатель).
4. Тот же режим доступа к файлу (чтение, запись или чтение/запись).
5. Те же флаги статуса файла (то есть оба дескриптора разделяют общие флаги статуса).
6. Ассоциированный с новым дескриптором флаг "закрыть при выполнении вызова `ехес`" устанавливается в состояние "оставить открытым при выполнении вызова `ехес`".

F_GETFD

Получить значение флага "закрыть при выполнении вызова `ехес`" для дескриптора файла `fildes`. Если младший бит возвращаемого значения равен нулю, то файл останется открытым, в противном случае при выполнении вызова `ехес` файл будет закрыт.

F_SETFD

Установить значение флага "закрыть при выполнении вызова `ехес`" для дескриптора файла `fildes` равным значению младшего бита (0 или 1) аргумента `arg`.

F_GETFL

Получить флаги статуса файла, ассоциированного с дескриптором `fildes`.

F_SETFL

Установить флаги статуса файла, ассоциированного с дескриптором `fildes`, равными значению аргумента `arg`. Могут быть установлены только некоторые флаги [см. [fcntl\(5\)](#)].

F_GETLK

Получить характеристики первой блокировки, мешающей установить новую блокировку, задаваемую структурой типа `flock` с адресом `arg`. Результирующая информация

возвращается в той же структуре. Если нет помех для создания нужной блокировки, то структура flock не изменяется за исключением поля типа блокировки, которому присваивается значение F_UNLCK.

F_SETLK

Установить или снять блокировку сегмента файла в соответствии со значением структуры типа flock, на которую указывает аргумент arg. [см. [fcntl\(5\)](#)]. Операция F_SETLK используется для установки блокировки на чтение (F_RDLCK) или запись (F_WRLCK), а также для снятия блокировки обоих типов (F_UNLCK). Если блокировка на чтение или запись не может быть установлена, то системный вызов fcntl завершается немедленно и возвращает -1.

F_SETLKW

Эта операция отличается от операции F_SETLK только тем, что при неудачной попытке установить блокировку на чтение или запись процесс переходит в состояние ожидания до тех пор, пока нужный сегмент файла не будет разблокирован.

Блокировка на чтение предотвращает блокировку защищаемой области каким-либо процессом на запись. Для данного сегмента файла могут одновременно существовать несколько блокировок на чтение. Дескриптор, используемый для установления блокировки на чтение, должен быть ассоциирован с файлом, открытым с правом чтения.

Блокировка на запись предотвращает блокировку защищаемой области на чтение или запись. Одновременно может существовать не более одной блокировки на запись данного сегмента файла. Дескриптор, используемый для установления блокировки на запись, должен быть ассоциирован с файлом, открытым с правом записи.

Структура типа flock содержит поля, определяющие для сегмента файла тип блокировки (l_type), начальное смещение (l_whence), относительное смещение (l_start), размер (l_len),

идентификатор системы PУФ (`l_sysid`), идентификатор процесса (`l_pid`). Идентификаторы процесса и системы используются только в случае операции `F_GETLK` для возврата характеристик блокировки. Начало и конец блокируемой области могут выходить за конец файла, но не за начало. Можно определить блокировку, всегда действующую до конца файла, если значение поля `l_len` равно 0. Если значения полей `l_whence` и `l_start` равны 0, то блокировка будет распространяться на весь файл. Изменение или снятие блокировки сегмента из середины большого защищенного сегмента приводит к появлению с обоих концов двух меньших защищенных сегментов. Блокировка сегмента, который уже блокирован вызывающим процессом, приводит к удалению старого и установке нового типа блокировки. Все блокировки, ассоциированные с файлом для данного процесса, удаляются, когда файл закрывается этим процессом или когда процесс завершается, не закрывая файл. Блокировки не наследуются порождаемым процессом при выполнении системного вызова [fork\(2\)](#).

Если блокировка доступа к файлу разрешена [см. [chmod\(2\)](#)], то системные вызовы `read` и `write` для этого файла выполняются с учетом действующих блокировок.

Системный вызов `fcntl` завершается неудачей, если выполнено хотя бы одно из следующих условий:

[EBADF]

Аргумент `fildes` не является корректным дескриптором открытого файла.

[EINVAL]

При операции `cmd`, равной `F_DUPFD`, значение аргумента `arg` либо отрицательно, либо больше или равно максимально допустимому для одного пользователя количеству дескрипторов открытых файлов.

[EINVAL]

При операции `cmd`, равной `F_GETLK`, `F_SETLK` или `F_SETLKW`, значение аргумента `arg` или информация, на которую указывает `arg`, некорректны.

[EACCES]

При операции `cmd`, равной `F_SETLK`, делается попытка заблокировать на чтение (`F_RDLCK`) сегмент файла, заблокированный другим процессом на запись, либо попытка заблокировать на запись (`F_WRLCK`) сегмент файла, заблокированный другим процессом на чтение или запись.

[ENOLCK]

При операции `cmd`, равной `F_SETLK` или `F_SETLKW`, превышает максимально допустимое системой количество блокировок.

[EDEADLK]

При операции `cmd`, равной `F_SETLKW`, ожидание возможности установить блокировку приводит к тупику.

[EFAULT]

При операции `cmd`, равной `F_SETLK`, аргумент `arg` указывает за пределы отведенного процессу адресного пространства.

[EINTR]

Во время выполнения системного вызова перехвачен сигнал.

[ENOLINK]

Дескриптор `fd` ассоциирован с файлом на удаленном компьютере, связи с которым в данный момент нет.

СМ. ТАКЖЕ

[close\(2\)](#), [creat\(2\)](#), [dup\(2\)](#), [exec\(2\)](#), [fork\(2\)](#), [open\(2\)](#), [pipe\(2\)](#), [fcntl\(5\)](#).

ДИАГНОСТИКА

При успешном завершении системного вызова в зависимости от операции `cmd` возвращаются следующие значения:

- **F_DUPFD** Новый дескриптор файла.
- **F_GETFD** Значение флага (определен только младший бит).
- **F_SETFD** Значение, отличное от `-1`.

- **F_GETFL** Значение флагов статуса файла.
- **F_SETFL** Значение, отличное от -1.
- **F_GETLK** Значение, отличное от -1.
- **F_SETLK** Значение, отличное от -1.
- **F_SETLKW** Значение, отличное от -1.

В случае ошибки возвращается -1, а переменной `errno` присваивается код ошибки.

ПРЕДОСТЕРЕЖЕНИЯ

Так как в будущем переменной `errno` будет присваиваться значение `EAGAIN` вместо `EACCES` в случае, если сегмент файла уже блокирован другим процессом, для достижения мобильности прикладных программ следует ожидать и обрабатывать оба кода ошибки.