

## Системный вызов open

### Прототип системного вызова

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

```
int open(char *path, int flags);  
int open(char *path, int flags, int mode);
```

### Описание системного вызова

Системный вызов **open** предназначен для выполнения операции открытия файла и, в случае ее удачного осуществления, возвращает файловый дескриптор открытого файла (небольшое неотрицательное целое число, которое используется в дальнейшем для других операций с этим файлом).

Параметр **path** является указателем на строку, содержащую полное или относительное имя файла.

Параметр **flags** может принимать одно из следующих трех значений:

**O\_RDONLY** - если над файлом в дальнейшем будут совершаться только операции чтения;

**O\_WRONLY** - если над файлом в дальнейшем будут осуществляться только операции записи;

**O\_RDWR** - если над файлом будут осуществляться и операции чтения, и операции записи.

Каждое из этих значений может быть скомбинировано посредством операции "побитовое или ( | )" с одним или несколькими следующими флагами:

**O\_CREAT** - если файл с указанным именем не существует, он должен быть создан.

**O\_EXCL** - применяется совместно с флагом **O\_CREAT**. При совместном их использовании и существовании файла с указанным именем, открытие файла не производится и констатируется ошибочная ситуация.

**O\_NDELAY** - запрещает перевод процесса в состояние **ожидание** при выполнении операции открытия и любых последующих операциях над этим файлом.

**O\_APPEND** - при открытии файла и перед выполнением каждой операции записи (если она, конечно, разрешена) указатель текущей позиции в файле устанавливается на конец файла.

**O\_TRUNC** - если файл существует, уменьшить его размер до 0, с сохранением

существующих атрибутов файла, кроме, быть может, времен последнего доступа к файлу и его последней модификации.

Кроме того, в некоторых версиях операционной системы UNIX могут применяться дополнительные значения флагов:

**O\_SYNC** - любая операция записи в файл будет блокироваться (т. е. процесс будет переведен в состояние **ожидание**) до тех пор, пока записанная информация не будет физически помещена на соответствующий низлежащий уровень hardware.

**O\_NOCTTY** - если имя файла относится к терминальному устройству, оно не становится управляющим терминалом процесса, даже если до этого процесс не имел управляющего терминала.

Параметр **mode** устанавливает атрибуты прав доступа различных категорий пользователей к новому файлу при его создании. Он обязателен в том случае, если среди заданных флагов присутствует флаг **O\_CREAT** и может быть опущен в противном случае. Этот параметр задается как сумма следующих восьмеричных значений:

<b>0400</b>	- Разрешено чтение для пользователя, создавшего файл.
<b>0200</b>	- Разрешена запись для пользователя, создавшего файл.
<b>0100</b>	- Разрешено исполнение для пользователя, создавшего файл.
<b>0040</b>	- Разрешено чтение для группы пользователя, создавшего файл.
<b>0020</b>	- Разрешена запись для группы пользователя, создавшего файл.
<b>0010</b>	- Разрешено исполнение для группы пользователя, создавшего файл.
<b>0004</b>	- Разрешено чтение для всех остальных пользователей
<b>0002</b>	- Разрешена запись для всех остальных пользователей
<b>0001</b>	- Разрешено исполнение для всех остальных пользователей

При создании файла реально устанавливаемые права доступа получаются из стандартной комбинации параметра **mode** и маски создания файлов текущего процесса **umask**, а именно - они равны **mode & ~umask**.

При открытии файлов типа FIFO системный вызов имеет некоторые особенности поведения по сравнению с открытием файлов других типов. Если FIFO открывается только для чтения, и не задан флаг **O\_NDELAY**, то процесс, осуществивший системный вызов, блокируется до тех пор, пока какой-либо другой процесс не откроет FIFO на запись. Если флаг **O\_NDELAY** задан, то возвращается значение файлового дескриптора, ассоциированного с FIFO. Если FIFO открывается только для записи, и не задан флаг **O\_NDELAY**, то

процесс, осуществивший системный вызов, блокируется до тех пор, пока какой-либо другой процесс не откроет FIFO на чтение. Если флаг **O\_NDELAY** задан, то констатируется возникновение ошибки и возвращается значение -1.

### **Возвращаемое значение**

Системный вызов возвращает значение файлового дескриптора для открытого файла при нормальном завершении и значение -1 при возникновении ошибки.