

**NAME**

**devname** - get device name

**LIBRARY**

Standard C Library (libc, -lc)

**SYNOPSIS**

```
#include <sys/stat.h>
```

```
#include <stdlib.h>
```

```
char *
```

```
devname(dev_t dev, mode_t type);
```

```
char *
```

```
devname_r(dev_t dev, mode_t type, char *buf, int len);
```

```
char *
```

```
fdevname(int fd);
```

```
char *
```

```
fdevname_r(int fd, char *buf, int len);
```

**DESCRIPTION**

The **devname**() function returns a pointer to the name of the block or character device in */dev* with a device number of *dev*, and a file type matching the one encoded in *type* which must be one of S\_IFBLK or S\_IFCHR. To find the right name, **devname**() asks the kernel via the *kern.devname* sysctl. If it is unable to come up with a suitable name, it will format the information encapsulated in *dev* and *type* in a human-readable format.

The **fdevname**() and **fdevname\_r**() function obtains the device name directly from a file descriptor pointing to a character device. If it is unable to come up with a suitable name, these functions will return a NULL pointer.

**devname**() and **fdevname**() return the name stored in a static buffer which will be overwritten on subsequent calls. **devname\_r**() and **fdevname\_r**() take a buffer and length as argument to avoid this problem.

**EXAMPLES**

```
int fd;
```

```
struct stat buf;
```

```
char *name;
```

```
    fd = open("/dev/tun");
    fstat(fd, &buf);
    printf("devname is /dev/%s\n", devname(buf.st_rdev, S_IFCHR));
    printf("fdevname is /dev/%s\n", fdevname(fd));
```

**SEE ALSO**

stat(2)

**HISTORY**

The **devname()** function appeared in 4.4BSD. The **fdevname()** function appeared in FreeBSD 8.0.