

NAME

makedev, major, minor - manage a device number

SYNOPSIS

```
#define _BSD_SOURCE /* See feature\_test\_macros\(7\)
*/
#include <sys/types.h>

dev_t makedev(unsigned int maj, unsigned int min);

unsigned int major(dev_t dev);
unsigned int minor(dev_t dev);
```

DESCRIPTION

A device ID consists of two parts: a major ID, identifying the class of the device, and a minor ID, identifying a specific instance of a device in that class. A device ID is represented using the type *dev_t*.

Given major and minor device IDs, **makedev()** combines these to produce a device ID, returned as the function result. This device ID can be given to [mknod\(2\)](#), for example.

The **major()** and **minor()** functions perform the converse task: given a device ID, they return, respectively, the major and minor components. These macros can be useful to, for example, decompose the device IDs in the structure returned by [stat\(2\)](#).

CONFORMING TO

The **makedev()**, **major()**, and **minor()** functions are not specified in POSIX.1, but are present on many other systems.

NOTES

These interfaces are defined as macros. Since glibc 2.3.3, they have been aliases for three GNU-specific functions: **gnu_dev_makedev()**, **gnu_dev_major()**, and **gnu_dev_minor()**. The latter names are exported, but the traditional names are more portable.

SEE ALSO

[mknod\(2\)](#), [stat\(2\)](#)

COLOPHON

This page is part of release 3.74 of the Linux *man-pages* project. A description of the project, information about reporting bugs, and the latest version of this page, can be found at <http://www.kernel.org/doc/man-pages/>.