NAME

sched_getcpu − determine CPU on which the calling thread is running

SYNOPSIS

#include <sched.h>

int sched_getcpu(void);

Feature Test Macro Requirements for glibc (see **feature_test_macros**(7)):

sched_getcpu():
Since glibc 2.14:
_GNU_SOURCE
Before glibc 2.14:
_BSD_SOURCE || _SVID_SOURCE
/* _GNU_SOURCE also suffices */

DESCRIPTION

sched_getcpu() returns the number of the CPU on which the calling thread is currently executing.

RETURN VALUE

On success, **sched_getcpu**() returns a nonnegative CPU number. On error, −1 is returned and *errno* is set to indicate the error.

ERRORS

ENOSYS
This kernel does not implement **getcpu**(2).

VERSIONS

This function is available since glibc 2.6.

ATTRIBUTES

For an explanation of the terms used in this section, see **attributes**(7).

Interface	Attribute	Value
sched_getcpu()	Thread safety	MT-Safe

CONFORMING TO

sched_getcpu() is glibc-specific.

NOTES

The call

```
cpu = sched_getcpu();
```

is equivalent to the following **getcpu**(2) call:

```
int c, s;
s = getcpu(&c, NULL, NULL);
cpu = (s == -1) ? s : c;
```

SEE ALSO

getcpu(2), **sched**(7)

COLOPHON

This page is part of release 5.05 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 https://www.kernel.org/doc/man−pages/.