



*Full credit is given to the above companies including the OS that this PDF file was generated!*

### ***Rocky Enterprise Linux 9.2 Manual Pages on command 'clock\_getcpuclockid.3'***

#### ***\$ man clock\_getcpuclockid.3***

CLOCK\_GETCPULOCKID(3) Linux Programmer's Manual CLOCK\_GETCPULOCKID(3)

#### NAME

clock\_getcpuclockid - obtain ID of a process CPU-time clock

#### SYNOPSIS

```
#include <time.h>
```

```
int clock_getcpuclockid(pid_t pid, clockid_t *clockid);
```

Link with -lrt (only for glibc versions before 2.17).

Feature Test Macro Requirements for glibc (see feature\_test\_macros(7)):

```
clock_getcpuclockid():
```

```
    _POSIX_C_SOURCE >= 200112L
```

#### DESCRIPTION

The clock\_getcpuclockid() function obtains the ID of the CPU-time clock of the process whose ID is pid, and returns it in the location pointed to by clockid. If pid is zero, then the clock ID of the CPU-time clock of the calling process is returned.

#### RETURN VALUE

On success, clock\_getcpuclockid() returns 0; on error, it returns one of the positive error numbers listed in ERRORS.

## ERRORS

**ENOSYS** The kernel does not support obtaining the per-process CPU-time clock of another process, and `pid` does not specify the calling process.

**EPERM** The caller does not have permission to access the CPU-time clock of the process specified by `pid`. (Specified in POSIX.1-2001; does not occur on Linux unless the kernel does not support obtaining the per-process CPU-time clock of another process.)

**ESRCH** There is no process with the ID `pid`.

## VERSIONS

The `clock_getcpuclockid()` function is available in glibc since version 2.2.

## ATTRIBUTES

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

??

?Interface      ? Attribute    ? Value    ?

??

?`clock_getcpuclockid()` ? Thread safety ? MT-Safe ?

??

## CONFORMING TO

POSIX.1-2001, POSIX.1-2008.

## NOTES

Calling `clock_gettime(2)` with the clock ID obtained by a call to `clock_getcpuclockid()` with a `pid` of 0, is the same as using the clock ID `CLOCK_PROCESS_CPUTIME_ID`.

## EXAMPLES

The example program below obtains the CPU-time clock ID of the process whose ID is given on the command line, and then uses `clock_gettime(2)` to obtain the time on that clock. An example run is the following:

```
$ ./a.out 1            # Show CPU clock of init process
```

```
  CPU-time clock for PID 1 is 2.213466748 seconds
```

```

#define _XOPEN_SOURCE 600

#include <stdint.h>

#include <stdio.h>

#include <unistd.h>

#include <stdlib.h>

#include <time.h>

int

main(int argc, char *argv[])

{

    clockid_t clockid;

    struct timespec ts;

    if (argc != 2) {

        fprintf(stderr, "%s <process-ID>\n", argv[0]);

        exit(EXIT_FAILURE);

    }

    if (clock_getcpuclockid(atoi(argv[1]), &clockid) != 0) {

        perror("clock_getcpuclockid");

        exit(EXIT_FAILURE);

    }

    if (clock_gettime(clockid, &ts) == -1) {

        perror("clock_gettime");

        exit(EXIT_FAILURE);

    }

    printf("CPU-time clock for PID %s is %jd.%09ld seconds\n",

        argv[1], (intmax_t) ts.tv_sec, ts.tv_nsec);

    exit(EXIT_SUCCESS);

}

```

## SEE ALSO

clock\_getres(2), timer\_create(2), pthread\_getcpuclockid(3), time(7)

## COLOPHON

This page is part of release 5.10 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

