NAME

clock_settime, clock_gettime, clock_getres – high-resolution clock operations

SYNOPSIS

cc [*flag* ...] *file* ... **–lposix4** [*library* ...]

#include <time.h>

int clock_settime(clockid_t *clock_id*, **const struct timespec** **tp***);**

int clock_gettime(clockid_t *clock_id***, struct timespec** **tp***);**

int clock_getres(clockid_t *clock_id***, struct timespec** **res***);**

```
struct timespec {
        time_t        tv_sec;        /* seconds */
        long          tv_nsec;       /* and nanoseconds */
};
```

MT-LEVEL

clock_gettime( ) is Async-Signal-Safe

DESCRIPTION

clock_settime( ) sets the specified clock, *clock_id,* to the value specified by *tp.* The calling process must have an effective user ID of **0.**

clock_gettime( ) returns the current value *tp* for the specified clock, *clock_id.*

The resolution of any clock can be obtained by calling **clock_getres( ).** If *res* is not NULL, the resolution of the specified clock is stored in *res.*

The *clock_id* for the real-time clock for the system is **CLOCK_REALTIME**. The values returned by **clock_gettime( )** and specified by **clock_settime( )** represent the amount of time (in seconds and nanoseconds) since 00:00 Universal Coordinated Time, January 1, 1970.

RETURN VALUES

clock_settime( ), clock_gettime( ), and **clock_getres( )** return **0** upon success, otherwise they return **-1** and set **errno** to indicate the error condition.

ERRORS

EINVAL *clock_id* does not specify a known clock.

The *tp* argument to **clock_settime( )** is outside the range for the given clock id.

The *tp* argument to **clock_settime( )** specified a nanosecond value less than zero or greater than or equal to 1,000,000,000.

ENOSYS **clock_settime( ), clock_gettime( ),** or **clock_getres( )** is not supported by this implementation.

EPERM The requesting process does not have the appropriate privilege to set the specified clock.

SEE ALSO

time(2), **ctime**(3C), **timer_gettime**(3R)

NOTES

Clock resolutions are implementation defined and are not settable by a process. Time values that are between two consecutive non-negative integer multiples of the resolution of the specified clock are truncated down to the smaller multiple of the resolution.