NAME

pthread_rwlock_timedwrlock - acquire a read-write lock for writing or give up after a specified period

LIBRARY

POSIX Threads Library (libpthread, -lpthread)

SYNOPSIS

#include <pthread.h>

int
pthread_rwlock_timedwrlock(*pthread_rwlock_t *restrict rwlock,*
*const struct timespec *restrict abs_timeout*);

DESCRIPTION

This function acquires a write lock on the read-write lock *rwlock*. However, if the lock cannot be acquired without waiting for another thread to unlock the lock, this wait shall be terminated when *abs_timeout* expires.

If the thread should be interrupted by a signal, the **pthread_rwlock_timedwrlock**() function will be automatically restarted after the thread returns from the signal handler.

The calling thread may deadlock if at the time the call is made it holds *rwlock*. The results are undefined if this function is called with an uninitialized read-write lock.

IMPLEMENTATION NOTES

To prevent writer starvation, writers are favored over readers.

RETURN VALUES

If successful, the **pthread_rwlock_timedwrlock**() function will return zero. Otherwise, an error number will be returned to indicate the error.

This function shall not return an error code of EINTR.

ERRORS

The **pthread_rwlock_timedwrlock**() function shall fail if:

[ETIMEDOUT] The lock could not be acquired before the specified timeout expired.

The **pthread_rwlock_timedwrlock**() function may fail if:

[EDEADLK] The calling thread already holds *rwlock*.

[EINVAL] The value specified by *rwlock* does not refer to an initialized read-write lock object, or the *abs_timeout* nanosecond value is less than zero or greater than or equal to 1 billion.

SEE ALSO

pthread_rwlock_init(3), pthread_rwlock_timedrdlock(3), pthread_rwlock_unlock(3)

STANDARDS

The **pthread_rwlock_timedwrlock**() function is expected to conform to ISO/IEC 9945-1:1996 ("POSIX.1").

HISTORY

The **pthread_rwlock_timedwrlock**() function first appeared in FreeBSD 5.2.