NAME

pthread_spin_lock, pthread_spin_trylock, pthread_spin_unlock − lock and unlock a spin lock

SYNOPSIS

#include <pthread.h>

int pthread_spin_lock(pthread_spinlock_t ****lock*);**
int pthread_spin_trylock(pthread_spinlock_t ****lock*);**
int pthread_spin_unlock(pthread_spinlock_t ****lock*);**

Compile and link with *−pthread*.

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

pthread_spin_lock(), **pthread_spin_trylock**():
_POSIX_C_SOURCE >= 200112L

DESCRIPTION

The **pthread_spin_lock**() function locks the spin lock referred to by *lock*. If the spin lock is currently unlocked, the calling thread acquires the lock immediately. If the spin lock is currently locked by another thread, the calling thread spins, testing the lock until it becomes available, at which point the calling thread acquires the lock.

Calling **pthread_spin_lock**() on a lock that is already held by the caller or a lock that has not been initialized with **pthread_spin_init**(3) results in undefined behavior.

The **pthread_spin_trylock**() function is like **pthread_spin_lock**(), except that if the spin lock referred to by *lock* is currently locked, then, instead of spinning, the call returns immediately with the error **EBUSY**.

The **pthread_spin_unlock**() function unlocks the spin lock referred to *lock*. If any threads are spinning on the lock, one of those threads will then acquire the lock.

Calling **pthread_spin_unlock**() on a lock that is not held by the caller results in undefined behavior.

RETURN VALUE

On success, these functions return zero. On failure, they return an error number.

ERRORS

pthread_spin_lock() may fail with the following errors:

EDEADLOCK
The system detected a deadlock condition.

pthread_spin_trylock() fails with the following errors:

EBUSY
The spin lock is currently locked by another thread.

VERSIONS

These functions first appeared in glibc in version 2.2.

CONFORMING TO

POSIX.1-2001.

NOTES

Applying any of the functions described on this page to an uninitialized spin lock results in undefined behavior.

Carefully read NOTES in **pthread_spin_init**(3).

SEE ALSO

pthread_spin_destroy(3), **pthread_spin_init**(3), **pthreads**(7)

COLOPHON

This page is part of release 5.02 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/.