NAME

pthread_mutex_trylock - attempt to lock a mutex without blocking

LIBRARY

POSIX Threads Library (libpthread, -lpthread)

SYNOPSIS

#include <pthread.h>

int
pthread_mutex_trylock(*pthread_mutex_t *mutex*);

DESCRIPTION

The **pthread_mutex_trylock**() function locks *mutex*. If the mutex is already locked, **pthread_mutex_trylock**() will not block waiting for the mutex, but will return an error condition.

RETURN VALUES

If successful, **pthread_mutex_trylock**() will return zero, otherwise an error number will be returned to indicate the error.

ERRORS

The **pthread_mutex_trylock**() function will fail if:

[EINVAL]	The value specified by *mutex* is invalid.
[EBUSY]	*Mutex* is already locked.
[EOWNERDEAD]	The argument *mutex* points to a robust mutex and the process containing the previous owning thread terminated while holding the mutex lock. The lock was granted to the caller and it is up to the new owner to make the state consistent.
[ENOTRECOVERABLE]	The state protected by the *mutex* is not recoverable.

SEE ALSO

pthread_mutex_consistent(3), pthread_mutex_destroy(3), pthread_mutex_init(3), pthread_mutex_lock(3), pthread_mutex_unlock(3)

STANDARDS

The **pthread_mutex_trylock**() function conforms to ISO/IEC 9945-1:1996 ("POSIX.1").