

**NAME**

**cnd\_timedwait** - wait for a condition with timeout

**SYNOPSIS**

```
library "threads"
```

```
#include <threads.h>
```

```
int
```

```
cnd_timedwait(cnd_t *cnd, mtx_t mtx, const struct timespec *ts);
```

**DESCRIPTION**

The function **cnd\_timedwait**() will stall the calling thread until a condition is being send to *cnd* from an other thread using either of **cnd\_signal**(3) or **cnd\_broadcast**(3) for up to a given absolute time specified by *ts* (based on **CLOCK\_REALTIME**) after which it will give up and continue execution. The function will release the mutex *mtx* atomically and hold it once the thread resumes execution.

A thread waiting with **cnd\_timedwait**() may be woken up by signals.

**RETURN VALUES**

Upon success **cnd\_timedwait**() returns *thrd\_success*, if a timeout occurred *thrd\_timeout* will be returned. On an error *thrd\_error* will be returned.

**SEE ALSO**

**cnd\_init**(3) **cnd\_destroy**(3) **cnd\_broadcast**(3) **cnd\_timedwait**(3) **cnd\_wait**(3) **mtx\_init**(3)

**HISTORY**

The **cnd\_timedwait**() function first appeared in the C11 standard ISO/IEC 9899:2011.

**AUTHORS**

Jan Adelsbach <jan@jadelsbach.de>