NAME

mq_receive – receive a message from a message queue

SYNOPSIS

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

```
#include <mqueue.h>

ssize_t mq_receive(mqd_t mqdes, char *msg_ptr, size_t msg_len,
        unsigned int msg_prio);

struct mq_attr {
        long    mq_flags;       /* message queue flags */
        long    mq_maxmsg;      /* maximum number of messages */
        long    mq_msgsize;     /* maximum message size */
        long    mq_curmsgs;     /* number of messages currently queued */
        ...
};
```

MT-LEVEL

MT-Safe

DESCRIPTION

mq_receive( ) is used to receive the oldest of the highest priority message(s) from the message queue specified by *mqdes*. If the size of the buffer in bytes, specified by *msg_len*, is less than the **mq_msgsize** attribute of the message queue, the function fails and returns an error. Otherwise, the selected message is removed from the queue and copied to the buffer pointed to by *msg_ptr*.

If *msg_prio* is not **NULL**, the priority of the selected message is stored in the location referenced by *msg_prio*.

If the specified message queue is empty and **O_NONBLOCK** is not set in the message queue description associated with *mqdes*, (see **mq_open**(3R) and **mq_setattr**(3R)), **mq_receive( )** blocks, waiting until a message is enqueued on the message queue, or until **mq_receive( )** is interrupted by a signal. If more than one process (or thread) is waiting to receive a message when a message arrives at an empty queue, then the process of highest priority that has been waiting the longest is selected to receive the message. If the specified message queue is empty and **O_NONBLOCK** is set in the message queue description associated with *mqdes*, no message is removed from the queue, and **mq_receive( )** returns an error.

RETURN VALUES

Upon successful completion, **mq_receive( )** returns the length of the selected message in bytes and the message will have been removed from the queue. Otherwise, no message is removed from the queue, the function returns a value of **-1**, and sets **errno** to indicate the error condition.

ERRORS

EAGAIN	**O_NONBLOCK** was set in the message description associated with *mqdes*, and the specified message queue is empty.
EBADF	*mqdes* is not a valid message queue descriptor open for reading.
EMSGSIZE	*msg_len* is less than the message size attribute of the message queue.
EINTR	**mq_receive( )** operation was interrupted by a signal.

ENOSYS **mq_receive( )** is not supported by this implementation.

SEE ALSO **mq_open**(3R), **mq_send**(3R), **mq_setattr**(3R)

BUGS In Solaris 2.5, these functions always return **–1** and set **errno** to **ENOSYS,** because this release does not support the Message Passing option. It is our intention to provide support for these interfaces in future releases.