



Full credit is given to the above companies including the OS that this PDF file was generated!

Rocky Enterprise Linux 9.2 Manual Pages on command 'mbrtowc.3'

\$ man mbrtowc.3

MBRTOWC(3) Linux Programmer's Manual MBRTOWC(3)

NAME

 mbrtowc - convert a multibyte sequence to a wide character

SYNOPSIS

```
#include <wchar.h>

size_t mbrtowc(wchar_t *pwc, const char *s, size_t n, mbstate_t *ps);
```

DESCRIPTION

The main case for this function is when `s` is not `NULL` and `pwc` is not `NULL`. In this case, the `mbrtowc()` function inspects at most `n` bytes of the multibyte string starting at `s`, extracts the next complete multibyte character, converts it to a wide character and stores it at `*pwc`. It updates the shift state `*ps`. If the converted wide character is not `L'\0'` (the null wide character), it returns the number of bytes that were consumed from `s`. If the converted wide character is `L'\0'`, it re-sets the shift state `*ps` to the initial state and returns 0.

If the `n` bytes starting at `s` do not contain a complete multibyte character, `mbrtowc()` returns `(size_t) -2`. This can happen even if `n >= MB_CUR_MAX`, if the multibyte string contains redundant shift sequences.

If the multibyte string starting at `s` contains an invalid multibyte sequence before the next complete character, `mbrtowc()` returns `(size_t) -1` and sets `errno` to `EILSEQ`. In this case, the effects on `*ps` are undefined.

A different case is when `s` is not `NULL` but `pwc` is `NULL`. In this case, the `mbrtowc()` function behaves as above, except that it does not store the converted wide character in memory.

A third case is when `s` is `NULL`. In this case, `pwc` and `n` are ignored.

If the conversion state represented by `*ps` denotes an incomplete multibyte character conversion, the `mbrtowc()` function returns `(size_t) -1`, sets `errno` to `EILSEQ`, and leaves `*ps` in an undefined state. Otherwise, the `mbrtowc()` function puts `*ps` in the initial state and returns `0`.

In all of the above cases, if `ps` is `NULL`, a static anonymous state known only to the `mbrtowc()` function is used instead. Otherwise, `*ps` must be a valid `mbstate_t` object. An `mbstate_t` object can be initialized to the initial state by zeroing it, for example using

```
memset(&a, 0, sizeof(a));
```

RETURN VALUE

The `mbrtowc()` function returns the number of bytes parsed from the multibyte sequence starting at `s`, if a non-`L'\0'` wide character was recognized. It returns `0`, if a `L'\0'` wide character was recognized.

It returns `(size_t) -1` and sets `errno` to `EILSEQ`, if an invalid multibyte sequence was encountered. It returns `(size_t) -2` if it couldn't parse a complete multibyte character, meaning that `n` should be increased.

ATTRIBUTES

For an explanation of the terms used in this section, see [attributes\(7\)](#).

[attributes\(7\)](#).

??

?Interface ? Attribute ? Value ?

??

?mbrtowc() ? Thread safety ? MT-Unsafe race:mbrtowc/!ps ?

??

CONFORMING TO

POSIX.1-2001, POSIX.1-2008, C99.

NOTES

The behavior of `mbrtowc()` depends on the `LC_CTYPE` category of the current locale.

SEE ALSO

`mbsinit(3)`, `mbsrtowcs(3)`

COLOPHON

This page is part of release 5.10 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/>.

GNU

2019-03-06

MBRTOWC(3)