NAME

stpcpy − copy a string returning a pointer to its end

SYNOPSIS

#include <string.h>

char *stpcpy(char **dest*, const char **src*);

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

stpcpy():
Since glibc 2.10:
_POSIX_C_SOURCE >= 200809L
Before glibc 2.10:
_GNU_SOURCE

DESCRIPTION

The **stpcpy**() function copies the string pointed to by *src* (including the terminating null byte ('\0')) to the array pointed to by *dest*. The strings may not overlap, and the destination string *dest* must be large enough to receive the copy.

RETURN VALUE

stpcpy() returns a pointer to the **end** of the string *dest* (that is, the address of the terminating null byte) rather than the beginning.

ATTRIBUTES

For an explanation of the terms used in this section, see **attributes**(7).

Interface	Attribute	Value
stpcpy()	Thread safety	MT-Safe

CONFORMING TO

This function was added to POSIX.1-2008. Before that, it was not part of the C or POSIX.1 standards, nor customary on UNIX systems. It first appeared at least as early as 1986, in the Lattice C AmigaDOS compiler, then in the GNU fileutils and GNU textutils in 1989, and in the GNU C library by 1992. It is also present on the BSDs.

BUGS

This function may overrun the buffer *dest*.

EXAMPLE

For example, this program uses **stpcpy**() to concatenate **foo** and **bar** to produce **foobar**, which it then prints.

```
#define _GNU_SOURCE
#include <string.h>
#include <stdio.h>

int
main(void)
{
    char buffer[20];
    char *to = buffer;

    to = stpcpy(to, "foo");
    to = stpcpy(to, "bar");
    printf("%s\n", buffer);
}
```

SEE ALSO

bcopy(3), **memccpy**(3), **memcpy**(3), **memmove**(3), **stpncpy**(3), **strcpy**(3), **string**(3), **wcpcpy**(3)

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/.