

NAME

`putc`, `putchar`, `fputc`, `putw` — put character or word on a stream

SYNOPSIS

```
#include <stdio.h>
```

```
int putc (c, stream)
```

```
char c;
```

```
FILE *stream;
```

```
putchar (c)
```

```
fputc (c, stream)
```

```
FILE *stream;
```

```
putw (w, stream)
```

```
FILE *stream;
```

DESCRIPTION

Putc appends the character *c* to the named output *stream*. It returns the character written.

Putchar(c) is defined as *putc(c, stdout)*.

Fputc behaves like *putc*, but is a genuine function rather than a macro. It may be used to save on object text.

Putw appends word (i.e. `int`) *w* to the output *stream*. It returns the word written. *Putw* neither assumes nor causes special alignment in the file.

The standard stream *stdout* is normally buffered if and only if the output does not refer to a terminal; this default may be changed by *setbuf(3S)*. The standard stream *stderr* is by default unbuffered unconditionally, but use of *freopen(3S)* will cause it to become unbuffered; *setbuf*, again, will set the state to whatever is desired. When an output stream is unbuffered information appears on the destination file or terminal as soon as written; when it is buffered many characters are saved up and written as a block. See also *fflush(3S)*.

SEE ALSO

`putc(3S)`, `fopen(3S)`, `getc(3S)`, `puts(3S)`, `printf(3S)`, `fwrite(3S)`, `ferror(3S)`

DIAGNOSTICS

These functions return the constant EOF upon error. Since this is a good integer, *ferror(3S)* should be used to detect *putw* errors.

BUGS

Because it is implemented as a macro, *putc* treats a *stream* argument with side effects improperly. In particular `'putc(c, *f++);'` doesn't work sensibly.