



**Full credit is given to the above companies including the Operating System (OS) that this PDF file was generated!**

### ***Rocky Enterprise Linux 9.2 Manual Pages on command 'outl.2'***

**\$ man outl.2**

OUTB(2)                      Linux Programmer's Manual                      OUTB(2)

#### **NAME**

outb, outw, outl, outsb, outsw, outsl, inb, inw, inl, insb, insw, insl, outb\_p, outw\_p, outl\_p, inb\_p, inw\_p, inl\_p - port I/O

#### **SYNOPSIS**

```
#include <sys/io.h>

unsigned char inb(unsigned short port);
unsigned char inb_p(unsigned short port);
unsigned short inw(unsigned short port);
unsigned short inw_p(unsigned short port);
unsigned int inl(unsigned short port);
unsigned int inl_p(unsigned short port);
void outb(unsigned char value, unsigned short port);
void outb_p(unsigned char value, unsigned short port);
void outw(unsigned short value, unsigned short port);
void outw_p(unsigned short value, unsigned short port);
void outl(unsigned int value, unsigned short port);
void outl_p(unsigned int value, unsigned short port);
void insb(unsigned short port, void *addr,
          unsigned long count);
void insw(unsigned short port, void *addr,
          unsigned long count);
void insl(unsigned short port, void *addr,
```

```

        unsigned long count);

void outsb(unsigned short port, const void *addr,
        unsigned long count);

void outsw(unsigned short port, const void *addr,
        unsigned long count);

void outsl(unsigned short port, const void *addr,
        unsigned long count);

```

## DESCRIPTION

This family of functions is used to do low-level port input and output. The `out*` functions do port output, the `in*` functions do port input; the `b`-suffix functions are byte-width and the `w`-suffix functions word-width; the `_p`-suffix functions pause until the I/O completes.

They are primarily designed for internal kernel use, but can be used from user space.

You must compile with `-O` or `-O2` or similar. The functions are defined as inline macros, and will not be substituted in without optimization enabled, causing unresolved references at link time.

You use `ioperm(2)` or alternatively `iopl(2)` to tell the kernel to allow the user space application to access the I/O ports in question. Failure to do this will cause the application to receive a segmentation fault.

## CONFORMING TO

`outb()` and friends are hardware-specific. The value argument is passed first and the port argument is passed second, which is the opposite order from most DOS implementations.

## SEE ALSO

`ioperm(2)`, `iopl(2)`

## 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/>.