### NAME

cpio − copy file archives in and out

### SYNOPSIS

**cpio −o**[**v**]

**cpio −i**[**drtuv**] [ pattern ]

**cpio −p**[**dlruv**] [ pattern ] directory

### DESCRIPTION

*Cpio −o* (copy out) reads the standard input for a list of pathnames and copies those files onto the standard output together with pathname and status information.

*Cpio −i* (copy in) extracts from the standard input, which is the product of a previous *“cpio −o”*, files whose names are selected by a *pattern* given in the name-generating syntax of *sh*(I). The*pattern* meta-characters ‘?’, ‘*’, ‘[...]’ will match ‘/’ characters. The *pattern* argument defaults to "*".

*Cpio −p* (pass) copies out and in in a single operation. Destination pathnames are interpreted relative to the named *directory.*

The options are:

**d** *Directories* are to be created as needed.

**r** Interactively *rename* files. If the user types a null line, the file is skipped.

**t** Print a *table of contents* of the input. No files are created.

**u** Copy *unconditionally* (normally, an older file will not replace a newer file with the same name).

**v** *Verbose:* causes a list of file names to be printed. When used with the **t** option, the table of contents looks like an ‘‘ls −l’’ (see *ls*(I)).

**l** Whenever possible, link files rather than copying them. Usable only with the **−p** option.

**m** Retain previous file modified time (only for the super-user).

The first example below copies the contents of a directory into an archive; the second duplicates a directory hierarchy:

```
ls | cpio −o >/dev/mt0

chdir olddir
find . −print | cpio −pdl newdir
```

### SEE ALSO

ar(I), cpio(V)

### BUGS

Path names are restricted to 128 characters.
If there are too many unique linked files, the program runs out of memory to keep track of them and subsequent linking information is lost.