NAME

cpio − copy files to and from archives

SYNOPSIS

cpio {−o|−−create} [−0acvABLV] [−C bytes] [−H format] [−M message] [−O [[user@]host:]archive] [−F [[user@]host:]archive] [−−file=[[user@]host:]archive] [−−format=format] [−−message=message] [−−null] [−−reset-access-time] [−−verbose] [−−dot] [−−append] [−−block-size=blocks] [−−dereference] [−−io-size=bytes] [−−quiet] [−−force−local] [−−help] [−−version] < name-list [> archive]

cpio {−i|−−extract} [−bcdfmnrtsuvBSV] [−C bytes] [−E file] [−H format] [−M message] [−R [user][:.][group]] [−I [[user@]host:]archive] [−F [[user@]host:]archive] [−−file=[[user@]host:]archive] [−−make-directories] [−−nonmatching] [−−preserve-modification-time] [−−numeric-uid-gid] [−−rename] [−−list] [−−swap-bytes] [−−swap] [−−dot] [−−unconditional] [−−verbose] [−−block-size=blocks] [−−swap-halfwords] [−−io-size=bytes] [−−pattern-file=file] [−−format=format] [−−owner=[user][:.][group]] [−−no-preserve-owner] [−−message=message] [−−force−local] −−absolute−filenames] [−−sparse] [−−only−verify−crc] [−−quiet] [−−help] [−−version] [pattern...] [< archive]

cpio {−p|−−pass-through} [−0adlmuvLV] [−R [user][:.][group]] [−−null] [−−reset-access-time] [−−make-directories] [−−link] [−−quiet] [−−preserve-modification-time] [−−unconditional] [−−verbose] [−−dot] [−−dereference] [−−owner=[user][:.][group]] [−−no-preserve-owner] [−−sparse] [−−help] [−−version] destination-directory < name-list

DESCRIPTION

This manual page documents the GNU version of **cpio**. **cpio** copies files into or out of a cpio or tar archive, which is a file that contains other files plus information about them, such as their file name, owner, timestamps, and access permissions. The archive can be another file on the disk, a magnetic tape, or a pipe. **cpio** has three operating modes.

In copy-out mode, **cpio** copies files into an archive. It reads a list of filenames, one per line, on the standard input, and writes the archive onto the standard output. A typical way to generate the list of filenames is with the **find** command; you should give **find** the −d option to minimize problems with permissions on directories that are unwritable or not searchable.

In copy-in mode, **cpio** copies files out of an archive or lists the archive contents. It reads the archive from the standard input. Any non-option command line arguments are shell globbing patterns; only files in the archive whose names match one or more of those patterns are copied from the archive. Unlike in the shell, an initial '.' in a filename does match a wildcard at the start of a pattern, and a '/' in a filename can match wildcards. If no patterns are given, all files are extracted.

In copy-pass mode, **cpio** copies files from one directory tree to another, combining the copy-out and copy-in steps without actually using an archive. It reads the list of files to copy from the standard input; the directory into which it will copy them is given as a non-option argument.

cpio supports the following archive formats: binary, old ASCII, new ASCII, crc, HPUX binary, HPUX old ASCII, old tar, and POSIX.1 tar. The binary format is obsolete because it encodes information about the files in a way that is not portable between different machine architectures. The old ASCII format is portable between different machine architectures, but should not be used on file systems with more than 65536 i-nodes. The new ASCII format is portable between different machine architectures and can be used on any size file system, but is not supported by all versions of **cpio**; currently, it is only supported by GNU and Unix System V R4. The crc format is like the new ASCII format, but also contains a checksum for each file which **cpio** calculates when creating an archive and verifies when the file is extracted from the archive. The HPUX formats are provided for compatibility with HPUX's cpio which stores device files differently.

The tar format is provided for compatibility with the **tar** program. It can not be used to archive files with names longer than 100 characters, and can not be used to archive "special" (block or character devices) files. The POSIX.1 tar format can not be used to archive files with names longer than 255 characters (less unless they have a "/" in just the right place).

By default, **cpio** creates binary format archives, for compatibility with older **cpio** programs. When extracting from archives, **cpio** automatically recognizes which kind of archive it is reading and can read archives created on machines with a different byte-order.

Some of the options to **cpio** apply only to certain operating modes; see the SYNOPSIS section for a list of which options are allowed in which modes.

OPTIONS

−0, −−null
In copy-out and copy-pass modes, read a list of filenames terminated by a null character instead of a newline, so that files whose names contain newlines can be archived. GNU **find** is one way to produce a list of null-terminated filenames.

−a, −−reset-access-time
Reset the access times of files after reading them, so that it does not look like they have just been read.

−A, −−append
Append to an existing archive. Only works in copy-out mode. The archive must be a disk file specified with the *−O* or *−F (−−file)* option.

−b, −−swap
In copy-in mode, swap both halfwords of words and bytes of halfwords in the data. Equivalent to *−sS*. Use this option to convert 32-bit integers between big-endian and little-endian machines.

−B Set the I/O block size to 5120 bytes. Initially the block size is 512 bytes.

−−block-size=BLOCK-SIZE
Set the I/O block size to BLOCK-SIZE * 512 bytes.

−c Use the old portable (ASCII) archive format.

−C IO-SIZE, −−io-size=IO-SIZE
Set the I/O block size to IO-SIZE bytes.

−d, −−make-directories
Create leading directories where needed.

−E FILE, −−pattern-file=FILE
In copy-in mode, read additional patterns specifying filenames to extract or list from FILE. The lines of FILE are treated as if they had been non-option arguments to **cpio**.

−f, −−nonmatching
Only copy files that do not match any of the given patterns.

−F, −−file=archive
Archive filename to use instead of standard input or output. To use a tape drive on another machine as the archive, use a filename that starts with ‘HOSTNAME:’. The hostname can be preceded by a username and an ‘@’ to access the remote tape drive as that user, if you have permission to do so (typically an entry in that user’s ‘˜/.rhosts’ file).

−−force-local
With *−F*, *−I*, or *−O*, take the archive file name to be a local file even if it contains a colon, which would ordinarily indicate a remote host name.

−H FORMAT, −−format=FORMAT
Use archive format FORMAT. The valid formats are listed below; the same names are also recognized in all-caps. The default in copy-in mode is to automatically detect the archive format, and in copy-out mode is "bin".

bin The obsolete binary format.

odc The old (POSIX.1) portable format.

newc The new (SVR4) portable format, which supports file systems having more than 65536 i-nodes.

crc The new (SVR4) portable format with a checksum added.

tar The old tar format.

ustar The POSIX.1 tar format. Also recognizes GNU **tar** archives, which are similar but not identical.

hpbin The obsolete binary format used by HPUX's cpio (which stores device files differently).

hpodc The portable format used by HPUX's cpio (which stores device files differently).

−i, −−extract
Run in copy-in mode.

−I archive
Archive filename to use instead of standard input. To use a tape drive on another machine as the archive, use a filename that starts with 'HOSTNAME:'. The hostname can be preceded by a username and an '@' to access the remote tape drive as that user, if you have permission to do so (typically an entry in that user's '~/.rhosts' file).

−k Ignored; for compatibility with other versions of **cpio**.

−l, −−link
Link files instead of copying them, when possible (usable only with the *−p* option).

−L, −−dereference
Dereference symbolic links (copy the files that they point to instead of copying the links).

−m, −−preserve-modification-time
Retain previous file modification times when creating files.

−M MESSAGE, −−message=MESSAGE
Print MESSAGE when the end of a volume of the backup media (such as a tape or a floppy disk) is reached, to prompt the user to insert a new volume. If MESSAGE contains the string "%d", it is replaced by the current volume number (starting at 1).

−n, −−numeric-uid-gid
In the verbose table of contents listing, show numeric UID and GID instead of translating them into names.

−−no-absolute-filenames
In copy-in mode, create all files relative to the current directory, even if they have an absolute file name in the archive.

−−no-preserve-owner
In copy-in mode and copy-pass mode, do not change the ownership of the files; leave them owned by the user extracting them. This is the default for non-root users, so that users on System V don't inadvertently give away files.

−o, −−create
Run in copy-out mode.

−O archive
Archive filename to use instead of standard output. To use a tape drive on another machine as the archive, use a filename that starts with 'HOSTNAME:'. The hostname can be preceded by a username and an '@' to access the remote tape drive as that user, if you have permission to do so (typically an entry in that user's '~/.rhosts' file).

−−only-verify-crc
When reading a CRC format archive in copy-in mode, only verify the CRC's of each file in the archive, don't actually extract the files.

−p, −−pass-through
Run in copy-pass mode.

−−quiet
Do not print the number of blocks copied.

−r, −−rename
Interactively rename files.

−R [user][:.][group], −−owner [user][:.][group]
In copy-out and copy-pass modes, set the ownership of all files created to the specified user and/or group. Either the user or the group, or both, must be present. If the group is omitted but the ":" or "." separator is given, use the given user's login group. Only the super-user can

change files' ownership.

−−sparse
In copy-out and copy-pass modes, write files with large blocks of zeros as sparse files.

−s, −−swap-bytes
In copy-in mode, swap the bytes of each halfword (pair of bytes) in the files.

−S, −−swap-halfwords
In copy-in mode, swap the halfwords of each word (4 bytes) in the files.

−t, −−list
Print a table of contents of the input.

−u, −−unconditional
Replace all files, without asking whether to replace existing newer files with older files.

−v, −−verbose
List the files processed, or with *−t*, give an 'ls −l' style table of contents listing. In a verbose table of contents of a ustar archive, user and group names in the archive that do not exist on the local system are replaced by the names that correspond locally to the numeric UID and GID stored in the archive.

−V −−dot
Print a "." for each file processed.

−−version
Print the **cpio** program version number and exit.

NAME

mt — magnetic tape manipulating program

SYNOPSIS

mt [**-f** *tapename*] *command* [*count*]

DESCRIPTION

The **mt** utility is used to give commands to a magnetic tape drive. By default **mt** performs the requested operation once. Operations may be performed multiple times by specifying *count*. Note that *tapename* must reference a raw (not block) tape device.

The available commands are listed below. Only as many characters as are required to uniquely identify a command need be specified.

weof Write *count* end-of-file marks at the current position on the tape.

smk Write *count* setmarks at the current position on the tape.

fsf Forward space *count* files.

fsr Forward space *count* records.

fss Forward space *count* setmarks.

bsf Backward space *count* files.

bsr Backward space *count* records.

bss Backward space *count* setmarks.

rdhpos Read Hardware block position. Some drives do not support this. The block number reported is specific for that hardware only. The count argument is ignored.

rdspos Read SCSI logical block position. Some drives do not support this. The count argument is ignored.

sethpos Set Hardware block position. Some drives do not support this. The count argument is interpreted as a hardware block to which to position the tape.

setspos Set SCSI logical block position. Some drives do not support this. The count argument is interpreted as a SCSI logical block to which to position the tape.

rewind Rewind the tape (Count is ignored).

offline, **rewoffl**
Rewind the tape and place the tape unit off-line (Count is ignored).

erase Erase the tape. A count of 0 disables long erase, which is on by default.

retension Re-tension the tape (one full wind forth and back, Count is ignored).

status Print status information about the tape unit. For SCSI magnetic tape devices, the current operating modes of density, blocksize, and whether compression is enabled is reported. The current state of the driver (what it thinks that it is doing with the device) is reported. If the driver knows the relative position from BOT (in terms of filemarks and records), it prints that. Note that this information is not definitive (only BOT, End of Recorded Media, and hardware or SCSI logical block position (if the drive supports such) are considered definitive tape positions).

errstat Print (and clear) error status information about this device. For every normal operation (e.g., a read or a write) and every control operation (e.g,, a rewind), the driver stores up the last command executed and it's associated status and any residual counts (if any). This command retrieves and prints this information. If possible, this also clears any latched error information.

blocksize Set the block size for the tape unit. Zero means variable-length blocks.

density Set the density for the tape unit. For the density codes, see below. The density value could be given either numerically, or as a string, corresponding to the "Reference" field. If the string is abbreviated, it will be resolved in the order shown in the table, and the first matching entry will be used. If the given string and the resulting canonical density name do not match exactly, an informational message is printed about what the given string has been taken for.

geteotmodel
Fetch and print out the current EOT filemark model. The model states how many filemarks will be written at close if a tape was being written.

seteotmodel
Set (from the *count* argument) and print out the current and EOT filemark model. Typically this will be *2* filemarks, but some devices (typically QIC cartridge drives) can only write *1* filemark. Currently you can only choose a value of *1* or *2*.

eom Forward space to end of recorded medium (Count is ignored).

eod Forward space to end of data, identical to **eom**.

comp Set compression mode. There are currently several possible values for the compression mode:

off	Turn compression off.
on	Turn compression on.
none	Same as *off*.
enable	Same as *on*.
IDRC	IBM Improved Data Recording Capability compression (0x10).
DCLZ	DCLZ compression algorithm (0x20).

In addition to the above recognized compression keywords, the user can supply a numeric compression algorithm for the tape drive to use. In most cases, simply turning the compression 'on' will have the desired effect of enabling the default compression algorithm supported by the drive. If this is not the case (see the **status** display to see which compression algorithm is currently in use), the user can manually specify one of the supported compression keywords (above), or supply a numeric compression value.

If a tape name is not specified, and the environment variable TAPE does not exist; **mt** uses the device /dev/nsa0.

The **mt** utility returns a 0 exit status when the operation(s) were successful, 1 if the command was unrecognized, and 2 if an operation failed.

The following density table was taken from the 'Historical sequential access density codes' table (A-1) in Revision 11 of the SCSI-3 Stream Device Commands (SSC) working draft, dated November 11, 1997.

The different density codes are as follows:

```
   0x0   default for device
   0xE   reserved for ECMA
```

Value	Width mm	Width in	Tracks	Density bpmm	Density bpi	Code	Type	Reference	Note
0x01	12.7	(0.5)	9	32	(800)	NRZI	R	X3.22-1983	2
0x02	12.7	(0.5)	9	63	(1,600)	PE	R	X3.39-1986	2
0x03	12.7	(0.5)	9	246	(6,250)	GCR	R	X3.54-1986	2
0x05	6.3	(0.25)	4/9	315	(8,000)	GCR	C	X3.136-1986	1
0x06	12.7	(0.5)	9	126	(3,200)	PE	R	X3.157-1987	2
0x07	6.3	(0.25)	4	252	(6,400)	IMFM	C	X3.116-1986	1
0x08	3.81	(0.15)	4	315	(8,000)	GCR	CS	X3.158-1987	1
0x09	12.7	(0.5)	18	1,491	(37,871)	GCR	C	X3.180	2

0x0A	12.7	(0.5)	22	262	(6,667)	MFM	C	X3B5/86-199	1
0x0B	6.3	(0.25)	4	63	(1,600)	PE	C	X3.56-1986	1
0x0C	12.7	(0.5)	24	500	(12,690)	GCR	C	HI-TC1	1,6
0x0D	12.7	(0.5)	24	999	(25,380)	GCR	C	HI-TC2	1,6
0x0F	6.3	(0.25)	15	394	(10,000)	GCR	C	QIC-120	1,6
0x10	6.3	(0.25)	18	394	(10,000)	GCR	C	QIC-150	1,6
0x11	6.3	(0.25)	26	630	(16,000)	GCR	C	QIC-320	1,6
0x12	6.3	(0.25)	30	2,034	(51,667)	RLL	C	QIC-1350	1,6
0x13	3.81	(0.15)	1	2,400	(61,000)	DDS	CS	X3B5/88-185A	5
0x14	8.0	(0.315)	1	1,703	(43,245)	RLL	CS	X3.202-1991	5
0x15	8.0	(0.315)	1	1,789	(45,434)	RLL	CS	ECMA TC17	5
0x16	12.7	(0.5)	48	394	(10,000)	MFM	C	X3.193-1990	1
0x17	12.7	(0.5)	48	1,673	(42,500)	MFM	C	X3B5/91-174	1
0x18	12.7	(0.5)	112	1,673	(42,500)	MFM	C	X3B5/92-50	1
0x19	12.7	(0.5)	128	2,460	(62,500)	RLL	C	DLTapeIII	6,7
0x1A	12.7	(0.5)	128	3,214	(81,633)	RLL	C	DLTapeIV(20)	6,7
0x1B	12.7	(0.5)	208	3,383	(85,937)	RLL	C	DLTapeIV(35)	6,7
0x1C	6.3	(0.25)	34	1,654	(42,000)	MFM	C	QIC-385M	1,6
0x1D	6.3	(0.25)	32	1,512	(38,400)	GCR	C	QIC-410M	1,6
0x1E	6.3	(0.25)	30	1,385	(36,000)	GCR	C	QIC-1000C	1,6
0x1F	6.3	(0.25)	30	2,666	(67,733)	RLL	C	QIC-2100C	1,6
0x20	6.3	(0.25)	144	2,666	(67,733)	RLL	C	QIC-6GB(M)	1,6
0x21	6.3	(0.25)	144	2,666	(67,733)	RLL	C	QIC-20GB(C)	1,6
0x22	6.3	(0.25)	42	1,600	(40,640)	GCR	C	QIC-2GB(C)	?
0x23	6.3	(0.25)	38	2,666	(67,733)	RLL	C	QIC-875M	?
0x24	3.81	(0.15)	1	2,400	(61,000)		CS	DDS-2	5
0x25	3.81	(0.15)	1	3,816	(97,000)		CS	DDS-3	5
0x26	3.81	(0.15)	1	3,816	(97,000)		CS	DDS-4	5
0x27	8.0	(0.315)	1	3,056	(77,611)	RLL	CS	Mammoth	5
0x28	12.7	(0.5)	36	1,491	(37,871)	GCR	C	X3.224	1
0x29	12.7	(0.5)							
0x2A									
0x2B	12.7	(0.5)	3	?	?	?	C	X3.267	5
0x41	12.7	(0.5)	208	3,868	(98,250)	RLL	C	DLTapeIV(40)	6,7
0x48	12.7	(0.5)	448	5,236	(133,000)	PRML	C	SDLTapeI(110)	6,8
0x49	12.7	(0.5)	448	7,598	(193,000)	PRML	C	SDLTapeI(160)	6,8

Code	Description
NRZI	Non return to zero, change on ones
GCR	Group code recording
PE	Phase encoded
IMFM	Inverted modified frequency modulation
MFM	Modified frequency modulation
DDS	DAT data storage
RLL	Run length limited
PRML	Partial Response Maximum Likelihood

Type	Description
R	Reel-to-reel
C	Cartridge
CS	Cassette

NOTES

1. Serial recorded.
2. Parallel recorded.
3. Old format known as QIC-11.
5. Helical scan.
6. This is not an American National Standard. The reference is based on an industry standard definition of the media format.
7. DLT recording: serially recorded track pairs (DLTapeIII and DLTapeIV(20)), or track quads (DLTapeIV(35) and DLTapeIV(40)).

8. Super DLT (SDLT) recording: 56 serially recorded logical tracks with 8 physical tracks each.

ENVIRONMENT

If the following environment variable exists, it is utilized by **mt**.

TAPE The **mt** utility checks the TAPE environment variable if the argument *tapename* is not given.

FILES

/dev/*wt*	QIC-02/QIC-36 magnetic tape interface
/dev/*sa[0-9]*	SCSI magnetic tape interface

SEE ALSO

dd(1), ioctl(2), ast(4), mtio(4), sa(4), environ(7)

HISTORY

The **mt** command appeared in 4.3BSD.

Extensions regarding the st(4) driver appeared in 386BSD 0.1 as a separate **st** command, and have been merged into the **mt** command in FreeBSD 2.1.

The former **eof** command that used to be a synonym for **weof** has been abandoned in FreeBSD 2.1 since it was often confused with **eom**, which is fairly dangerous.