NAME

cdrecord – records audio or data compact discs (CDs) from a master

SYNOPSIS

cdrecord [-general options] [-dev=*device*]... [-track options] [track1 . . . trackn]

GENERAL OPTIONS

General options must be entered before any track file name or track option.

`-atip`	Retrieves and prints out the ATIP (absolute time in pregroove) information of a CD recordable or CD rewritable media. With this option, `cdrecord` tries to retrieve the ATIP info. If the actual drive does not support to read the ATIP info, it may be that only a reduced set of information records or even nothing is displayed. Only a limited number of MMC compliant drives support reading the ATIP info. If `cdrecord` is able to retrieve the lead-in start time for the first session, it tries to decode and print the manufacturer information from the media.	
`-blank=` *`type`*	Blanks a CD-RW and exits or blanks a CD-RW before writing. The blanking type may be one of the following:	
	`all`	Blanks the entire disk. This may take a long time.
	`fast`	Minimally blanks the disk. This results in erasing the PMA, the TOC and the pregap.
	`help`	Displays a list of possible blanking types.

session	Blanks the last session.
track	Blanks a track.
trtail	Blanks the tail of a track.
unclose	Uncloses last session.
unreserve	Unreserves a reserved track.

If used together with the `-force` flag, this option may be used to blank CD-RW disks that otherwise cannot be blanked. Note that you may need to specify `-blank=`*all* because some drives do not continue with certain types of bad CD-RW disks. Note also that `cdrecord` does its best if the `-force` flag is used, but it finally depends on the drive's firmware whether the blanking operation succeeds or not.

`-checkdrive`

Checks if a driver for the current drive is present and exits. If the drive is a known drive, `cdrecord` uses exit code `0`.

`-dao`

Sets disk-at-once mode. This currently only works with MMC drives that support non-raw `Session At Once` mode.

`-debug=`*#*
`-d`

Set the *debug* value to *#* (with `-debug=`*#*) or increment the *debug* level by one (with `-`*d*). Specifying `-`*dd* is equal to `-debug=`*2*. This may help to find problems while opening a driver for `libscg` as well as with sector sizes and sector types. Using `-debug` slows down the process and may be the reason for a buffer underrun.

`-defpregap=`*#*

Sets the default pregap size for all tracks except track number `1`. This option currently only makes sense with the TEAC drive when creating track-at-once disks without the 2 second silence before each track. This option may not be available in future.

`-dev=` *`target`*

Sets the SCSI target for the CD-Recorder (see notes above). A typical device specification is `-dev=`*`6,0`*. If a filename must be provided together with the numerical target specification, the filename is implementation-specific. The correct filename in this case can be found in the system-specific manuals of the target operating system. On a FreeBSD system without CAM support, you need to use the control device (that is, `/dev/rcd0.ctl`). A correct device specification in this case may be `-dev=`*`/dev/rcd0.ctl:@`*.

On Linux, drives connected to a parallel port adapter are mapped to a virtual SCSI bus. Different adapters are mapped to different targets on this virtual SCSI bus.

If no *`dev`* option is present, `cdrecord` tries to get the device from the `CDR_DEVICE` environment.

’If the argument to the `-dev=` option does not contain the characters `,`, `/`, `@` or `:`, it is interpreted as an label name that may be found in the file `/etc/default/cdrecord`. See the FILES section for more information.

`-driver=` *`name`*

Uses a user-supplied driver name for the device. To get a list of possible drivers use `-driver=`*`help`*. The reason for the existence of this option is to allow users to use `cdrecord` with drives that are similar to supported drives but not known directly by `cdrecord`. Use this option with extreme care. If a wrong driver is used for a device, the possibility of creating corrupted disks is high. The minimum problem related to a wrong driver is that the `-speed` or `-dummy` does not work.

There is a special driver entry in the list: *`cdr_simul`*. This driver is designed to make timing

	tests at any speed or timing tests for drives that do not support the `-dummy` option. The simulation driver implements a drive with a buffer size of 1MB that can be changed with the *`CDR_SIMUL_BUFSIZE`* environment variable. The simulation driver correctly simulates even a buffer underrun condition. If the `-dummy` option is present, the simulation is not aborted in case of a buffer underrun.
`-driveropts=` *`option list`*	Sets driver-specific options. The options are specified with a comma-separated list. To get a list of valid options use `-driveropts=`*`help`* together with the *`checkdrive`* option. Currently only the *`burnproof`* option is implemented to support Buffer Underrun Proof writing with drives that use the Sanyo BURN-Proof technology.
`-dummy`	The CD-Recorder goes through all steps of the recording process, but the laser is turned off during this procedure. It is recommended to run several tests before actually writing to a compact disk if the timing and load response of the system is not known.
`-eject`	Ejects disk after doing the work. Some devices (such as Philips) need to eject the medium before creating a new disk. Doing a `-dummy` test and immediately creating a real disk would not work on these devices.
`-fix`	The disk is only fixated (that is, a TOC for a CD-Reader is written). This may be used if for some reason the disk has been written but not fixated. This option currently does not work with old TEAC drives (CD-R50S and CD-R55S).
`-nofix`	Does not fixate the disk after writing the tracks. This may be used to create an audio disk in steps. An unfixated disk usually cannot be used on a non CD-writer type drive but there are audio CD players that are able to play such a disk.

`-force`

Forces command to continue on some errors. This option currently implements some tricks that allow you to blank out bad CD-RW disks.

`-fs=#`

Sets the fifo (ring buffer) size to *#*. You may use the same method as in `dd`(1), `sdd`(1) or `star`(1). The number representing the size is taken in bytes unless otherwise specified. If a number is followed directly by the letter `b`, `k`, `m`, `s` or `f`, the size is multiplied by 512, 1024, 1024*1024, 2048 or 2352 respectively. If the size consists of numbers separated by `x` or `*`, multiplication of the two numbers is performed. Thus `-fs=10x63k` species a fifo size of 630 kBytes.

The size specified by the `-fs=` argument includes the shared memory that is needed for administration. This is at least one page of memory. If no `-fs=` option is present, `cdrecord` tries to get the fifo size value from the `-CDR_FIFOSIZE` environment. The default fifo size is currently 4 MB.

The fifo is used to increase buffering for the real-time writing process. It allows the system to run a pipe from `mkisofs` directly into `cdrecord`. If the fifo is active and a pipe from `mkisofs` into `cdrecord` is used to create a CD, `cdrecord` aborts prior to doing any modifications on the disk if `mkisofs` dies before it starts writing. The recommended fifo size is between 4 and 32 MBytes. As a rule of thumb, the fifo size should be at least equal to the size of the internal buffer of the CD-Recorder and no more than half of the physical amount of RAM available in the machine. If the fifo size is big enough, the fifo statistics prints a fifo empty count of zero and the fifo min fill is not below 20%. It is not wise to use too much space for the fifo. If you need more than 8 MB to write a CD on an idle machine, your machine is either underpowered, has hardware problems or is misconfigured. If you have buffer underruns or similar problems and observe a zero *`fifo empty count`*, you have hardware problems. The fifo size in this case is sufficient.

`-ignsize`	Ignores the known size of the medium. This option should be used with extreme care; it exists only for debugging purposes and should not be used for other reasons. It is not needed to write disks with more than the nominal capacity.
`-inq`	Does an inquiry for the drive, prints the inquiry information, and exits.
`-kdebug`*#* `-kd=`*#*	Tells the `-scg` driver to modify the kernel debug value while SCSI commands are running.
`-load`	Loads the media and exits. This only works with a tray loading mechanism but seems to be useful when using the Kodak disk transporter.
`-mcn=`*`med_cat_nr`*	Sets the Media Catalog Number of the CD to `med_cat_nr`.
`-msinfo`	Retrieves multisession information in a form suitable for `mkisofs 1.10` or later. This option only makes sense with a CD that contains at least one closed session and is appendable (not yet finally closed). Some drives create error messages if you try to get the multisession information for a disk that is not suitable for this operation.
`-multi`	Allows multisession CDs to be made. Use only for recording multisession CDs. This flag needs to be present on all sessions of a multisession disk, except when you want to create a session that is the last session on the media. The fixation allows the CD-Recorder to append additional sessions later. This is done by generating a table of contents with a link to the next program area. Media generated in this way is not 100% compatible to manufactured CDs, except for CDplus. If this option is present, the default track type is `CD-ROM XA mode 2`. The Sony drives have no

hardware support for `-CD-ROM XA mode 2`. You have to specify the `-data` option in order to create multisession disks on these drives.

Because `cdrecord` does not have a coder for converting data sectors to audio sectors, you need to force `CD-ROM` sectors by including the `-data` option if you want to record a multisession disk in `DAO/SAO` mode. Not all drives allow multisession CDs in `DAO/SAO` mode.

`-packet`	Sets packet writing mode. This is an experimental interface.
`-pktsize=`*#*	Sets the packet size to *#*, which forces fixed-packet mode. This is an experimental interface.
`-prcap`	Prints the drive capabilities for SCSI-3/mmc compliant drives as obtained from mode page `0x2A`. Values marked with `kB` use 1000 bytes as kilobyte, values marked with `KB` use 1024 bytes as Kilobyte.
`-scanbus`	Scans all SCSI devices on all SCSI busses and prints the inquiry strings. This option may be used to find the SCSI address of the CD-Recorder on a system. The numbers printed out as labels are computed as `bus * 100 + target`
`-silent`	Does not print out a status report for failed SCSI commands.
`-speed=`*#*	Sets the speed factor of the writing process to *#*, an integer, representing a multiple of the audio speed. This is about 150 KB/s for CD-ROM and about 172 KB/s for CD-Audio. If no `-speed` option is present, `cdrecord` tries to get the speed value from the `CDR_SPEED` environment. If your drive has problems with `-speed=`*2* or `-speed=`*4*, you should try `-speed=`*0*.

`-timeout=#`	Sets the default SCSI command timeout value to # seconds. The default SCSI command timeout is the minimum timeout used for sending SCSI commands. If a SCSI command fails due to a timeout, you may try to raise the default SCSI command timeout above the timeout value of the failed command. If the command runs correctly with a raised command timeout, report the better timeout value and the corresponding command to the author of the program. If no `-timeout` option is present, a default timeout of 40 seconds is used.
`-toc`	Retrieves and prints out the table of contents or PMA of a CD. With this option, `cdrecord` works with CD-R drives and with CD-ROM drives.
`-useinfo`	Uses `*.inf` files to overwrite audio options. If this option is used, the pregap size information is read from the `*.inf` file that is associated with the file that contains the audio data for a track.
`-v`	Increments the level of general verbosity by one. This displays the progress of the write process.
`-V`	Increments the verbose level in the SCSI command transport by one. This helps to debug problems that occur in the CD-Recorder during the write process. If you get incomprehensible error messages, use this option for more detailed output. `-VV` shows data buffer content. Using the `-V` or `-VV` option slows down the process and may cause a buffer underrun.
`-version`	Prints version information and exits.
`-waiti`	Waits for input to become available on standard input before trying to open the SCSI driver. This allows `cdrecord` to read its input from a pipe even when writing additional sessions to a multisession disk. When writing another session to a multisession disk, `mkisofs` needs to read the old session from the

device before writing output. This cannot be done if `cdrecord` opens the SCSI driver at the same time.

TRACK OPTIONS

Track options may be mixed with track file names.

`-audio` If this flag is present, all subsequent tracks are written in CD-DA (similar to Red Book) audio format. The file with data for these tracks should contain stereo, 16-bit digital audio with 44100 samples/s. The byte order should be the following: MSB left, LSB left, MSB right, LSB right, MSB left and so on. The track should be a multiple of 2352 bytes. It is not possible to put the master image of an audio track on a raw disk because data is read in multiple of 2352 bytes during the recording process.

If a filename ends in `.au` or `.wav`, the file is considered to be a structured audio data file. The `cdrecord` command assumes that the file in this case is a Microsoft `.wav` file and extracts the audio data from the files by skipping over the non-audio header information. In all other cases, `cdrecord` only works correctly if the audio data stream does not have any header. Because many structured audio files are not an integral number of blocks (1/75th second) in length, it is often necessary to specify the `-pad` option as well. The `cdrecord` command recognizes that audio data in a `.wav` file is stored in Intel (little-endian) byte order, and automatically byte-swaps the data if the CD recorder requires big-endian data. The `cdrecord` command rejects any audio file that does not match the Red Book requirements of 16-bit stereo samples in pcm coding at 44100 samples/second.

Using other structured audio data formats as input to `cdrecord` usually works if the structure of the data is the structure described above (raw pcm data in big-endian byte order). However, if the data format includes a header, you hear a click at the start of a track.

If neither `-data` nor `-audio` have been specified, `cdrecord` defaults to `-audio` for all filenames that end in `.au` or `.wav` and to `-data` for all other files.

`-cdi` If this flag is present, all subsequent tracks are written in CDI format. The data is a multiple of 2048 bytes.

`-data`	If this flag is present, all subsequent tracks are written in CD-ROM mode 1 (Yellow Book) format. The data is a multiple of 2048 bytes. The file with track data should contain an ISO-9660" or "Rock Ridge" filesystem image (see `mkisofs` for more details). If the track data is an UFS filesystem image, fragment size should be set to 2 KB or more to allow CR-drives with 2 KB sector size to be used for reading. If no other flag is present, `-data` is the default. If neither `-data` nor `-audio` have been specified, `cdrecord` defaults to `-audio` for all filenames that end in `.au` or `.wav` and to `-data` for all other files.
`-index=` *`list`*	Sets an index list for the next track. An index list is a comma separated list of numbers counting from index 1. The first entry in this list must contain a 0; the following numbers must be an ascending list of numbers (counting in 1/75 seconds) that represent the start of the indices. An index list in the form: 0,7500,15000 sets index 1 to the start of the track, index 2 to 100 seconds from the start of the track and index 3 to 200 seconds from the start of the track.
`-isosize`	Uses the ISO-9660 file system size as the size of the next track. This option is needed if you want to read the image of a track from a raw disk partition or on a master CD. In the first case the option `-isosize` is needed to limit the size of the CD to the size of the ISO filesystem. In the second case the option `-isosize` is needed to prevent `cdrecord` from reading the two run-out blocks that are appended by each CD-recorder in track-at-once mode. These two run-out blocks cannot be read and would cause a buffer underrun that would cause a defective copy. Do not use this option if `cdrecord` reads the track data from *`stdin`*. This option currently cannot be used to determine the size of a file system if the multisession option is present.
`-isrc=` *`Circumambulate`*	Sets the International Standard Recording Number for the next track to *`Circumambulate`*.
`-mode2`	If this flag is present, all subsequent tracks are written in CD-ROM mode 2 format. The data is a multiple of 2048 bytes.

`-pad`	If the track is a data track, 15 sectors of zeroed data is added to the end of this and each subsequent data track. In this case, the `-pad` option is superseded by the `-padsize=` option. It remains, however, as a shorthand for `-padsize=15s`. If the `-pad` option refers to an audio track, `cdrecord` pads the audio data to be a multiple of 2352 bytes. The audio data padding is done with binary zeroes which is equal to absolute silence. The `-pad` option remains valid until disabled by `-nopad`.
`-nopad`	Does not pad the following tracks. This is the default.
`-padsize=#`	Sets the amount of data to be appended as padding to the next track to #. Opposed to the behavior of the `-pad` option, the value for `-padsize=` is reset to zero for each new track. See the `-fs=` option for possible arguments. Use this option if your CD-drive is not able to read the last sectors of a track or if you want to be able to read the CD on a Linux system with the ISO-9660 filesystem read-ahead bug. If an empty file is used for track data, this option may be used to create a disk that is entirely made of padding.
`-preemp`	If this flag is present, all TOC entries for subsequent audio tracks indicate that the audio data has been sampled with 50/15 µsec preemphasis. The data, however is not modified during the process of transferring from file to disk. This option has no affect on data tracks.
`-nopreemp`	If this flag is present, all TOC entries for subsequent audio tracks indicates that the audio data has been mastered with linear data. This is the default.
`-pregap=#`	Sets the pregap size for the next track. This option currently only makes sense with the TEAC drive when creating track-at-once disks without the 2 second silence before each track. This option may go away in future.
`-shorttrack`	Allows all subsequent tracks to violate the Read Book track-length standard, which requires a minimum track length of 4 seconds. This option is only useful when used in DAO

mode. Not all drives support this feature. The drive must accept the resulting CUE sheet.

`-noshort-track` Enforces to the Red Book track length standard. Tracks must be at least 4 seconds.

`-swab` If this flag is present, audio data is assumed to be in byte-swapped (little-endian) order. Some types of CD-Writers, for example, Yamaha, Sony and the SCSI-3/mmc drives require audio data to be presented in little-endian order, while other writers require audio data to be presented in the big-endian (network) byte order normally used by the SCSI protocol. The `cdrecord` command knows if a CD-Recorder needs audio data in big- or little-endian order, and corrects the byte order of the data stream to match the needs of the recorder. You only need the *`swab`* flag if your data stream is in Intel (little-endian) byte order.

Note that the verbose output of `cdrecord` shows you if swapping is necessary to make the byte order of the input data fit the required byte order of the recorder. The `cdrecord` command does not show you if the *`swab`* flag was actually present for a track.

`-tsize=`*`#`* If the master image for the next track has been stored on a raw disk, use this option to specify the valid amount of data on this disk. If the image of the next track is stored in a regular file, the size of that file is taken to determine the length of this track.

If the track contains an ISO 9660 filesystem image use the `-isosize` option to determine the length of that filesystem image. In *`disk-at-once`* mode and with some drives that use the TEAC programming interface, even in *`track-at-once`* mode, `cdrecord` needs to know the size of each track before starting to write the disk. The `cdrecord` command now checks this and aborts before starting to write. If this happens you need to run `mkisofs -print-size` and use the output as an argument to the `-tsize=` option of `cdrecord`. See the `-fs=` option for possible arguments.

`-xa1`	If this flag is present, all subsequent tracks are written in CD-ROM XA mode 1 format. The data is a multiple of 2048 bytes.
`-xa2`	If this flag is present, all subsequent tracks are written in CD-ROM XA mode 2 format. The data is a multiple of 2048 bytes.

DESCRIPTION

The `cdrecord` command is used to record data or audio compact discs on an Orange Book CD-Recorder.

The device refers to *`scsibus`*/*`target`*/*`lun`* of the CD-Recorder. Operating systems use a library simulation of the SCSI general driver. Possible syntax is: `-dev=` *`scsibus`*,*`target`*,*`lun`* or `-dev=` *`target`*,*`lun`*. In the latter case, the CD-Recorder has to be connected to the default SCSI bus of the machine. The parameters *`scsibus`*, *`target`* and *`lun`* are integer numbers. Some operating systems or SCSI transport implementations may require to specify a filename in addition. In this case the correct syntax for the device is: `-dev=` *`devicename`*:*`scsibus`*,*`target`*,*`lun`* or `-dev=` *`devicename`*:*`target`*,*`lun`*. If the name of the device node that has been specified on such a system refers to exactly one SCSI device, a shorthand in the form `-dev=` *`devicename`*:*`@`* or `-dev=` *`devicename`*:*`@`*,*`lun`* may be used.

To access remote SCSI devices, prepend the SCSI device name by a remote-device indicator. The remote device indicator is either `-REMOTE:`*`user@host:`* or `-REMOTE:`*`host:`* A valid remote SCSI device name may be `-REMOTE:`*`user@host:`* to allow remote SCSI bus scanning or `-REMOTE:`*`user@host:1,0,0`* to access the SCSI device at *`host`* connected to SCSI bus # 1,target 0, lun 0.

To make `cdrecord` portable to all UNIX platforms, the syntax `-dev=` *`devicename`*:*`scsibus`*,*`target`*,*`lun`* is preferred as it hides operating-system specific knowledge about device names from the user. A specific operating system must not necessarily support a way to specify a real device file name nor a way to specify *`scsibus`*,*`target`*,*`lun`*.

The default SCSI bus on the machine is *`scsibus 0`*. Watch the boot messages or look at`-/var/adm/messages` for more information about the SCSI configuration of your machine. If you have problems figuring out what values for *scsibus*,*target*,*lun* should be used, try the `-scanbus` option of `cdrecord`.

If a file `/etc/default/cdrecord` exists, the parameter to the `-dev=` option may also be a drive name label in said file (see FILES section).

On SVr4 compliant systems, `cdrecord` uses the real time class to get the highest scheduling priority that is possible (higher than all kernel processes). On systems with POSIX realtime scheduling, `cdrecord` also uses real time scheduling but may not be able to gain a priority that is higher than all kernel processes.

In *track-at-once* mode, each track corresponds to a single file that contains the prepared data for that track. If the argument is `-`, standard input is used for that track. Only one track may be taken from `stdin`.

EXAMPLES

For all examples below, it is be assumed that the CD-Recorder is connected to the primary SCSI bus of the machine. The SCSI target id is set to 2.

To record a pure CD-ROM at double speed using data from the file *cdimage.raw*:

```
# cdrecord −v speed=2 dev=2,4,0 cdimage.raw
```

To create an image for a ISO 9660 filesystem with Rock Ridge extensions:

```
# mkisofs −R −o cdimage.raw /home/joerg/master/tree
```

To check the resulting file before writing to CD on Linux:

```
# mount cdimage.raw −r −t iso9660 −o loop /mnt
# ls −lR /mnt     umount /mnt
# umount /mnt
```

If the overall speed of the system is sufficient and the structure of the filesystem is not too complex, `cdrecord` runs without creating an image of the ISO 9660 filesystem. Simply run the pipeline:

```
# mkisofs −R /master/tree | cdrecord −v fs=6m speed=2 dev=2,4,0 -
```

The recommended minimum fifo size for running this pipeline is 4 MBytes. As the default fifo size is 4 MB, the `-fs=` option needs only be present if you want to use a different fifo size. If your system is loaded, you should run `mkisofs` in the real-time class. To raise the priority of `mkisofs` replace the command

```
# mkisofs –R /master/tree
```

To record a pure CD-DA (audio) at single speed with each track contained in a file named `track01.cdaudio`, `track02.cdaudio`, etc:

```
# cdrecord –v speed=1 dev=2,4,0 -audio track*.cdaudio
```

To check if you can use double speed for the example above, use the dummy write option:

```
# cdrecord –v –dummy speed=2 dev=2,4,0 –audio track*.cdaudio
```

To handle drives that need to know the size of a track before starting to write, first run

```
# mkisofs -R -q -print-size /master/tree
```

and then run

```
# mkisofs -R /master/tree | cdrecord speed=2
dev=2,4,0 tsize=XXXs -
```

where *XXX* is replaced by the output of the previous run of `mkisofs`.

To copy an audio CD in the most accurate way, first run

```
# cdda2wav -v255 -D2,4,0 -B -Owav
```

and then run

```
# cdrecord -v dev=2,4,0 -dao -useinfo *.wav
```

ENVIRONMENT

CDR_DEVICE	This may either hold a device identifier that is suitable to the open call of the SCSI transport library or a label in the file `/etc/default/cdrecord`.
CDR_FIFOSIZE	Sets the default size of the fifo (see also the `-fs=` option).
CDR_SPEED	Sets the default speed value for writing (see also the `-speed` option).

FILES

Default values can be set for the following options in `/etc/default/cdrecord`.

CDR_DEVICE	This may either hold a device identifier that is suitable to the open call of the SCSI transport library or a label in the file `/etc/default/cdrecord` that it allows to identify a specific drive on the system.
CDR_SPEED	Sets the default speed value for writing (see also the `-speed` option). For example, `CDR_SPEED=2`.
CDR_FIFOSIZE	Sets the default size of the fifo (see also the `-fs=` #option). For example, `CDR_FIFOSIZE=8m`.
Any other label	is an identifier for a specific drive on the system. Such an identifier may not contain the four characters comma (`,`), slash (`/`), at (`@`), or colon (`:`).

Each line that follows a label contains a TAB-separated list of items. Currently, three items are recognized: the SCSI ID of the drive, the default speed that should be used for this drive and the default fifo size that should be used for this drive. The values for `-speed` and `-fifosize` may be set to `-1` to tell `cdrecord` to use the global defaults. A typical line may look this way:

```
# teac1= 0,5,0 4 8m
# yamaha= 1,6,0 -1 -1
```

This tells `cdrecord` that a drive named *`teac1`* is at `scsibus 0, target 5, lun 0` and should be used with `-speed=`*4* and a `-fs=`*`8 MB`*. A second drive may be found at `scsibus 1, target 6, lun 0` and uses the default speed and the default fifo size.

NOTES

Disks made in *`track-at-once`* mode are not suitable as masters for direct mass production by CD manufacturers. You need the *`disk-at-once`* option to record such disks. Nevertheless the disks made in *track-at-once* mode normally are read in all CD players. Some old audio CD players however may produce a two second click between two audio tracks.

The minimal size of a track is 4 seconds or 300 sectors. If you write smaller tracks, the CD-Recorder adds dummy blocks. This is not an error, even though the SCSI-error message looks this way.

The `cdrecord` command has been tested on an upgraded Philips CDD-521 recorder at single and double speed on a SparcStation 20/502 with no problems; slower computer systems should work also. The newer Philips/HP/Plasmon/Grundig drives as well as Yamaha CDR-100 and CDR-102 work also. The Plasmon RF-4100 works but has not been tested in multisession. A Philips CDD-521 that has not been upgraded does not work. The Sony CDU-924 has been tested, but does not support XA-mode2 in hardware. The Sony therefore cannot create conforming multisession disks. The Ricoh RO-1420C works, but some people seem to have problems using them with `-speed=`*2*; try `-speed=`*0* in this case.

The Yamaha CDR-400 and all new SCSI-3/mmc conforming drives are supported in single and multisession.

You should run several tests in all supported speeds of your drive with the `-dummy` option turned on if you are using `cdrecord` on an unknown system. Writing a CD is a realtime process. NFS does not always deliver constantly the needed data rates. If you want to use `cdrecord` with CD-images that are located on an NFS mounted filesystem, be sure that the fifo size is big enough. It is recommended that you leave the system as lightly loaded as possible while writing a CD. If you want to make sure that buffer underruns are not caused by your source disk, use the command:

```
# cdrecord -dummy dev=2,4,0 padsize=600m /dev/null
```

to create a disk that is entirely made of dummy data. The `cdrecord` command needs to run as root to get access to the */dev/scg?* device nodes and to be able to lock itself into memory.

If you do not want to allow users to become root on your system, `cdrecord` may safely be installed suid root. This allows all users or a group of users with no root privileges to use `cdrecord`. The `cdrecord` command in this case checks to determine if the real user would have been able to read the specified files. To give all user access to use `cdrecord`, enter:

```
# chown root /usr/local/bin/cdrecord
# chmod 4711 /usr/local/bin/cdrecord
```

To give a restricted group of users access to `cdrecord`, enter:

```
# chown root /usr/local/bin/cdrecord
# chgrp cdburners /usr/local/bin/cdrecord
# chmod 4710 /usr/local/bin/cdrecord
```

and add a group *cdburners* on your system.

Never give write permissions for the */dev/scg?* devices to nonroot users unless you would allow anybody to read/write/format all your disks.

Do not connect old drives that do not support disconnect/reconnect to either the SCSI bus that is connected to the CD-Recorder or the source disk.

A compact disc can have no more than 99 tracks.

When creating a disc with both audio and data tracks, the data should be on track 1 otherwise you should create a CDplus disk which is a multisession disk with the first session containing the audio tracks and the following session containing the data track.

Many operating systems are not able to read more than a single data track or need special software to do so.

More information on the SCSI command set of a HP CD-Recorder can be found at:

`http://www.hp.com/isgsupport/cdr/index.html`

If you have more information or SCSI command manuals for currently unsupported CD-Recorders please contact the author.

The Philips CDD 521 CD-Recorder (even in the upgraded version) has several firmware bugs. Some of them force you to power cycle the device or to reboot the machine.

When using `cdrecord` with the broken Linux SCSI generic (sg) driver. You should note that `cdrecord` uses a hack that tries to emulate the functionality of the sg driver. Unfortunately, the sg driver on Linux has several severe bugs:

- It cannot see if a SCSI command could not be sent at all.
- It cannot get the SCSI status byte. The `cdrecord` command for that reason cannot report failing SCSI commands in some situations.

- It cannot get a real DMA count of transfer. The `cdrecord` command cannot tell you if there is an DMA residual count.
- It cannot get the number of bytes valid in auto sense data. The `cdrecord` command cannot tell you if device transfers no sense data at all.
- It fetches too little data in auto request sense (CCS/SCSI-2/SCSI-3 needs >= 18).

The fifo percent output is computed just after a block of data has been written to the CD-Recorder. For this reason, there is never 100% fifo fill while the fifo is in streaming mode.

DIAGNOSTICS

You have 9 seconds to type `^C` to abort `cdrecord` after you see the message:

```
Starting to write CD at speed %d in %s mode for %s session.
```

A typical error message for a SCSI command looks like the following:

```
cdrecord: I/O error. test unit ready: scsi sendcmd: no error
CDB:  00 20 00 00 00 00
status: 0x2 (CHECK CONDITION)
Sense Bytes: 70 00 05 00 00 00 00 0A 00 00 00 00 25 00 00 00 00 00
Sense Key: 0x5 Illegal Request, Segment 0
Sense Code: 0x25 Qual 0x00 (logical unit not supported) Fru 0x0
Sense flags: Blk 0 (not valid)
cmd finished after 0.002s timeout 40s
```

The first line gives information about the transport of the command. The text after the first colon gives the error text for the system call from the view of the kernel. It usually is `I/O error` unless other problems happen. The next words contain a short description for the SCSI command that fails. The rest of the line tells you if there were any problems for the transport of the command over the SCSI bus. The output `fatal error` means that it was not possible to transport the command (that is, no device is present at the requested SCSI address).

The second line prints the SCSI command descriptor block for the failed command.

The third line gives information on the SCSI status code returned by the command if the transport of the command succeeds. This is error information from the SCSI device.

The fourth line is a hexadecimal dump of the auto-request sense information for the command.

The fifth line is the error text for the sense key if available, followed by the segment number that is only valid if the command was a *`copy`* command. If the error message is not directly related to the current command, the text `deferred error` is appended.

The sixth line is the error text for the sense code and the sense qualifier, if available. If the type of the device is known, the sense data is decoded from tables in *`scsierrs.c`*. The text is followed by the error value for a field-replaceable unit.

The seventh line prints the block number that is related to the failed command and text for several error flags. The block number may not be valid.

The eighth line reports the timeout set up for this command and the time that the command really needed to be finished.

The following message is not an error:

```
Track 01: Total bytes read/written: 2048/2048 (1 sectors).
cdrecord: I/O error. flush cache: scsi sendcmd: no error
CDB:  35 00 00 00 00 00 00 00 00 00
status: 0x2 (CHECK CONDITION)
Sense Bytes: F0 00 05 80 00 00 27 0A 00 00 00 00 B5 00 00 00 00 00
Sense Key: 0x5 Illegal Request, Segment 0
Sense Code: 0xB5 Qual 0x00 (dummy data blocks added) Fru 0x0
Sense flags: Blk -2147483609 (valid)
cmd finished after 0.002s timeout 40s
```

It simply notifies that a track that is smaller than the minimum size has been expanded to 300 sectors.

RESTRICTIONS

The `cdrecord` currently only warns if the input data does not fit on the disk. If you do not abort the command you get unpredictable results.

AUTHOR

This page has been adapted from information provided by:

Joerg Schilling

Seestr. 110 D-13353

Berlin Germany

Additional information can be found at **`http://www.fokus.gmd.de/research/cc/glone/employees/jeorg.schilling/private/cdrecord.html`**

SEE ALSO

Commands: `mkisofs`(8)