Name

mformat - add an MSDOS filesystem to a low-level formatted floppy disk

Note of warning

This manpage has been automatically generated from mtools's texinfo documentation, and may not be entirely accurate or complete. See the end of this man page for details.

Description

The `mformat` command is used to add an MS-DOS file system to a low-level formatted diskette. Its syntax is:

`mformat` [`-t` *cylinders*|`-T` *tot_sectors*] [`-h` *heads*] [`-s` *sectors*]
[`-f` *size*] [`-1`] [`-4`] [`-8`]
[`-v` *volume_label*]
[`-F`] [`-S` *sizecode*]
[`-M` *software_sector_size*]
[`-N` *serial_number*] [`-a`]
[`-C`] [`-H` *hidden_sectors*] [`-I` *fsVersion*]
[`-r` *root_sectors*] [`-L` *fat_len*]
[`-B` *boot_sector*] [`-k`]
[`-m` *media_descriptor*]
[`-K` *backup_boot*]
[`-R` *nb_reserved_sectors*]
[`-c` *clusters_per_sector*]
[`-d` *fat_copies*]
[`-X`] [`-2` *sectors_on_track_0*] [`-3`]
[`-0` *rate_on_track_0*] [`-A` *rate_on_other_tracks*]
drive:

`Mformat` adds a minimal MS-DOS file system (boot sector, FAT, and root directory) to a diskette that has already been formatted by a Unix low-level format.

The following options are supported: (The S, 2, 1 and M options may not exist if this copy of mtools has been compiled without the USE_2M option)

The following options are the same as for MS-DOS's format command:

Options

`v` Specifies the volume label. A volume label identifies the disk and can be a maximum of 11 characters. If you omit the -v switch, mformat will assign no label to the disk.

`f` Specifies the size of the DOS file system to format. Only a certain number of predefined sizes are supported by this flag; for others use the -h/-t/-s flags. The following sizes are supported:

- 160 160K, single-sided, 8 sectors per track, 40 cylinders (for 5 1/4 DD)
- 180 160K, single-sided, 9 sectors per track, 40 cylinders (for 5 1/4 DD)
- 320 320K, double-sided, 8 sectors per track, 40 cylinders (for 5 1/4 DD)
- 360 360K, double-sided, 9 sectors per track, 40 cylinders (for 5 1/4 DD)
- 720 720K, double-sided, 9 sectors per track, 80 cylinders (for 3 1/2 DD)
- 1200 1200K, double-sided, 15 sectors per track, 80 cylinders (for 5 1/4 HD)
- 1440 1440K, double-sided, 18 sectors per track, 80 cylinders (for 3 1/2 HD)
- 2880 2880K, double-sided, 36 sectors per track, 80 cylinders (for 3 1/2 ED)

`t` Specifies the number of tracks on the disk.

`T` Specifies the number of total sectors on the disk. Only one of these 2 options may be specified (tracks or total sectors)

`h` The number of heads (sides).

`s` Specifies the number of sectors per track. If the 2m option is given, number of 512-byte sector equivalents on generic tracks (i.e. not head 0 track 0). If the 2m option is not given, number of physical sectors per track (which may be bigger than 512 bytes).

`1` Formats a single side (equivalent to -h 1)

`4` Formats a 360K double-sided disk (equivalent to -f 360). When used together with -the 1 switch, this switch formats a 180K disk

`8` Formats a disk with 8 sectors per track.

MS-DOS format's `q`, `u` and `b` options are not supported, and `s` has a different meaning.

The following options are specific to mtools:

`F` Format the partition as FAT32.

`S` The size code. The size of the sector is 2 ^ (sizecode + 7).

`X` formats the disk as an XDF disk. See section XDF, for more details. The disk has first to be low-level formatted using the xdfcopy utility included in the fdutils package. XDF disks are used for instance for OS/2 install disks.

`2` 2m format. The parameter to this option describes the number of sectors on track 0, head 0. This option is recommended for sectors bigger than normal.

`3` don't use a 2m format, even if the current geometry of the disk is a 2m geometry.

`0` Data transfer rate on track 0

`A` Data transfer rate on tracks other than 0

`M` software sector size. This parameter describes the sector size in bytes used by the MS-DOS file system. By default it is the physical sector size.

`N` Uses the requested serial number, instead of generating one automatically

`a` If this option is given, an Atari style serial number is generated. Ataris store their serial number in the OEM label.

`C` creates the disk image file to install the MS-DOS file system on it. Obviously, this is useless on physical devices such as floppies and hard disk partitions, but is interesting for image files.

`H` number of hidden sectors. This parameter is useful for formatting hard disk partition, which are not aligned on track boundaries (i.e. first head of first track doesn't belong to the partition, but contains a partition table). In that case the number of hidden sectors is in general the number of sectors per cylinder. This is untested.

`I` Sets the fsVersion id when formatting a FAT32 drive. In order to find this out, run minfo on an existing FAT32 drive, and mail me about it, so I can include the correct value in future versions of mtools.

`c` Sets the size of a cluster (in sectors). If this cluster size would generate a FAT that too big for its number of bits, mtools automatically increases the cluster size, until the FAT is small enough. If no cluster size is specified explicitly, mtools uses a default value as described in section "Number of sectors per cluster" below.

`d` Sets the number of FAT copies. Default is 2. This setting can also be specified using the `MTOOLS_NFATS` environment variable.

`r` Sets the size of the root directory (in sectors). Only applicable to 12 and 16 bit FATs. This setting can also be specified using the `MTOOLS_DIR_LEN` environment variable.

`L` Sets the length of the FAT.

`B` Use the boot sector stored in the given file or device, instead of using its own. Only the geometry fields are updated to match the target disks parameters.

`k` Keep the existing boot sector as much as possible. Only the geometry fields and other similar file system data are updated to match the target disks parameters.

`K` Sets the sector number where the backup of the boot sector should be stored (only relevant on FAT32).

`R` Sets the number of reserved sectors for this filesystem. This must be at least 1 for non-FAT32 disks, and at least 3 for FAT disks (in order to accommodate the boot sector, the info sector and the backup boot sector).

`m` Use a non-standard media descriptor byte for this disk. The media descriptor is stored at position 21 of the boot sector, and as first byte in each FAT copy. Using this option may confuse DOS or older mtools version, and may make the disk unreadable. Only use if you know what you are doing.

To format a diskette at a density other than the default, you must supply (at least) those command line parameters that are different from the default.

`Mformat` returns 0 on success or 1 on failure.

It doesn't record bad block information to the Fat, use `mbadblocks` for that.

Number of sectors per cluster

If the user indicates no cluster size, mformat figures out a default value for it.

For FAT32 it uses the following table to determine the number of sectors per cluster, depending on the total number of sectors on the filesystem.

more than 32*1024*1024*2: 64 sectors
between 16*1024*1024*2 and 32*1024*1024*2: 32 sectors
between 8*1024*1024*2 and 16*1024*1024*2: 16 sectors
between 260*1024*2 and 81024*1024*2: 1 sectors

This is derived from information on page 20 of Microsoft's `fatgen103` document, which currently can be found at the following address:

`https://staff.washington.edu/dittrich/misc/fatgen103.pdf`

For FAT12 and FAT16, mformat uses an iterative approach, where it starts with a set value, which it doubles until it is able to fill up the disk using that cluster size and a number of cluster less than the maximum allowed.

The starting value is 1 for disks with one head or less than 2000 sectors, and 2 for disks with more than one head, and more than 2000 sectors.

The number of sectors per cluster cannot go beyond 128.

See Also

Mtools' texinfo doc

Viewing the texi doc

This manpage has been automatically generated from mtools's texinfo documentation. However, this process is only approximative, and some items, such as crossreferences, footnotes and indices are lost in this translation process. Indeed, these items have no appropriate representation in the manpage format. Moreover, not all information has been translated into the manpage version. Thus I strongly advise you to use the original texinfo doc. See the end of this manpage for instructions how to view the texinfo doc.

* To generate a printable copy from the texinfo doc, run the following commands:

./configure; make dvi; dvips mtools.dvi

* To generate a html copy, run:

./configure; make html

A premade html can be found at `'http://www.gnu.org/software/mtools/manual/mtools.html'`

* To generate an info copy (browsable using emacs' info mode), run:

./configure; make info

The texinfo doc looks most pretty when printed or as html. Indeed, in the info version certain examples are difficult to read due to the quoting conventions used in info.