

**Name**

mpartition - partition an MSDOS hard 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 `mpartition` command is used to create MS-DOS file systems as partitions. This is intended to be used on non-Linux systems, i.e. systems where `fdisk` and easy access to SCSI devices are not available. This command only works on drives whose partition variable is set.

```
mpartition -p drive
mpartition -r drive
mpartition -I [-B bootSector] drive
mpartition -a drive
mpartition -d drive
mpartition -c [-s sectors] [-h heads]
[-t cylinders] [-v [-T type] [-b
begin] [-l length] [-f]
```

Mpartition supports the following operations:

- p Prints a command line to recreate the partition for the drive. Nothing is printed if the partition for the drive is not defined, or an inconsistency has been detected. If verbose (`-v`) is also set, prints the current partition table.
- r Removes the partition described by *drive*.
- I Initializes the partition table, and removes all partitions.
- c Creates the partition described by *drive*.
- a "Activates" the partition, i.e. makes it bootable. Only one partition can be bootable at a time.
- d "Deactivates" the partition, i.e. makes it unbootable.

If no operation is given, the current settings are printed.

For partition creations, the following options are available:

- s *sectors*  
The number of sectors per track of the partition (which is also the number of sectors per track for the whole drive).
- h *heads*  
The number of heads of the partition (which is also the number of heads for the whole drive). By default, the geometry information (number of sectors and heads) is figured out from neighboring partition table entries, or guessed from the size.
- t *cylinders*  
The number of cylinders of the partition (not the number of cylinders of the whole drive).
- b *begin*  
The starting offset of the partition, expressed in sectors. If *begin* is not given, `mpartition` lets the partition begin at the start of the disk (partition number 1), or immediately after the end of the previous partition.
- l *length*  
The size (length) of the partition, expressed in sectors. If *end* is not given, `mpartition` figures out the size from the number of sectors, heads and cylinders. If these are not given either, it gives the

partition the biggest possible size, considering disk size and start of the next partition.

The following option is available for all operation which modify the partition table:

- f** Usually, before writing back any changes to the partition, mpartition performs certain consistency checks, such as checking for overlaps and proper alignment of the partitions. If any of these checks fails, the partition table is not changes. The **-f** allows you to override these safeguards.

The following options are available for all operations:

- v** Together with **-p** prints the partition table as it is now (no change operation), or as it is after it is modified.
- vv** If the verbosity flag is given twice, mpartition will print out a hexdump of the partition table when reading it from and writing it to the device.

The following option is available for partition table initialization:

**B** *bootSector*

Reads the template master boot record from file *bootSector*.

### Choice of partition type

Mpartition proceeds as follows to pick a type for the partition:

- FAT32 partitions are assigned type 0x0C (“Win95 FAT32, LBA”)
- For all others, if the partition fits entirely within the 65536 sector of the disk, assign 0x01 (“DOS FAT12, CHS”) for FAT12 partition and 0x04 (“DOS FAT16, CHS”) for FAT16 partitions
- If not covered by the above, assign 0x06 (“DOS BIG FAT16 CHS”) if partition fits entirely within the first 1024 cylinders (CHS mode)
- All remaining cases get 0x0E (“Win95 BIG FAT16, LBA”)

If number of fat bits is not known (not specified in drive’s definition), then FAT12 is assumed for all drives with less than 4096 sectors, and FAT16 for those with more than 4096 sectors.

This corresponds more or less to the definitions outlined at [https://en.wikipedia.org/wiki/Partition\\_type#List\\_of\\_partition\\_IDS](https://en.wikipedia.org/wiki/Partition_type#List_of_partition_IDS) and [https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-2000-server/cc977219\(v=technet.10\)](https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-2000-server/cc977219(v=technet.10)), with two notable differences:

- If fat bits are unknown, the reference documents consider drives with less than 32680 sectors to be FAT12. Mtools uses 4096 sectors as the cutoff point, as older versions of DOS only support FAT12 on disks with less than 4096 sectors (and these older versions are the ones which would be most likely to use FAT12 in the first place).
- The reference documents use a 8GB (wikipedia) or a 4GB (Microsoft) cutoff between 0x06 (DOS BIG FAT16 CHS) and 0x0E. Mtools uses 1024 cylinders. This is because any partition beyond 1024 cylinders must be LBA and cannot be CHS. 8GB works out to be the biggest capacity which can be represented as CHS (63 sectors, 255 heads and 1024 cylinders). 4GB is the capacity limit for windows 2000, so it makes sense that a documentation for windows 2000 would specify this as the upper limit for any partition type.

### 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.