

Welcome to FIPS
The First nondestructive Interactive Partition Splitting program

Version 2.0
May 11, 1998

Copyright 1993-98 Arno Schaefer
Parts of the code Copyright 1997/98 Gordon Chaffee

0. What you need to use FIPS
1. Introduction
2. What FIPS does
3. Safety
4. Restrictions
5. Before you start
6. Use with a multitasking OS
7. Using FIPS
8. After splitting the partition
9. Commandline Switches
10. Troubleshooting
11. Credits

In file SPECIAL.DOC:

- S1. Use with Stacker/SuperStor/Doublespace etc.
- S2. Use with OS/2
- S3. Use with OnTrack Disk Manager and similar drivers

FIPS is a program designed to split an existing DOS partition without deleting the data on it.

FIPS is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2, or (at your option) any later version.

FIPS is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with FIPS; see the file COPYING. If not, write to the Free Software Foundation, 675 Mass Ave, Cambridge, MA 02139, USA.

Report problems and direct all questions to:

schaefer@rbg.informatik.th-darmstadt.de

0. What you need to use FIPS

You need a defragmentation program in order to move all data to the beginning of the hard disk. FIPS will only split your partition if you have enough free space at the end. Windows 95 and later MS-DOS versions contain DEFRAG, which

is suitable for this purpose. Other suitable programs are Norton Speedisk, PCTools' Compress, DISKOPT in Novell DOS 7, or the shareware programs ORG, DOG or SAFPAK (available by anonymous FTP from any SimTel mirror in the diskutil directory). I did not test these however, so don't blame me if they don't work for you.

You may also want to use a program like Norton Disk Doctor (or 'scandisk' in DOS 6.*/Windows 95) to check your harddisk before and after using FIPS.

FIPS was developed under DOS 5.0. It should work fine with anything above 3.0, perhaps even with 2.0. However it will not be of much use with older DOS versions, since the large partition sizes are only available since DOS 4. It has been reported to work with DOS 6.0 and 6.2 and Novell DOS 7. I have also received reports about successful use with Windows 95 partitions. Note that the long filenames are no problem for FIPS, because it works on a lower level where directory structure is of no concern. It is probably sensible to boot from a DOS disk before using FIPS in such a setup (see 6.)

1. Introduction

The program was inspired by the Linux Project. When installing Linux on a PC that was used for DOS / Windows, many people want to retain a smaller partition for their DOS software. However, since most Harddisks contain only one large partition, you would normally be required to do a complete backup, erase the partition and build two (or more) new partitions. Then you would restore the backup to one of the new partitions. On today's systems with hard disk capacities of usually 500MB or more, a complete backup becomes practically infeasible without large media like streamer tapes once the drive has filled up to some considerable fraction. Even though many people backup their most valuable data (a practice which I would highly recommend in any case), the process of reinstalling all the software packages takes many hours or even days.

FIPS was written to remedy this problem. You can now split a partition without losing any data, provided there is enough free space for the new partition at the end of the old one.

2. What FIPS does

FIPS reduces the size of a partition by changing some values in the partition table and boot sector. It does not change the formatting of the partition, especially not the cluster size and the size of the file allocation table (FAT). Therefore the reduced partition will have a FAT that is in part unused, but this is not a problem for DOS.

From the free space that is won by this, FIPS creates a new `_primary_` DOS partition.

If you want to use the new partition under a different OS (e.g. Linux), use its supplied fdisk program to make any necessary changes (refer to the OS manuals).

If you want to use the new partition under DOS/Windows, you can use it as it is (after formatting!), but be aware of the following:

According to the official references, DOS can only have one primary partition. All DOS versions (at least from v5.0 on) will happily work with multiple primary partitions, but this is an 'undocumented feature'. If you want to follow the official rules, you can delete the new partition with fdisk and create an extended partition in its place.

There are reasons for having multiple primary partitions, among them the possibility to boot from different partitions by changing the active partition with fdisk.

Known problems with multiple primary partitions are:

- If you accidentally delete one primary partition with fdisk, you can not easily recreate it, since fdisk will refuse to. There may be other fdisk programs around that work, and if all else fails you can boot from a Linux boot disk to run Linux' fdisk, but it is always a hassle.
- There exist some software packages that work with the partition table and which may be confused by multiple primary partitions. Among them was SFS, the 'Secure file system' by Peter Gutmann. I think Peter has made a change to his program to accept some unusual configuration, but there may exist other software packages that will have problems.

3. Safety

FIPS was specifically designed to provide a maximum of safety. On startup it checks the Partition Table, Boot Sector and FAT for any inconsistencies. If it finds anything suspicious, it will tell you so. If there are errors, FIPS will not proceed.

You have the possibility to write backup copies of your root and boot sector to a floppy disk before proceeding. If something goes wrong, you may restore these with the program 'restorb.exe' (see section 5). I strongly recommend making use of this feature. It will also enable you to reverse the partition split afterwards, so it might be a good idea to save the root/boot sector image in a safe place.

FIPS checks for free space on the partition and will let you choose the new start cylinder accordingly.

After having calculated the new partition table and boot sector, FIPS will check everything again, so that possible bugs in the calculation may be detected. Only if everything is ok, FIPS will ask for permission to write the new root and boot sector.

4. Restrictions

FIPS will only work with Hard Disk BIOSes that use interrupt 13h for low level harddisk access. This is true for almost all PCs. I have received a report that an older Adaptec SCSI controller had a software driver that worked on a higher level. In this case you can only hope to get a new driver from Adaptec. This is no problem any more for all newer Adaptec's (I personally use one).

FIPS will only work on disks with a sector size of 512 bytes. It seems that DOS is prepared to deal with different sector sizes, but so far I have never seen this.

FIPS will not split partitions with 12 bit FATs (you would not want to split

partitions with less than 10 MB, would you?).

FIPS will only split DOS partitions. Partition table and boot sector must conform to the MSDOS 3.0+ conventions. This is marked by the system indicator byte in the partition table, it must have the value 4 (16 bit sector number) or 6 (32 bit sector number).

It will especially **not** split Linux partitions.

NEW: Windows 95 FAT32 partitions are now also supported.

FIPS does not work on extended DOS partitions.

FIPS will not work if you already have four partitions, since it needs one free partition entry.

FIPS will not reduce the original partition to a size with less than 4085 clusters, because this would imply rewriting the 16 bit FAT to a 12 bit FAT.

5. Before you start

Run CHKDSK or SCANDISK on the partition you want to split.

If you have Norton Disk Doctor or something similar, you may use it alternatively. Make sure there remain no 'dead' clusters on the disk.

Prepare a bootable floppy disk in drive A:. Under DOS this is usually done by giving the command 'sys a:' or 'format a:/s'. Under Windows NT or OS/2 this may be different, if in doubt check your manual or boot from a boot disk from a DOS PC.

Copy the FIPS files RESTORRB.EXE, FIPS.EXE and ERRORS.TXT to this disk.

Test booting from the prepared floppy disk. Read your manual or ask a local guru if you can't boot from floppy disk or if you can not access your hard disk after booting (test this by giving the command 'dir c:', you should see your hard disk's root directory). If all else fails, try using FIPS after booting normally from the hard disk (a bit more risky, but sometimes the last resort).

When you start FIPS (later!), you will be given the opportunity to write backup copies of your root and boot sector to a file on drive A: called ROOTBOOT.00x (where x stands for a digit from 0 to 9). If anything goes wrong while using FIPS, you can restore the original configuration by booting from the floppy and running RESTORRB. Please note: if you use FIPS more than once (this is normally not necessary, but it may happen), more than one ROOTBOOT file is written to the floppy disk. RESTORRB lets you choose which configuration file to restore. The file RESTORRB.000 contains your original configuration. Try not to confuse the versions.

You will need this backup file (ROOTBOOT.00x) if you want to undo the partition split later.

But before starting FIPS you must now defragment your Harddisk. All of the space that will be used for the new partition must be free. Be aware that the Windows Swapfile will not be moved by most defragmentation programs. You must uninstall it (in the 386enhanced part of the Windows Control Panel) and reinstall it after using FIPS.

If you use IMAGE or MIRROR, the last sector of the hard disk contains a hidden system file with a pointer to your mirror files. You must delete this

file before using FIPS (it will be recreated the next time you run mirror).
Do 'attrib -r -s -h image.idx' or 'attrib -r -s -h mirrorsav.fil' in the root directory, then delete the file.

If FIPS does not offer as much disk space for creation of the new partition as you would expect it to have, this may mean that

- a. You still have too much data in the remaining partition. Consider making the new partition smaller or deleting some of the data.
- b. There are hidden files in the space of the new partition that have not been moved by the defragmentation program. You can find the hidden files on the disk by typing the command 'dir /a:h /s' (and 'dir /a:s /s' for the system files). Make sure to which program they belong. If a file is a swap file of some program (e.g. NDOS) it is possible that it can be safely deleted (and will be recreated automatically later when the need arises). See your manual for details.

If the file belongs to some sort of copy protection, you must uninstall the program to which it belongs and reinstall it after repartitioning.

I can't give you more aid in this - if you really can't figure out what to do, contact me directly.

Attention! Do not try to move DOS' hidden system files (ibmbio.com & ibmdos.com or something similar). You may end up with a hard disk that will not boot any more. Since these files are already in the first sectors of the partition, it is not necessary to move them. In DOS 5.0 and later, at least one of the system files may be moved, but it is nevertheless a good idea to leave it alone.

When you have run FIPS, you must reboot, so make sure to disable any programs in the config.sys and autoexec.bat that write to the disk, in particular mirror or image. Consider temporarily renaming these files.

Be aware that the location of your DOS partitions in the partition table may change after using FIPS. If you use the new partition under DOS and you have an extended partition and/or two drives, this means that the names of the partitions may change (e.g. D: may become E:). I have taken care that C: always remains C:, so that you will still be able to boot.

Also if you have a CD-ROM, it will in most cases change its drive letter. If you use the /L switch to explicitly specify the drive letter in the call to MSCDEX in the autoexec.bat file, you might have to change it accordingly. If you don't use the /L switch, MSCDEX will automatically choose the first free letter, so that you do not need to change anything.

Apropos drive letters: DOS uses the following order to assign them:

1. The first primary partition on each drive
2. The volumes inside the extended partitions on all drives
3. The remaining primary partitions on all drives

Example:

C: first primary partition on first HD
D: first primary partition on second HD
E: first volume in extended partition on first HD
F: second volume in extended partition on first HD
G: first volume in extended partition on second HD

H: second primary partition on second HD
I: CD ROM

I am not aware of an easy way to change this scheme. In particular you can not assign an arbitrary drive letter to a partition. You can however influence the ordering by changing a primary partition into an extended one.

For Linux users:

The possible change of the location of the DOS partition in the partition table also means that the device number of the DOS partition under Linux may change (e.g. /dev/hda3 may become /dev/hda1). Any existing Linux partitions will not change, so that you will have no trouble booting. You just need to edit your /etc/fstab file if you mount your DOS partition on bootup.

6. Use with a multitasking OS

You should not use FIPS in multitasking environments like OS/2, Desqview, Windows, Novell Task Manager or the Linux DOS Emulator. These systems might still write to the disk after FIPS has changed the hard disk structure, which may result in corrupting the disk. This is not necessarily so, I'd suppose that in most cases it would work nevertheless. But since safety is my first concern with FIPS, I would recommend booting from a DOS boot disk and then running FIPS, that should be safe.

In version 1.0 I added some code by Dave McCaldon to detect Windows and Desqview (thanks, Dave!). OS/2 and Novell Task Manager are not yet detected. I had to remove the code for detecting the Linux DOS emulator because it caused a hangup on many machines.

7. Using FIPS

If you have prepared a bootable floppy disk as described in section 5, boot from it now.

Important! Make sure not to have a disk cache program like Smartdrive running. It has been reported that in some cases the changes FIPS made were only written to the disk in part, which resulted in hard disk corruption later. I think this may be caused by the use of Smartdrive, which in the default configuration delays the disk writes for some seconds. If you reboot too fast, some of the changes may be lost.

You start FIPS by typing FIPS at the DOS prompt, followed by <ENTER>. You may exit from the program at any time by pressing <CTRL-C>.

FIPS will first try to detect under which OS it is running. If it is Windows or Desqview, it will complain and tell you to boot from a floppy disk. You can proceed nevertheless, but this is at your own risk (see section 8).

Then FIPS will detect your hard disks, if you have more than one, it will ask you which one you want to work on.

In previous releases, FIPS failed to detect the correct number of hard disks with some BIOSes (esp. in Gateway Pentium machines). I hope to have corrected this. If FIPS fails to detect the correct number of disks, please let me know. In the meantime you may use the '-n' switch to select the drive by hand (see below).

FIPS then reads the root sector of the hard disk and displays the partition table.

Example:

Part.	bootable	Start Head Cyl. Sector	End System Head Cyl. Sector	Start Sector	Number of Sectors	MB
1	yes	0 148 1	83h 15 295 63	149184	149184	72
2	no	1 0 1	06h 15 139 63	63	141057	68
3	no	0 140 1	06h 15 147 63	141120	8064	3
4	no	0 0 0	00h 0 0 0	0	0	0

If you don't know what to make of this, don't worry too much. You may just use the number of Megabytes to identify your partitions.

The root sector is then checked for errors.

If you have more than one partition on the disk, you will be asked which one you want to split.

The boot sector of the chosen partition is read and some information is displayed.

Example:

Bytes per sector: 512
Sectors per cluster: 8
Reserved sectors: 1
Number of FATs: 2
Number of rootdirectory entries: 512
Number of sectors (short): 0
Media descriptor byte: f8h
Sectors per FAT: 145
Sectors per track: 63
Drive heads: 16
Hidden sectors: 63
Number of sectors (long): 141057
Physical drive number: 80h
Signature: 29h

FIPS checks if this information is consistent with the partition table and tries to detect other errors.

It then verifies if the two copies of the FAT are identical, if they are not, FIPS will exit with an error message.

If everything checks out ok, FIPS now looks for free space at the end of the partition. The new partition must have at least one cylinder, so if the last cylinder is not free, you have no chance of splitting the partition: FIPS will exit with an error message. Probably you forgot to remove a mirror or image file (see above).

You must now enter on which cylinder the new partition should start. Use the cursor keys right/left to increase or decrease the cylinder count. The size of the remaining partition and the new partition are

displayed in the process, so you will have no trouble choosing the right cylinder. With cursor up/down you can change the count in steps of 10. When ready, press enter to continue.

FIPS will check again if the space for the new partition is empty - this is an additional security measure and should never show an error, since the free space was already determined before.

After this, FIPS will calculate the changes to the root sector, check the changes and display the new partition table. You may now choose to reedit the partition table (this will return you to the point where you select the partition) or to continue. If you type 'c', FIPS will calculate the changed boot sector, check it again and prompt you if you want to proceed. If you type 'y' then, FIPS will write the changes to the disk and exit.

8. After splitting the partition

Make sure that the changes have been written to the disk. If you used a disk cache like Smartdrive (despite my explicit recommendation not to use it :-), wait for some seconds so that the cache is flushed to disk.

Your new partition will be recognized by DOS after you rebooted. **DO NOT WRITE ANYTHING TO THE DISK BEFORE REBOOTING.** After rebooting, use CHKDSK or Norton Disk Doctor to make sure your old (now smaller) partition is ok.

There have been cases where the changes made by FIPS were not correctly written. To make sure that this is not the case, run FIPS again with the -t (test mode) switch after rebooting. Select the drive and partition that you were splitting. If FIPS displays no error until the cylinder selection dialogue, there is no problem and you can stop the program with CTRL-C. Otherwise you should undo the changes with 'restorb' and contact me by email.

If you don't find any errors, you may now reboot with your normal config.sys and autoexec.bat. Start some programs and make sure you can still read your data.

If you want to use your new partition under DOS, you must format it. If you have multiple partitions, make sure to format the right one, the drive names may have changed!

If you want to use the partition under Linux, you may now change the system indicator byte with Linux' fdisk, then use MKFS.

If you want to split the new partition again in two smaller ones, you must first format it under DOS, otherwise FIPS will complain.

9. Commandline Switches

Here is the explanation of FIPS' commandline switches. If you prefer the DOS style, you may use '/' instead of '-' as the switch character. The switches may be arbitrarily combined. Type 'FIPS -help' to get a list of the switches. Here is a more detailed explanation:

-t or -test : test mode (no writes to disk)

This doesn't need much explanation.

-d or -debug : debug mode

In this mode, a complete transcript of your session along with some additional information is written to the file FIPSINFO.DBG in the current directory. You can send this file to me in case of trouble (see below).

-h or -help or -? : help page

A short summary of the switches

-n<num> : select drive <num>

Preselect the drive number with this switch. Valid numbers are 128 to 255. This may also be used to override the automatic drive detection - if for any reason the drive is not found by FIPS, you may try this switch.

10. Troubleshooting

FIPS is still somewhat experimental, although it has been used by many people successfully and without serious problems. When in doubt I usually decided to stay safe and display error messages when encountering suspicious configurations. Some of the minor errors may be overridden.

Please make sure you have read this doc carefully and also look in the file FIPS.FAQ that covers some frequently asked questions.

If you can't resolve a problem yourself, or have a configuration not supported by FIPS, or if you suspect a bug in FIPS, make a transcript of your session using the -d switch and send the full FIPSINFO.DBG file along with a short comment to schaefer@rbg.informatik.th-darmstadt.de. Possibly your problem has already been solved.

11. Credits

FIPS is based on the procedure described by Drew Eckhardt in Linux digest132. Most of what I know about Harddisk structures comes from the excellent german book 'Scheibenkleister II' by Claus Brod and Anton Stepper. It is for the Atari ST, but much of it applies to PCs also.

Information on the Harddisk Interrupts was drawn from Ralf Brown's Interrupt List. Thanks to Hamish Coleman for some useful info and to Paul Smith for his good suggestions. Gunnar Hilmarsson suggested the procedure for stacked drives, and Miguel Alvarez helped me improve the partition ordering. Chetan Patil, Rand Phares and Eric Jung pointed me at bugs in the program and documentation. Stefan Andreasen provided important informations about OS/2. Keith Crews suggested some additions to the documentation concerning the preparation of the boot disk and new features of DOS 6.x. Dave McCaldon wrote the code for detecting the OS FIPS is running under. Scott Ellentuch and Billy Patton provided info about OnTrack Disk Manager.

A very valuable piece of info on EIDE drives and address translation is the Linux EIDE-Mini-HOWTO by Patrick LoPresti.

Many thanks to Gordon Chaffee for the FAT32 support.

Thanks to all others who sent me feedback and suggestions. FIPS would not be the same program without the innumerable emails I received.

Arno Schaefer
schaefer@nacamar.de