

DiskMon.DOC

COLLABORATORS

	<i>TITLE :</i> DiskMon.DOC		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		September 19, 2022	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	DiskMon.DOC	1
1.1	DiskMon V2.6t Manual	1
1.2	what is diskmon ?	2
1.3	copyright	3
1.4	warranty	3
1.5	requirements	3
1.6	read	4
1.7	checksum	4
1.8	write	4
1.9	Drive	4
1.10	edit	4
1.11	Search	5
1.12	readme	5
1.13	Repair (ONLY Floppy Disks)	5
1.14	quit	5
1.15	display	5
1.16	check	6
1.17	Gadget Block	6
1.18	Cyl. - Cylinder	6
1.19	R - RootBlock gadget	6
1.20	head	6
1.21	auto	6
1.22	Sec. - Sector	7
1.23	ndos	7
1.24	'Hidden' gadgets 'Header/Parent,Next/1.Data/DirCache'	7
1.25	error	7
1.26	checkok	7
1.27	type	7
1.28	name	7
1.29	driveinfo	8

1.30	The MFM-Editor (ONLY Floppy Disks)	8
1.31	mfm read	9
1.32	mfm write	9
1.33	mfm indexsync	9
1.34	mfm writelen	9
1.35	mfm prewrite	9
1.36	mfm cyl.	10
1.37	mfm head	10
1.38	mfm arrange sn	10
1.39	mfm arrange so	10
1.40	mfm sync	10
1.41	mfm find	10
1.42	mfm sync,find	10
1.43	mfm scroll data	10
1.44	mfm quit	11
1.45	mfm error	11
1.46	BAM-Editor (ONLY 1760 block drives)	11
1.47	File-Editor	11
1.48	filename	12
1.49	fileread	12
1.50	filewrite	12
1.51	fileblocknr	12
1.52	fileauto	12
1.53	fileedit	12
1.54	filedisplay	12
1.55	filesearch	13
1.56	filequit	13
1.57	fileerror	13
1.58	How to get a registered version	13
1.59	My Address	13

Chapter 1

DiskMon.DOC

1.1 DiskMon V2.6t Manual

```

000:0000000 0000000 0000000 0000000 .....
  Read
Checksum
  Write
010:0000000 0000000 0000000 0000000 .....
Drive ...
Edit <mode>
020:0000000 0000000 0000000 0000000 .....
Search
  ReadMe
  Repair
030:0000000 .....
Ed Mfm
  FileEd
  Ed BAM
040:0000000 This manual was designed .....
Quit
  Display
  Check
050:0000000 for using with MultiView, ..... Block

$

060:0000000 window size 640x245 (PAL). ..... Cyl.

+
-
R
070:0000000 It looks differnt using ..... Head

+
-
Auto
080:0000000 the original AmigaGuide. ..... Sec.

+
-
NDOS

```

```

090:00000000 .....
Checksum:Right? - Used?
0A0:00000000 00000000 00000000 00000000 .....
Type      :Type of Block
0B0:00000000 00000000 00000000 00000000 .....
Parent    :<Block number>
0C0:00000000 00000000 00000000 00000000 .....
Next      :<Block number>
0D0:00000000 00000000 00000000 00000000 .....
Name      :<Name of file>
0E0:00000000 00000000 00000000 00000000 ..... or ←
      <Directory>
0F0:00000000 00000000 00000000 00000000 .....
      DRV: Unit # of ...
      .
      About 'DiskMon' Version 2.6t
      ShareWare
      .
:
      © Copyright 1990-1994 by
      Jörg Strohmayer
      :
1F0:00000000 00000000 00000000 00000000 .....
      Error:pay Shareware fee

```

1.2 what is diskmon ?

DiskMon is a DiskMonitor for the Amiga including

- BlockEditor

: search/view/edit data on block-orientated devices such as FloppyDisk (Amiga 880KB, Amiga 1.71MB, CrossDos 720KB, CrossDos 1.44MB, ...), HardDisk (including Rigid-Disk-Blocks), RamDisk (RAD:, ...) and many others.

- FileEditor

: search/view/edit data of any file.

- MfmEditor

: search/view/edit data at low level on FloppyDisks.

- BamEditor

: view/edit BAM (Block-Availible-Map) on FloppyDisks.

- RepairTrack

: recover lost data (read/write error) on FloppyDisks.

DiskMon does not need any libraries or other files to run, simply copy it to any directory you want.

DiskMon should work on any Amiga.

DO NOT USE THIS PROGRAM IF YOU DON'T KNOW WHAT YOU ARE DOING ←

!!!!!!!!!!!!!!

1.3 copyright

DiskMon is NOT Public Domain, it is © by
Jörg Strohmayer

.
ONLY this version 2.6t (with the 'Pay Shareware'-Requesters)
is freely distributable. If you find this program usefull and want to
use it, you'll have to pay the shareware fee to get a registered version
of the program, see
Shareware

.
See also

Warranty
.

1.4 warranty

This program is provided "as is" without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The entire risk as to the results and performance of this program is assumed by you. Should the program prove defective, you alone assume the entire cost of all necessary servicing, repair, or correction. Further, the author of this program neither warrants, guarantees, or makes any representations regarding the use of, or the results of the use of, the program in terms of corrections, accuracy, reliability, currentness, or otherwise; and you rely on the program and results solely at your own risk. The author of this program can in no event be held responsible for any data or information which may be lost or rendered inaccurate by this program, even if the author of this program has been advised of the possibility of such damages.

1.5 requirements

DiskMon should work on any Amiga with KickStart/OS 1.2 or greater.

Successfull tests were made using

- Amiga 1000 256 KB Kickstart 1.2 (V33.180)
- Amiga 1000 512 KB Kickstart 1.3 (V34.5)
- Amiga 1000 2.0 MB OS 2.00 (V36.207)
- Amiga 500+ 1.5 MB OS 2.04 (V37.175)
- Amiga 4000/EC030 5.5 MB OS 3.0 (V39.106)

DiskMon uses a 640x256 non-interlaced screen on PAL Amigas and on NTSC Amigas it has to open an interlaced screen. IF you have a ECS- or AA-Amiga and DiskMon opens an interlaced NTSC-Screen, select PAL in the Boot-Menu and you will get a PAL screen.

Using OS 2.0 or greater DiskMon uses the ASL-Requesters to select the FileName for the File-Editor, otherwise you'll get a simple string gadget to enter the FileName.

1.6 read

DiskMon reads and displays the selected block of the selected Drive .

1.7 checksum

DiskMon calculates the checksum of the displayed block if the block type is

- Root Block
 - Directory
 - File Header
 - File List
 - Boot Block (only if 'Check' is 'on')
 - old filesystem Data Block
 - Directory Cache Block
- DOES NOT (yet) WORK WITH
- Rigid-Disk-Blocks (RDSK, PART, FSHD, LSEG, ...)

Only use it if the block type is not '?? Unknown ??'.

1.8 write

DiskMon writes the block to the selected block number and drive .

Remember to correct the checksum if required.

1.9 Drive ...

If the gadget 'Drive ...' is clicked, you get a list of all available drives/partitions. Select the drive you wish to edit by clicking the correct line.

1.10 edit

Selects the mode of editing. HEX means you have to enter the data as hexadecimal (half-)bytes, ASCII for entering characters. On pre-2.0 systems you can't move the cursor with the cursor keys if editmode 'ASCII' is selected, use mouse instead.

1.11 Search ...

A requester is displayed where you can select the startblock, endblock and the text you wish to search. 'Stop' aborts while 'Search' starts searching. If the text is found you can 'Stop' searching or continue searching by clicking 'Search' again.

Search is case sensitive and no patterns are used.

If you want to search hexadecimal data you can do this by entering '\$' as the first character (for example '\$AB cd 12' which is equal to '\$ abc d12' but ' \$AbcD12' is wrong because of the space before the '\$').

1.12 readme

The Documentation of DiskMon as ASCII-Text.

You can scroll inside the text with the cursor keys and the mouse.

```
cursor up      : 1 line up
cursor down    : 1 line down
shift cursor up : 1 page (32 lines) up
shift cursor down: 1 page (32 lines) down
alt cursor up   : top if text
alt cursor down : botton of text
<ESC>          : quit
<S>           : save/print
```

1.13 Repair (ONLY Floppy Disks)

If there is a read/write error on the disk you can't read the whole track, but in most cases there is only one of the 11/22 blocks destroyed. Diskmon reads the selected track and tries to recover as much data as possible. It will be displayed which block is ok ('repaired') and which block could not be corrected ('not repaired'). Now you are asked if you want to write the data back to disk or not. You may change the disk to write the results to another disk. If you select 'Yes' the recovered data is written to the disk, 'No' aborts.

ATTENTION !!! if you write the data back to the same disk and there are blocks which are not repaired there is no chance of getting lost data back.

1.14 quit

The self-destruction-button of DiskMon.

1.15 display

This gadget selects which characters are displayed.

```
Default: ASCII characters 32-127
1.Click: Visible characters 32-127 and 160-255
2.Click: All characters 0-255
3.Click: ASCII ...
```

1.16 check

Toogles the range checking of the block number.

on : Default

off: The block number given is send directly to the device.

On Harddisks for example, you get the rigid-disk-block if 'check' is off and you enter block number 0.

ATTENTION !!!: You cant calculate the checksum of RDSK,PART,FSHD,LSEG... blocks with the gadget 'Checksum'. (If you want to calculate it manually, it is the 3rd longword)

ATTENTION !!!: By changing the rigid-disk-blocks you may loose all the data on all your partion of the Harddisk.

ATTENTION !!!: If you get outside the range of a RAM-Disk like RAD: you view/edit any part of the memory which can cause a system crash.

Use 'Check off' with extreme caution and at your own risk.

1.17 Gadget Block

You can enter the block number decimal (1.Gadget) or hexadecimal (2.Gadget).

Use the '+' and '-' gadgets of
Sec.
to increase/decrease
the block number.

1.18 Cyl. - Cylinder

You can enter the cylinder number decimal, increase '+' or decrease '-' it.

1.19 R - RootBlock gadget

The Gadget 'R' gives you the Root-Block of the disk/partion.
If 'Check' is off you get block number 0.

1.20 head

You can enter the head number decimal, increase '+' or decrease '-' it.

1.21 auto

If 'on', any change to the block number will read and display the new block imediately. If 'off', you have to click 'Read' to get the new block.

1.22 Sec. - Sector

You can enter the sector number decimal, increase '+' or decrease '-' it.
Use '+' and '-' of Sec. to increase/decrease the block number.

1.23 ndos

If 'on', no other task can use the drive you are viewing/editing. ↔
If 'off',
other tasks have access to the drive too (dangerous if you change data, if
you only want to view/search data you may set NDOS to 'off').
You have to switch it 'off' if you want to use the
FileEditor
on the
selected drive.

1.24 'Hidden' gadgets 'Header/Parent,Next/1.Data/DirCache'

These gadgets are placed below 'Type' and have no borders but if the
texts 'Header', 'Parent', 'Next', '1.Data' or 'DirCache' are displayed you
can click on it to get the block-number displayed after the ':'.

1.25 error

Errors are displayed on the last line at the right side.

1.26 checkok

DiskMon displays if the CheckSum is right and used in the displayed block.
Checksum: 'Right' or 'Wrong'
'Used !!!' or 'Not used'

1.27 type

Type : Type of the block or '?? unknown ??' if not a valid type. On
Fast-File-System disks '?? unknown ??' blocks may be data blocks.

1.28 name

Name : name of file/dir/disk or nothing

```
00260:0000 0000 0000 0000 0000 0000 0000 0000 0000
    <
        1 Bit
    >
00280:0000 0000 0000 0000 0000 0000 0000 0000 0000
    <
        2 Bits
    >
00300:0000 0000 0000 0000 0000 0000 0000 0000 0000
    <
        4 Bits
    >
00320:0000 0000 0000 0000 0000 0000 0000 0000 0000
    <

Bytes
    >
00340:0000 0000 0000 0000 0000 0000 0000 0000 0000
    Quit MFM-Editor
00360:0000 0000 0000 0000 0000 0000 0000 0000 0000
Error: pay fee !!!
```

1.31 mfm read

Reads a track from disk.

1.32 mfm write

Writes the track back to disk.

1.33 mfm indexsync

Toogles the indexsync mode. AmigaDOS does not use indexsync but PC-DOS does.

1.34 mfm writelen

Number of bytes (decimal) to write to the disk. On 880KB disks, AmigaDOS uses 11968 bytes of data (11*1088) and the rest of about 700 bytes are the gap.

1.35 mfm prewrite

Number of bytes (decimal) to write before the real data. These bytes are 0xAA which is decoded 0x00. PreWrite is used to delete possible syncs in the gap because the real data is less than would fit on the disk. PreWrite is not used if IndexSync is on.

1.36 mfm cyl.

Number of the Cylinder to read/write.

1.37 mfm head

Select head 0 or 1.

1.38 mfm arrange sn

The sectors are arranged by the sector number. This means at offset 0 will be sector 0, offset 1088 sector 1, ..., offset 10880 sector 10.

1.39 mfm arrange so

The sectors are arranged by the sector offset like they are written to the disk by AmigaDOS. In the error-line (right last line below the gadgets) it is displayed in which order the sectors are arranged, '-' means sector not found. Additionally the gap is arranged after the last sector.

1.40 mfm sync

DiskMon searches the given SYNC-word and if found the data will be arranged, that the SYNC-word is at offset 0.

1.41 mfm find

Searches the given word at 4-Bit boundary. If found, a requester appears where you can continue searching or stop searching.

1.42 mfm sync,find

Enter the word to find/sync.

1.43 mfm scroll data

Cycles the data 1,2,4 bit or a given number of bytes left or right.

1.44 mfm quit

Return to main (block) editor.

1.45 mfm error

Errors are displayed on the last line at the right side.

1.46 BAM-Editor (ONLY 1760 block drives)

In this editor you can mark blocks as used or free on a disk.
 The BAM (Block-Availible-Map, Sectormap) is loaded and displayed.
 '+' means the block is used and '.' is a free block. Change the status
 by clicking on the cursor or pressing return.
 Gadgets: 'Write BAM' writes the changes back to the disk and
 'Quit BAM' returns to the main editor without writing the changes.

1.47 File-Editor

```

          000: 00000000 00000000 00000000 00000000 .....
          File:<name>
          010: 00000000 00000000 00000000 00000000 .....
020: 00000000 00000000 00000000 00000000 .....
          Read Block
          030: 00000000 00000000 00000000 00000000 .....
          Write Block
          040: 00000000 00000000 00000000 00000000 .....
050: 00000000 00000000 00000000 00000000 ..... Filesize:<#> Blocks
060: 00000000 00000000 00000000 00000000 .....
          +
          -
          070: 00000000 00000000 00000000 00000000 .....
080: 00000000 00000000 00000000 00000000 .....
          Auto Read on/off
          090: 00000000 00000000 00000000 00000000 .....
0A0: 00000000 00000000 00000000 00000000 .....
          EditMode= <mode>
          0B0: 00000000 00000000 00000000 00000000 .....
0C0: 00000000 00000000 00000000 00000000 .....
          Display = <mode>
          0D0: 00000000 00000000 00000000 00000000 .....
0E0: 00000000 00000000 00000000 00000000 .....
          Search
          !!! The file you want to edit must not be read/write protected ←
          !!!
1F0: 00000000 00000000 00000000 00000000 .....
          Error: pay fee !!!

```

1.48 filename

Select another file to edit.

1.49 fileread

Reads and displays the selected block.

1.50 filewrite

Writes the displayed block back to the file.

1.51 fileblocknr

Increase, enter, decrease the block number to view/edit.

1.52 fileauto

If 'on', any change to the block number will read and display the new block immediately. If 'off', you have to click 'Read' to get the new block.

1.53 fileedit

Selects the mode of editing. HEX means you have to enter the data as hexadecimal (half-)bytes, ASCII for entering characters. On pre-2.0 systems you can't move the cursor with the cursor keys if editmode 'ASCII' is selected, use mouse instead.

1.54 filedisplay

This gadget selects which characters are displayed.

Default: ASCII characters 32-127
1.Click: Visible characters 32-127 and 160-255
2.Click: All characters 0-255
3.Click: ASCII ...

1.55 filesearch

A requester is displayed where you can select the startblock, endblock and the text you wish to search. 'Stop' aborts while 'Search' starts searching. If the text is found you can 'Stop' searching or continue searching by clicking 'Search' again.

Search is case sensitive and no patterns are used.

If you want to search hexadecimal data you can do this by entering '\$' as the first character (for example '\$AB cd 12' which is equal to '\$ abc d12' but ' \$AbcD12' is wrong because of the space before the '\$').

1.56 filequit

Return to main editor.

1.57 fileerror

Errors are displayed on the last line at the right side.

1.58 How to get a registered version

DiskMon is Shareware, if you want to use it you have to pay the shareware fee. You will receive a disk with the latest version of DiskMon, and a message whenever a major update of DiskMon is available. For further updates, registered users only will have to pay a small amount of money to cover the cost for disks and postage. If you want to get automatic updates whenever they are available, please send more money and you'll get updates as long as the money you send covers the costs.

Shareware fee:	Cash	Check
Germany	DM 20	DM 30
Europe	DM 25 US\$ 17	DM 35 US\$ 24
Other Countries	DM 30 US\$ 20	DM 45 US\$ 30

1.59 My Address

Send comments, suggestions, bug reports, etc. and the shareware fee to:

Jörg Strohmayer
Im Bachacker 10

D-35252 Dautphetal
Germany