

mfm_read can analyze and/or read a MFM disk and write raw MFM transition data, decoded data, or emulation data to files.

mfm_util can read transition and convert to emulation or decoded data. It can also read emulation file data and convert to decoded data.

mfm_read and mfm_util both use similar command options.

--sectors -s #[,#]

The number of sectors per track, lowest sector number.

--heads -h #

The number of heads.

--cylinders -c #

The number of cylinders.

--header_crc -g #h,#h,#h[,#h]

The CRC/ECC parameters for the sector header. Initial value, polynomial, polynomial length, maximum ECC span.

--data_crc -d #h,#h,#h[,#h]

The CRC/ECC parameters for the sector data area. Initial value, polynomial, polynomial length, maximum ECC span.

--format -f WD_1006 | OMTI_5510 | DEC_RQDX3 | Xebec_104786

The track format.

--sector_length -l #

The sector data area length in bytes. Default is 512.

--unbuffered_seek -u

Use unbuffered/ST506 seeks. Default is buffered/ST412.

--interleave -i # | #,#,#,#,...

The logical sector numbers from header in physical sector order or the interleave value

--head_3bit -3

Selects header 3 bit head encoding used by WD 1003 controller. Default is 4 bit. This will not be detected by analyze. The wrong number of heads may be selected.

--retries -r #

Select number of retries on read errors. Only valid for read command. Default 50.

--analyze -a

Analyze disk format.

--quiet -q #h

Bit mask to select which messages don't print. 0 is print all messages. Default is 1 (no debug messages). Higher bits are more important messages in general.

--transitions_file -t filename

File name to write raw MFM transitions to. No file created if not specified. Only valid for read command.

--extracted_data_file -e filename

File name to write decoded data to. No file created if not specified.

--emulation_file -m filename

File name to write emulation bit data to. No file created if not specified

--drive -j #

Drive number to select for reading. Only valid for read command.

--version -v

Print program version number.

Long options can be abbreviated to the shortest unique name. Long option values can't have spaces.

is a number. #h is a number which may be decimal, octal if starts with a 0, or hex starting with 0x. The Cyclic Redundancy Check (CRC) / Error Correcting Code (ECC) parameters consists of a initial value which the CRC register is set to before starting the CRC, a CRC polynomial, and a maximum ECC span. The ECC span should be zero if ECC correction should not be used.

32 bit or longer polynomials may be usable for ECC even if the original controller only used them for CRC. The quality of the polynomial chosen determines the miss-correction probability. Most 32 bit polynomials specify 5 bit correction though some say they can be used for up to 11 bit correction.

Emulation file is for use by the mfm_emu program to emulate a disk drive. For mfm_read it is an output. For mfm_util it is an output if a transitions file is specified otherwise it is an input.

If extract file is specified mfm_read will attempt to decode the track data. If the disk format is known an extract file should be specified even if one is not needed since mfm_read will then reread tracks that have errors until it exceeds the error count or gets a successful read of the track.

Examples:

To read an unknown format drive

```
mfm_read --analyze --transitions_file raw_data --extracted_data_file extracted_data
```

Analyze will not enable ECC. If the polynomial supports ECC rerun with the command line printed and change the CRC parameter ,0 to the desired ECC span.

To store raw transitions without decoding

```
mfm_read --transitions_file raw_data --drive 1 --heads 4 --cylinders 5
```

To process previously read raw transitions data

```
mfm_util --sectors 17,0 --heads 6 --cylinders 640 --header_crc 0x2605fb9c,0x104c981,32,0  
--data_crc 0xd4d7ca20,0x104c981,32,0 --format OMTI_5510 --sector_length 512 --interleave  
0,3,6,9,12,15,1,4,7,10,13,16,2,5,8,11,14 --transitions_file raw_out2 --extracted_data_file  
/tmp/decoded_out
```

Analysis typical messages. Explanation in italic:

AM33XX

File prucode0.bin open passed

Informational messages from PRU access routines.

Found drive at select 2

Informational. Select line drive responded to.

Returning to track 0

Informational. This operation may take a while.

Drive RPM 3594.4

Informational. Drive should be close to 3600 RPM

Found multiple matching header parameters. Will use largest matches:

Matches 1 controller DEC_RQDX3 Polynomial 104c981 length 32 initial value 2605fb9c

Matches 17 controller OMTI_5510 Polynomial 104c981 length 32 initial value 2605fb9c

Possible problem. We found multiple possible methods for decoding the track. The program will

choose the largest match. In this example since one matches all the sectors it is the correct format. The code doesn't know how many sectors/track at this point. Some drives (DEC RQDX3) actually do use multiple formats. This code does not deal well with that. You can manually read with the different formats and put the data back together.

Header CRC Information:

Polynomial 104c981 length 32 initial value 2605fb9c

Controller type OMTI_5510

Sector length 512, Data CRC Information:

Polynomial 104c981 length 32 initial value d4d7ca20

What we picked so far for the format

Retrying on a different cylinder and head

Since we found multiple possible matches we will try a different cylinder and head.

Header CRC Information:

Polynomial 104c981 length 32 initial value 2605fb9c

Controller type OMTI_5510

Sector length 512, Data CRC Information:

Polynomial 104c981 length 32 initial value d4d7ca20

Informational. What we found on the second try. Only one match this time.

Number of heads 6 number of sectors 17 first sector 0

Informational. What more we have determined about the format.

Interleave: 0 3 6 9 12 15 1 4 7 10 13 16 2 5 8 11 14

Informational. If it can't determine interleave the disk can still be read and decoded.

Drive supports buffered seeks (ST412)

Informational.

No sectors readable from cylinder 640

Stopping end of disk search due to two unreadable tracks in a row

Number of cylinders 640, 33.4 MB

Informational. The method we used to determine the number of cylinders and size determined.

Command line to read disk:

```
--sectors 17,0 --heads 6 --cylinders 640 --header_crc 0x2605fb9c,0x104c981,32,0 --data_crc  
0xd4d7ca20,0x104c981,32,0 --format OMTI_5510 --sector_length 512 --retries 50 --drive 2  
--interleave 0,3,6,9,12,15,1,4,7,10,13,16,2,5,8,11,14
```

This is the options needed to decode the disk with mfm_read. For mfm_util remove --retries and --drive from options. Change header_crc and/or data_crc last parameter if you wish to use ECC corrections.

It is recommended to verify results of analysis since it can make mistakes.

Decoding messages:

AM33XX

File prucode0.bin open passed

Informational messages from PRU access routines in mfm_read.

Returning to track 0

Informational. This operation may take a while. Only mfm_read.

Retries failed cyl 22 head 0

Unable to recover all data from the specified track. Only mfm_read.

Bad sectors on cylinder 22 head 0: 3 15H

Indicates that for the specified track that data of sector 3 has uncorrected errors and the header for sector 15 has errors. If the header has an error the data will be all zero.

ECC Corrections on cylinder 56 head 5: 1(2) 13(4) 15(1H)

Informational. Indicates that for the specified track that data of sectors 1 and 13 were corrected and the header of sector 15 corrected. The number in the parenthesis is the length of the bit pattern corrected. Unless the ECC correction had a false correction the data is good.

Cyl 59 head 2 Missed sector between 12(4) and 1(6)

Informational. Indicates that the sectors found didn't match the interleave specified. The first number is the sector header number and the number in parenthesis the physical sector starting with 0. A bad sector message will be printed if the data is bad. Some disks vary the interleave. Try again without the interleave option.

Good data after 20 Retries cyl 219 head 0

Informational. Indicates we recovered good data by retrying the read.

Found cyl 0 to 639, head 0 to 5, sector 0 to 16

Informational. Should match what was specified

Expected 65280 sectors got 65276 good sectors, 0 bad header, 4 bad data

11 sectors corrected with ECC. Max bits in burst corrected 5

Summary of errors during the read. We have 4 sectors with bad data.

Errors:

Command 4 fault 300 status 1002c

Not Write fault

Not Seek complete

Not Index

Ready

Drive selected

Not Track 0

This indicates a command failed. See cmd.h for definitions. Status is the drive status bits which are decoded in text below. This error was the drive did not give seek complete in the expected time. These are normally fatal and the program will exit. Some are recovered from during analysis and are informational.