mfm_emu emulates a MFM disk drive. It can use a data file created from a real drive by mfm_read or can create a empty emulation file.

--begin_time -b #
      The number of nanoseconds to delay from index to start reading track Only needed if initialize specified. Default is zero if not specified.
--cylinders -c #
      The number of cylinders. Only needed if initialize specified.
--drive -d #[,#]
      Drive number to emulate. For revision B or C boards specify 0 or 1 or 1,2. Use 0 for drive to always be selected (radial select). 1 and 2 select the first and second jumper block to use for drive select. One or two drive numbers may be specified to emulate one or two drives. Only one may be specified if drive number is 0. For revision A boards specify the drive number to emulate 1-4.
      First drive specified uses J1. The second uses J6.
--file -f filename[,filename]
      Emulation filename. First filename corresponds to first drive number specified.
--heads -h #
      The number of heads. Only needed if initialize specified.
--initialize -i
      If given create/overwrite specified file with empty data. Heads and cylinders must be specified.
--note -n "string"
      Description to store in the emulation file. Only used if initialize specified.
--options -o "string"
      Options for mfm_util in decoding such as –data_crc. This saves having to type each time you wish to decode file. Only used if initialize specified and not required. No validation is performed so use mfm_util to verify file is valid after creating.
--pool -p #,#.#
      The first parameter is the size of the of track buffer pool to use. and the second is the maximum delay. The delay will increase linearly as the buffers fill to the maximum delay. Default is 75,0.6
--quiet -q #h
      Bit mask to select which messages don't print. 0 prints all messages. Default is 1 (no debug messages). Higher bits are more important messages in general.
--rate -r #
      Bit rate in Hz for the MFM clock and data bits. Only needed if initialize specified. Default is 10,000,000. if not specified. For most SA1000 controllers specify 8680000. Only needed when –initialize specified.
--rpm -R #
      Drive RPM. Default is 3600 unless rate is close to 8680000 where the default is 3125 for SA1000 drives. For Quantum Q2000 drives specify 3000 if you wish to emulate the real drive RPM. Only needed when –initialize specified.
--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 buffer pool option controls buffering used to prevent controller timeouts when writing data to the

file. Writing to flash has large delays at times which the buffers hide. The maximum delay should be set shorter than the controller timeout for a seek. The maximum number of buffers needs to be low enough that they can be written out before the holdup capacitors are drained.

Begin_time is for drives that have a sector straddle the start of the index pulse. For emulation to work properly all the data must be read consecutively. Set this parameter to the time in nanoseconds the first physical sector is delayed from the index pulse. It is needed for Corvus model H and NorthStar Advantage drives.

When this program is run it appends to logfile.txt in the current directory. It logs when it started, stopped, how long it was executing, maximum seek time, minimum free buffers, and how many seeks and writes were done. The shutdown time is from when the program was told to shut down to the emulation file closed and written to storage. The operating system will take about 5 more seconds to shut down.

If the log file minimum free buffers is zero you may wish to either increase the number of buffers or maximum delay. If the log file shutdown time is close to capacitor holdup time or not all the message were written to the log file the number of buffers should be decreased or wait 45 seconds after significant write activity before powering off the MFM emulator.
Example:
    mfm_emu –drive 1 –file disk_file
This used file disk_file to emulate a drive on select line 1.

    mfm_emu –drive 1 –file disk_file –initialize –cylinders 306 -heads 4 –note "Games disk" --options "--sectors 17,0 --heads 6 --cylinders 640 --header_crc 0x2605fb9c,0x104c981,32,0 --data_crc 0xd4d7ca20,0x104c981,32,0 --format OMTI_5510"

This creates/overwrites file disk_file with empty data for a disk with 306 cylinders and 4 heads. It then emulates a drive on select line 1. You will need to use the host computer low level format program to write the sector headers before the emulated drive can be read or written normally.

Currently the program prints various stuff to see what it's doing. This will change as testing goes on.
bad pattern count 0
Read queue underrun 0
Write queue overrun 0
Ecapture overrun 0
glitch count 0
glitch value 0
0:test 0 0
0:test 1 0
0:test 2 0
0:test 3 0
0:test 4 0
1:test 0 0
1:test 1 0
1:test 2 0
1:test 3 0
1:test 4 0
    The named values are various errors/unexpected conditions. The test values are used to show

internal variables from the PRU's. See the source for current purpose.

select 0 head 0
  Current select and head line state

Waiting, seek time 3.9 ms max 3.9
  IARM is waiting for PRU. The values are the last and maximum time from PRU requesting the next cylinder to the data being returned. **If other than first seek time printed is zero you are using a buggy version of am335x_pru_package and data is likely to be corrupted.**

Cyl 0,400 select 1, head 4 dirty 0
  Last and next requested cylinder, select and head lines. Dirty 1 indicates the sector data was written to.

I have found that if unrecoverable read errors occur operating the drive in a different orientation may allow the data to be read. I have also had some luck especially on Seagate drives with pushing on the shaft of the head stepper motor while it is retrying. This seems to get the head at a slightly different position where it can read the data. This has a risk of drive damage so make sure you have read as much as you can before trying this.