

# Converting .TD0 files to .img or .imd format

---

These notes cover the tool `td02imx` which is designed to convert Teledisk .TD0 files to either .img or .imd formats. They also provide some background on the implementation decisions, which differ from other similar tools. These may be of wider interest as they address some of the anomalies seen in .TD0 files.

A separate document `teledisk.md` provides information on the internal .TD0 file format.

## td02imx Usage

---

The basic usage of the tool is as shown below.

```
Usage: td02imx -v | -V | [options] td0file [outfile | .ext]
```

where `-v` shows version info and `-V` shows extended version info

If present `outfile` or `.ext` are used to specify the name of the converted file

The `.ext` option is a short hand for using the input file with its extent replaced by `.ext`

If the `outfile` ends in `.imd`, the IMD file format is used instead of a binary image

Options are

- `-c dd` Only cylinders before the number `dd` are written
- `-k[aAcddgt]*` Keep spurious information
  - `a` Attempt to use auto generated sectors - use with caution
  - `A` As 'a' but also tag with CRC error in IMD images
  - `c` Use data from a bad CRC sectors - implied for IMD
  - `d` Use data from deleted data sectors - implied for IMD
  - `g` Keep sector gaps - implied for IMD
  - `t` Use spurious tracks
- `-oo` Process all cylinders from side 1 then side 2 both in ascending order (Out-Out)
- `-ob` As `-to` but side 2 cylinders are processed in descending order (Out-Back)

Track sorting defaults to cylinders in ascending order, alternating head

For image file skipped and missing data sectors are always replaced with dummy sectors

Duplicates of valid sectors are ignored, any with differing content are noted

Unlike Teledisk the `td0file` does not need to end in `.td0` however if multi volume disks are used the file name should end in a digit and the follow on file must have the same name but with the digit incremented by one. In normal Teledisk usage the files would be `file.td0` and `file.td1` ...

Note the code implements the multi volume disk sets feature of Teledisk but due to lack of example disks it hasn't been tested.

## Displayed information

---

In addition to any generated file, the tool displays the following information

1. **General information:** The name of the file being processed and information similar to td0check, with comment text trimmed of trailing newlines and spaces. The original td0check information differs slightly between versions, so there is an additional debug line showing drive type, stepping and density values in hex from the td0 file header.
2. **Track information:** For each tracked processed the following information is shown: cylinder/head sector\_count [sector\_size] sector\_ids\_in\_order e.g.

```
0/0 MFM 10 [512] 1 2 3 4 5 6 7 8 9 10
```

To handle unusual disk formats and td0 anomalies the following variations to the above may also be shown.

If not all sectors are used then sector\_count is shown as usable\_count/sector\_count and excluded sectors are prefixed by a '#' e.g.

```
14/0 FM 16/17 [128] 1 #1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
```

A change of sector size causes [size] to be inserted e.g.

```
0/0 MFM 10 [128] 1 [512] 2 3 4 5 6 7 8 9 10
    here the first sector is 128 bytes, the rest are 512 bytes
```

A cylinder or head mismatch or change causes {cyl/new} to be inserted e.g.

```
0/1 MFM 10 [512] {0/0} 10 18 13 16 11 19 14 17 12 15
78/0 MFM 16 [512] {77/0} 1 5 9 13 2 6 10 14 3 7 11 15 4 8 12 16
```

Errors relating to a specific sector are tagged by one or more letters namely:

```
P - indicates problem: duplicate sector id but different content
D - duplicate sector flag was set - doesn't appear to be used anymore
c - crc sector flag was set
d - deleted data flag was set
u - dos unused flag was set
n - no data flag was set
? - sector has generated IDAM - only ever occurs on first sector
```

Examples

```
48/0 MFM 15 [512] 1 2 3 4 5 6 7 8 9 10 11 12 13c 14 15
0/0 FM 26 [128] 1 2 3 4 5 6 7 8 9r 10r 11r 12r 13r 14r 15r
...
```

If there are no sectors then "No usable sectors" is show e.g.

```
40/0 No usable sectors
```

Due to what appears to be a bug in Teledisk, if there are no IDAMs, it will occasionally generate a single spurious sector with an auto generated IDAM with sector id 100, usually with cylinder/head of the previous real track read. This is reported as spurious e.g.

```
40/0 MFM 1 [256] {39/1} 100 - spurious no real Address Markers
```

Track specific errors are flagged as the end of the sector list and would normally cause the track to be skipped as spurious, the -kt flag changes this to a warning. The cases currently processed are

```
< 3 valid Address Markers (and valid sectors are < 8192 bytes)
duplicate track - there is a cyl/head mismatch and the content for valid
sectors matches the track indicated by the cyl/head
no good sectors on track - all the sectors on a track have a CRC error
Examples
  40/0 FM    1  [128]  1c - Spurious < 3 Address Markers
  79/0 MFM   9  [1024] {76/0} 1 2 3 4 5 6 7 8 9 - spurious
duplicate track
  24/0 MFM 10/11 [512] #110? 1c 3c 5c 7c 9c 11c 13c 15c 17c 19c -
Spurious no good sectors on track
```

### 3. Disk sizing

Following the track information a guess is done as to the disk format.

Note: The code to do this attempts to account for missing sectors using the same information that is used insert missing sectors in image files; see [Identifying Missing Sectors](#) . There are scenarios where this may be wrong, thankfully for most cases it makes a sensible guess.

Examples:

```
[40x8 512 MFM sector 160K] 160K
[39x17 256 MFM sector 165K] [1x18 128 FM sector 2K] 168K
[40x2x9 512 MFM sector 360K] 360K
[(0:38x5 1:40x5) 1024 MFM sector 390K] [0:2x10 mixed size MFM sector 18K]
408K
The last example reflects head 0-38 tracks & head 1-40 tracks with 5 1024
byte sectors and head 0-2 tracks with mixed size sectors.
```

### 4. Warnings during file creation

Dependent on which file format is created additional warnings may be issued. The various warnings are noted under File Generation below.

## Handling Anomalies

It is clear from examining many .TD0 files that occasionally there were problems reading the source disk, which unfortunately results in anomalies in the .TD0 file. These basically break into 3 common groups of problem and the sections below describe how they are handled in td02imx before any output file is generated.

- When Teledisk reads the IDAMs on a track it captures more than one revolution of the disk and in theory could read up to 100 IDAMs. In processing these it looks for the the repeat of the first IDAM to determine the correct number of sectors. However if the remaining IDAMs don't confirm that they are repeats, possibly due to read errors, it will include all the IDAMs read. In most cases this will result in duplicate sectors.
- Related to the above, when attempting to synchronise the IDAMs to the start of a track, if Teledisk fails to match the first sector read by the floppy disk "Read Track" command, it will add an auto-generated sector.

Note: If the sector read with the "read track" command has a data error or a sector length different from the first IDAM read, Teledisk is unlikely to find a match.

- Missing sectors - where the original could not be read or its address mark could not be read. This is mainly an issue for creating image files, as the IMD format contains information about used sectors so missing sectors can be detected if an attempt is made to read it.
- Unusual IDAM reads - The most common cause of this is when a drive has been stepped beyond the normal end of the drive and physically is reading the last track more than once. The use of a protection mechanism is another possible reason.
- Apparent bug in Teledisk. Occasionally the auto-generated sector is not tagged as such. Two scenarios are
  - The only sector has id 100 and is untagged.
  - The first sector is  $100 + \text{number of other sectors} + 1$ .

These two scenarios are fixed.

## Identifying Usable Sectors

The algorithm to identify usable sectors is relatively simple and processes each td0 sector in order as follows:

- If the sector is auto generated ignore it.
  - If the sector id of the sector hasn't occurred before add it to the list of usable sectors
  - Otherwise both the sector and the previous usable sector with the same sector id are given a score (see below) and if the sector has a higher score it replaces the previous usable one.
- If both sectors have valid data but different cylinder or head or size or content, the unused one is marked as a problem duplicate.

Note the list of usable sectors contains indexes into the list of td0 input sectors, so does not reflect the sector id directly.

The sector score is currently determined as shown in the table below. The only potentially contentious decision is to give Normal sectors a higher score than Deleted data sectors even if there is a CRC error. In practice a Normal and Deleted data sector on the same track with the same sector id is unlikely to occur.

normal sector	16
duplicate	15
with CRC	14
duplicate with CRC	13
deleted data sector	12
duplicate	11
with CRC	10
duplicate with CRC	09
skipped sector	08
duplicate	07
No data	04
duplicate	03
No address mark	00

## Identifying Missing Sectors

The key assumption made in identifying missing sectors is that all tracks with a given encoding (FM/MFM), sector size and head are formatted the same and that sorted sectors ids are spaced evenly. There is also an assumption that at least one track with each used combination that has a complete set of good sectors. How exceptions to this are handled when writing image files are documented in the Image File section below.

The separate tracking by head caters for sector ids that continue in sequence from head 0 to head 1.

Using this assumption, for each track processed a rounded average of first, last and sector spacing is calculated for each encoding, sector size and head combination. Tracks with bad sectors, intra track cylinder, head or sector size changes are excluded, as are tracks where there appear to be more than two sectors missing.

After the calculation above, these averages are used to modify the sector range of each track unless it would remove sectors before the first or after the last, in which case a warning is given and the original range of sector ids is unchanged.

Tracks with mixed sectors sizes, more than two missing sector ids are tagged as non-standard and no missing sectors are added in a .IMG file.

If the -ka or -kA option is specified and there is an auto generated sector and only one missing sector Id, the auto generated sector is used in its place.

The information above is also used to help guess the disk structure.

Note: Where there are many warnings generated it is recommended to use IMD format, since apart from -ka or -kA processing, for IMD files no missing sectors are inserted.

## Dealing With Unusual Address Marker

The following processing is currently done to help with Unusual Address Markers

- If there are less than three usable address markers on a track, a warning is issued and the track is skipped. The -kt option force includes the track.
- If all of the usable sectors in a track record the same cylinder but it is different from the physical cylinder, the physical track is compared to the track corresponding to the sector's recorded cylinder. For both tracks if all the sectors with valid data match and there is at least one match, then a duplicate track warning is issued and the track is skipped. As above the -kt option will force include the track.

No other automated handling of unusual address markers is done, however

- The displayed track information shows cylinder and head differences and changes
- Image file production ignores cylinder and head differences and changes

## File Generation

---

If no output file is specified or there is no data to save then the tool exits displaying "No file produced".

# IMD File

The .imd file is created as follows from the .TD0 file

1. Header time is copied from the td0 comment field if present else the current time is used.
2. The comment includes a note of the source file, minus any path component, followed by the td0 comment trimmed of trailing spaces and newlines. If this removes all the comment text it is replaced by "Comment text is blank". If there is no td0 comment record the text "No comment information" is used.
3. Missing tracks or tracks with mixed size sectors are skipped as are spurious tracks unless -kt is specified.
4. Mode is derived from the td0 file header density and td0 track header encoding marker (top bit of td0 head value).  
Note as IMD does not currently support 1000kps recording, so this is mapped to 500kps.
5. Sector, cylinder, head maps are generated from the list of usable sectors, by indexing into the td0 list of sectors to get the relevant information. These are written as necessary.  
Sectors are not generated for missing sectors except when the -ka or -kA option is used, see Identifying Missing Sectors above. If -kA is specified then the sector is marked as having a CRC error.  
Warning: Treat autogenerated sectors with caution as their content isn't CRC checked and may have the wrong size.
6. The sector data is derived from the td0 sector flags as follows

```
No data          -> type 0
DOS allocated    -> type 2 + repeated 0xe5 byte
Deleted data     -> type 3 + sector data or type 4 + repeated byte
                  + CRC error -> type 7 + sector data or type 8 + repeated byte
Normal          -> type 1 + sector data or type 2 + repeated byte
                  + CRC error -> type 5 + sector data or type 6 + repeated byte
```

## Image File

Unlike the IMD format, image files do not contain information on cylinder, head or sector. As they are linear files any gaps in the image need to be filled so that an application can easily seek to the correct sector. It also cannot support non-standard cylinder and head maps.

## Track Ordering

The image file is created by writing each track in order. The td02imx application supports three common track orderings as follows:

1. For each cylinder from zero upwards, write the track with head 0 followed by the track with head 1, if it exists. This is the default order (ALTERNATE)
2. If option -soo is specified, the order is, for each cylinder from zero upwards, write the track with head 0. Once all of the cylinders are written, repeat for each cylinder from zero upwards, but write the track with head 1. (OUT-OUT)
3. If option -sob is specified, the order is, for each cylinder from zero upwards, write the track with head 0. Once all of the cylinders are written, repeat for each cylinder but from the last cylinder downwards, and write the track with head 1. (OUT-BACK)

## Writing Sectors

For each track to be written, if there is no sector data then a warning is issued and no data is written.

Otherwise:

1. The sector range to be written is calculated from the table created during the missing sectors analysis based on encoding, head and sector size. Sector spacing is also calculated.
2. If the -ka or -KA option is specified and there is both an autogenerated sector and one missing sector, the autogenerated sector is assumed to have the sector id of the missing sector.  
Warning: Treat autogenerated sectors with caution as their content isn't CRC checked and may have the wrong size.
3. A table is constructed in sector id order from the list of usable sectors.
4. The constructed table is then processed for each sector id in the ordered range and spacing calculated above:
  - if no usable sector exists
    - If the track is tagged as non-standard no warning is issued. Warning previously given.
    - else a dummy sector containing repeated text "\*\*\* missing \*\* " is written. Warning issued
  - else if CRC flag is set, a warning is issued
    - if the -kc option is specified the sector is written
    - else a dummy sector containing repeated text "\*\*\* bad crc \*\* " is written
  - else if the DOS allocated flag is set, a warning is issued
    - a dummy sector containing repeated text "\*\*\* unused \*\* " is written
  - else if Deleted Address Mark flag is set, a warning is issued
    - if the -kd option is specified the sector is written
    - else a dummy sector containing repeated text "\*\*\* deleted \*\* " is written
  - else if the No Data flag is set, a warning is issued
    - a dummy sector containing repeated text "\*\*\* missing\*\* " is written
  - else the sector data is written

Note all cylinder and head information recorded for a sector is ignored.

## Performance of td02imx

---

As part of testing td02imx I have used a collection of .td0 files, from multiple sources, with the largest collection coming from Don Maslin's archive. In total there are 717 files, after de-duplication and removal of two test files. Using these files I have ranked the handling of the .TD0 anomalies in one of four categories

1. No problem. The file was processed and produced no warnings
2. Other ok. The file was processed and the warnings can all be explained by de-duplication of sectors and reading beyond the end of a disk

3. Possibly ok. These were files where the warnings indicate a small number of missing/extra sectors or CRC errors which, dependent on whether they are accessed may or may not cause a problem. Track 0 missing is included as in may cases it isn't accessed except for booting. Even if the problem sectors are used, it should hopefully be possible to recover most of the information.
4. Bad. These are files where there are multiple missing sectors and CRC, with in no obvious indication of how to correct when creating an .IMG file. If these disks are still needed, I suggest generating and IMD format file, which may work if a genuine non-standard sector layout that is used.

	No Problem	Other ok	Possibly ok	Bad
Count & percent	545 - 76%	89 - 12%	60 - 8%	23 - 3%

This indicates 88% are fully usable, rising to 97% being potentially usable.

For those of you with access to the Don Maslin archive the following are the "Bad" files.

File	Identified problem
compat32.td0	Odd numbered Cylinders are different from even numbered ones
cpm_2_2_03_Disk1.td0	Cylinders 31,34,36, and 38 have additional sector 1 although cylinder 34's has a CRC error
dyn-alec.td0	Too many missing IDAMs and CRC errors
xor.td0	Too many missing IDAMs
800basic.td0	Cylinders 20,21,31,32,39 missing IDAM sector Ids 4,2,2,16,18
if800cpm.td0	1/0 sector 1 is 256 bytes the reset are 512. Cylinder 24 and later all sectors are corrupt. Also some corrupt sectors on cylinder 23
hz8937-2.td0	Cylinders 0,1,2,3,4,9,13,18,22,23,26 all missing sector 1,
csr30nrm.td0	Side two all CRC errors. Side one cylinder 2-27 small number of CRC errors. Rest CRC errors. Read beyond end of disk
kpro-ii.td0	Info beyond end of disk. Multiple CRC errors
multidos.td0	Cylinder 17 missing sectors 5,11 and 16. Cylinder 18 missing sectors 0,5,6,11 and 17. Cylinder 38 unreadable
coco2.td0	Info beyond end of disk. Cylinders 0, 14, 23, 24 and 30 have extra sector 1
rmc14b31.td0	All cylinders have missing or corrupt sector 0. Cylinder 10 missing sector 10, Cylinder 36 missing sector 8
rmc14b56.td0	All cylinders have missing or corrupt sector 0.
rmc22c31.td0	All tracks have sector 0 with a CRC error.
rmc22c56.td0	All cylinders have missing or corrupt sector 0.



File	Identified problem
rml14b31.td0	Cylinder 0, 1 and 14 have no sector 0, unlike rest of disk. Cylinder 14 has a duplicate sector 1
rml14b56.td0	Cylinders 0,1,16 don't have sector 0. Cylinder 4 is missing sector id 11. Cylinder 16 has a duplicate sector 1
rml22c31.td0	Cylinder 24 cannot be read. Cylinder 23 missing sector 3. Cylinder 28 has missing sector 10, and a duplicate sector 1
rml22c3r.td0	Cylinders 0-5 have sector 0, the rest of the disk doesn't
rml22c56.td0	Cylinder 0, and 1 have no sector 0, unlike rest of disk. Cylinder 19 has missing sector 11 and a duplicate sector 1
oii40d.td0	Cylinders 57 & 58 have missing sector 25. Cylinder 67 sector 25 has a CRC error
8202trn5.td0	Multiple tracks are missing or have corrupt sector 1

Update: 17-Oct-2024 Mark Ogden