SEAGATE
F3 architecture

Table of contents

Introduction

By the year 2008 Seagate evolved from manufacture of the classic 3.5" Barracuda HDD line (3.5" Barracuda - … 7200.7, 7200.8, 7200.9, 7200.10, Barracuda ES; 2.5" Momentus - … 5400.2, 5400.3, 5400.4 …) to the new F3 architecture intended to unify the production of ATA and SCSI drives. That new architecture is mostly based on mechanics and the firmware kernel from the ATA hard drives, but a large part of firmware microarchitecture has been revised in accordance with the design used in SCSI Seagate drives. ROM has been extended and supplemented with the code enabling the controller board to function in ATA mode without the HDA attached. Furthermore, the manufacturer moved to ROM adaptive data and HDD ID information (due to that fact a board disconnected from HDA can return via the ATA interface HDD ID containing correct HDD information[1]). However, together with useful effects of that combination the new architecture also inherited a number of problems from both its parent lines. In particular, it includes a number of problems related to translator corruption (issues typical of SCSI HDD), the Pending Bug problem, etc. The new architecture also encountered some issues related to the integration of previously independent platforms and addition of new functionality. In this guide we describe the Arch F3 platform, typical problems of its models, drive diagnostics, and the methods available for work with these HDD in the PC-3000 suite.

[1] FW in the ROM of some Seagate drives contains the default HDD ID, which is updated with the information from disk surface in individual drives. Moreover, when the firmware on disk surface appears to be inaccessible for some reason, a drive may return the HDD ID template with the default model name and serial number (while the actual serial number is preserved in the ROM data structures).

Barracuda 7200.12 HDD

Fig. 1.1. A Barracuda 7200.12 HDD

1 − manufacturer – Seagate; 2 − drive family − Barracuda 7200.12;
3 − serial number − 5VM428EF; 4 − model – ST3160318AS;
5 − FW version – CC37; 6 - Site code (code of the manufacture location) – WU.

Barracuda 7200.11 HDD

Fig. 1.2. A Barracuda 7200.11 HDD

1 − manufacturer – Seagate; 2 − drive family − Barracuda 7200.11;
3 − serial number − 9QM040C5; 4 − model – ST3500320AS;
5 − FW version – SD04; 6 - Site code (code of the manufacture location) – KRATSG.

Barracuda ES.2 HDD

Fig. 1.3. A Barracuda ES.2 HDD

1 – manufacturer – Seagate; 2 – drive family – Barracuda ES.2;
3 – serial number – 9QM7D0QK; 4 – model – ST3500320NS;
5 – FW version – SN05; 6 - Site code (code of the manufacture location) – KRATSG.

Desktop HDD

Fig. 1.4. A Desktop HDD

1 − manufacturer – Seagate; 2 − drive family − Desktop HDD;
3 − serial number − W300H8J4; 4 − model – ST4000DM000;
5 − FW version – CC52; 6 - Site code (code of the manufacture location) – WU.

Laptop SSHD

Fig. 1.5. A Laptop SSHD

1 − manufacturer – Seagate; 2 − drive family − Laptop SSHD;
3 − serial number − W38051SR; 4 − model – ST1000LM014;
5 − FW version – SM11; 6 - Site code (code of the manufacture location) – WU.

Momentus 7200.3 HDD

Fig. 1.6. A Momentus 7200 HDD

1 – manufacturer – Seagate; 2 – drive family – Momentus 7200.3;
3 – serial number – 5TH0BV0S; 4 – model – ST9250421AS;
5 – FW version – SD13; 6 - Site code (code of the manufacture location) – WU.

2. Supported drive families

Drive family	Model
2.5"	
Momentus 5400.5	ST9320320AS ST9320320ASG ST9250320AS ST9200321AS ST9160310AS ST9160310ASG ST9120310AS ST9120310ASG ST980310AS ST980310ASG
Barracuda 5400.6	ST9500325AS ST9400326AS ST9320325AS ST9250315AS ST9160314AS ST9160301AS ST9120315AS ST980313AS ST9500325ASG ST9320325ASG ST9250315ASG ST9160314ASG ST9120315ASG ST980313ASG
Momentus 7200.3	ST9320421AS ST9320421ASG ST9250421AS ST9250421ASG ST9200421AS ST9200423ASG ST9160411AS ST9160411ASG ST9120411AS ST9120411ASG ST980411AS ST980411ASG
Momentus 7200.4	ST9500420AS ST9500420ASG ST9320423AS ST9320423ASG ST9250410AS ST9250410ASG ST9160412AS ST9160412ASG ST9120410AS ST9120410ASG
Constellation,	ST91000640NS ST9500620NS

Constellation.2	ST9250610NS ST9160511NS
Momentus XT	ST95005620AS
FreePlay	ST1500LM003 ST91000LM010 ST91000430AS ST9888430AS ST9750430AS
Momentus Thin	ST500LT012 ST320LT020 ST320LT007 ST500LT012 ST250LT012 ST250LT007

Drive family	Model
3.5"	
ES.2	ST31000340NS ST3750330NS ST3500320NS ST3250310NS
Barracuda 7200.11	ST31500341AS ST31000340AS ST31000333AS ST3750630AS ST3750330AS ST3640323AS ST3500620AS ST3500320AS ST3320613AS ST3160813AS
Barracuda 7200.12	ST31000528AS ST3750528AS ST3500418AS ST3500410AS ST3320418AS ST3250318AS ST3160318AS
Bara LP / Green	ST2000DL003 ST1500DL003 ST32000542AS ST31500541AS ST31000520AS
Barracuda XT	ST32000641AS
Constellation ES / ES.2	ST2000NM0011 ST2000NM0031 ST2000NM0051 ST1000NM0011 ST1000NM0031 ST1000NM0051 ST500NM0011 ST500NM0031 ST500NM0051 ST33000650NS ST33000651NS

	ST33000652NS ST32000645NS ST32000646NS ST32000647NS

3. Getting started

For details about preparation for work with the utility and HDD connection to the suite, please refer to *chapter 3 of Seagate documentation, part 1*.

Attention! HDD ROM contains adaptive data. Check to ensure that the controller board matches the heads and disk assembly (HDA). To do that, you can even use a board disconnected from HDA, which returns HDD ID information via the ATA interface anyway. Having read HDD ID, you can compare the model and its serial number to the information printed on HDA label. If the returned HDD ID contains a template, use the tools for operations with ROM described further to obtain the serial number from it *(see section 7.2, Work with Flash ROM image file and Correction of HDD ID data in ROM*). In addition, in cases when the terminal command mode is available you can extract the serial number by sending the [Ctrl]+[R] and [Ctrl]+[L] pair of commands; the HDD will respond then with a report similar to the following:

GrenadaBP LuxorPlus2 4K, GPF1 MC with new serpent DFW
Product FamilyId: 58, MemberId: 03
HDA SN: W1F23RHF, RPM: 7199, Wedges: 188, Heads: 6, OrigHeads: A, Lbas: 00002BB5420C, PreampType: CC 06
Bits/Symbol: C, Symbols/UserSector: BEB, Symbols/SystemSector: 1A5
PCBA SN: 0000E3246TRJ, Controller: LUXORLPLUS_2_0(203)(FF-FF-FF-F, Channel: STX_DUNRAVEN, PowerAsic: Unknown Rev 00, BufferBytes: 4000000
Package Version: GPG17D.CCD2.AX0851.CC24, Package P/N: 100714019, Package Global ID: 00188399,

Attention! Please keep in mind that if a non-native controller board is used, then any operation involving recording to disk surface may cause irreversible corruption of service data resulting in inaccessibility of user data.

4. Launching the utility

As soon as the utility starts, it displays the drive family selection dialog. The family is recognized automatically based on the HDD ID information. If a drive returns no HDD ID (permanent BSY, ABR response in case of firmware damage), you can select its family manually.

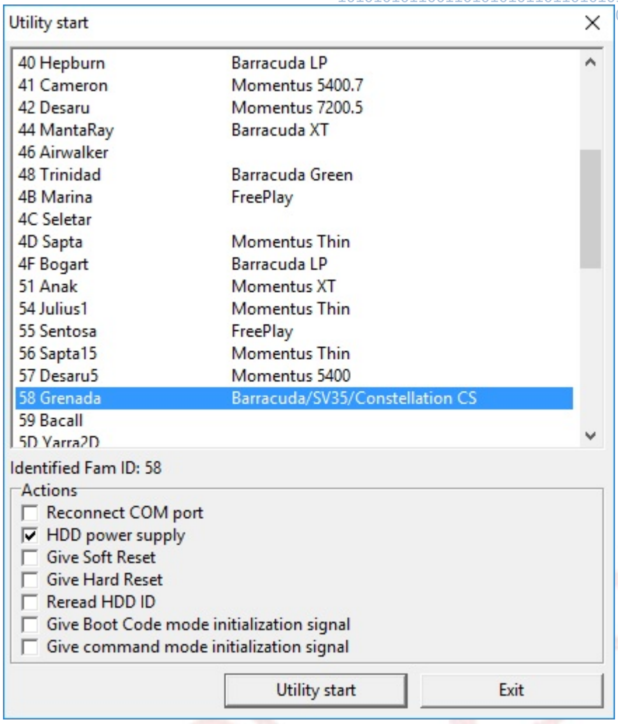


Fig. 4.1

As soon as you click the 'Utility start' button, the utility will access the drive for automatic configuration requesting the following information:

```
Family ID: 58
Selected family......................... : 58, Grenada
Model by ID............................. : ST2000DM001-9YN164

Loading default settings!
Use the options dialog to review and save utility settings.
Then press OK!


Family default settings loading!


Tech Key...

Requesting FW Pkg ver ........ GR753C.CCD4.BP01BG.CC4H


Detecting Phys Sct Size...
Result.................................. : 4096

Detecting Max Head number...
Phys. heads............................. : 4

Obtaining Saved Mode Pages File information...
Reading Saved Mode Pages...
Parsing Saved Mode Pages...

LBA alignment........................... : 0

Obtaining Registry File information...
Reading Registry File...
Parsing Registry File...

Media Cache
===========
MC       : disabled
MC Size  : 0x44C850
MCMT Ver : MCMTV03
```

```
Collecting Sys Files information...
  ID        Len
===============
   0         36  Reserved Defect Table (SA defect table)
   1          2  Drive Information File (DIF)
   2         16  Performance Parameter File (PPF)
   3       7372  Primary Defect File (P-List))
   5          4  Manufacturing Information File (MIF)
   8       4800
   B       1000
   C     122880  Serial Port Test Results File (Self Scan Log)
   D        160
   E        432  Data Integrity Check (DIC), HEAD 0
   F        223  DIC Dummy File
  13          8  Disc Registry File (settings, security, etc)
  15        100  BGMS File (Directed Offline Scan)
  16          2
  17         40
  19         24  Log File
  1A         33  Log File
  1B          1  Log File
  1C          9
  1D        128  Physical Overlay File 0
  1E        128  Physical Overlay File 1
  22         80  Drive Self Test (DST), Head 0
  23         80  Drive Self Test (DST), Head 1
  24         80  Drive Self Test (DST), Head 2
  25         80  Drive Self Test (DST), Head 3
  2A          9  Saved Mode Pages (HDD settings)
  2B       9862  R/W Operating Parameters (translator, etc)
  2C        432  Data Integrity Check (DIC), HEAD 1
  2D        432  Data Integrity Check (DIC), HEAD 2
  2E        432  Data Integrity Check (DIC), HEAD 3
  33       4576
  34        640  Packed CONGEN XML definition
  39       8192
  3A          4
  42       4096
  43       8192
Collecting Sys File Infos completed

Collecting DT Infos...
ROM:
  ID     MemOffs         Len
===========================
   0    40030200       37888  Read Adaptive Parameters File (RAP)
   1    40030000         512  Controller Adaptive Parameters File (CAP)
   4       1D400        7168  Servo Adaptive Parameter File (SAP)
        40002000       13312
              total    20480
   5       6C000       81920  Servo Firmware Table
   A      2574C0        1024  Seek Profile
  13    4011EA98        1440  Dependent Zone Table for User Area
  14    4011F038          60  Dependent Zone Table for System Area
  15           0      524288  Flash ROM
  1D    40030220         424  VBAR Configuration File
  1E    400305B8        4016  Zone Format Parameters File
  26    401179A0       28920
  27    401178B0         240

RAM:
  ID     MemOffs         Len
===========================
   0    40030200       37888  Read Adaptive Parameters File (RAP)
   1    40030000         512  Controller Adaptive Parameters File (CAP)
   2    400303C8         496  Zone Table
   4       1D400        7168  Servo Adaptive Parameter File (SAP)
        40002000       13312
              total    20480
   5       6C000       81920  Servo Firmware Table
   6      A63D00       81964  Servo Flaw Table
   A      2574C0        1024  Seek Profile
  13    4011EA98        1440  Dependent Zone Table for User Area
  14    4011F038          60  Dependent Zone Table for System Area
  15           0      524288  Flash ROM
  16    40030200          32  Drive Configuration Table
  17    400319A0        2016  Temperature Control Configuration
  1D    40030220         424  VBAR Configuration File
  1E    400305B8        4016  Zone Format Parameters File
  1F    4020D858        3300  RW working user preamp table
  20    4020E540         110  RW working system preamp table
```

```
   26     401179A0       28920
   27     401178B0         240

Collecting DT Infos completed


Reading autotuning buffer...............   Ok
Searching module table in RAM... .......   Ok
Searching volume 3 file table in RAM....   Ok
Searching volume 0 file table in RAM....   Ok
Searching SMP RAM FID in RAM............   Ok
Searching Non Resident G-List FID in RAM   Ok
Searching CFW OVL FID in RAM............   Ok
Searching DT table(RAM, WR) in RAM......   Ok
Searching DT table(RAM, RD) in RAM......   Ok

System Files I/O method detection... ATA(direct)
```

If errors are encountered during initialization, respective messages will be logged, and the utility will automatically provide troubleshooting advice (see Fig.4.2 for an example).

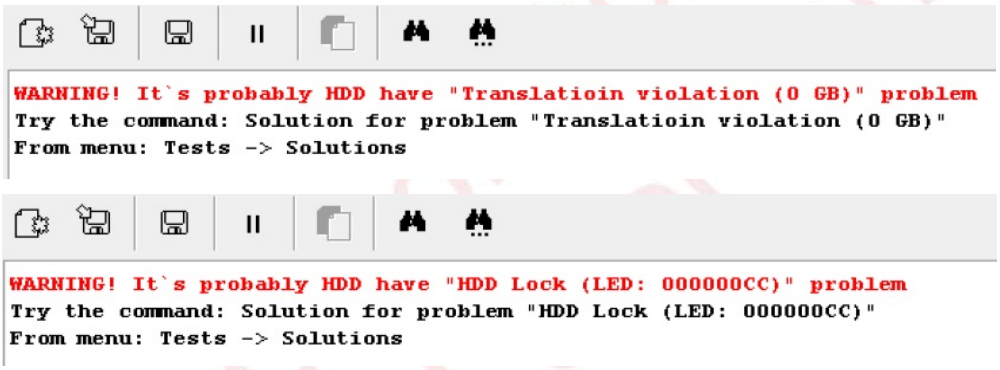


Fig. 4.2

After completion of the device start and initialization procedure the main window of the utility (including terminal) will become accessible (*Fig. 4.3*).

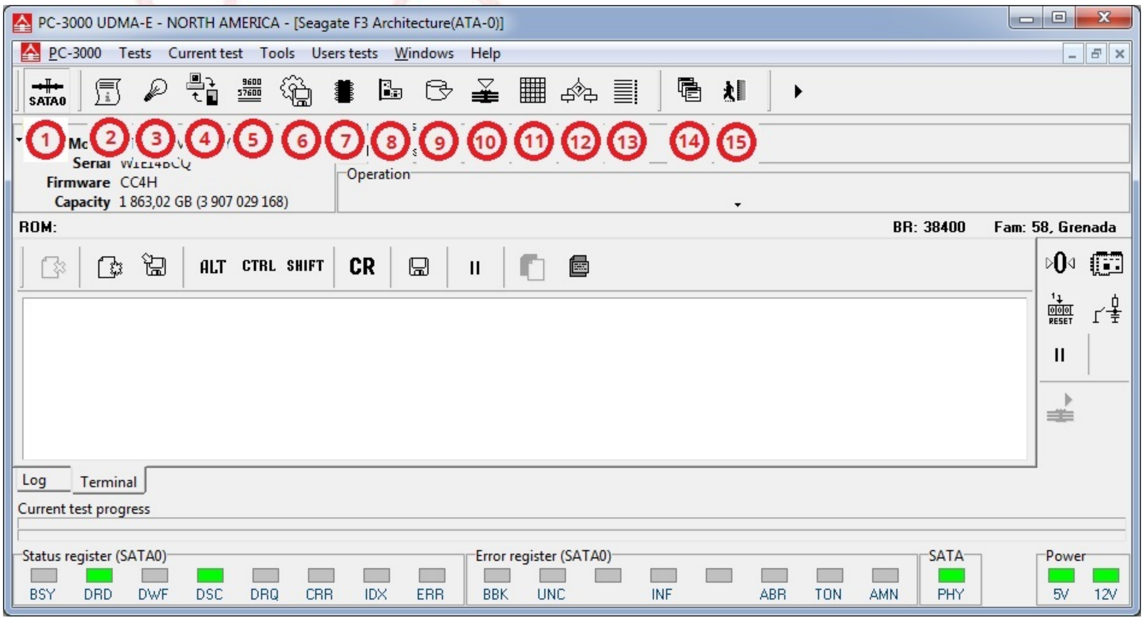


Fig. 4.3

Toolbar buttons:

1 – drive power management;

2 – loader (Update Microcode) creation and uploading;

3 – utility status dialog (which allows, among other things, to re-read the lists of modules and DT);

4 – reconnection to COM port (necessary when the USB to COM adapter freezes);

5 – detection and switching of the rate of data exchange with HDD via terminal;

6 – ROM reading/writing menu;

7 – group of features for operations with the controller: DT reading/writing, modification of the heads map, SAP flags, and HDD ID in drive RAM;

8 – group of features for operations with the service area (backup of service information, reading/writing of HDD modules and system files, HDD ID modification, translator recalculation and restoration, resetting of passwords);

9 – logical scanning of disk surface;

10 – work with HDD defect lists: Non Resident G-List, P-List, Non Resident Primary DST List;

11 – group of features for automatic repair of typical malfunctions;

12 – a set of custom commands;

13 – switch between PC-3000 windows;

14 – exit from the utility.

We should separately mention the list of automated solutions for HDD problems available in the utility (*Fig. 4.4*):

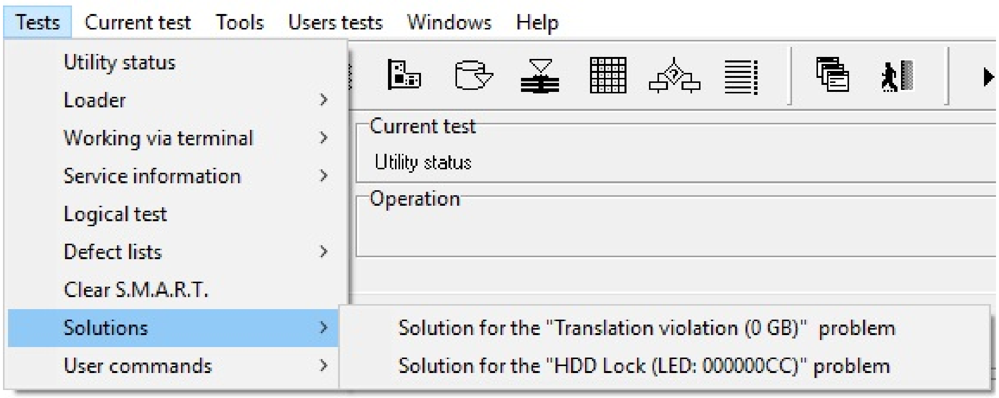


Fig. 4.4

- A drive may be detected in the system, but report zero capacity (0 Gb) (except for the situations when zero capacity is caused by a damaged controller board / heads or non-native controller).
- Host system cannot detect HDD presence because the drive permanently remains in the BSY state - HDD is locked (LED: 000000CC).

5. Frequent malfunctions

5.1. Motor seizure

If a drive motor seizure occurs, then a HDD cannot spin up the disks even if the controller board and motor winding are functional (confirmed by replacing the board with a known good board from another drive of the same family having the same revision and checking the coils (as described in *section 11, Appendix 3. Testing the motor winding coils*). That malfunction can be identified by typical tactile vibration of the HDA when power is switched on. When the HDA is opened and the heads inside are not stuck, it is very difficult or impossible to turn the disks manually.

To solve the problem, move the disks stack into a donor HDA. In that case you can use the controller board, heads and magnets from the damaged drive (if native heads get damaged, they can be replaced).

5.2. "Stuck" heads

"Stuck" heads have manifestations identical to the motor seizure case. When HDA is opened, the heads turn out to be outside the parking area. They stick to the disk surface because of molecular attraction forces.

Unfortunately, there are no universal methods "releasing" the heads. You should rely on your experience and the expertise of your colleagues.

5.3. HDD returns HDD ID (the host system detects it), but reports capacity = 0

This malfunction can be caused by the following reasons:

- non-native controller board;
- damaged read/write heads in the HDA;
- broken contact between the controller board and HDA in the connector of the commutator preamplifier (oxidized contacts, etc.) (*Fig. 5.2*);
- damaged service information (SMART, G-List, translator…).

Fig. 5.2

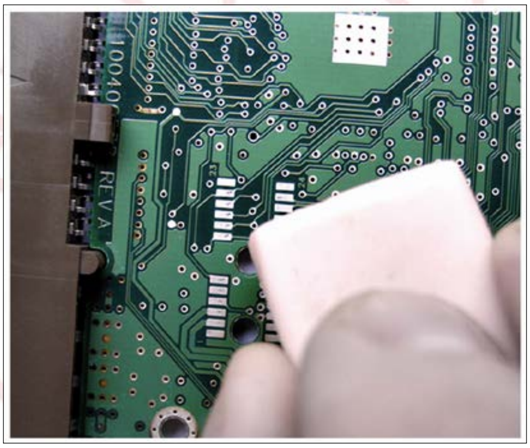


Fig. 5.1

Section 0 of this manual describes the method that can be used to ensure that the board is native. At present there are no methods for drive restoration if its native board is lost.

If a contact is broken, try cleaning the connector, for example, using a common office eraser (*see Fig. 5.1*).

If magnetic heads are damaged, a drive usually produces knocking sounds when powered up. In that case it is recommended to replace the malfunctioning heads stack.

If service data get corrupted, use the "Troubleshoot "Invalid translation (0 GB)"" menu item (*Fig. 5.3*). Selection of that item makes the utility perform automatically a sequence of operations to fix the problem (*Fig. 5.4*) resulting in restoration of the access to user data.

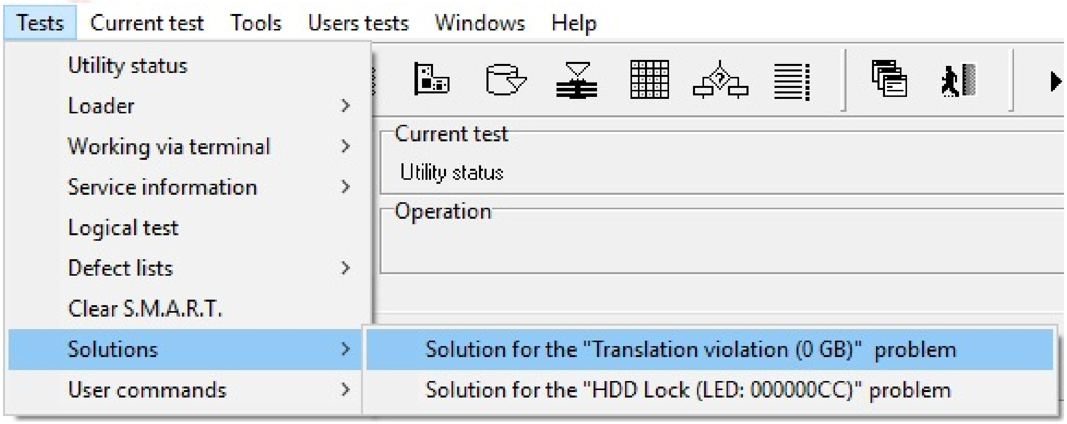


Fig. 5.3

```
WARNING! It`s probably HDD have "Translatioin violation (0 GB)" problem
Try the command: Solution for problem "Translatioin violation (0 GB)"
From menu: Tests -> Solutions



Test   : Solution for problem "Translatioin violation (0 GB)"
HDD capacity restrict. check (SetMax)...
HDD capacity restrict. check (SetMax)... : Error
SMART erase............................. : Ok
HDD capacity restrict. check (SetMax)...
HDD capacity restrict. check (SetMax)... : Error
G-List erase............................ : Ok
HDD capacity restrict. check (SetMax)...
HDD capacity restrict. check (SetMax)... : Error
 Max Wr Retries = 00, Max Rd Retries = 00, Max ECC T-Level = 14, Max Certify

User Partition Format  10% complete, Zone 00, Pass 00, LBA 00004339, ErrCode
User Partition Format  10% complete, Zone 00, Pass 00, LBA 00008DED, ErrCode
User Partition Format Successful - Elapsed Time 0 mins 10 secs

F3 T>
Translator regeneration................. : Ok
HDD capacity restrict. check (SetMax)...
HDD capacity restrict. check (SetMax)... : Ok
Solution for problem "Translatioin violation (0 GB)"........ : Ok
```

Fig. 5.4

5.4. Host system does not detect a HDD, which permanently remains BSY

This malfunction can be caused by the following reasons:

- hardware HDD malfunctions (malfunction of the controller board, heads, etc.);
- damaged service information (SMART, G-List, translator, Media Cache data structure[1]).

In the former case, you should check the HDD controller board using the method described in *section 10, Appendix 2. Testing the HDD controller board*. The latter situation can be resolved using software tools. If the terminal command mode is available when the drive is powered, perform malfunction diagnostics as described in section *5.6, Diagnostics using the list of 32 recent RW commands*. If the command mode cannot be invoked because of HDD freeze or LED error, select the "Solution for problem "HDD Lock (LED: 000000CC)"" menu item (*Fig. 5.5*) having bridged the reading channel contacts. If the freeze of a powered drive is accompanied with the "LED: 000000CC" message output to terminal, the procedure should be performed till the end. However, if the freeze has other symptoms, then once you obtain access to terminal using the automated solution for the LED: 000000CC problem, click 'Cancel' in the main window guiding you through the read channel bridging and continue HDD diagnostics as advised in *section 5.6, Diagnostics using the list of 32 recent RW commands*. For details on unlocking the HDD locked with the LED: 000000CC error please see *chapter 8, Unlocking HDD in case of LED: CC error* .

[1] Media Cache is a drive's data cache located in its fastest disk surface area. Initially it was introduced in hybrid HDD where it was located in NAND Flash on the controller board. As the technology was adjusted for better affordability, the cache was relocated to the outer tracks of HDD surface, where linear rate of data exchange reaches its maximum.

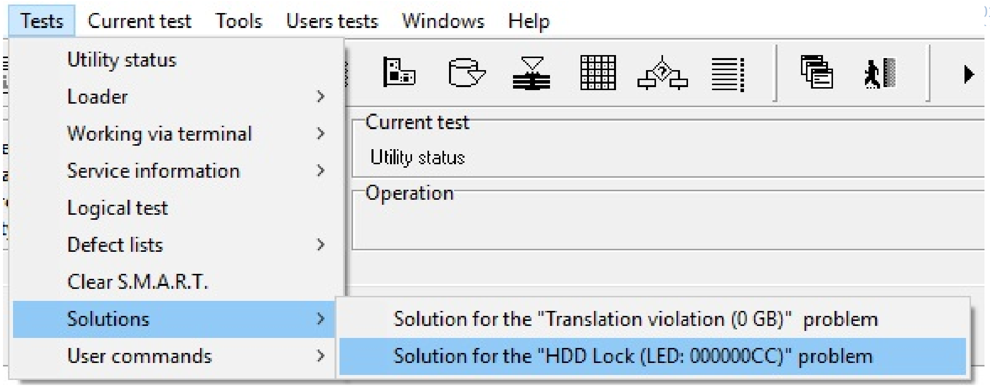


Fig. 5.5

5.5. Host system detects a HDD, drive beginning is visible but starting with a certain LBA continuous space until the end is filled with UNC errors

When such HDD arrives for diagnostics, the host system recognizes it correctly, but the drive allows reading access just to a portion of its surface. Beginning with a certain LBA, and until the end of disk space it returns UNC errors while reading.

This malfunction is caused by data corruption in the translator. To explain the nature of corruption, a few words about the HDD factory testing procedure are necessary. During the first stage Self Scan process builds the main list of drive defects (P-List). It is used to generate the initial translator. Then during the presale preparation additional surface testing is performed to reveal the areas where a HDD demonstrates slower or unstable performance. However, these areas are relocated using directly the translator instead of P-List. Besides, when defects are added to the translator, information about them is appended to Non Resident G-List, which allows to supplement the translator in series losing no information about additional defects hidden earlier. An attempt to recalculate the translator based on P-List only discards information about those defects both in the translator and in Non Resident G-List. At the same time ECC for each sector is generated using its LBA number under which it will be recorded. When the information about defects found during post-processing gets lost, the numbering of sectors becomes shifted beginning with the first "lost" defect: LBA used to read a sector no longer matches the actual address where the sector data have been recorded. As a result, UNC errors will be returned at an attempt to read from the surface starting with a certain LBA until the end of disk space.

Corruption of translator data can have various causes. The main of them are as follows:

- Unlocking of LED 000000CC using the methods described in the Internet without prior backup of the translator module.
- Logical corruption of the translator module caused by its overwriting by the drive during G-List update affected by writing malfunctions (in that case the drive may freeze during firmware start or return zero capacity).
- HDD errors occurring during translator recalculation. In particular, in some cases translator recalculation may add erroneous data to Non Resident G-List; moreover, the flags field of such defect records may contain code 20 preventing removal of such corrupted records by the recalculation procedure and thus causing shifts in translation.
- Mistakes made during work with the translator, Non Resident G-List, and P-List.

While the first situation results from unsafe operations performed by the user or repair technicians, the second case may occur "for no reason" when a user is working with the device.

For troubleshooting of this issue, three approaches exist (for details please refer to the corresponding chapters):

- Automatic solutions available from the menu 'Tests → Service information → Work with service area → Translator → Recover translator and Single LBA position adjustment'.
- Manual translator restoration using the tools available in the utility.
- Creation of a DE task in translator restoration mode with the 'Read from active PC-3000 Utility' option enabled for the reading command (the Seagate F3 utility must be running at the same time).

In the first and the second case the existing translator will be supplemented with the "translation forks" revealed during the testing procedure (lost defects). In the third case Data Extractor will read data "as is" and ignore the translation shifts; then it will build a virtual translator to access the data.

Attention! Before any operation with the HDD service information and with the translator in particular, please back up the service information to a separate folder in the HDD profile to be able to roll back any erroneous changes.

We should also mention that besides the loss of information about Post-Process defects[1] the translator can be corrupted by addition of "extra" defects to Non resident G-List or P-List. Although the utility contains a feature clearing Non resident G-List, using it before translator restoration cannot be recommended unconditionally. As a matter of fact, Non resident G-List may contain tracks of HDD surface representing thousands of sectors. Thus, manual analysis and repair of the defect lists may appear quite useful. Its structure and the methods of manual restoration will be discussed in detail in the *section 5.5.4.1. "Delicate" work with Non Resident G-List.* "Extra" defects can also be introduced into P-List when the command transferring G-List → P-List (implemented only in some firmware versions) is invoked. In this case you should use the P-List editor of the utility having located physical coordinates of the records with problems. It should be mentioned that in case of "extra" defects translation recovery using the automated solution in the utility is impossible because the automatic method adds defect records on its own until it reaches a complete translation match. As a result, the utility will discover data both to the left from the translator divergence location and to the right, in the area unreadable with regular ATA commands, output a message about ambiguous condition, and terminate the process abnormally. In that case you should analyze the divergence area manually, correct the information about defects and either perform the restoration procedure manually or restart the automated solution.

5.5.1. Automatic translator restoration (for the entire LBA space)

This solution is available from the menu 'Tests → Service information → Work with service area → Translator → Recover translator'. When the procedure starts, you have to select the appropriate profile for the drive being restored and check the restoration settings in the dialog displayed next.

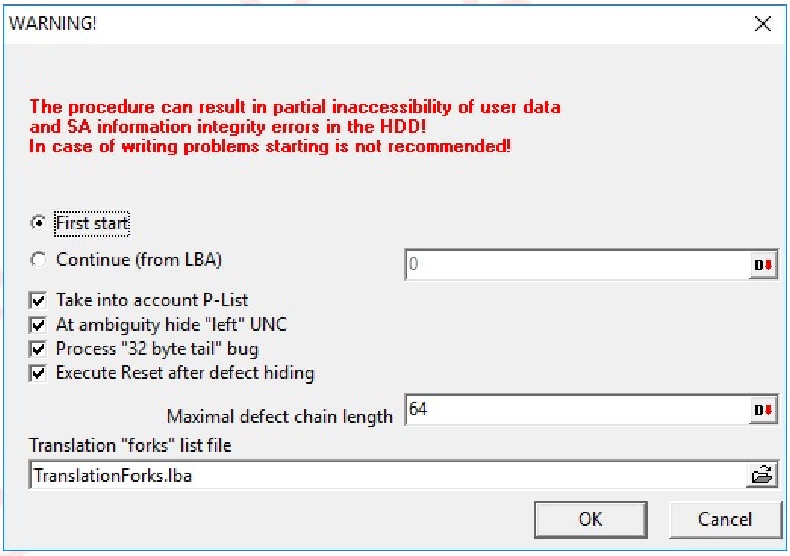


Fig. 5.6

If the 'First start' mode is selected, the utility will suggest saving a backup copy of the service information (specific items and the destination folder for subsequent backup procedure), disable in the SA Autoreassign, Deferred defect hiding, offline self-test, identify the type of command used to hide defects, and recalculate the base translator. Then it will start surface scan beginning with LBA = 0.

[1] Factory detection of defects in Seagate Arch F3 uses several passes. The first pass occurs during the Self Scan procedure. Subsequent passes are performed in a stand under external software control. Post-process defects are specifically the defects revealed during these final passes. We should mention that they are added directly to the translator instead of P-List. Therefore, when the translator is recalculated using P-List only, information about these defects may be lost irreversibly.

If the option to 'Continue (from LBA)' is selected, the utility will resume scanning from the LBA entered in the dialog. If you interrupt the testing process, the utility will save in the profile data the LBA where the procedure was paused and offer to resume it from that location the next time it is started.

The 'Take into account P-List' option must be selected if the drive has a valid P-List. If the P-List is corrupted (e.g., with erroneous data transferred from G-List) the option should be deselected.

The option 'At ambiguity hide "left" UNC' allows automatic handling of the situations when the algorithm cannot reliably recognize the "fork" type because the area with translation deviation is preceded by some unreadable (UNC) sectors (and so the algorithm cannot access their content for analysis).

The option to 'Process "32 byte tail" bug' prevents the utility from analyzing the last 32 bytes of each sector, which some FW versions may misinterpret during reading in factory mode.

The option to 'Execute Reset after defect hiding' configures the utility to send a Soft Reset after each addition of defects to the translator in order to update it.

In the dialog you can also define the file name for the list of "translation forks" where the scanning process will store found nodes. If later a need arises to recalculate the translator considering P-List only, this list will allow you to restore the initial translation by hiding the defects into the translator based on the file data.

The algorithm is automatic and requires operator participation only in cases when it cannot detect precisely the type of node being hidden in the translator. In that case the utility will display a corresponding notification with a suggestion to resolve the ambiguity manually[1].

When the algorithm completes its work, the utility will display the list of found translation "forks". **IT SHOULD NOT BE HIDDEN TO THE TRANSLATOR!** It is intended for two purposes **ONLY**: for your information and quick restoration of the initial condition if for some reason translator recalculation based on P-List has been performed and consequently the result of algorithm operation gets lost.

5.5.2. Automatic location correction for a specific LBA (beginning with a certain LBA)

This solution is available from the menu 'Tests → Service information → Work with service area → Translator → Single LBA position adjustment'. When the procedure starts, you have to select the appropriate profile for the drive being restored and check the restoration settings in the dialog displayed next.

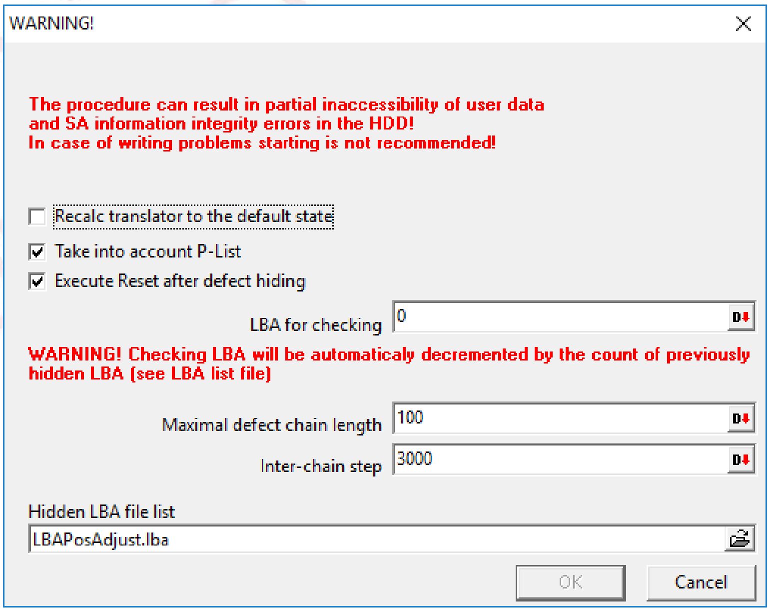


Fig. 5.7

1 Please remember while doing that to append the list of found nodes. The step will allow you to continue correct automatic translator restoration later.

The feature is necessary in situations when it is impossible to restore translation correctly for the entire disk surface or when such restoration appears to be complicated for some reason. In particular, it may prove helpful in case of multiple BAD sectors preceding the area with the sought data (e.g., when reading via the utility discovers the sought second partition or the body of the necessary file). In that case it may be possible to keep hiding sectors preceding the required (and constantly tracked) surface location until the location becomes readable using logical level access. The feature described here serves specifically for that purpose. Once the procedure completes successfully, the beginning of the necessary area will be readable already. Then you may attempt restoring the translation further automatically using the feature described in the previous section[1].

The option 'Recalc translator to the default state' allows recalculating the translator to its initial state when the mode starts. It is necessary when the mode is initiated for the first time and the results of earlier failed restoration attempts have to be discarded.

The option 'Take into account P-List' indicates to the utility whether the P-List information should be used during the procedure. In case of considerable discrepancies between the available P-List and the working one the option allows you to exclude the list from the process and then use the same mode while restoring the translator in completely automatic mode.

The option to 'Execute Reset after defect hiding' configures the utility to send a Soft Reset after each addition of defects to the translator in order to update it.

'LBA for checking' defines the location for constant tracking while the procedure is being performed.

'Maximal defect chain length' defines the largest allowed size for a contiguous area to be hidden. The option is necessary to prevent automatic hiding of an entire track by the drive when a certain number of hidden LBA is reached because it would disrupt the algorithm operation. When the corresponding number of hidden LBA is reached, the utility will automatically shift the LBA relocation area (before the one being monitored) to the left by the 'Inter-chain step' value.

5.5.3. Data Extractor task

When this solution is selected, you have to specify during creation of a new DE task that it should be generated in the translator recalculation mode. Once the task is created, open its settings dialog and use the command to 'Read from active PC-3000 Utility'. Please keep in mind that data will be transferred using SCT[2] with intermediate conversion to PCHS and data access in sector-by-sector mode[3]. Therefore, data reading will be considerably slower than in UDMA mode!

Once the task configuration detailed above is specified, you can proceed to data restoration following the guidelines in Data Extractor documentation describing the translator recalculation mode.

Attention! Some FW versions return incorrect data while reading in factory mode. In particular, the last 32 bytes of each sector may be garbled.

5.5.4. Manual translator restoration

This method requires knowledge of some peculiarities in the operation of translator in Seagate F3 drives.

As we have noted above, during ECC calculation for sector data the drive uses the number of the LBA that it is working with. Therefore, beginning with the translator "fork" point the drive will detect reading errors. On the ATA command level it will return then the UNC error informing about unrecoverable data. The drive itself can discern in such case translation errors and situations when it cannot recover incorrectly read data. Unfortunately, no methods are known at present that could be used to obtain from a drive the LBA number that has been used to record such data[4].

---

[1] To do that, you have to start the procedure and configure it to continue from the LBA already mapped to its original location.

[2] SCT (SMART Command Transfer) implies transfer of additional commands via the data sectors of standard ATA commands for operations with SMART logs (for details please refer to the ATA specification).

[3] Reading specifics in case of a corrupted translator means that each sector will be read separately and an error status (translation disruption) will be returned after that.

[4] While reading a sector producing errors, the drive firmware outputs to the terminal messages with the codes: 03110081 – actual UNC sector; 04090082 – addressing error (translator "mismatch"). No information is output about the LBA used to write the sector.

Thus during translator restoration indirect methods have to be used to identify the type of the fork to hide. It can be both a "left" or a "right" fork. To understand the types please keep in mind that hiding of post-processing defects occurs in several passes. Different passes may reveal defects before or after the previously hidden ones. Besides, during each recovery step you may discover "forks" added during the current test cycle, i.e. a "left fork" when the area hidden at the factory is readable because it was written to these addresses but immediately followed by unreadable data, because the LBA where it was supposed to be written will match the starting LBA of the "left" branch. **Therefore, a "left" fork should be hidden in the direction of LBA number decrease UNTIL THE UNC OCCURRENCE LOCATION.** Testing may also reveal "forks" hidden during earlier scanning steps. These forks have been written before hiding with larger LBA number than they occupy at the moment. In this situation reading of user data terminates abruptly with a UNC error followed by the hidden area, then user data continue. Please keep in mind in this connection that post-processing testing used to be based on recording of a plain pattern – the entire sector was filled with a single byte. Initially drives were tested using sector filling with byte 0x77, then manufacturer started filling the sectors with zeroes. While in the first case it is quite easy to recognize the 0x77 code pattern (the probability that user data will match it and occur in the "fork" location is rather low), filling with zeroes causes ambiguity – before sale the entire drive surface is filled with zeroes and there is no way to discern by sector content a sector written and hidden during surface tests from a sector in the user space of the HDD. The difference can only be noticed if non-zero user data are present in the area in question. Following from the above, you can examine the "fork" area and make assumptions about the position and size of the hidden area based on the data placement. While you can read the data preceding the LBA with UNC error using a regular ATA command, reading the LBA with UNC error and the following sectors requires the extended functionality of the utility. To do that, use the sector editor from the Tools menu. To access the data in the UNC area, you have to switch the reading mode to reading via the utility. The utility will display a prompt asking whether data should be returned in case of a real reading error (UNC). To identify translation problems, you should respond 'No' (responding 'Yes' will allow you to obtain the uncorrected data in case of a corrupted sector, but the mode is unusable for the purpose of translator restoration).

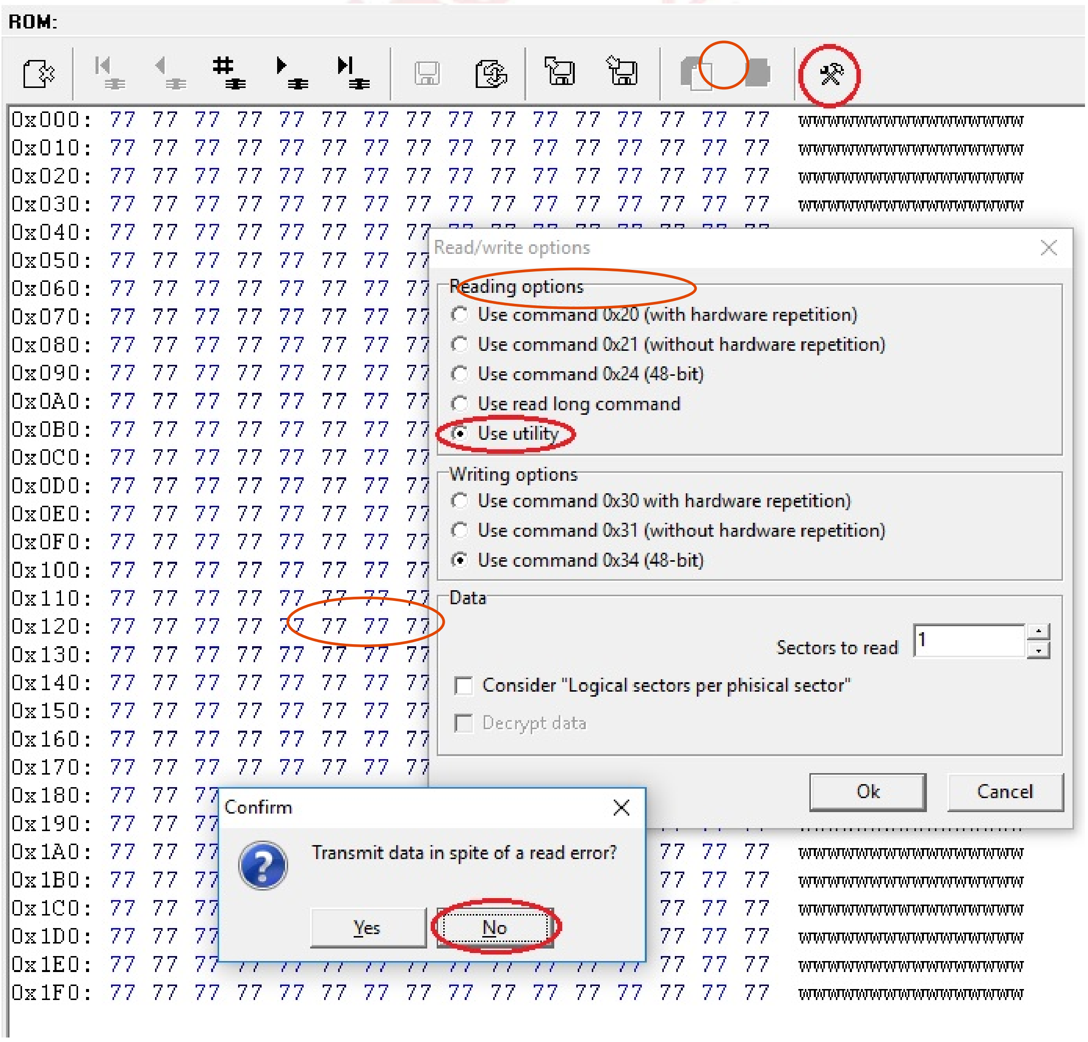


Fig. 5.8

Once the position of the hidden area relatively to UNC is identified, you should add the information about that area to the translator using the defects editor of the utility.

Create in the list a defect record with the corresponding starting point and length, then invoke the 'Hide to Slip-List' command from the shortcut menu (*Fig. 5.9*). When the operation completes, user data up to the next "fork" will become accessible for reading provided the hidden area has been selected correctly.

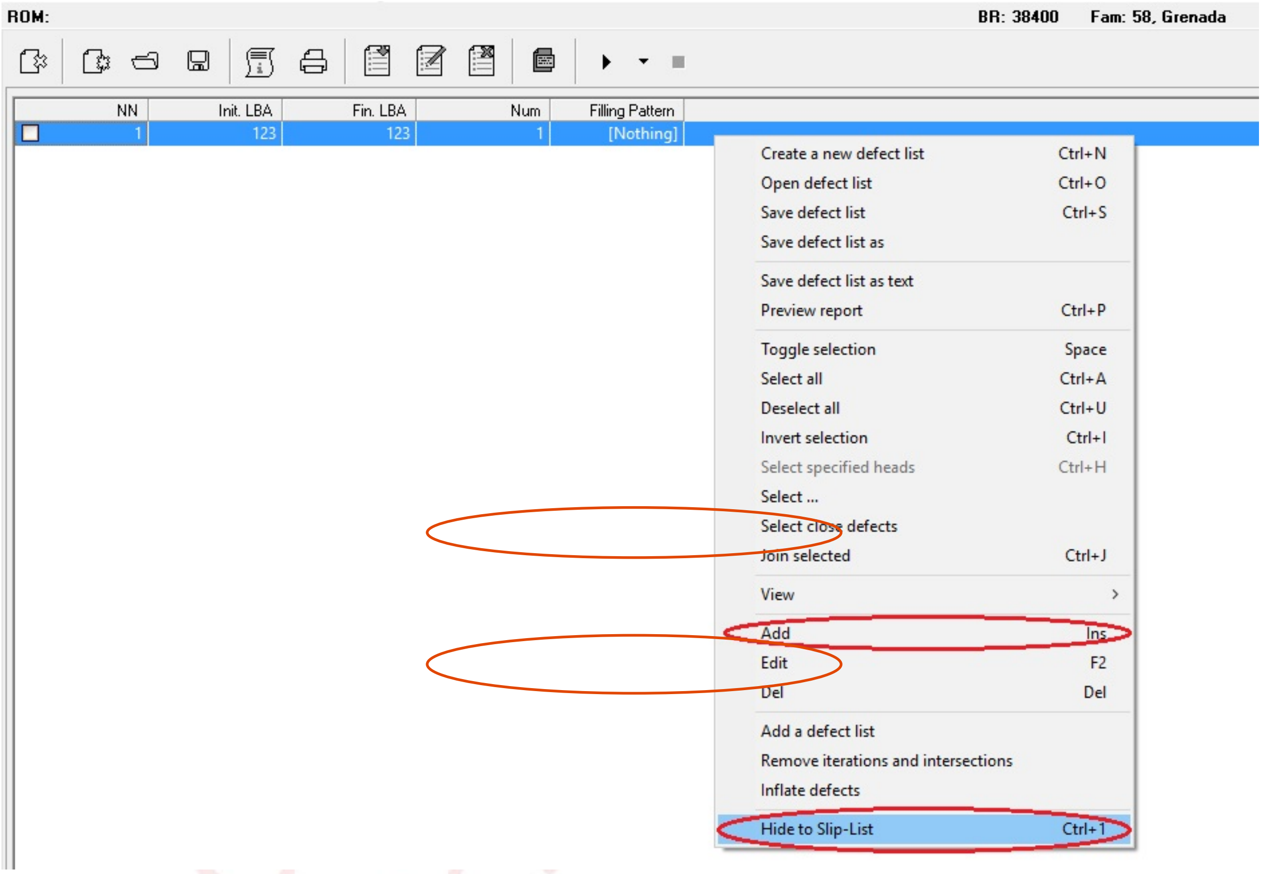


Fig. 5.9

The actual searching for the "fork" should be performed using the logical scanning test of the utility with verification enabled ('Tests → Logical test'). The maximum number of defects to be revealed before the test is stopped should be set in the test options to 1. Thus the test will stop immediately at the supposed fork location enabling creation of a defects list pointing to that LBA.

Attention! Please keep in mind while analyzing the test results that apart from the translation "forks" the drive may contain real bad blocks.

The case of "extra" defects is worth mentioning separately. Essentially, the problem is as follows. For some reason a drive may have added to P-List or Non Resident G-List[1] additional defect records disrupting translation. Automatic translator recovery procedure (both methods) additionally hides sectors in the process and thus obviously cannot "fix" an area already containing sectors, which have been hidden in error. In that case you should try locating and removing the extra records using both the data from surface obtained with regular reading and reading via the utility. The utility provides the following opportunities for that purpose:

- P-List editing. The editor allows to extract from a drive the list of defects stored in P-List, and then use the data from surface and the flags of defect records (in particular, records with the G-List -> P-List flag are suspicious) to identify and remove the records disrupting translation. Please keep in mind that a single defect record may describe a chain of sectors, some of which may belong to a correct list while the rest are

[1] Meaning the records with the flag other than 0 (most frequently, 0x20), as the records with the 0 flag are removed during translator recalculation based on P-List.

invalid. When such record is deleted, the user data space gets ***supplemented*** with a certain number of extra sectors, which can be hidden in the translator automatically or manually.

- Non Resident G-List clearing. In many cases that list of defect records supplementing the translator is empty. However, an error during translator recalculation or other corruptions of service information may cause in it records with a flag other than 0, which will prevent the translator recalculation procedure from clearing the list and thus include that invalid information in the translation process. The feature allows you to force clearing of Non Resident G-List, after which you can again initiate automatic restoration in the 'First start' mode[1].
- Non Resident G-List editing. Since the list can contain both correct and invalid records, the utility allows you to work with them selectively in manual mode.

Attention! Records in Non Resident G-List may be described both as single sectors or chains thereof. A chain may span across several tracks. Therefore, a backup copy of Non Resident G-List should be saved prior to any manipulations with it. At the same time, clearing it before restoration cannot be recommended with certainty in all cases as it may result in a time-consuming process hiding thousands of sectors of a defective track.

As tracks hidden in Non Resident G-List are not uncommon, we shall describe further the method for their identification and quick hiding. If automatic restoration stops because the maximum length of a hidden chain is reached, you can suppose one of two scenarios - an "extra" defect and a long, often track-sized chain. Let us examine the second case. Suppose that the translator has diverged at LBA = 10 000 000. For further diagnostics we need to convert the number in hexadecimal notation. LBA = 10 000 000 = 0x00989680. First, we identify physical location of the LBA. To do that, use the terminal command A>**F**aaaa,bbbb,,10, i.e. command **F** on level A> with the aaaa and bbbb parameters defining the LBA[2]. Here bbbb represents the last 4 characters of the LBA number in hexadecimal notation, aaaa – the next 4 characters. The 0x prefix is omitted. If an LBA in hexadecimal notation is represented with less than 4 characters, they are used in the bbbb position as they are, with 0 substituted in the "a" position. Thus the number 0x00989680 is split into two groups – 98 and 9680; therefore, the necessary command is as follows:

A>**F98,9680,,10**

Here is an example of its execution by a drive:

```
F3 A>F98,9680,,10

Track Info:
Partition PhyCyl   LogCyl   NomCyl   RadiusMils   LogHd Zn LogicalTrack FirstLba
User      0000111B 0000111B 00001068 +1.771750E+3 04    01 FFFFFFFFFFFF 0000009895D9

FirstPba     LogSecs PhySecs WdgSkw SecPerFrm WdgPerFrm
00000098AE80 0165    0165    0067   0033      0038

Sector Info:
LBA          PBA          LogSec PhySec Wdg  SFI      Split     Burst
000000989680 00000098AF27 00A7   00A7   011E 000C84DF 06D8:0513 011F
000000989681 00000098AF28 00A8   00A8   011F 000C9123 05C5:0626 0120
000000989682 00000098AF29 00A9   00A9   0120 000C9D67 04B2:0739 0121
000000989683 00000098AF2A 00AA   00AA   0121 000CA9AB 039F:084C 0122
000000989684 00000098AF2B 00AB   00AB   0122 000CB5EF 028C:095F 0123
000000989685 00000098AF2C 00AC   00AC   0123 000CC233 0179:0A72 0124
000000989686 00000098AF2D 00AD   00AD   0124 000CCE77 0066:1(0AF1):0094 0125
000000989687 00000098AF2E 00AE   00AE   0126 000CDAFB 0A44:01A7 0127
000000989688 00000098AF2F 00AF   00AF   0127 000CE73F 0931:02BA 0128
000000989689 00000098AF30 00B0   00B0   0128 000CF383 081E:03CD 0129
00000098968A 00000098AF31 00B1   00B1   0129 000CFFC7 070B:04E0 012A
00000098968B 00000098AF32 00B2   00B2   012A 000D0C0B 05F8:05F3 012B
00000098968C 00000098AF33 00B3   00B3   012B 000D184F 04E5:0706 012C
00000098968D 00000098AF34 00B4   00B4   012C 000D2493 03D2:0819 012D
00000098968E 00000098AF35 00B5   00B5   012D 000D30D7 02BF:092C 012E
00000098968F 00000098AF36 00B6   00B6   012E 000D3D1B 01AC:0A3F 012F
F3 A>
```

---

1 Alternatively, you can recalculate the translator based on P-List, delete the FoundForks.lba file and start the automatic procedure with LBA = 0.

2 Parameter 10 defines the number of LBA for translation in hexadecimal notation. In this case, 10 means that 0x10 = 16 sectors will be translated.

Here we can see that the sought LBA corresponds to track PhyCyl = 111B, head 4, sector Log Sec = A7, and the track contains LogSecs = 165 logical sectors. Now we need to analyze the information collected. If the 0xA7 LBA span (i.e. from LBA = 0x9895D9 to LBA = 0x989680) until the translator divergence point is filled with zeroes while the space preceding it by a single LBA (LBA = 0x9895D8) contains data, then it is likely that we have found a track hidden by the post-processing testing. To check that assumption, add the track beginning LBA to its length: 0x9895D9 + 0x165 = 0x98973E. Reading the obtained LBA via the utility[1] should reveal to us continuation of the user data[2].

As another example, let us assume that non-zero user data follow until LBA = 9999832, then an unreadable area or a short area filled with zeroes follows, then the translation divergence is observed. Now we can analyze it. LBA = 9999832 = 9895D8. The appropriate command should be sent to translate the LBA to a physical location:

```
F3 A>F98,95D8,,10

Track Info:
Partition PhyCyl   LogCyl   NomCyl   RadiusMils   LogHd Zn LogicalTrack FirstLba
User      000010E0 000010E0 0000107B +1.771687E+3 03    01 FFFFFFFFFFFF 000000989428

FirstPba     LogSecs PhysSecs WdgSkw SecPerFrm WdgPerFrm
00000098ACCF 01B1    01B1     0181   005F      0056

Sector Info:
LBA          PBA          LogSec PhySec Wdg  SFI
0000009895D8 00000098AE7F 01B0   01B0   0180 00145908

Track Info:
Partition PhyCyl   LogCyl   NomCyl   RadiusMils   LogHd Zn LogicalTrack FirstLba
User      0000111B 0000111B 00001068 +1.771750E+3 04    01 FFFFFFFFFFFF 0000009895D9

FirstPba     LogSecs PhysSecs WdgSkw SecPerFrm WdgPerFrm
00000098AE80 0165    0165     0067   0033      0038


    Sector Info:
    LBA          PBA          LogSec PhySec Wdg  SFI      Split     Burst
    0000009895D9 00000098AE80 0000   0000   0067 000480BF 0AF1:00FA 0068
    0000009895DA 00000098AE81 0001   0001   0068 00048D03 09DE:020D 0069
    0000009895DB 00000098AE82 0002   0002   0069 00049947 08CB:0320 006A
    0000009895DC 00000098AE83 0003   0003   006A 0004A58B 07B8:0433 006B
    0000009895DD 00000098AE84 0004   0004   006B 0004B1CF 06A5:0546 006C
    0000009895DE 00000098AE85 0005   0005   006C 0004BE13 0592:0659 006D
    0000009895DF 00000098AE86 0006   0006   006D 0004CA57 047F:076C 006E
    0000009895E0 00000098AE87 0007   0007   006E 0004D69B 036C:087F 006F
    0000009895E1 00000098AE88 0008   0008   006F 0004E2DF 0259:0992 0070
    0000009895E2 00000098AE89 0009   0009   0070 0004EF23 0146:0AA5 0071
    0000009895E3 00000098AE8A 000A   000A   0071 0004FB67 0033:1(0AF1):00C7 0072
    0000009895E4 00000098AE8B 000B   000B   0073 000507EB 0A11:01DA 0074
    0000009895E5 00000098AE8C 000C   000C   0074 0005142F 08FE:02ED 0075
    0000009895E6 00000098AE8D 000D   000D   0075 00052073 07EB:0400 0076
    0000009895E7 00000098AE8E 000E   000E   0076 00052CB7 06D8:0513 0077
    F3 A>
```

Here we can see that the only readable LBA is located on one track while the next 15 are on the following track. Such situation is one of the signs indicating a hidden track. Similarly to the previous example, we need to check for the presence of data one track further – at LBA = `0x9895D8` + 0x165 (LogSecs) = `0x98973D`.

Then we can use the Non Resident G-List editor to add the track record from PBA = (`FirstPba of track beginning`) 0x98AE80 with the length of (Log Secs) = 165, then clear G-List and recalculate the translator taking into account Non Resident G-List in order to apply the changes it contains. To simplify the work, you can use the 20 flag to mark all defects. In that case they will not be erased during translator recalculation based on P-List. Then we need to switch the HDD to ATA mode and send the Soft Reset signal or toggle the drive power supply.

Unfortunately, the method is not applicable in cases when the area of translator divergence contains no recorded non-zero user data.

[1] Please keep in mind that in command mode of the terminal a HDD cannot execute ATA commands and so you need to switch it using the [Ctrl] + [R] or [Ctrl] + [T] command.

[2] If more than one track is hidden, the check may require looking further. The search is based on track-sized stepping, the data should appear right in the track beginning.

Attention! All the LBA used above are designated in drive sectors; therefore, if a HDD sector contains 4096 bytes, i.e. 8 host sectors, the ATA LBA should be expressed as Native LBA divided by 8. Furthermore, if a drive's sector size is other than 512 bytes, LBA = 0 can be shifted by the so-called alignment factor (1 or 0) and by Media Cache size when it is placed in the beginning of the user data area.

5.5.4.1. "Delicate" work with Non Resident G-List

Translator divergence is often caused by the loss of information about post-processed defects during recalculation based on P-List. Information from Non Resident G-List in such cases may be lost incompletely. As a matter of fact, in the simplest case the only operation with Non Resident G-List performed during translator recalculation based on P-List is clearing of the records counter. The counter value itself in such cases can often be restored from preserved data. Please note that if Non Resident G-List viewing reveals records flagged differently from 0 (most frequently = 20), it means that the corresponding information about defects has overwritten during relocation to the list beginning the original records with the 0 flag initially located there. Moreover, if defects hiding via the translator had been performed, that operation also caused overwriting of the list beginning. The part that remained intact can be used then to speed up translator recalculation, especially if the original Non-Resident G-List contained track records. E.g., you can restore the original overwritten records up to the position where the lists match and then correct the records counter so that the resulting list includes both the revealed initial records and the final records remaining intact. For better understanding of the mechanisms involved in manipulations with Non Resident G-List, a description of its structure is appropriate. Please see the Non Resident G-List dump file below.

```
0x000000   aa aa 00 00 00 00 00 00 bb bb bb bb cc cc cc cc
0x000010   FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
0x000020   FF FF FF FF FF FF FF FF 00 00 00 00 00 00 00 00
0x000030   00 10 6E 00 19 00 69 00 C8 00 81 46 F2 7C FF FF
0x000040   33 31 33 46 FF FF FF FF 94 79 08 00 FF FF FF FF
0x000050   FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
...
0x200040   FF FF FF FF FF FF FF FF 00 FF FF FF FF FF FF FF
0x200050   FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
...
```

Two aaaa bytes at offset 0 make up the word of the records counter. The bbbbbbbb (4 bytes) field at offset 8 forms a double word of the offset for the list of defect locations (PBA). In this example, it is 0x00000048. The next cccccccc (double word, 4 bytes) field represents the offset of the table containing the flags of defect records. In this example, it is 0x00200048. A single PBA record (4 bytes) corresponds to one byte among the flags; therefore, you can check the size of PBA records area containing no fill pattern (0xFF) and divide it by 4 to obtain the number of records in Non Resident G-List. You can verify the assumption by calculating the size of the flags area containing no fill pattern. In the example above we can see that prior to clearing Non Resident G-List contained one record at LBA = 0x00087994 with the flag 0. Please keep in mind that Non Resident G-List records describe defects in Native PBA, i.e. they use logical coordinates based on a single physical sector acting as the measurement unit. You may also encounter FW of the drives with the approximate capacity of 2TB using a 64-bit PBA field (8 bytes). They can be recognized visually quite easily, and a check of the occupied portion size in the flags area will allow you to determine conclusively the value of the records counter field.

5.6. Diagnostics using the list of 32 recent RW commands

You can use the list of 32 last RW commands executed by a drive for diagnostics of its current status. To obtain the list, you need to enter the utility terminal and toggle the HDD power off and then on, wait for the malfunction to manifest itself, switch the drive terminal to Online or Diag mode using the [Ctrl]+[R] or [Ctrl]+[Z] commands respectively. Then send the [Ctrl]+[X] command, after which the HDD will output a report similar to the following:

```
ASCII Online mode

ATA  15 Cmds
Ts(ms)     dT(ms)   Op Cnt  LBA
         0 99999999 00 0000 000000000000
         0        0 00 0000 000000000000
...
         0        0 00 0000 000000000000

RW  32 Cmds
Ts(ms)     dT(ms)   xT(ms)  Type Option Mode  St EC       Info
      1098 99999999        1 04  005041 000000 1 00000080 SRV_MEM Type 8 Addr 04001998 Cnt 0004
      1099        1        0 04  005041 000000 1 00000080 SRV_MEM Type 6 Addr 0000001D Cnt 0001
      1099        0        0 04  005041 000000 1 00000080 SRV_MEM Type 8 Addr 040019A0 Cnt 0004
```

```
 1099       0         0 04  005041 000000 1 00000080 SRV_MEM Type 6 Addr 000000E7 Cnt 0001
 1099       0         1 04  005041 000000 1 00000080 SRV_MEM Type 8 Addr 040019A8 Cnt 0004
 1100       1         0 04  005041 000000 1 00000080 SRV_MEM Type 6 Addr 0618003B Cnt 0001
 1100       0         0 04  005041 000000 1 00000080 SRV_MEM Type 6 Addr 00000025 Cnt 0001
 1100       0         0 04  005041 000000 1 00000080 SRV_MEM Type 8 Addr 04001930 Cnt 0004
 1100       0         1 04  005041 000000 1 00000080 SRV_MEM Type 6 Addr 00000007 Cnt 0001
 1101       1         0 04  005041 000000 1 00000080 SRV_MEM Type 8 Addr 04001940 Cnt 0004
 1101       0         0 04  005041 000000 1 00000080 SRV_MEM Type 6 Addr 000000A6 Cnt 0001
 1101       0         0 04  005041 000000 1 00000080 SRV_MEM Type 8 Addr 04001958 Cnt 0004
 1101       0         1 04  005041 000000 1 00000080 SRV_MEM Type 6 Addr 000000A7 Cnt 0001
 1102       1         0 04  005041 000000 1 00000080 SRV_MEM Type 8 Addr 04001968 Cnt 0004
 1112      10         8 12  005041 000000 1 00000080 TEMP 28 degC
 1121       9         8 02  005041 000000 1 00000080 XFR WR SYS LBA B 0000000075AA L 00000001 …
 1129       8         3 02  002241 000000 1 00000080 XFR RD SYS LBA B 000000008E55 L 00000001 …
 1143      14         8 02  002241 000000 1 00000080 XFR RD SYS LBA B 000000005702 L 00000001 …
 1151       8         8 02  002241 000000 1 00000080 XFR RD SYS LBA B 000000005702 L 00000001 …
 1170      19         6 02  002241 000000 1 00000080 XFR RD SYS LBA B 000000005717 L 00000002 …
 1176       6         8 02  005041 000000 1 00000080 XFR WR SYS LBA B 000000005717 L 00000002 …
 1195      19         6 02  002241 000000 1 00000080 XFR RD SYS LBA B 000000005702 L 00000001 …
 1201       6         8 02  005041 000000 1 00000080 XFR WR SYS LBA B 000000005702 L 00000001 …
 1220      19         6 02  002241 000000 1 00000080 XFR RD SYS LBA B 000000005702 L 00000001 …
 1226       6         8 02  005041 000000 1 00000080 XFR WR SYS LBA B 000000005702 L 00000001 …
 1245      19         8 02  002241 000000 1 00000080 XFR RD SYS LBA B 0000000061B7 L 00000008 …
 1253       8         5 02  005041 000000 1 00000080 XFR WR SYS LBA B 0000000063A8 L 00000002 …
 1258       5         3 02  005041 000000 1 00000080 XFR WR SYS LBA B 0000000061B7 L 00000008 …
 1262       4         8 02  005041 000000 1 00000080 XFR WR SYS LBA B 00000002BBC7 L 00000008 …
 1281      19         3 02  002241 000000 1 00000080 XFR RD SYS LBA B 0000000060A4 L 00000006 …
 1284       3         8 02  005041 000000 1 00000080 XFR WR SYS LBA B 0000000060A4 L 0000000A …
60113   58829         9 34  002221 000000 1 00000080 TCC 0
```

In the output, we need the last section: RW 32 Cmds. Please note the EC column during its analysis. It is used to output the command completion code: 80 indicates success, other codes – errors. Info column shows the operation type: XFR – data transfer, RD – reading, WR – writing, SYS means work with the service area, USR – with the user data area. 'LBA' is followed by the address in LBA space of the service or user data areas respectively. To identify the damaged system files (sys file), you need to find the "**XFR RD SYS**" operations, which completed with errors. Once you have LBA coordinates of the operation completed with an error (LBA field), you have to find the file associated with it. That can be accomplished using any of the following two methods.

1) Using the terminal command T>**y**[1]. After you send the command, the drive will output to terminal a report similar to the following.

```
F3 T>y

File  Vol  FD        Location      Size      Cylinder  Hd  Sector
----  ---  --------  ------------  --------  --------  --  ------
0000  000  fc000042  00000003750a  00000001  00039963  00  0001fa
0001  000  fc001043  00000009a04c  00000001  00037479  01  0001fa
0002  000  fc002044  0000000fcb8e  00000001  0003284b  02  0001fa
0003  000  fc003045  00000015f6d0  00000001  000374ad  03  0001fa
0004  000  fc004046  0000001c2212  00000001  0003829c  04  0001fa
0005  000  fc005047  000000224d54  00000001  00035fab  05  0001fa
0006  000  fc006048  000000287896  00000001  00039b13  00  0001fa
…

File  Vol  FD        Location      Size      Cylinder  Hd  Sector
----  ---  --------  ------------  --------  --------  --  ------
0000  001  13100201  000040328040  000003c0  ffffffff  ff  00ffff
…
0039  001  fd139064  000040606600  00004000  ffffffff  ff  00ffff
003a  002  3c23a000  000001423280  00004800  ffffffff  ff  00ffff
…
004d  002  fc24d084  000000180000  04000000  ffffffff  ff  00ffff


File  Vol  FD        Location      Size      Cylinder  Hd  Sector
----  ---  --------  ------------  --------  --------  --  ------
0000  003  fd30021b  0000000408b5  001d97f0  00039984  00  000335
0001  003  fd30121c  0000003db094  000c5684  ffffffff  ff  00ffff
0002  003  fc30221d  0000000c5684  0003737c  00032784  02  000000
0003  003  fc30321e  0000001281c6  0003737c  000373e6  03  000000
0004  003  fc30421f  00000018ad08  0003737c  000381d5  04  000000
```

[1] ***Attention!*** The command may be unsupported in the FW of your drive (an Invalid Diag Cmd message will be output to terminal). In that case you should use the second method.

```
0005  003  fc305220  0000001ed84a  0003737c  00035ee4  05  000000
0006  003  fc306221  00000025038c  0003737c  00039a4c  00  000000
…

File  Vol  FD        Location      Size      Cylinder  Hd  Sector
----  ---  --------  ------------  --------  --------  --  ------
0000  004  fc400100  000040180000  00008000  ffffffff  ff  00ffff
0001  004  fc401101  000040188000  00008000  ffffffff  ff  00ffff
0002  004  fc402102  000040190000  00008000  ffffffff  ff  00ffff
0003  004  fc403103  000040198000  00008000  ffffffff  ff  00ffff
…
0046  005  80546208  0000011e2000  00000200  ffffffff  ff  00ffff
0047  005  3854701b  0000011e2000  003ccc00  ffffffff  ff  00ffff
0048  005  4254801c  0000011e2000  0004c800  ffffffff  ff  00ffff
0049  005  4554901d  0000011e2000  00038400  ffffffff  ff  00ffff
004a  005  4954a01e  00000121a400  00014400  ffffffff  ff  00ffff
…

File  Vol  FD        Location      Size      Cylinder  Hd  Sector
----  ---  --------  ------------  --------  --------  --  ------
0000  009  02900208  000040030000  00000200  ffffffff  ff  00ffff
0001  009  07901001  000040030200  00009ff0  ffffffff  ff  00ffff
0002  009  09902002  000040030200  00000020  ffffffff  ff  00ffff
0003  009  fc903004  0000400303c8  000001f0  ffffffff  ff  00ffff
0004  009  13904005  0000400303c8  00000010  ffffffff  ff  00ffff
…
0012  009  fc912096  000000000000  00000100  ffffffff  ff  00ffff
0013  00a  fca1303f  00000007b000  00001000  ffffffff  ff  00ffff
0014  00a  fca1432a  000000000000  00080000  ffffffff  ff  00ffff
```

The Location column contains information about the file location, Size – about its size. For our purposes, we need the lists of vol 0 and vol 3 files. Once we identify the necessary file by its position and size, we can find information about its file descriptor (FD column) and the partition where it is located (Vol). Three last characters of the descriptor represent the file identifier (FID).

2) To identify a corrupted sys file, you may also use the utility features available in the dialog for viewing of the service information objects[1].

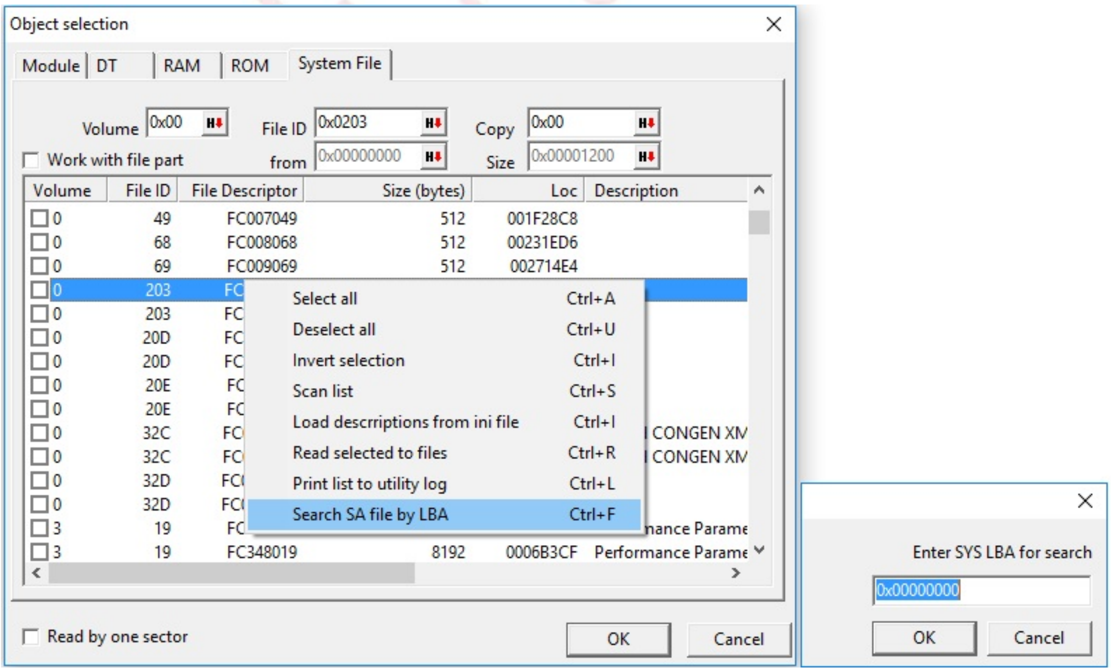


Fig. 5.10

The 'Search SA file by LBA' command from the context menu allows to find the files that the selected LBA belongs to.

1 Menu 'Tools – Utility extensions – View and edit HDD resources'.

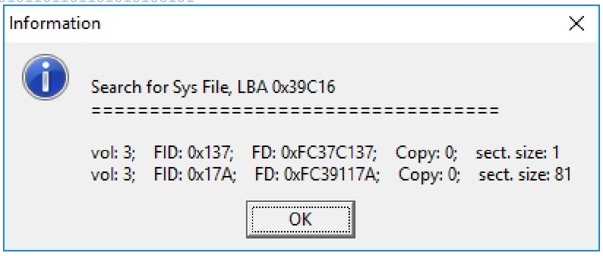


Fig. 5.11

If multiple files are displayed for a Sys LBA, it means that some of them (the largest ones) are the so-called ***Superfiles*** – container files with ***SubFiles*** inside.

Once the list of corrupted files is created, you have to try restoring them using the utility tools and the information about the purpose / uniqueness / structure and methods for work with various service information files. If a file is not unique, it may be initialized with HDD commands, overwritten with a copy borrowed from a donor resource or sometimes even recorded as a file with the corresponding size filled with zeroes. If a unique file is damaged, you should take the steps necessary for its restoration using sector-by-sector reading from copies, information from files with similar purpose (which may contain the data from corrupted blocks in the same or a different format). Please keep in mind that sometimes damage affects unique files, which are not required during the drive loading procedure. For example, the P-List file may get damaged, but in that case HDD initialization and complete access to user data are still possible provided that the translator is intact. However, its further corruption makes translator recalculation impossible.

6. Tests menu

The menu contains the operations necessary for work with an HDD, arranged into functional groups. Let us examine some of them.

6.1. Utility status

This menu item opens the dialog that displays the status of the utility and allows you to re-read the HDD ID information, DT lists from RAM and ROM, the list of modules, restart autotuning, and modify some runtime settings of the utility (*Fig. 6.1*). Clicking the "refresh" buttons makes the utility read again the service information of the drive. This is necessary in cases when non-standard methods are used to start a HDD (with controller isolation or channel short connection).

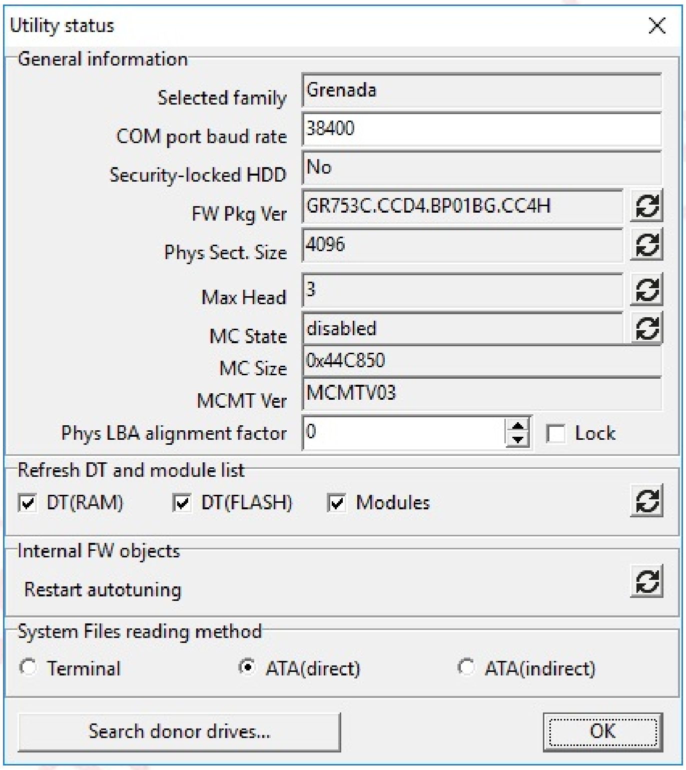


Fig. 6.1

Media Cache information reflects MC State (whether Media Cache is enabled / disabled / unsupported), MC Size (size of the area allocated for Media Cache), and MCMT Ver (Media Cache support module version).

The 'Phys LBA alignment factor' field is intended for drives based on physical sectors with the user data capacity larger than 512 bytes (e.g., 4096), its purpose is to associate USER LBA = 0 with the physical native LBA [1]. In most cases the utility identifies the field value correctly. However, if the recognized value is wrong, you can specify it manually and Lock it using the corresponding checkbox.

The 'Restart autotuning' button initiates reading of drive memory ranges via ATA and their subsequent analysis, which the utility then uses to obtain the information about Sys File Volume = 0, 3, and some other system information objects of the HDD being examined.

The 'System Files reading method' radio button determines how the utility will read the HDD system files. When the utility starts, it detects the appropriate method automatically. In this dialog utility selection can be modified as necessary. 'Terminal' option means that the utility was unable to access Sys Files via ATA and therefore further

[1] E.g., for a while standard practice implied such data alignment, where USER LBA = 63 would match the beginning of a physical sector of 4096 bytes. In that case the alignment factor = 1. USER LBA = 8 at that was aligned with the beginning of the 8th physical sector.

attempts to work with these items should be performed in the terminal. 'ATA(direct)' option means that the drive features a native command for access to Sys Files; therefore, the utility will use it for further work. 'ATA(indirect)' option means that the firmware has no internal factory commands for operations with Sys Files. In that case the utility will attempt to access the service information using the commands for operations with modules as a "bypass".

6.2. Loader

The submenu contains two commands: Create loader and Update Microcode.

6.3. Create loader

This menu item allows you to build from the service information of a connected HDD a data file in Update Microcode command format; the file will consist of the Disk FW Overlay (ROM overlay) and firmware overlays. That *.lod file can be used then to restore corrupted overlays in a compatible drive or to load some of them to HDD memory. At present Disk FW Overlay recording is only possible to disk surface while the firmware overlays can be loaded to HDD RAM without their recording to the SA. The fact allows in some cases to start a drive with impaired writing functionality. The generated file does not contain complete or partial ROM data; therefore, the loader created by the utility cannot be employed for automatic firmware updates.

6.4. Update Microcode

This menu item can be used to start the files loading / updating the drive firmware with the standard Update Microcode (92h) ATA command. Files in that format include both native Seagate firmware update files and the overlay loading files generated by the utility documented here.

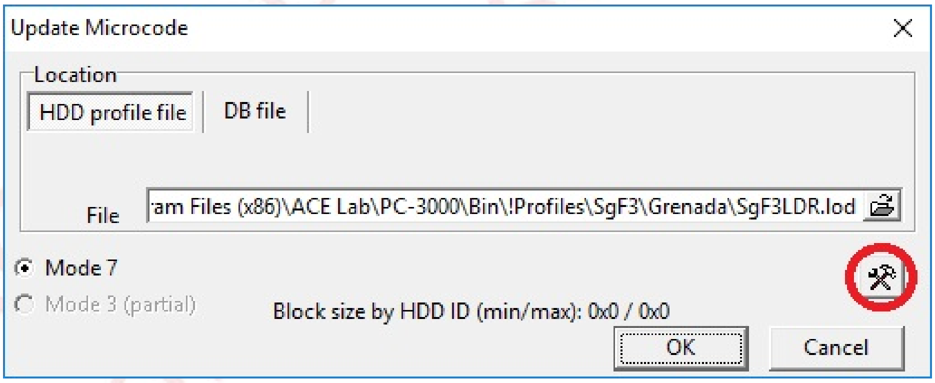


Fig. 6.2

The dialog allows to choose the source file in a profile or database, and the mode that will be used to upload it - mode 7[1] (loading in a single block) or mode 3 (multiple blocks). The latter mode has been introduced in the ATA standard to support large update files, which a drive cannot apply completely in one cycle. Support of that mode is indicated in the HDD ID, which the utility checks to enable or disable the block loading mode. In addition, the dialog allows to open the content editor for Seagate Arch F3 *.lod files (the button in red circle). Below you can see a sample editing dialog containing a file for a 7200.12 Pharaoh drive update to CC49 version.

[1] Please refer to the ATA standard for the Update Microcode (92h) command specification.

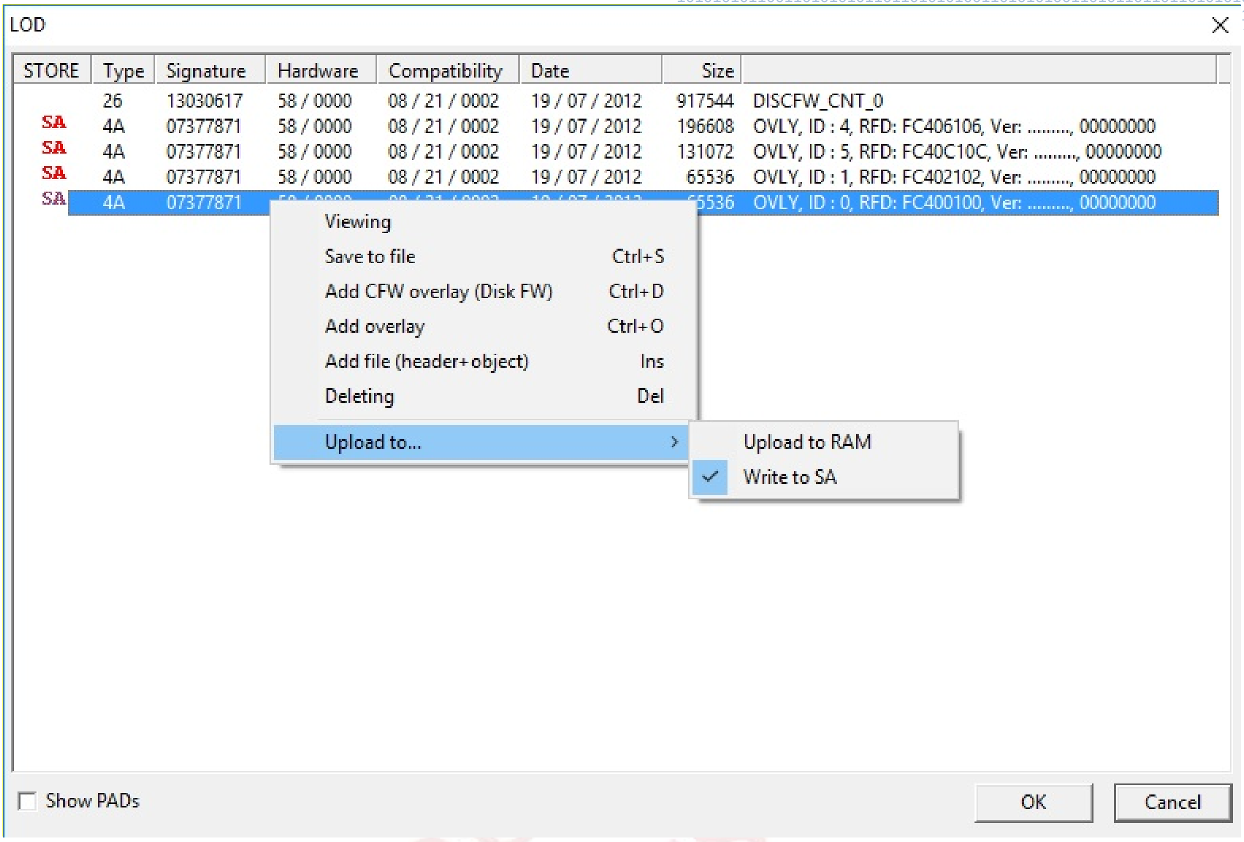


Fig. 6.3

The editor allows to remove the objects unnecessary for the current task from the loader, to add Disk FW overlay and firmware overlays from files (in particular, the feature allows creating a loader from HDD resource files saved earlier), to modify the destination of overlays selecting their recording to the SA or only uploading them to RAM. In addition, you can save and view individually the objects contained in a loader. During loader modification the editor adds respective headers, alignment blocks (pads), and corrects the checksums for the loader data. However, when its components are modified individually, their **internal** checksums and configuration headers should be generated using specialized tools outside the editor (for example, the hex editor plug-ins).

6.5. Work with terminal

This submenu contains the set of commands used to control the terminal connection with a drive (*Fig. 6.4*).

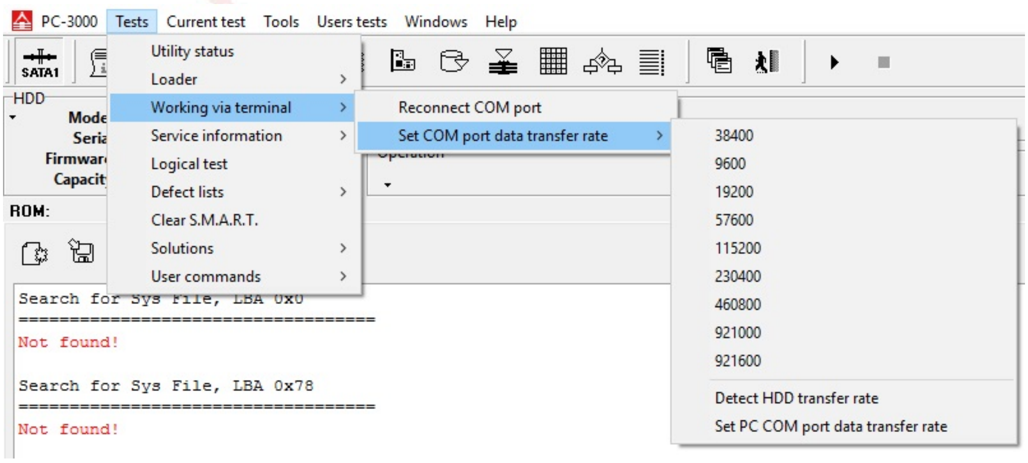


Fig. 6.4

The command to 'Reconnect COM port' allows you to reconnect to the COM port selected in the settings when USB Terminal has to be re-initialized (disconnected from USB and connected again) after its "freeze". The functionality for switching of the COM port baud rate allows changing the data exchange rate of interaction with the drive terminal.

6.6. Service information

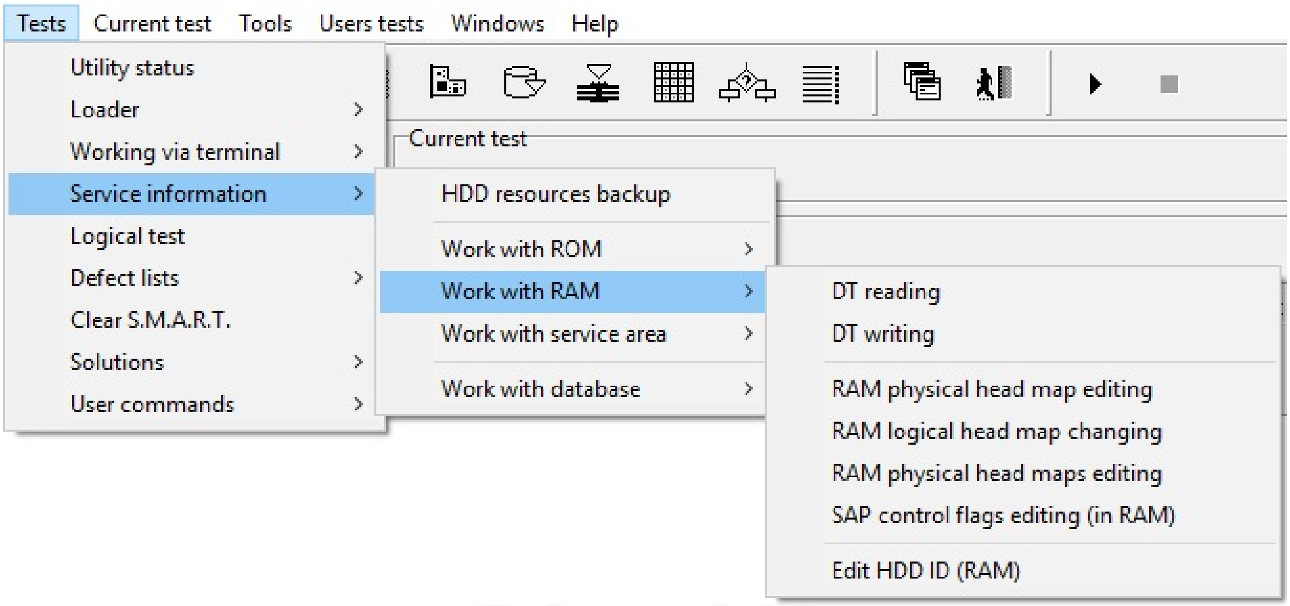


Fig. 6.5

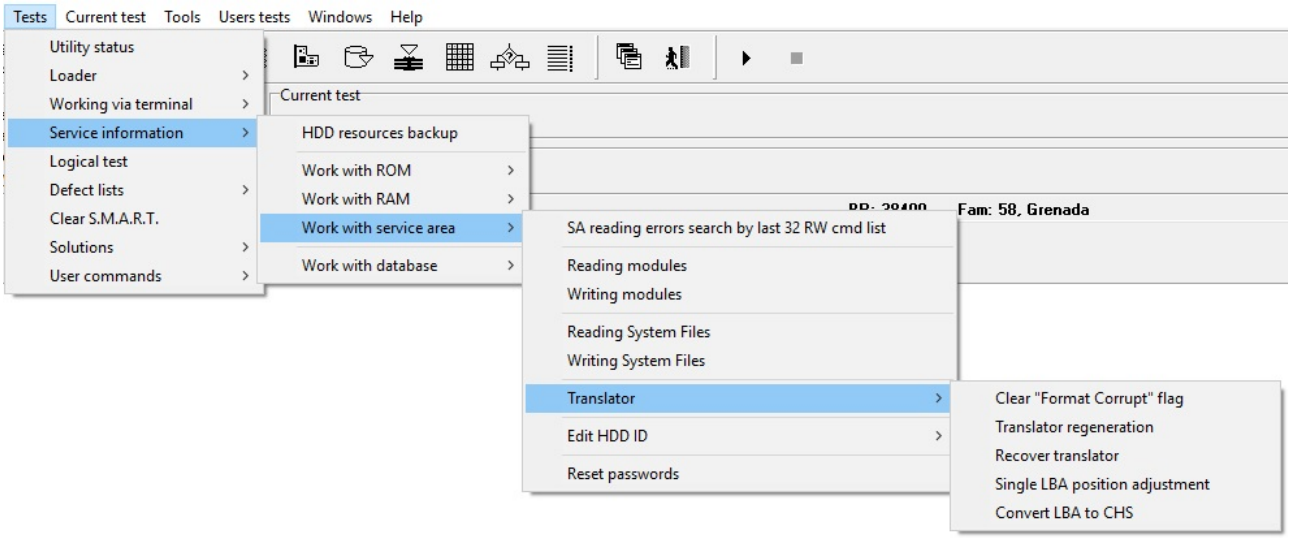


Fig. 6.6

The menu contains features intended for operations with the service information of a drive in the controller board and on disk surface in the HDA.

6.6.1. HDD resources backup

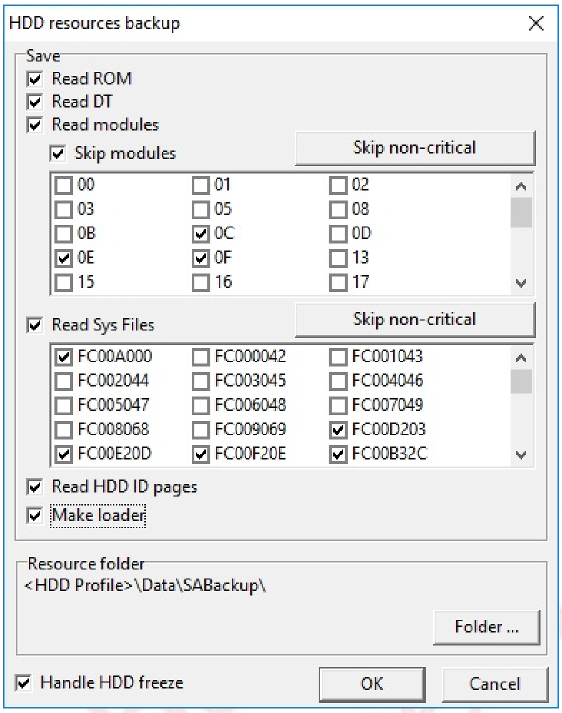


Fig. 6.7

This operation is available in the 'Tests → Service information' menu. Selecting it allows you to save a backup copy of the drive's service information to a corresponding folder in its profile (by default, SABackup).

Attention! To bypass the modules and Sys Files files, which are not essential for a HDD and may cause a freeze[1] at an attempt to read them, the dialog offers the corresponding buttons to 'skip non-critical' items. In certain cases, when the HDD condition is critical, it makes sense to restrict the volume of data to back up even further by saving just the ROM content, P-List (module 0x03), translator (module 0x2B) and the Non Resident G-List file (its file descriptor may vary in different FW, in recent versions it is fixed as vol 3, FID = 0x35). Drive overlays are not unique, they can be borrowed from another HDD. The same is also true for configuration modules: Saved Mode Pages (module 0x2A), and Registry (module 0x13).

6.6.2. Work with ROM

This menu contains the commands for ROM reading and writing to a file in drive profile or database. Reading can be performed via ATA (for an initialized drive that reports on readiness), or via Boot Code (for HDD which cannot reach the ready status). ROM writing is performed in the Boot Code mode.

If you select an operation that must be performed via Boot Code, the utility will prompt the steps necessary to initialize the HDD.

[1] The main FW portion is designed for work with drives based on HDA containing up to eight heads. However, there are also “slim” HDD based on HDA containing fewer heads. Therefore, the manufacturer uses for such purposes FW unable to read certain modules pertaining to the logs and testing modules for heads 2 – 7 (or 4 - 7). An attempt to read them from the surface results in a HDD switching to a LED error status and freezing.

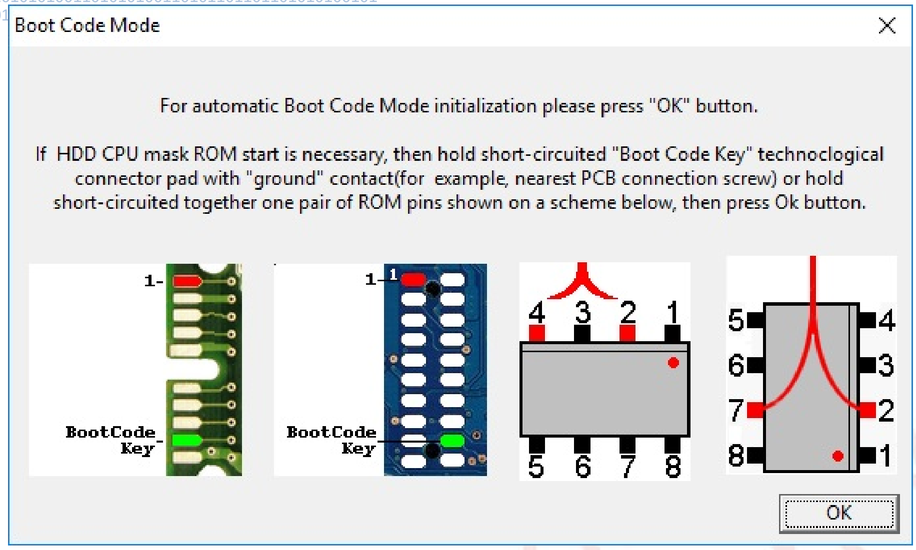


Fig. 6.8

Click OK for automatic switching. If the utility is unable to switch the HDD to Boot Code mode automatically, you need to either short-circuit the pad of the drive controller indicated in the figure with the HDD "ground" contact, or bridge the ROM contacts indicated above (*see Fig. 6.8*). The need to use a forced switch may be indicated by an error message output to log and containing the "Bogus ISR" text.

```
Write ROM...
Flash size is: 0x00080000
Flash type is: STM
Erasing...
Bogus ISR
CPSR = 0x600000D3
R14 = 0x00100A83
```

Attention! Reading in Boot Code mode is performed through the terminal! The operation requires selecting a supported baud rate. If during Boot Code initialization the utility outputs a message about terminal switching error, try selecting a lower baud rate.

6.6.3. Work with RAM

The menu contains features for operations with the HDD service information:

- DT (Drive Tables) reading
- DT (Drive Tables) writing
- RAM head map editing
- RAM logical head map changing
- RAM physical head maps editing
- SAP (Servo Adaptives Parameters) control flags editing (in RAM)
- HDD ID editing (in RAM)

As the name implies, DT reading and writing allow to read and write Drive Tables[1]. The operations for modification of the heads map in RAM allow editing the corresponding heads maps in drive RAM[2]. The 'RAM physical head maps editing' feature allows you to modify the active map in RAM on-the-fly (provided the HDD FW supports that) and combine heads into pairs, if necessary. The 'RAM logical head map changing' feature allows you to modify the map of

[1] DT (Drive Tables) used to be referred to earlier as CP – Config Pages.

[2] The drives in question contain several maps of heads. For details on their structure and operation please see section 6.6.3.1, Heads maps in Seagate F3 drives.

correspondence between the logical and physical heads (provided the HDD FW supports that) and combine heads into pairs, if necessary.

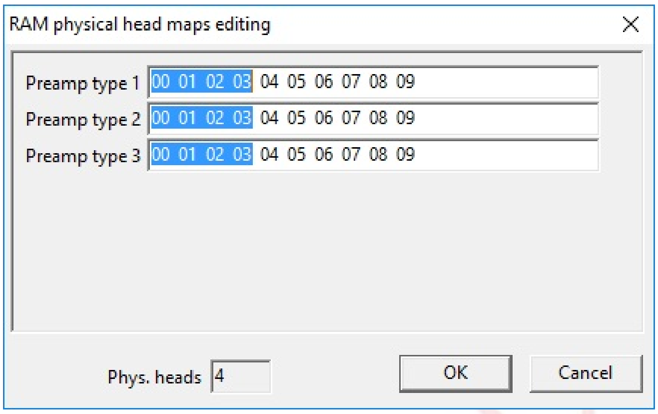


Fig. 6.9

If FW of a HDD does not support the translation map associating logical heads with physical ones and returns the pointer to the general maps array for all commutator types instead of a pointer to the current table for translation of physical heads into commutator codes, then the 'RAM physical head maps editing' feature should be used. It opens the dialog for editing of all available maps for translation of physical heads into commutator codes in RAM. To pair up heads, you have to perform identical operations with each table (line) (e.g., to pair, copy the cell i to cell j).

The feature for modification of the SAP control flags in RAM allows editing such SAP control variables as the Seek Mode, maximum logical head number, RRO (Repeatable Runout) mode.

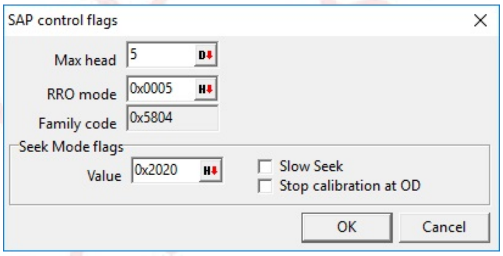


Fig. 6.10

Enabling the Slow Seek and Stop calibration at OD flags activates slower positioning. When enabled, these flags facilitate more reliable functioning of drives with mechanical damage / replaced heads / replaced stack of disks due to the activation of specialized configuration tables.

Having restricted the maximum head number in ROM (e.g., to 0), you can start a HDD with fewer heads and use the RAM heads map editor together with modification of the maximum head number to find out, which heads are damaged.

The RRO mode variable allows control over the Repeatable Runout mode correcting the position of the magnetic heads stack. Research is currently in progress to determine the correspondence between the mode numbers and specific correction procedures.

Editing of HDD ID in drive RAM allows, among other things, to disable Autoreassign in HDD memory if recording to surface is impossible or if there is a risk of service information corruption. The list of options available in the dialog is identical to that in the HDD ID editing dialog for further saving of the changes to the service area.

6.6.3.1. Heads maps in Seagate F3 drives

Firmware of Seagate F3 drives uses several levels of head maps. Support or lack of such support for each level are determined by the flags used during compilation of HDD FW.

The lowest part of the hierarchy contains the tables used for conversion of the physical head number into the selection code for the preamplifier commutator channel. There is an individual table for each commutator type, selection of the active translation table occurs during HDD initialization when the power is switched on. Some firmware versions support requesting the active map address, others return the starting address of the translation maps block. If a HDD returns specifically the active map address, it may be corrected individually in RAM. If the drive returns the address of maps array, correction requires editing of the entire array (because there is no way to tell, which of them is being used). On that level the heads are numbered sequentially from 0 to the last head number. Nothing can be skipped in the middle.

The map for translation of logical heads into physical ones occupies a higher level in the hierarchy. It makes it possible to create HDD models with heads disabled in the middle of the stack provided their firmware supports this table. The map should be taken into account while selecting a donor heads stack. If the map is supported, you can view it on a functional HDD or in ROM image. According to the table, the "logical" head numbers are converted into "physical" ones as indexes in the table for conversion into the commutator channel selection code.

E.g.:

Logical head, table index	0	**1**	**2**	3
Physical head	0	**1**	**4**	5

→

Physical head, table index	0	**1**	2	3	**4**	5
Preamplifier code	a	**b**	c	d	**e**	f

In the provided example heads 2 and 3 are disabled (skipped). Let us assume that logical head 2 is damaged. Consequently, to obtain the physical head number you have to use the number in cell **2** from the logical heads translation table. The number is **4**. It means that we have to deal with the physical head 4. The number in cell under index 4 from the table for conversion into the preamplifier codes produces code **e**. It means that to exclude the damaged head from the translation process you have to replace in the map for conversion of logical heads into physical ones the number **4** with the number corresponding to a live head (e.g., **1** = log. head 1), or replace in the preamplifier code selection table the code **e** with a code corresponding to a live head (e.g., **b** = phys. head 1 = log. head 1). Thus, the resulting head maps will look as follows:

Logical head, table index	0	**1**	**2**	3
Physical head	0	**1**	**1**	5

→

Physical head, table index	0	**1**	2	3	**4**	5
Preamplifier code	a	**b**	c	d	**b**	f

Attention! While working with the maps of drive heads, please keep in mind that firmware is located in areas accessible with the heads 0 and 1. Therefore, you cannot exclude both of them because it will prevent normal functioning of the HDD.

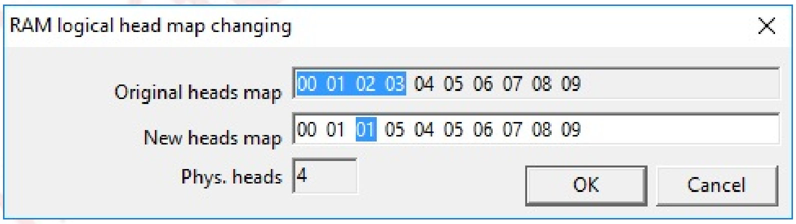


Fig. 6.11

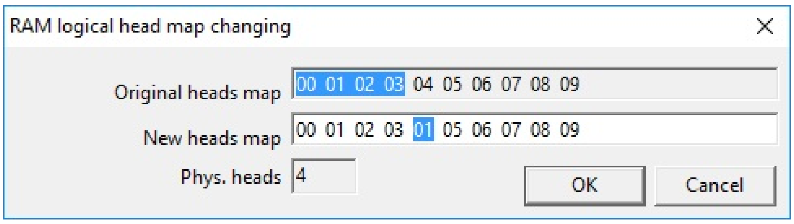


Fig. 6.12

6.6.4. Work with service area

The menu contains features for operations with the HDD service information:

- reading modules
- writing modules
- reading System Files
- writing System Files
- operations with the translator – recalculation, restoration, position correction for an individual LBA
- HDD ID editing
- password resetting

Menu items for reading and writing of modules / System Files allow you to read / record modules / System Files respectively using the structure of HDD profiles and the product database. The menu for operations with translator allows recalculating the translator via ATA based on P-List or taking certain steps for its restoration. For details on the translator restoration modes please refer to *section 5.5, Host system detects a HDD, drive beginning is visible but starting with a certain LBA continuous space until the end is filled with UNC errors.*

HDD ID editing allows you to modify some parameters affecting the drive operation[1]. In particular, the dialog allows the operator to disable Offline scan known earlier for its connection with the Pending Bug manifestation. In addition, disabling of the options pertaining to Autoreassign and Deferred defect hiding allows you to prevent a HDD from switching to ABRT condition in case of its unstable operation. The condition is manifested as follows: HDD reads until a certain location is reached and then responds to any command with ABRT until power is switched, after which continues standard functioning for a while and then switches to the ABRT state again. The option to 'Switch HDD power supply OFF/ON before saving' allows the operator to bypass HDD freeze in cases of Pending Bug right after the drive's power supply is turned on. In that case a few commands sent immediately after the power is switched on may "pass", then the HDD freezes. Additional power switching allows the utility to overwrite the drive configuration module before the freeze.

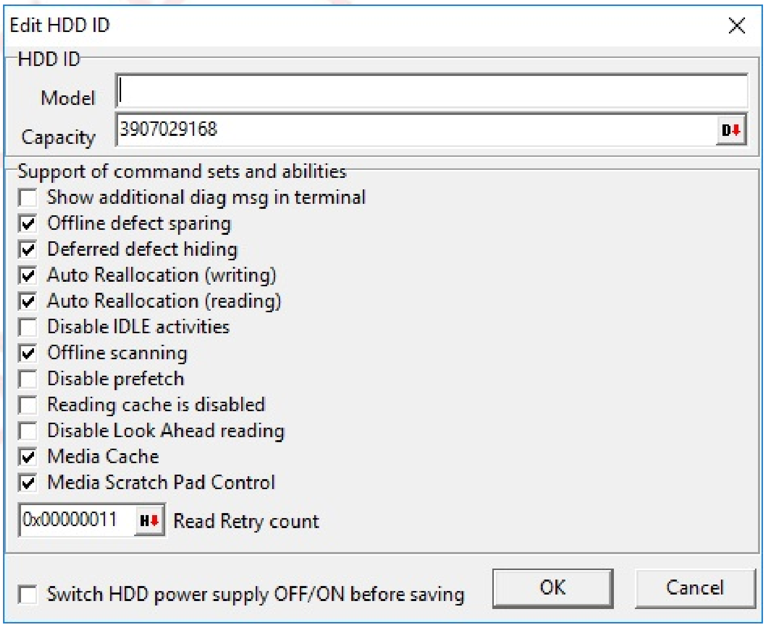


Fig. 6.13

Password resetting allows you to unlock a password-protected HDD. Due to the progress of the solutions for individual data protection, HDD manufacturers tend to be using more sophisticated schemes of drive initialization preventing

[1] Resulting changes will be saved in the service area and the new settings will become effective after the HDD power supply is toggled, provided that the corresponding configuration modules have not been corrupted in the process of recording.

unauthorized access to user data. In particular, Seagate F3 HDD with enabled password protection block factory mode commands making password resetting more complicated [1].

While developing the utility we have designed and implemented three methods for accessing the service information: short connection of the read-write channel, isolation of the spindle motor connector, access to the security subsystem via the terminal. Unlocking through the terminal is the simplest method that requires no additional manipulations and is therefore suggested as the recommended approach (other methods will require bridging the read channel when necessary or disconnecting the HDD controller board). When it is invoked, the utility reads a certain amount of service information from HDD, analyzes it and then tries to use the obtained data to unlock the drive.

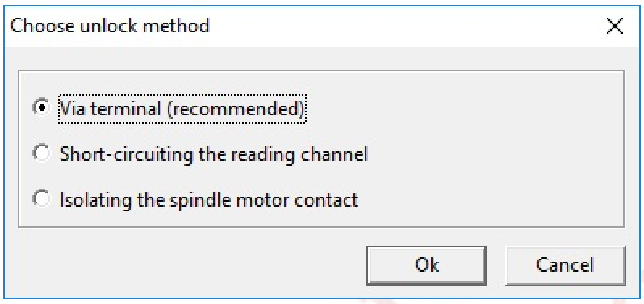


Fig. 6.14

```
Test  : Reset passwords
HDD power supply switching OFF/ON...
Found passwords:
Master:
  Text: 112233
  Hex : 31 31 32 32 33 33 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
User  :
  Text: 112233
  Hex : 31 31 32 32 33 33 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Checking...
HDD power supply switching OFF/ON...
Unlocking...
Unlocked!
Collecting Sys Files information...
```

Fig. 6.15

In all cases the utility opens a HDD and resets the password with standard ATA commands ("open drive", "reset password") using the found password data automatically. If the utility fails to find the password information block, the users are advised to save the registry module to a file for subsequent individual analysis to be performed manually.

6.7. Logical test

A drive gradually wears out while operating. As a result, BAD blocks may appear on its surface and an attempt to read them may cause the drive to return various errors. Unfortunately, such powerful tool of drive recovery as Self Scan is currently unavailable for Seagate F3 Architecture HDD. Therefore, for relocation of defective LBA the utility offers just the logical test feature with subsequent hiding of revealed defects and slower access areas.

[1] The description does not cover the drives belonging to the FDE (Full Data Encryption) series, which represent a topic worthy of a separate study.

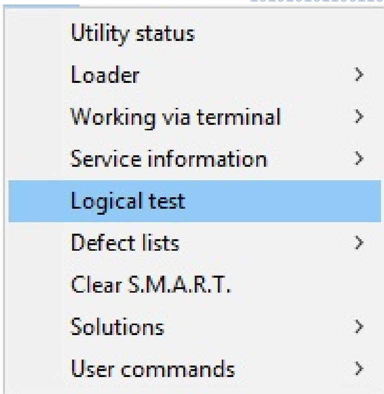


Fig. 6.16

Selection of the corresponding menu item starts the complex test of the universal utility; test results will be used to generate the list of found defects.

Attention! You are advised to disable Autoreassign and Deferred defect hiding before test to prevent the drive's self-testing subsystem from affecting the scan results.

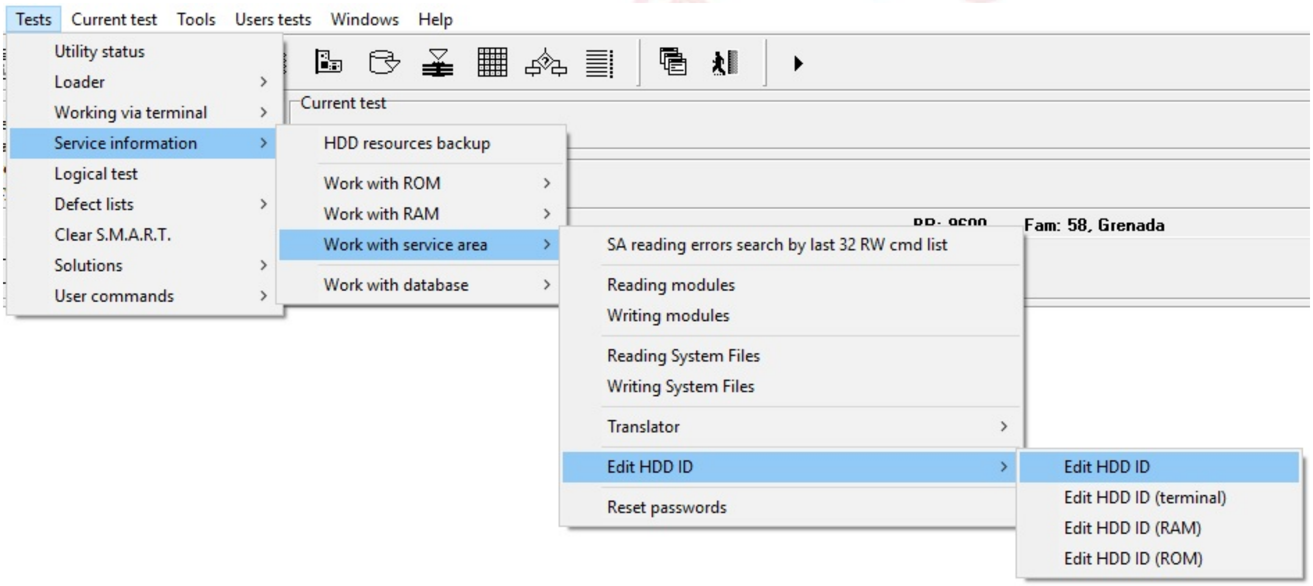


Fig. 6.17

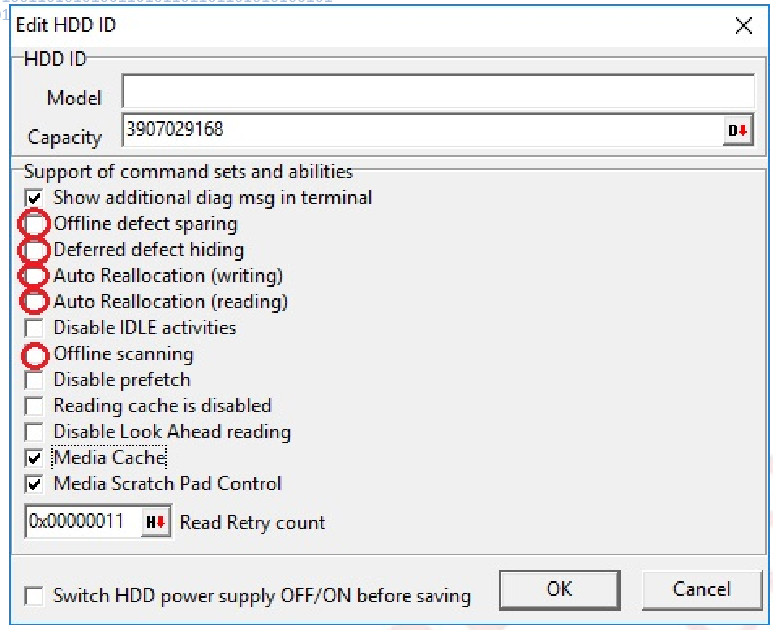


Fig. 6.18

After scanning completion the utility will output to a separate 'Defects' tab the list of found defects and slower access areas. Then use the shortcut menu of the list (displayed after right-clicking it) to select hiding of the revealed defects to the HDD translator. The P-List module will not be modified at that. Defects will be hidden specifically in the translator (as it happens during the last stage of factory testing). To hide the found defects, select in the menu the 'Hide to Slip-List' (Ctrl+1) command.

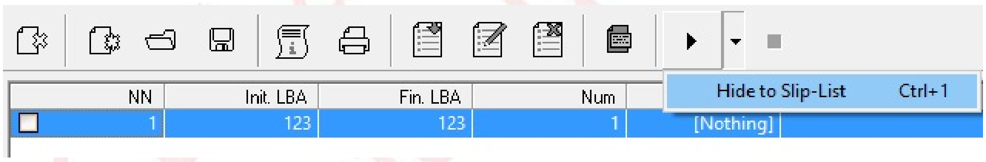


Fig. 6.19

6.8. Defect lists

The menu allows you to edit (and, in particular, clear) such defect lists as Non Resident G-List, P-List, and Non Resident Primary DST List. If you select an editing feature, the utility will decode the corresponding defect list of the drive and open the obtained list in the internal editor; then you can modify the list and save it to a file or the drive.

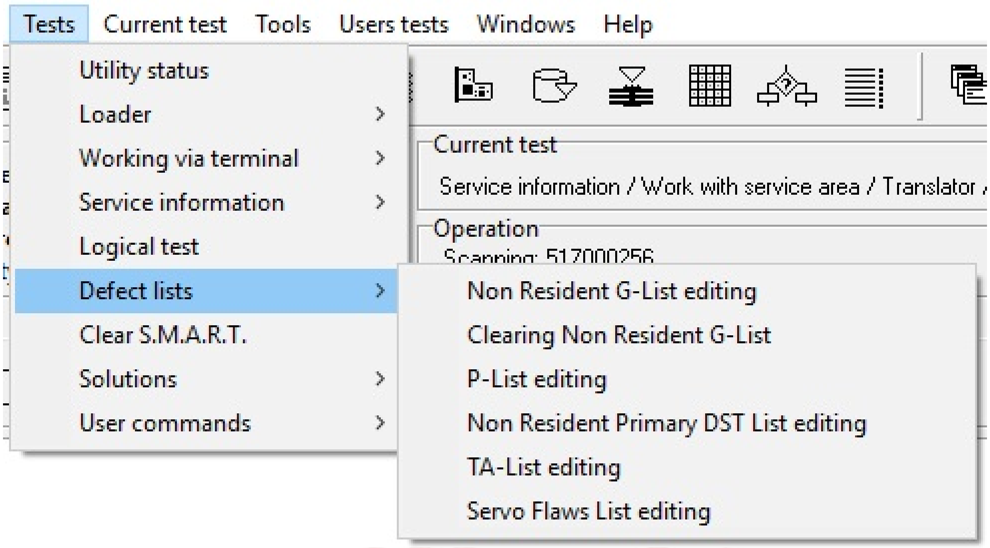


Fig. 6.20

6.9. S.M.A.R.T. erase

This menu item invokes resetting of S.M.A.R.T tables in a drive to the "factory defaults" state.

6.10. Solutions

The menu contains a group of features for automatic repair of typical HDD malfunctions. These features are discussed in detail in *sections 5.3, HDD returns HDD ID (the host system detects it), but reports capacity = 0,* and *5.4, Host system does not detect a HDD, which permanently remains BSY.*

6.11. User commands

The menu contains a list of user-defined commands for work with a HDD via terminal (*Fig. 6.21*). The list can be modified from the settings dialog of the utility (for details pertaining to the modification of the user commands list see the first part of documentation).

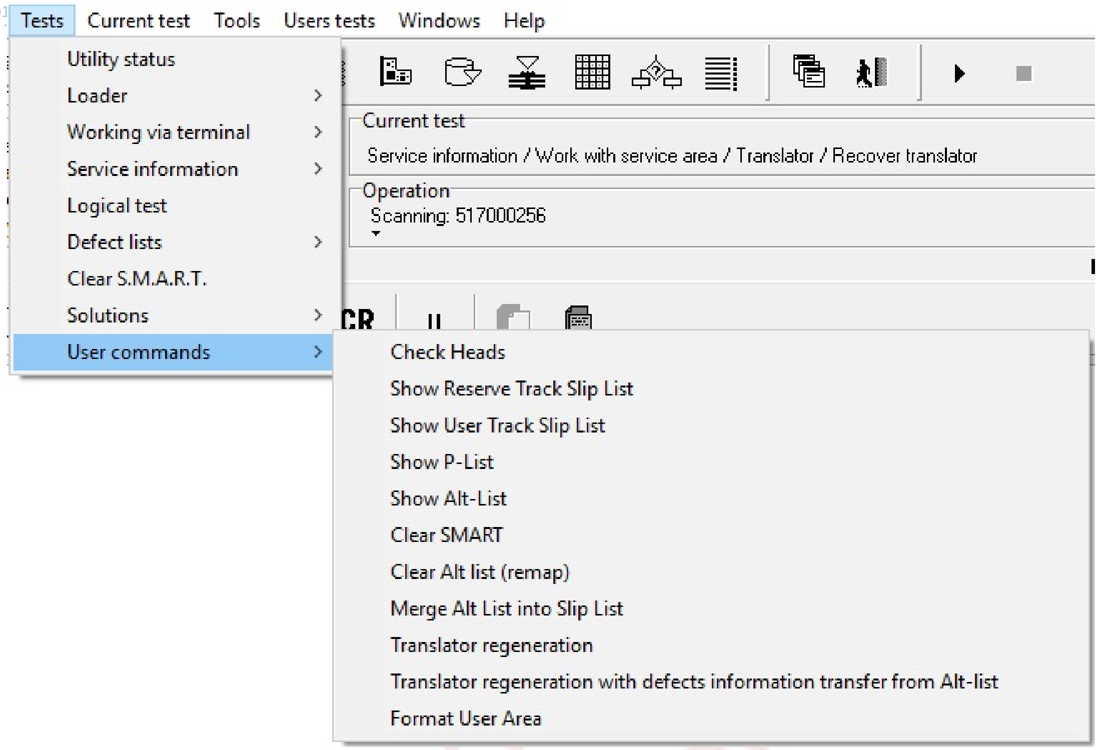

Tests
Current test
Tools
Users tests
Windows
Help
Utility status
Loader
Working via terminal
Service information
Logical test
Defect lists
Clear S.M.A.R.T.
Solutions
User commands
Current test
Service information / Work with service area / Translator / Recover translator
Operation
Scanning: 517000256
Check Heads
Show Reserve Track Slip List
Show User Track Slip List
Show P-List
Show Alt-List
Clear SMART
Clear Alt list (remap)
Merge Alt List into Slip List
Translator regeneration
Translator regeneration with defects information transfer from Alt-list
Format User Area

Fig. 6.21

7. Specialized tools of the utility

The tools described further are accessible from the 'Tools → Utility extensions' menu.

7.1. View and edit HDD resources

The tool allows you to work with various service information items of a hard disk drive (*Fig. 7.1*).

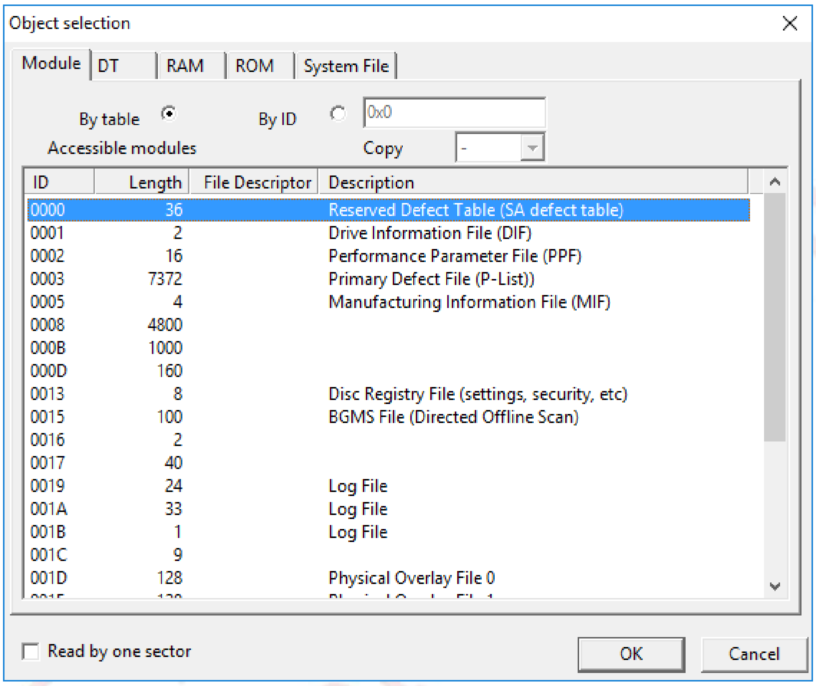


Fig. 7.1

The Module tab can be used to read/write the service information modules of a HDD. Modules can be accessed 'By table' - from the list of scanned (during start or from the utility status dialog) modules or directly by their identifiers – 'By ID'. In the latter case you can specify the copy to access ("1st" or "2nd"). Sector-by-sector reading method is supported for recovery of corrupted modules.

Once a module is selected, the utility reads it to the standard hex editor for viewing and editing (manually or using plug-ins). The procedure for work with resources in the internal hex editor of the utilities is universal, it is described in the main part of product documentation.

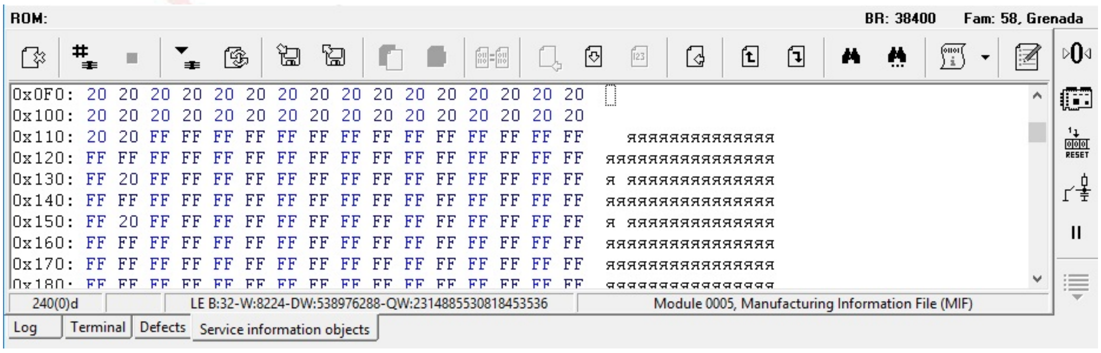


Fig. 7.2

The DT tab can be used to read/write Drive Tables in HDD RAM and ROM.

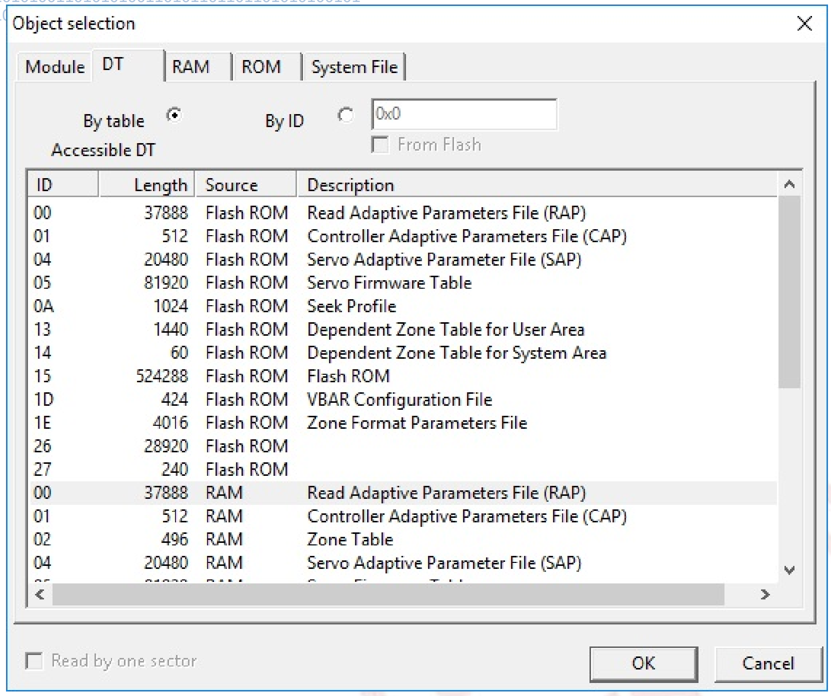


Fig. 7.3

DT can be accessed 'By table' - from the list of scanned (during start or from the utility status dialog) DT or directly by their identifiers – 'By ID'. In the latter case you have to select additionally the DT source: ROM or RAM.

The RAM tab provides access to the objects in the random-access memory of the controller.

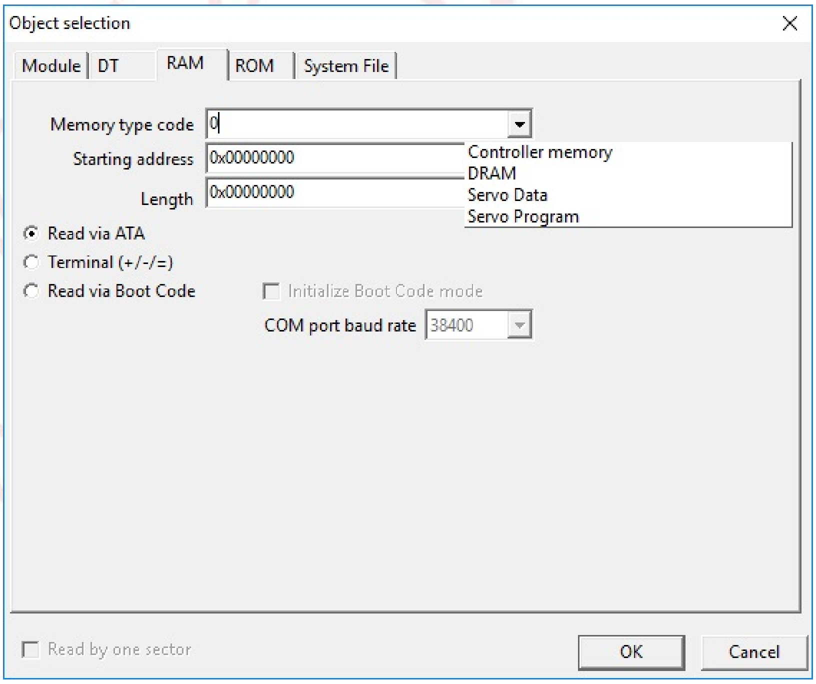


Fig. 7.4

The following parts of address space are distinguished at that:

- Controller memory (general memory)
- DRAM (controller memory array)
- Servo Data (servo program address space)
- Servo Program (servo program address space)

The utility supports reading both via ATA (if the HDD is initialized and able to perform factory mode commands) and in the Boot Code mode. In the latter case you have to specify the terminal baud rate.

The ROM tab allows reading the Flash ROM of a drive to the hex editor. The utility supports reading both in ATA or in Boot Code mode (terminal). In the latter case you have to specify the terminal baud rate. Writing to ROM is performed in Boot Code mode.

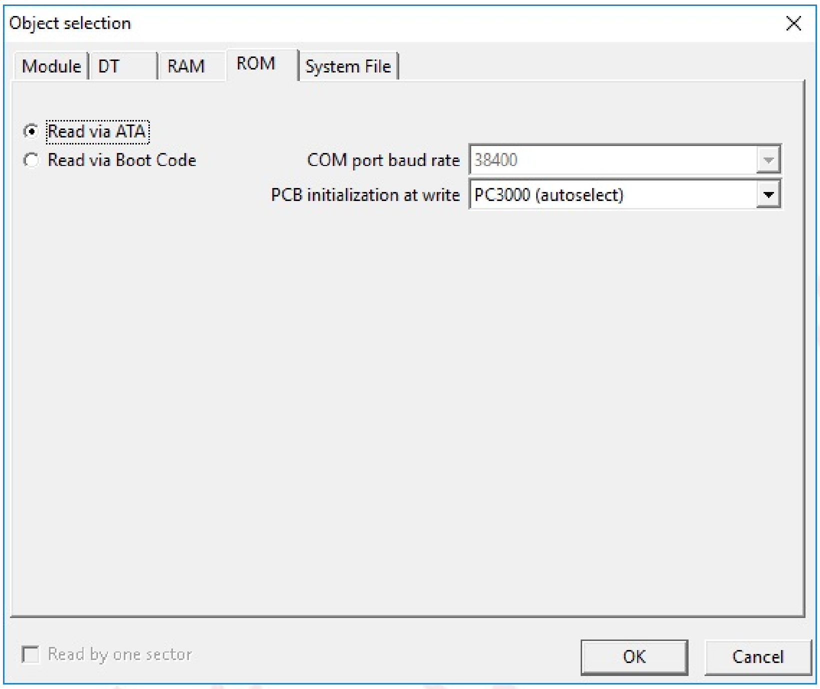


Fig. 7.5

The System File tab can be used to access the service information files of a drive.

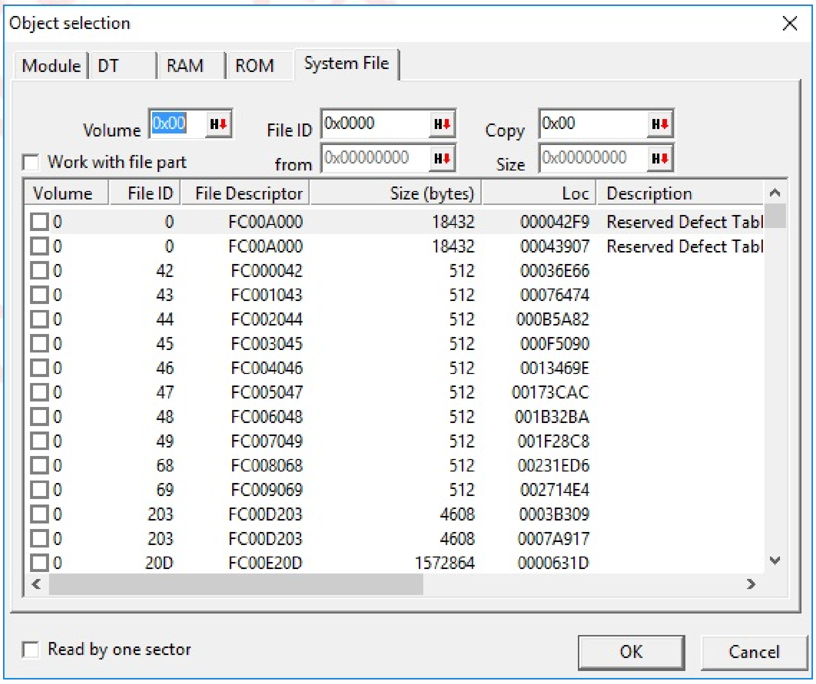


Fig. 7.6

The file system supports several Volumes containing service information files selectable by their File ID and copy number. At present the purpose of the following volumes is known:

- Volume = 0 – service information on disk surface (Disc Root)
- Volume = 3 – service information on disk surface (Sys Disc)
- Volume = 4 and 5 – RAM volumes
- Volume = 9, Volume = 0x0A – service information in controller ROM

Sector-by-sector reading method is supported for corrupted files. The reading mode (Terminal / ATA(direct) / ATA(indirect)) is specified in the utility status dialog. It supports access to the System Files both via the terminal and via ATA, in the latter case the utility allows work with files even if the drive FW contains no internal commands for operations with these objects.

You can right-click the list of files to initiate scanning for available FID:

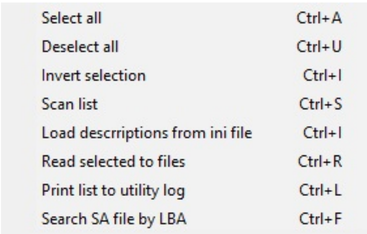


Fig. 7.7

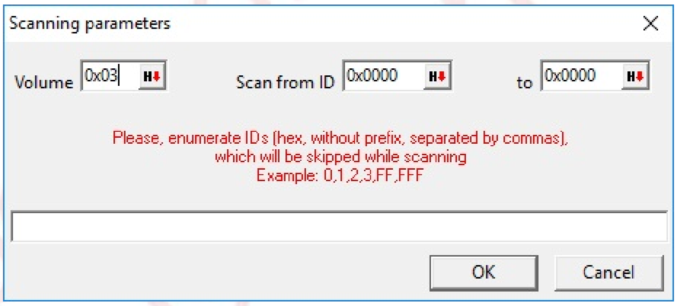


Fig. 7.8

When you select the command to scan the list, the utility displays a dialog suggesting to define the volume and File ID range for scanning as well as the list of File ID to skip (when information about certain FW files is requested, a LED: xxxx exception may occur resulting in firmware freeze unwanted during automatic scanning). The list of files to skip will be saved to the requested ini file.

Besides, modern firmware versions (in particular, FW of Barracuda LP Green drives) are able to output to terminal the list of available files after the "T>y" command. Sample list returned in response to the command:

```
F3 T>y
File  Vol  FD        Location      Size      Cylinder  Hd  Sector
----  ---  --------  ------------  --------  --------  --  ------
0000  000  fd000000  0000000246df  0000000c  0002a4ab  00  00001f
0001  000  fc001203  00000002900f  00000009  0002a4bd  00  00026f
0002  000  fc002204  000000029020  00000003  0002a4bd  00  000280
…
```

Here FD stands for File Descriptor. FD format is xxYxxZZZ, where Y = Volume, ZZZ = File ID.

You can also print the list of Sys Files in the dialog to the utility log and use the Sys LBA to identify the corresponding file it belongs to. The opportunity allows quick recognition of damaged files based on the report about recent RW commands executed by the drive. For details on the diagnostics method please see the *section 5.6, Diagnostics using the list of 32 recent RW commands.*

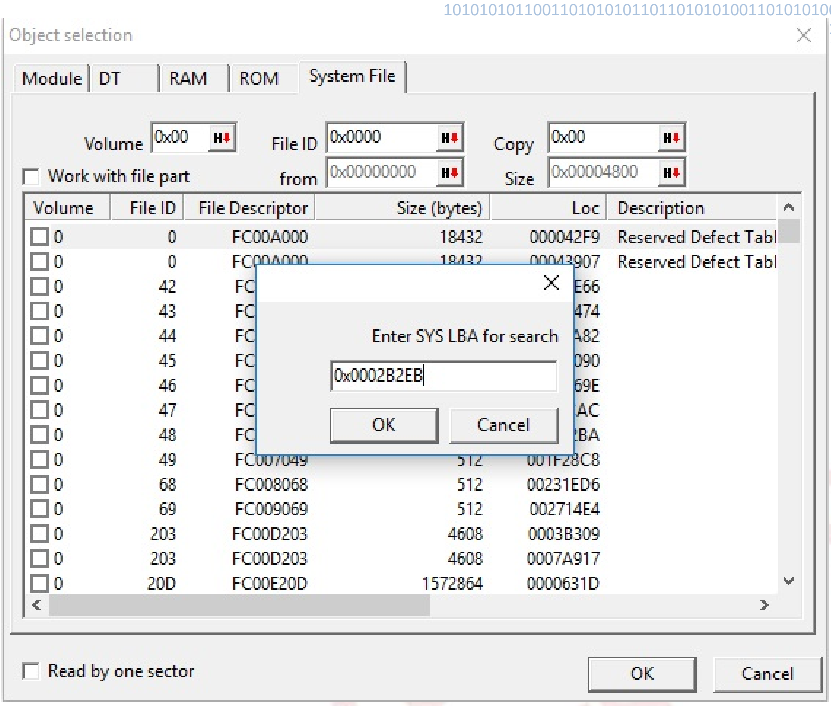


Fig. 7.9

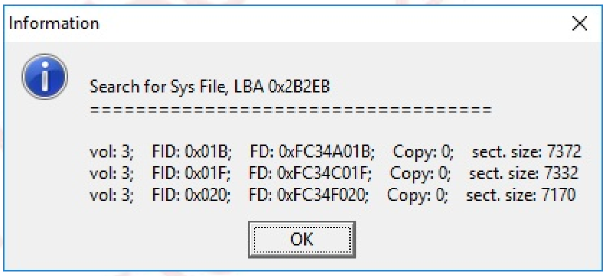


Fig. 7.10

7.2. Work with Flash ROM image file

If a drive enters the ready state normally, its ROM can be read directly via ATA. If a HDD does not reach readiness but allows access to the terminal command mode and supports work with System Files, the data can be read from Volume = 0x0A, File ID = 0x032A. Otherwise ROM reading should be performed via terminal[1] in Boot Code mode or in a programmer device (having unsoldered the ROM chip from the controller first). ROM writing is performed in the Boot Code mode in the utility. You can also write a corrected image in a programmer device.

ROM itself is a complex object based on block structure. It contains servo code that controls positioning and reading, Custom FW code working with the general ATA commands, and a few blocks of adaptive data. Among the blocks of adaptive data please note CAP (Controller Adaptive Parameters), RAP (Read Adaptive Parameters), SAP (Servo Adaptive Parameters), and IAP (Interface Adaptive Parameters). CAP, RAP, SAP are equivalent to the corresponding DT and data blocks in correspondingly named service information blocks (minor differences are possible – the CAP module may contain no model name, modules of servo adaptive data may contain more detailed settings). The IAP data block is a container for flags, such as SpinUp at Power ON (the option is supported by some drives), used during HDD operation. The container may also be empty if no flag switching took place.

Work with damaged drives often suggests tasks related to the correction of ROM information. These include:

- correction of the heads map for work with a damaged magnetic heads stack or scratched surface (pairing the heads up);

[1] You have to specify the terminal baud rate.

- correction of the available heads counter during the damage diagnostics stage;
- correction of model name / serial number, etc. in case of repair;
- transfer of adaptive data from a damaged ROM to a live donor chip;
- transfer of adaptive data from a ROM with vendor-recognized software error preventing the HDD from reaching readiness (or accessing it through the terminal, etc.), to a donor **updated ROM** from the firmware library;
- transfer of adaptive information associated with drive heads during replacement of the heads assembly with MHA borrowed from a donor drive.

These tasks can be performed using the tool 'Work with Flash ROM image file', available in the 'Tools → Utility extensions' menu.

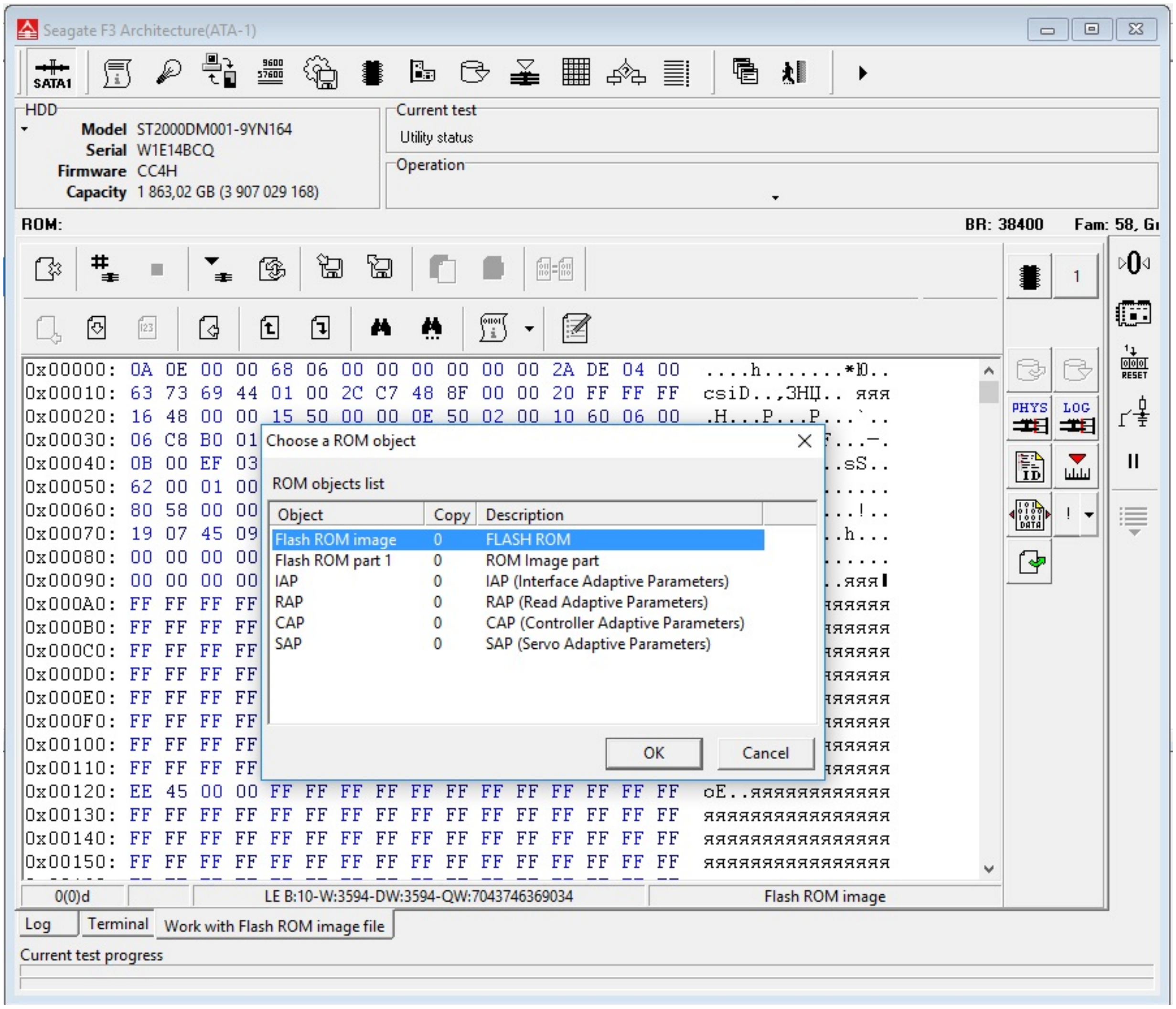


Fig. 7.11

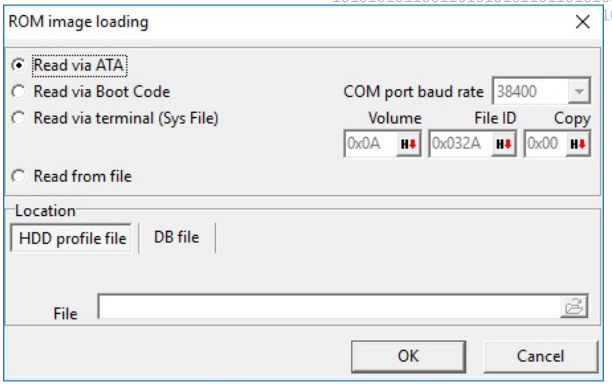


Fig. 7.12

When started, the wizard loads ROM image from a file or drive (you may also perform the procedure using the button when necessary (see the loading dialog above), and allows you to correct the information in ROM manually or with the integrated tools. Tools available in the wizard are:

- Load data of the object being edited from drive memory and write it back via the interface for operations with DT. Available for CAP, RAP, and SAP.

PHYS LOG - Correct the head maps in ROM. The tool consequently supports the maps for conversion of physical head numbers to the preamplifier channel selection codes (editing of all available maps) and the map for conversion of logical head numbers to physical ones. For details please refer to the *section 6.6.3.1, Heads maps in Seagate F3 drives*. Editing of the maps for conversion of physical head numbers to the preamplifier channel selection codes is performed in the dialog shown in *Fig. 7.13*.

Editing of the map for conversion of logical head numbers to physical ones occurs in the dialog shown in *Fig. 7.14*.

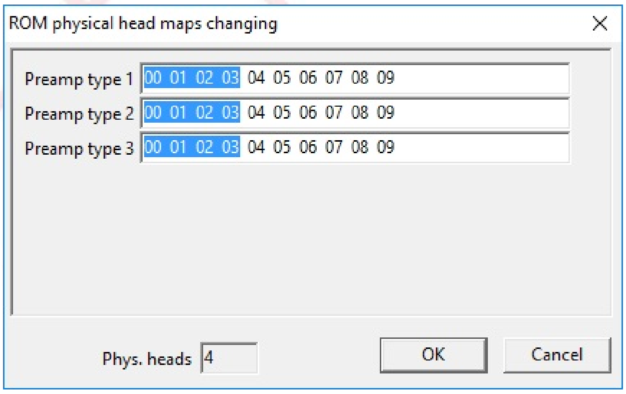


Fig. 7.13

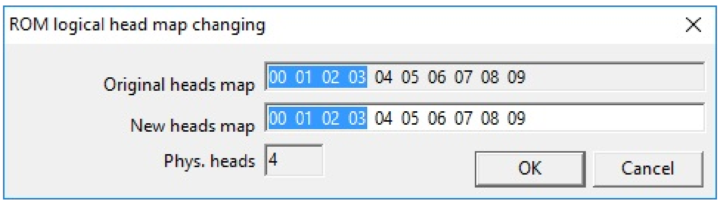


Fig. 7.14

- Correction of HDD ID data in ROM. The button displays a dialog that allows you to modify the serial number of the HDA, controller, WWN, manufacture date, model name, and capacity (if the field is stored in ROM).

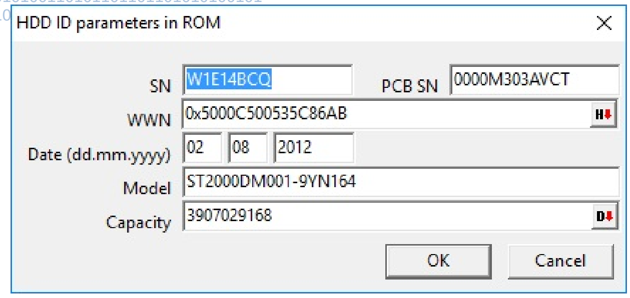


Fig. 7.15

- Modification of the SAP control flags in ROM image. The feature is identical to the corresponding feature for work with RAM. For details please refer to the section *6.6.3, Work with RAM.*

- Import adaptive data – when this feature is invoked, the utility requests the ROM image file which should be used as the import source and then transfers from it the blocks of adaptive parameters (CAP, RAP, SAP) or (at your option) adaptive information about the heads to the ROM being edited.

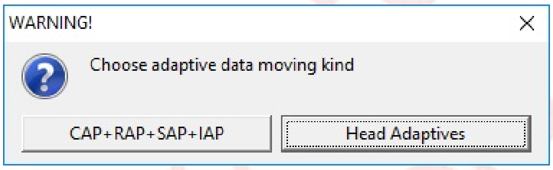
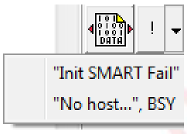


Fig. 7.16

After manipulations with a ROM image you have to either write it to a drive (via the terminal in Boot Code mode) or save it to a file, then write to the ROM chip in a programmer device and solder the chip back to the controller board of the "patient" HDD. ROM image can be saved to a file or drive whenever you click the button, which applies the changes. The selected item in ROM image can also be saved to file when you click the button to save the hex editor content to file.



– The button opens the list of ROM code modifications available in this utility version. At present the following modifications are available:

- "Init SMART Fail". Some FW versions require a SMART reset after translator recalculation (e.g., the Pharaoh drive family). If module 35 is unreadable at that, SMART clearing will complete with an error and HDD will switch to cyclic output of the LED: 000000CC[1] error. If module 35 cannot be overwritten, the "Init SMART Fail" fix should be used to disable the SMART reset feature in drive firmware.
- "No HOST…", BSY. During start a drive attempts to validate critical files of its firmware and if they are corrupted, it outputs a "No HOST PhysReadyStatusFlags…" message to terminal and stops initialization of the ATA mode. Access to terminal commands may be possible in such cases. The fix disables firmware damage control, as a result you may often manage to access the factory mode commands and perform drive analysis via ATA. The list of available features includes the Update Microcode (92h) command, which can be used to load CFW / Diag / Tech Overlays. At present we are working on implementing a mechanism for loader-based approach, which will allow to upload the above overlays and obtain access to the diagnostic factory mode via ATA even if they are corrupted on the SA surface.

– Utility initialization based on a loaded ROM image or CFW overlay. The utility uses the provided data to find and load the modules table, service information volumes 0 and 3, and also some other information that it needs.

[1] It should not be confused with the LED 0000000CC error caused in 7200.11 drives by the translator or S.M.A.R.T. corruption.

8. Unlocking HDD in case of LED: CC error

In cases when a HDD fails to reach readiness, it outputs to the terminal a cycling message like "LED: 000000CC" and the drive cannot be switched to the terminal command mode, it is blocked both for ATA and for terminal commands. This condition is typical of some firmware versions of Seagate F3 drives[1, 2]. To solve the problem you need to achieve just partial initialization of the drive (to avoid reaching the step when blocking occurs), diag overlay must be loaded and the terminal command mode must be active. Initialization may be interrupted either by disconnecting the controller board from HDA or by short connection of the read channel at a certain moment.

Disconnection of the controller board on-the-fly while the spindle is rotating is not allowed because it will result in heads stuck in the user area and may damage the preamplifier chip (in the HDA) or the read channel chip (on the controller board). Therefore, the only way is to disconnect the controller board in advance, before the device is powered on[3].

We know of three FW groups behaving differently during PCB initialization without a HDA:

- 7200.11 for FW versions CD04 through CD15, SD1A, AD14, ES.2, FW SN01 through SN04 – when powered on, the controller reaches readiness, terminal and factory mode commands are available.
- 7200.11, newer FW versions and ES.2 beginning with version SN05 – during initialization with a disconnected motor the controller reaches readiness but the terminal and factory mode commands remain unavailable.
- 7200.12 and newer (some firmware versions) does not reach readiness, in terminal it outputs the message No HOST Fis-ReadyStatusFlags[4]…

Thus, not all the drives allow obtaining access to the command mode after initialization with a controller board disconnected from HDA. Therefore if PCB disconnection has not helped unlock the HDD, you should use the method for short connection of the read channel. The method is somewhat more complicated because it imposes certain restrictions on the tool used in the procedure[5], but it works on all Seagate Arch F3 HDD[6].

Attention! The scheme for HDD unlocking implemented in our utility earlier, methods published in the Internet, and the scheme implemented in the Salvation Data suite **do not include backup of service data and force translator recalculation, which may result in incorrect translator generation!** The possibility results from the fact that widely known command T>m0,6,2,,,,,22 and its alternative variations use just P-List for translator recalculation. However, most drives in recent years pass the so-called post-processing before they leave the factory, i.e. an additional logical scanning with relocation of defects using a shifting scheme. The information about defects then is added directly to the translator (but not to P-List) and gets lost during its recalculation forced by the command above (process logs are also typically erased as a part of the pre-sale preparation procedure). As a result, the drive surface becomes unreadable beginning with an arbitrary LBA (the smallest LBA number relocated during post-processing) and the drive returns the UNC error.

The unlocking algorithm implemented currently in the utility backs up the service information and restores the translation module in its original state thus preventing the potentially unsafe translator recalculation operation.

Further we shall examine closely two methods available to make a HDD reach readiness and obtain access in terminal command mode for standard unlocking. The first suggested method of spindle motor connector isolation is simpler, but unfortunately possible in some cases only.

---

[1] Classic and F3 Barracuda drives can be easily distinguished by the FW version appearance. While FW of classic drives contains a dot (e.g., **3.06**, **3.AAD**), FW version of F3 drives contains no dots (e.g., **SD15**, **0002BSM1**).

[2] LED: CC lock described in this section develops because of CE Log file overflow in a drive. However, the solution method used in the utility can be employed to solve the problem of locked drives caused by other reasons as well.

[3] It is possible to isolate just the spindle motor connector without removal of the entire controller board.

[4] In this drive family LED: CC locking does not occur after SMART overflow; however, it is possible in case of translation problems. The way to make the devices reach readiness is being researched.

[5] The aspect will be discussed in detail in the corresponding section.

[6] The possibilities of reaching readiness on 7200.12 drives using short connection of the read channel are being investigated.

8.1. Disconnection of the controller board from HDA (isolation of the spindle motor connector)

The procedure for drive initialization according to this method is as follows:

- While the power is off, remove the screw fixing the controller board near the motor connector, and insulate the connector (e.g., using a sheet of dense paper or a plastic card).
- After a while the drive will reach readiness (as shown by the status register indicators on the utility panel).
- While the power is still on, remove the insulating layer and return the screw fixing the controller board near the motor connector.
- Then follow the basic unlocking algorithm.

To start the mode in the utility:

- Select the 'Solution for problem "HDD Lock (LED: 000000CC)"' menu item.
- Confirm that the mode should be started.
- Specify the backup folder for service data storage.
- Select 'Isolating spindle motor contact' as the unlocking method.
- Follow the guidelines displayed in the main window in that mode.

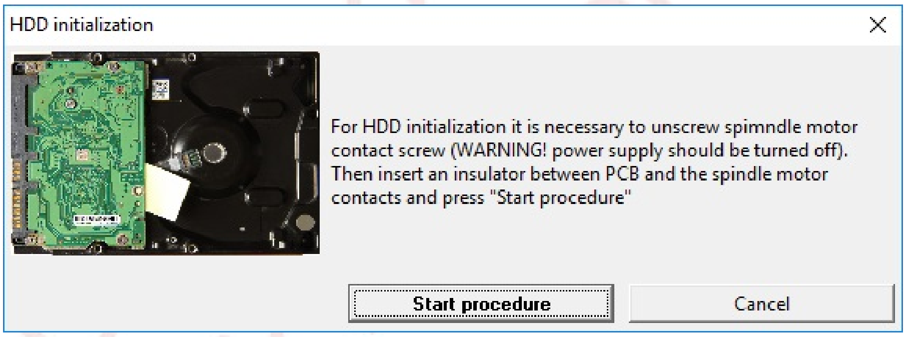


Fig. 8.1

Before this window is displayed, power supply of the corresponding PC3000 channels will be turned off, i.e. you can remove the screw and insulate the connector immediately after reading the instructions. Once you have insulated the connector and clicked the 'Start procedure' button, the utility will perform a series of necessary operations and ask at a certain moment to remove the insulating layer and put back the screw. When the algorithm completes, the drive will be unlocked.

8.2. Short connection of the read channel

This method is universal because it does not depend on the HDD family or FW version. Therefore it is used in the utility as the recommended mode for unlocking in "LED: 000000CC" cases. The main operation of the method is bridging of the differential contacts pair in the reading channel[1]. Short-circuit connection should be performed after the terminal command mode becomes active (the terminal must return the F3 T> prompt), but before the LED: 000000CC error occurs. Since the tool used for connection is going to function in a high-frequency circuit, certain restrictions apply: its impedance (active and reactive resistance parts) must be minimal, the contact points must not be oxidized, the

[1] In the end of this description you can find photographs of controller boards with indicated connection points. However, identification of these points is sufficiently simple. They are actually the transitional openings of the differential contacts pair of the read channel lines to the external side of the board. The differential pair is represented by two parallel conductors from the preamplifier connector towards the processor with a resistor soldered across them (there is no resistor on the write channel pair).

short connection ark size must be minimal. The simplest solution in such situations would be small tweezers with sharp edges holding a metal paper clip (see *Fig. 8.2*).

Fig. 8.2

If your selected tool complies with the described requirements, short-circuit connection makes the HDD produce knocking sounds and after a while reach the ready state with its motor stopped. You are advised to pick the appropriate tool in advance by bridging the read channel of a functional drive and switching on the HDD power supply.

The procedure for HDD unlocking using a short connection of the read channel is as follows:

- Switch off the HDD power supply.
- Switch on the HDD power supply and check that the signal indicating activation of the terminal command mode is output to the terminal.
- As soon as the *F3 T>* prompt appears in terminal, perform the short connection of the read channel according to the description above.
- After a while the HDD will reach readiness with its motor stopped (before the drive stops its spindle and reaches readiness, it must produce knocking sounds).
- Then follow the basic unlocking algorithm.

To start the mode in the utility:

- Select the 'Solution for problem "HDD Lock (LED: 000000CC)"' menu item.
- Confirm that the mode should be started.
- Specify the backup folder for service data storage.
- Select 'Short-circuiting the reading channel' as the unlocking method.
- Follow the guidelines displayed in the main window in that mode.

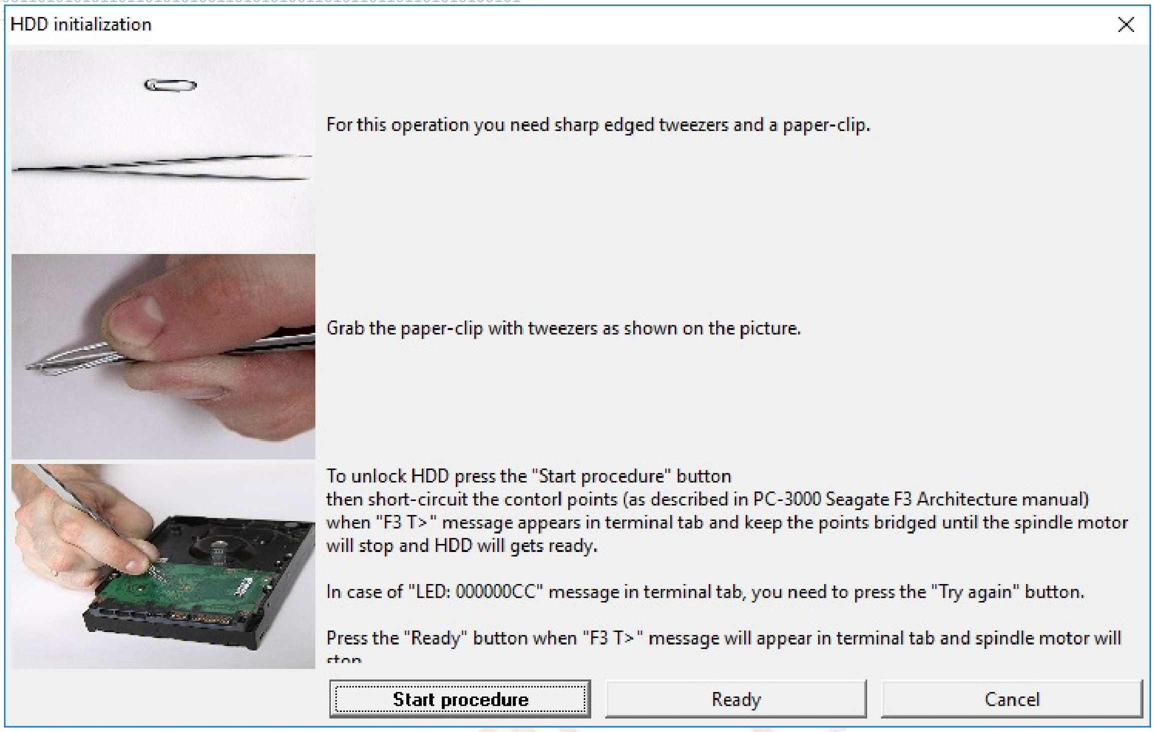


Fig. 8.3

Here you should click the 'Start procedure' button, monitor the terminal output[1], and connect the appropriate points on the controller board when the HDD displays the **F3 T>** prompt. When the drive reaches readiness, remove the connecting tool and click the 'Ready' button.

The mode can be also quite convenient for bypassing of some other LED locks. In such cases, click the 'Cancel' button as soon as access to terminal is obtained, then proceed to the analysis of drive status via the terminal.

8.3. Operating principle of the main part of the unlocking procedure

This part of the algorithm implemented in the utility does not require user participation; it is described here for better understanding of the processes in the HDD being unlocked.

When one of the lock bypass algorithms (spindle motor insulation or short connection of the read channel) completes, the utility analyzes service information, backs up the HDD modules and ROM, resets S.M.A.R.T. and overwrites the translator module. This sequence allows it to preserve the original drive condition (except for CE-log) and skip the translator recalculation step.

Attention! At present the majority of HDD after Self Scan pass post-processing consisting of logical scanning with relocation of defects beyond P-List. As a result, translator recalculation based on P-List (described in articles available in the Internet, used in the previous revision of our suite and the Salvation Data tools) will end up in translation mismatch and inaccessible user data beginning with a certain LBA. Automatic solution for the translation mismatch problem is implemented in our utility, you can invoke it from the menu 'Tests → Service information → Work with service area → Recover translator'[2] (for details please see *section 5.5, Host system detects a HDD, drive beginning is visible but starting with a certain LBA continuous space until the end is filled with UNC errors).*

[1] It is easier to monitor the situation using the RxD indicator of the PC USB Terminal adapter instead of the terminal window: when it starts blinking frequently it means that the drive returns the command mode prompt. The adapter can be conveniently placed near the HDD within the field of vision while examining the connection points.

[2] We should also mention that translator corruption resulting from G-List appending in case of writing problems becomes increasingly frequent and can be encountered independently from the LED: CC lock because of CE Log overflow. The problem at that may be initially manifested as zero capacity in the HDD ID.

8.4. Searching for the short connection points

The lines of the read channel differential pair are highlighted on the figure below. You can see there is a resistance inserted between the lines next to the microcontroller, it distinguishes this pair from that of the write channel. In addition, there are transition openings between the resistor connection points and microcontroller. Since the PCB on the drive is fixed to the HDA, these are the transitional openings, which should be short-circuited on the external side of the board.

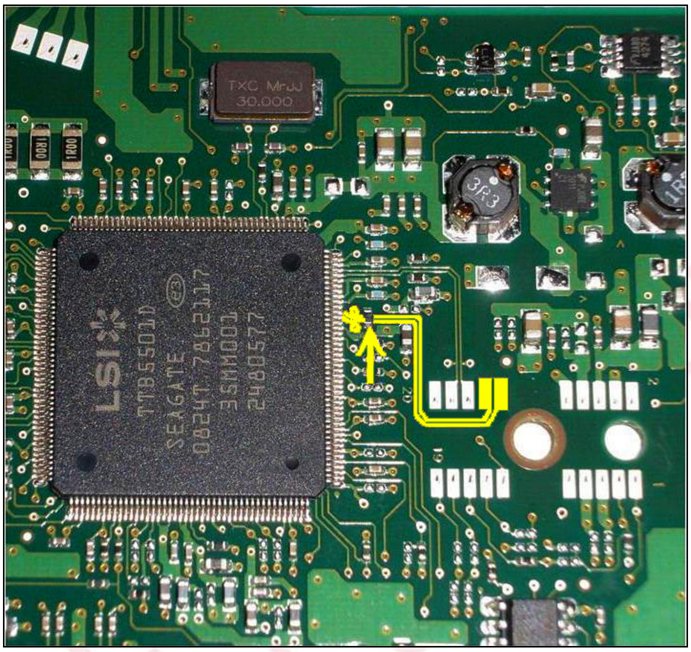


Fig. 8.4

Please keep in mind that if the above-mentioned resistor is not present on the differential pair of the read channel, you can identify the pair experimentally. To do that, you have to identify on the controller board two differential pairs - those of the read and write channels. Then you need to short-circuit one of them and try starting the drive.

Attention! **During the experimental start keep the selected differential pair bridged all the time to avoid damaging the service information!** If the read channel is bridged, after power-up the drive will spin up the spindle, knock for a while and then stop the motor. If the write channel is bridged, the drive will produce no knocking sounds. If you cannot identify the differential pair of the read channel during the first attempt, try the other one.

9. Appendix 1. Replacement of the controller board

Attention! ROM of the drives (ROM label is 25FW406A) contains adaptive information. Therefore, during replacement of the controller board you have to move the ROM data from the damaged board to the donor one (resolder the ROM chip from the damaged board to the borrowed one). See *Fig. 9.1*.

Fig. 9.1. ROM chip is indicated in the figure.

Please keep in mind that if you need to restore a drive using a non-native controller board, the drive at the start will be unable to access the firmware data on disk surface. That behaviour can be erroneously recognized as a problem with the heads or reading channel, so please check carefully to ensure that the controller board matches the HDA. To do that, you can even use a board disconnected from HDA, which returns HDD ID information via the ATA interface anyway. Having read HDD ID, you can compare the model and its serial number to the information printed on HDA label[1].

Attention! Please keep in mind that if a non-native controller board is used, then any operation involving recording to disk surface may cause irreversible corruption of service data resulting in inaccessibility of user data.

[1] Unfortunately, as we have mentioned before, in some Barracuda ES.2 drives the FW portion in ROM contains the same standard data for all HDD making identification a more complex task.

10. Appendix 2. Testing the HDD controller board

To check normal board functionality, there is an additional method besides visual inspection and electric layout analysis. It is based on the fact that a connection to the SATA adapter must be established when a functional board is powered on. Established connection is indicated by the glowing PHY RDY indicator on the PC PATA – SATA adapter. If the indicator is off, the board is malfunctioning. Please note also that a functional board disconnected from its HDA enters the readiness state in a while after being powered-on and returns HDD ID (for the 7200.11 drive family it also returns correct SN). Besides, to analyze the controller board condition, you can use the PCB swap method. Boards with the same revision should be used for the swap procedure (*Fig. 10.1*)..

Fig. 10.1. HDD controller board, revision 100496208 REV A.

Even in that condition the donor board is suitable for initial estimation of the drive status (does it establish SATA connection, does it spin up the spindle motor, whether terminal response is available). For further analysis including service data inspection you should transfer to the donor board adaptive data from the malfunctioning PCB. To do that, you have to resolder the ROM chip (see *section 9, Appendix 1. Replacement of the controller board*).

11. Appendix 3. Testing the motor winding coils

Motors in Seagate drives use the triangular layout (*Fig. 11.1.*), so you have to check three pairs of connection points for drive coils. Resistance between pairs of the coils connection points in a functional drive is ~ 3.4 Ohm. You can determine the value more precisely for each individual case by measuring the coils resistance on a functional drive belonging to the same family.

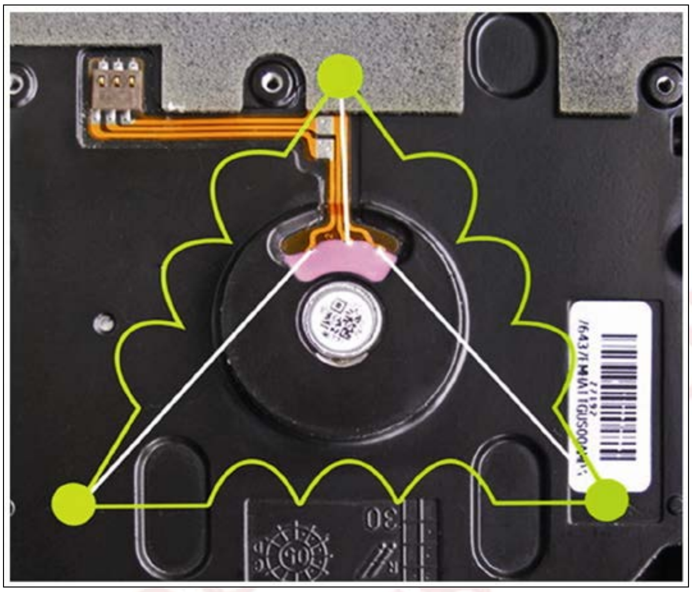


Fig. 11.1.

12. Appendix 4. Selecting a donor heads stack

To select a suitable donor heads stack, the following criteria should be used:

- Names of the donor and patient drives must match to ensure that the drive family and the number of used magnetic heads are the same[1].
- The number of magnetic heads and their map in the patient and donor drives must match.
- To test suitability of the donor heads, install the original controller board from the patient drive onto the potential donor HDA, switch on the power and watch over the start of the produced "hybrid" device. If the HDD produces strong and quick knocking sounds with the heads and stops the spindle motor, these heads are incompatible. If the drive does not stop and recalibration attempts are noticeable, you can try using the set as a donor part. Unfortunately, success cannot be guaranteed in all cases because the parameters of heads vary widely not only between different HDD, but even within a single magnetic heads stack. Still, the heads stack selected using this method will have the highest chance of compatibility.

Attention! Back up the ROM content prior to any modification! ROM contains unique settings of the drive, their loss or corruption will result in irreversible damage to the HDD.

[1] Smaller drives belonging to the same family may be used, but in that case you will have to perform additional steps for modification of information in ROM and some work with the HDA mechanics. If you have no successful practical experience of such operations, using donor HDD with fewer magnetic heads than in the patient drive is strongly discouraged.

13. Appendix 5. Location of the read channel checkpoints on Barracuda F3 3.5" HDD

13.1. PCB 100466725 REV A (DLAJ-4)

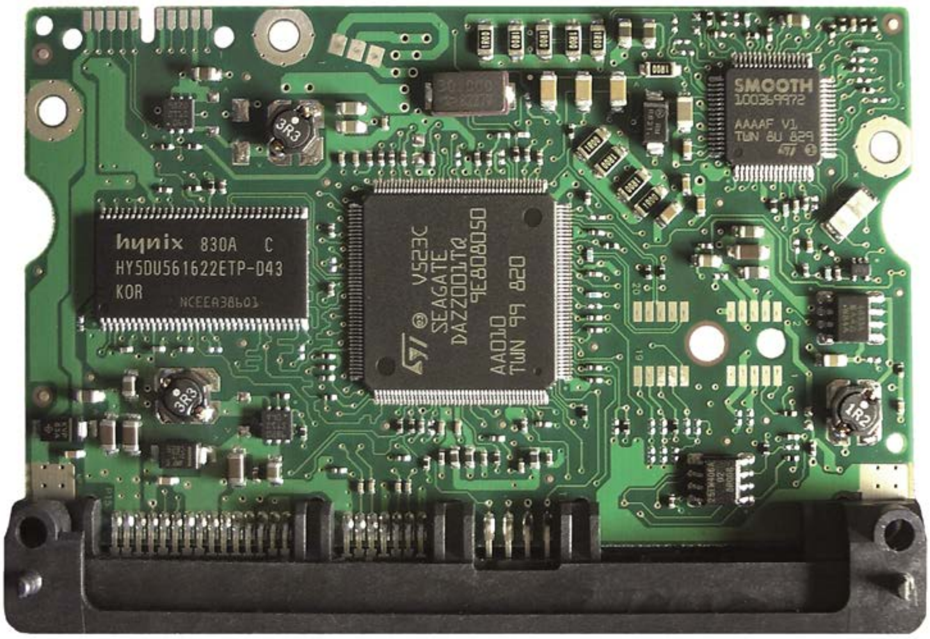


Fig. 13.1. Board view from the components side.

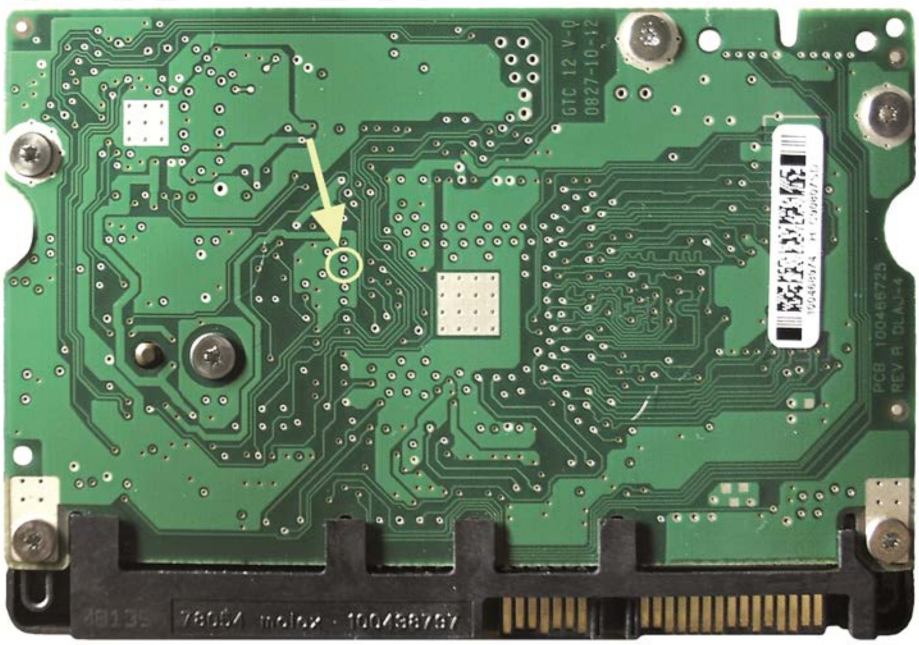


Fig. 13.2. Board view from the external side. Short connection points are highlighted in yellow.

13.2. PCB 100466824 REV A (UJAJ-6)

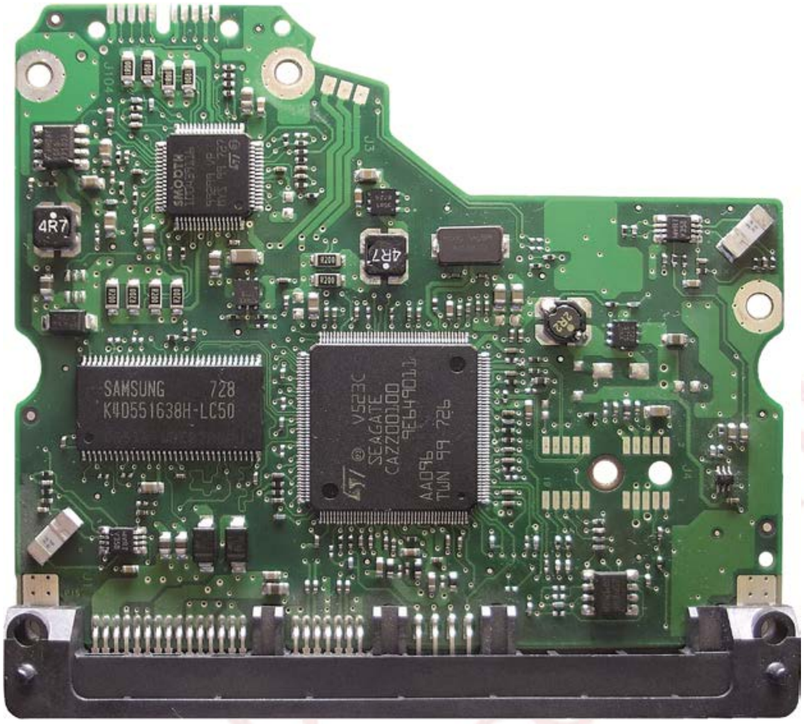


Fig. 13.3. Board view from the components side.

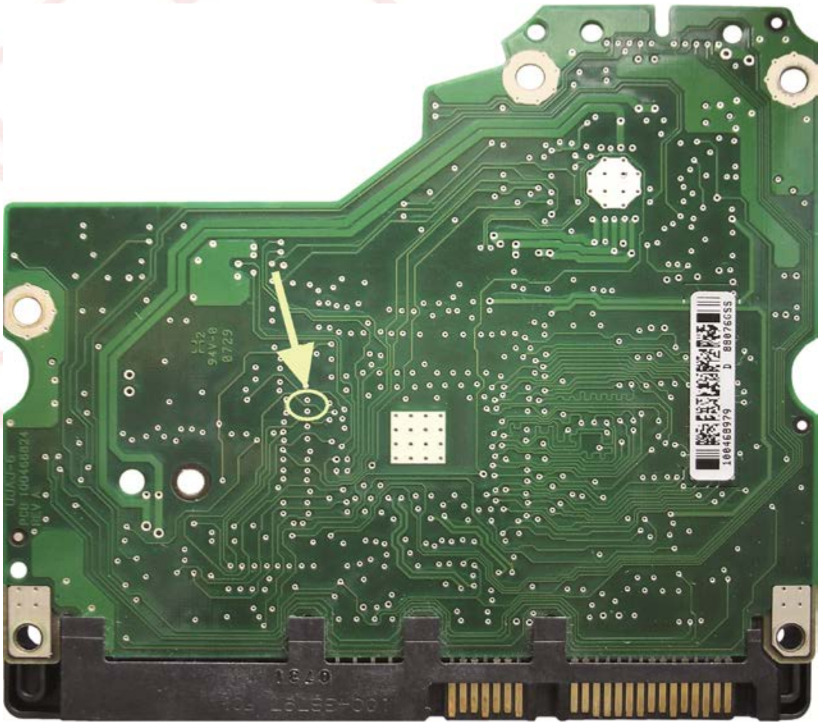


Fig. 13.4. Board view from the external side. Short connection points are highlighted in yellow.

13.3. PCB 100466824 REV B (UJAJ-6)

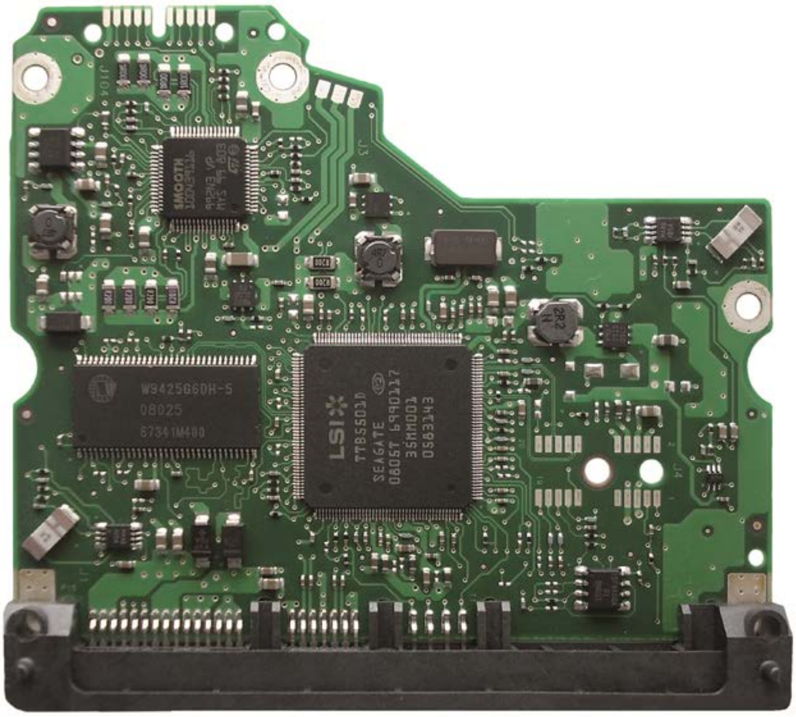


Fig. 13.5. Board view from the components side.

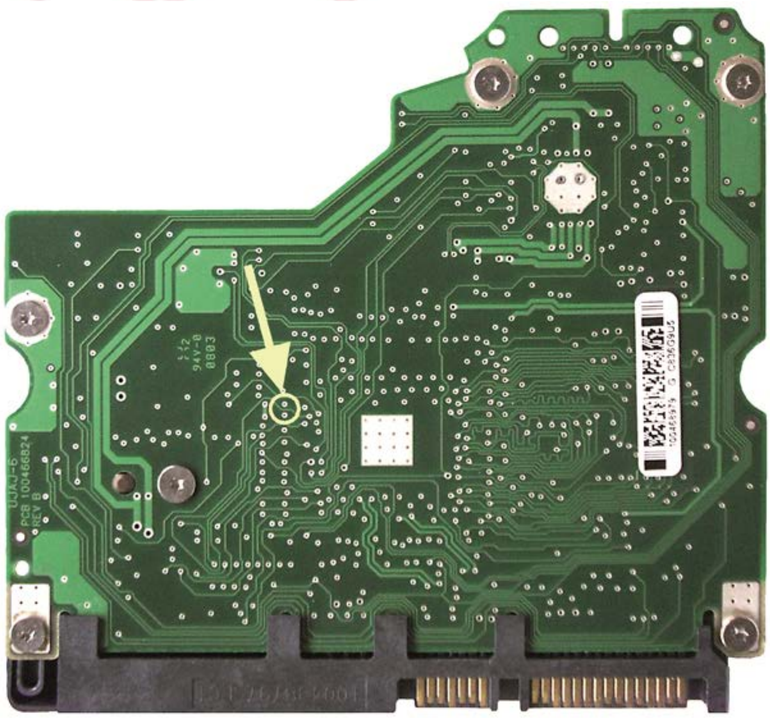


Fig. 13.6. Board view from the external side. Short connection points are highlighted in yellow.

13.4. PCB 100466824 REV C (UJAJ-6)

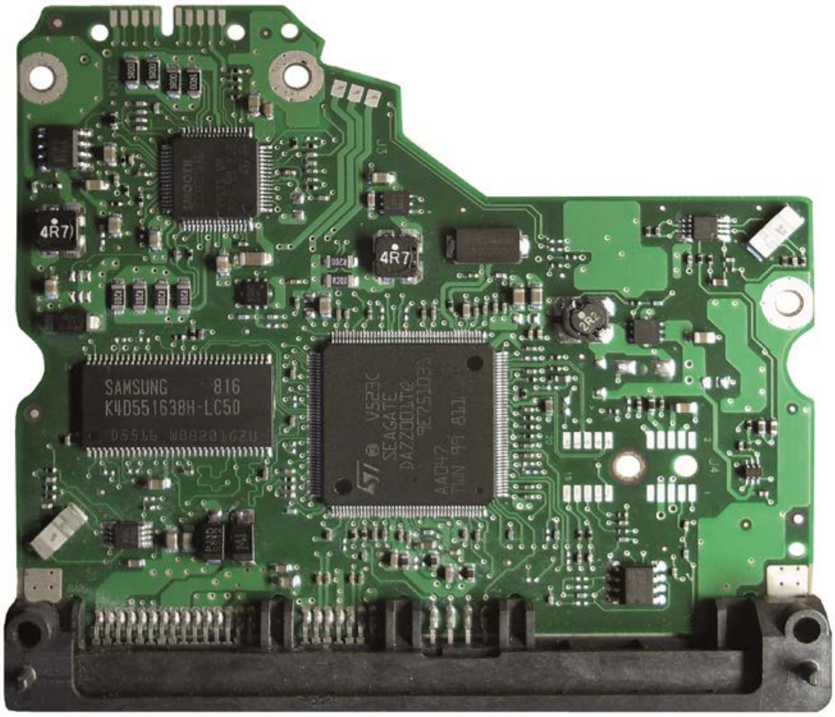


Fig. 13.7. Board view from the components side.

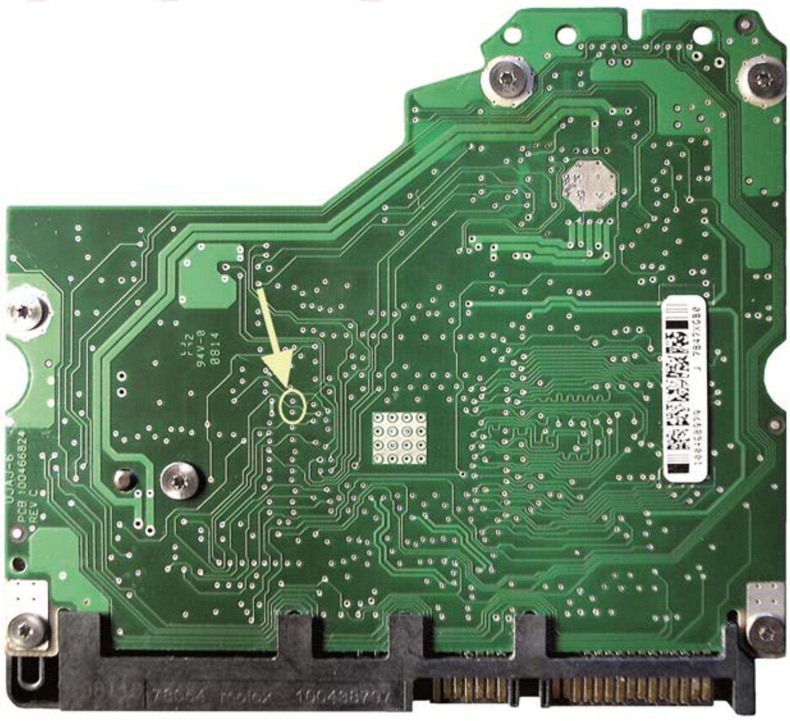


Fig. 13.8. Board view from the external side. Short connection points are highlighted in yellow.

13.5. PCB 100496208 REV A

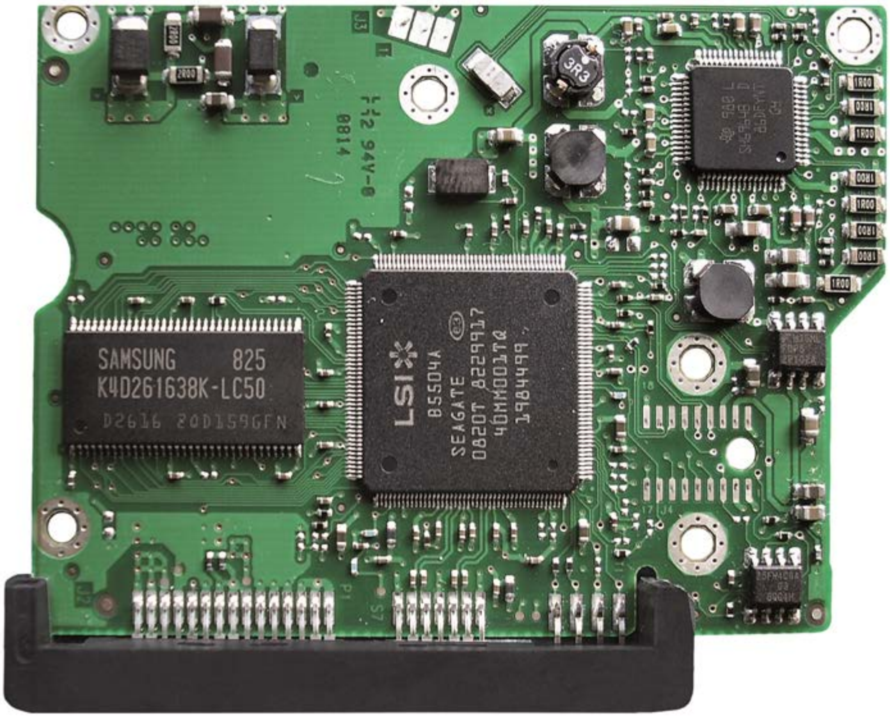


Fig. 13.9. Board view from the components side.

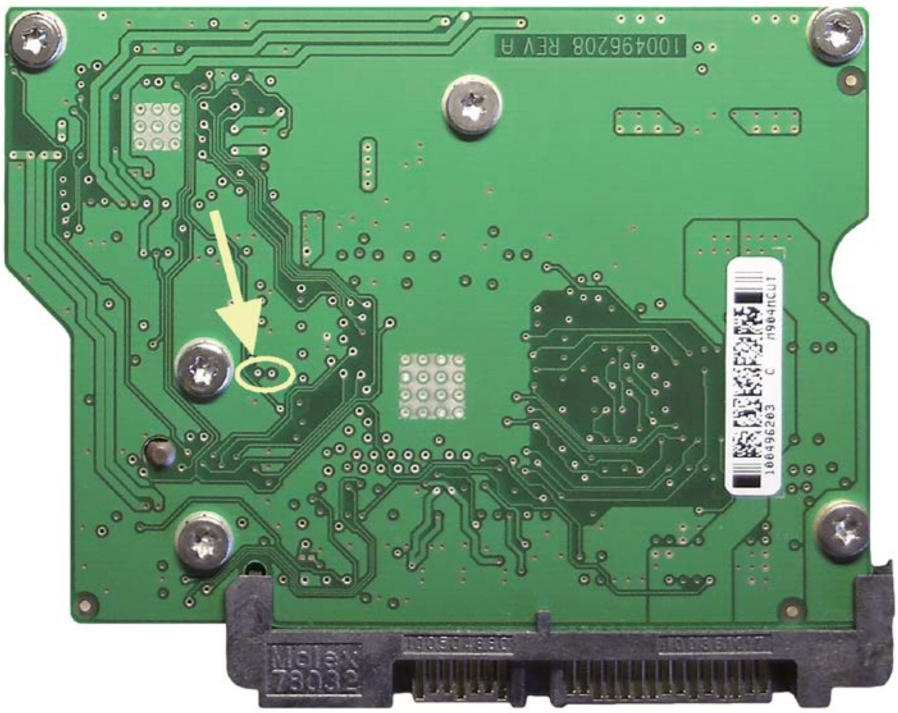


Fig. 13.10. Board view from the external side. Short connection points are highlighted in yellow.

13.6. PCB 100504364 REV B

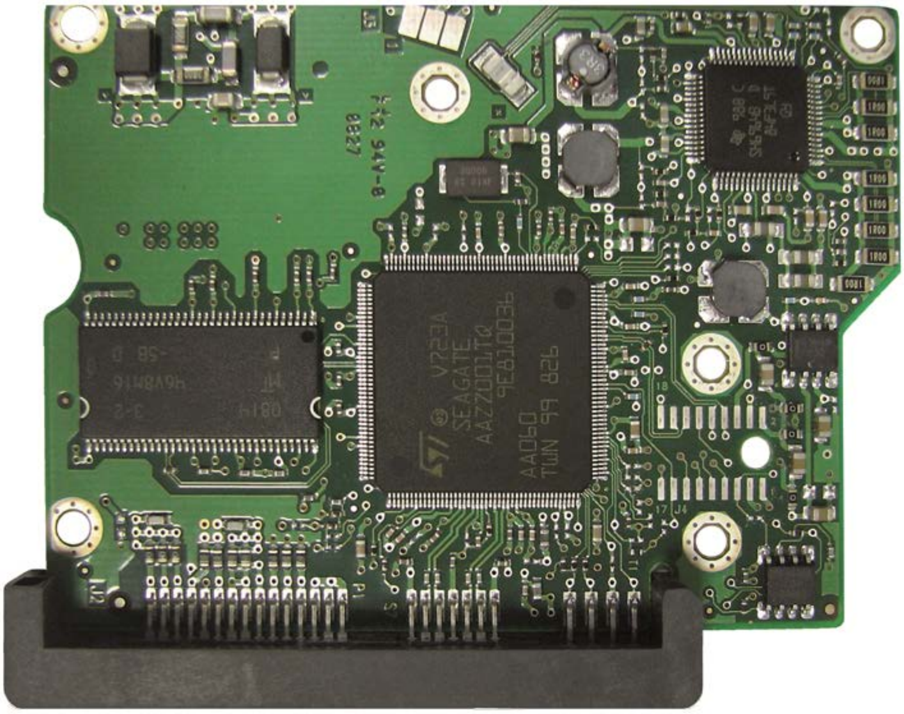


Fig. 13.11. Board view from the components side.

Fig. 13.12. Board view from the external side. Short connection points are highlighted in yellow.

13.7. PCB 100512588 REV A

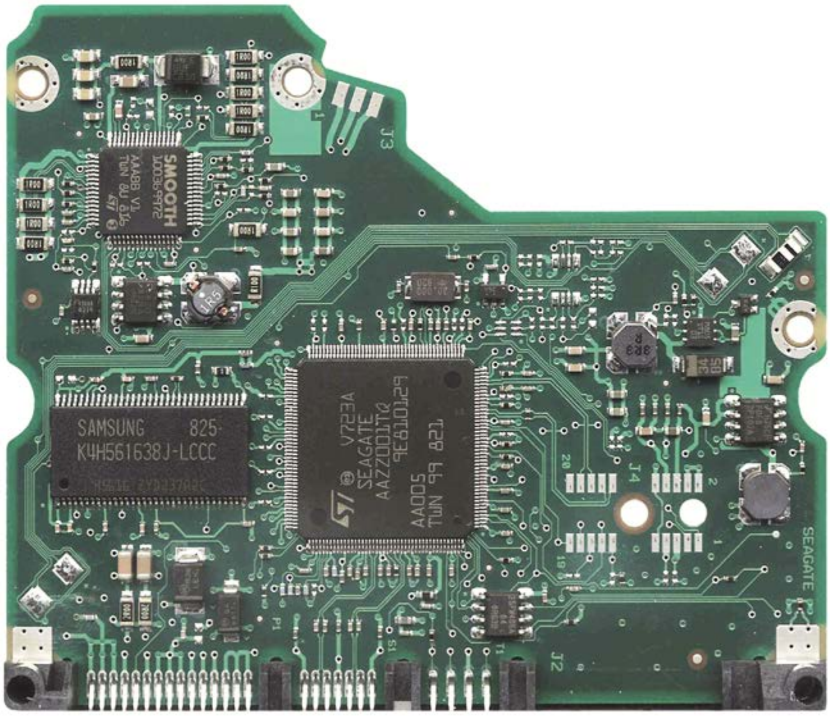


Fig. 13.13. Board view from the components side.

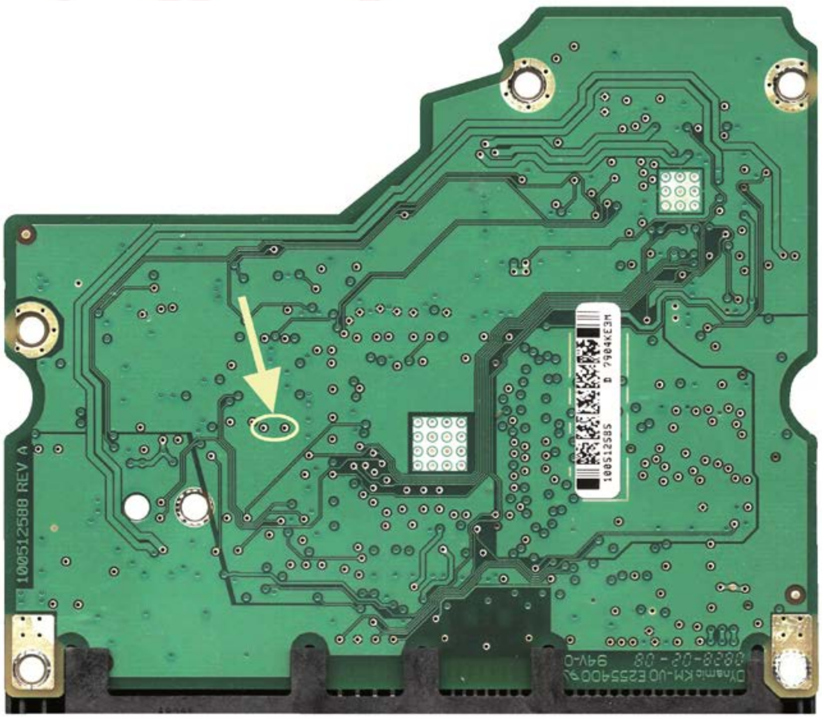


Fig. 13.14. Board view from the external side. Short connection points are highlighted in yellow.

13.8. PCB 100475720 REV A (ZKAJ-7)

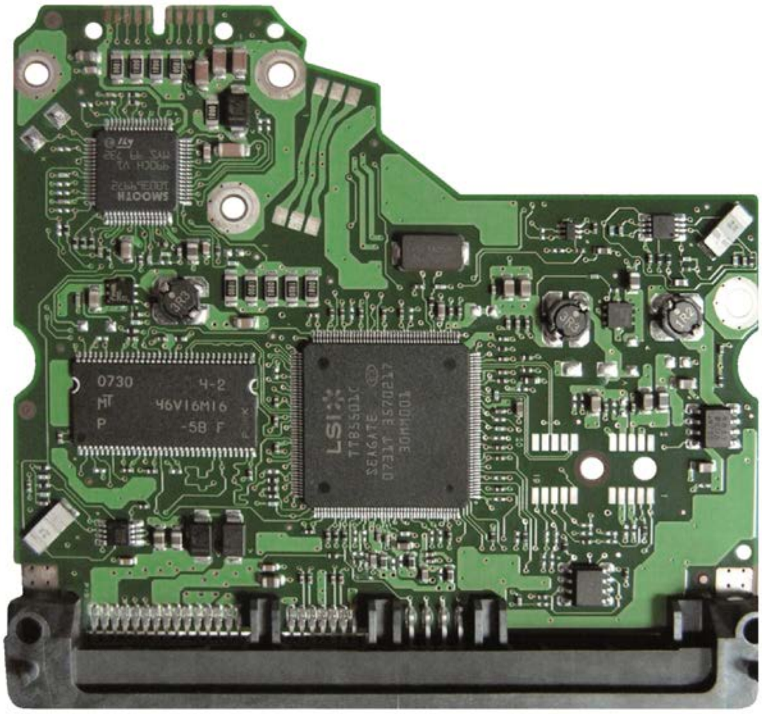


Fig. 13.15. Board view from the components side.

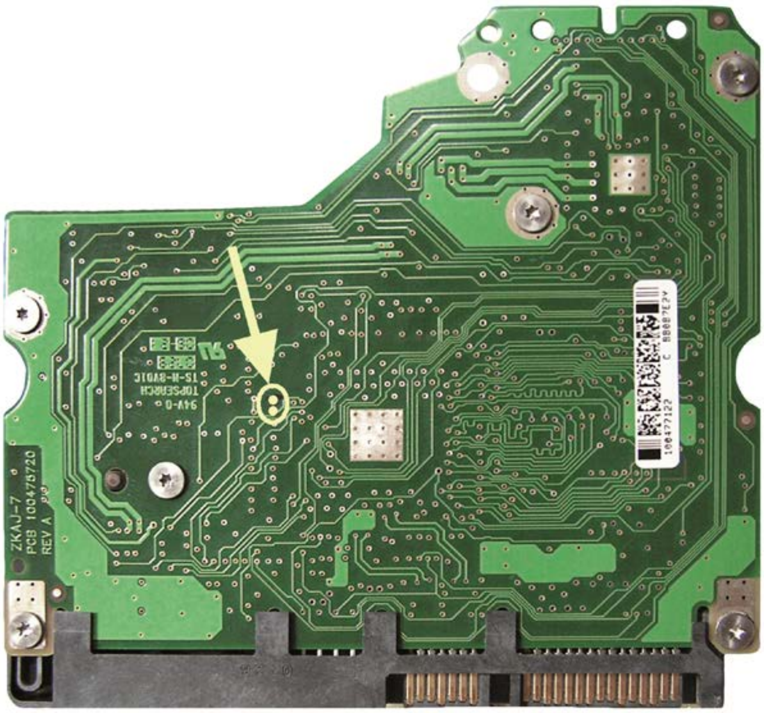


Fig. 13.16. Board view from the external side. Short connection points are highlighted in yellow.

14. Appendix 6. Location of the checkpoints on Barracuda F3 2.5" HDD

14.1. PCB 100536286 REV E

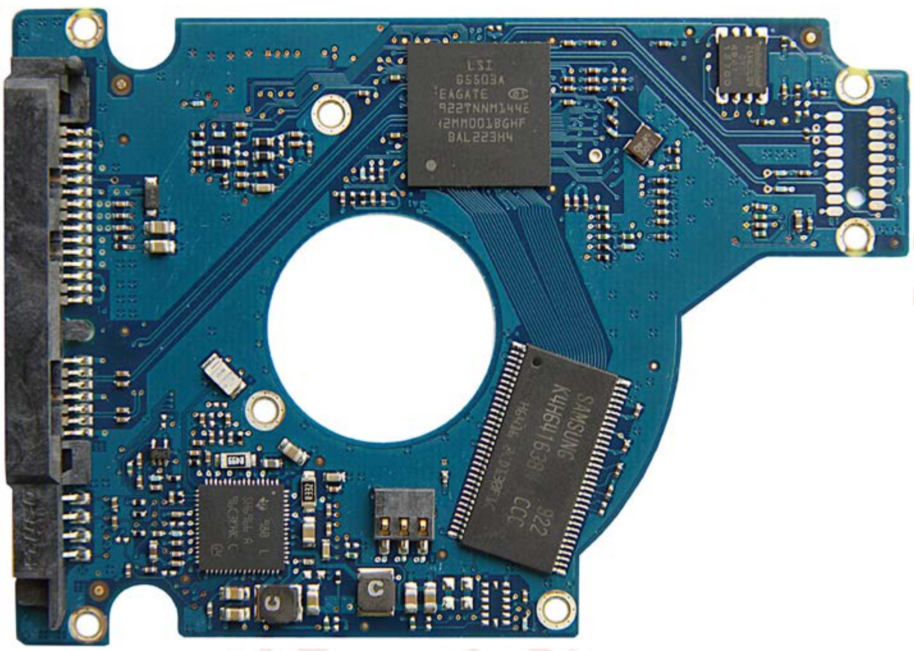


Fig. 14.1. Board view from the components side.

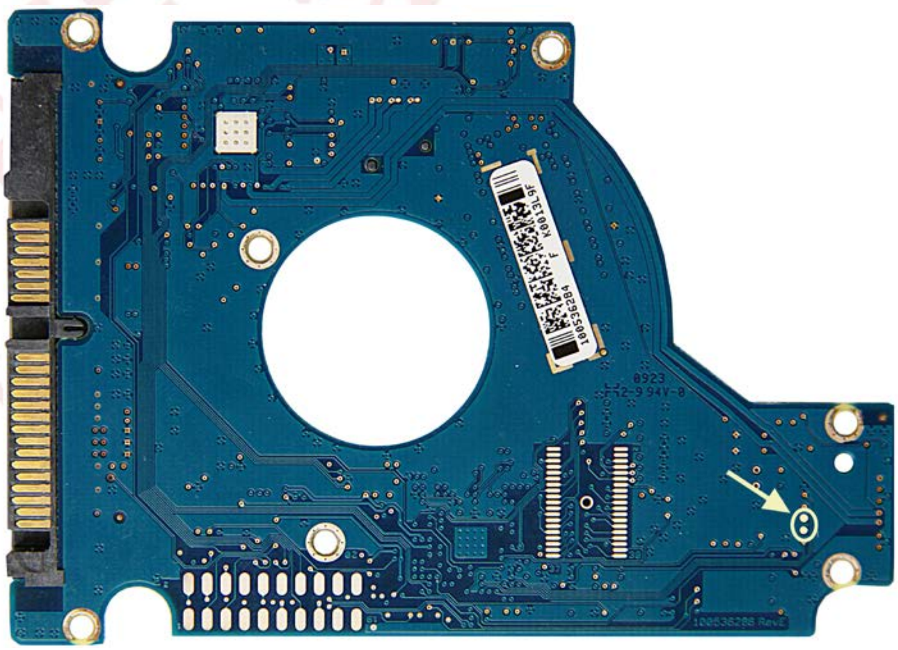


Fig. 14.2. Board view from the external side. Short connection points are highlighted in yellow.

14.2. PCB 100513573 REV B

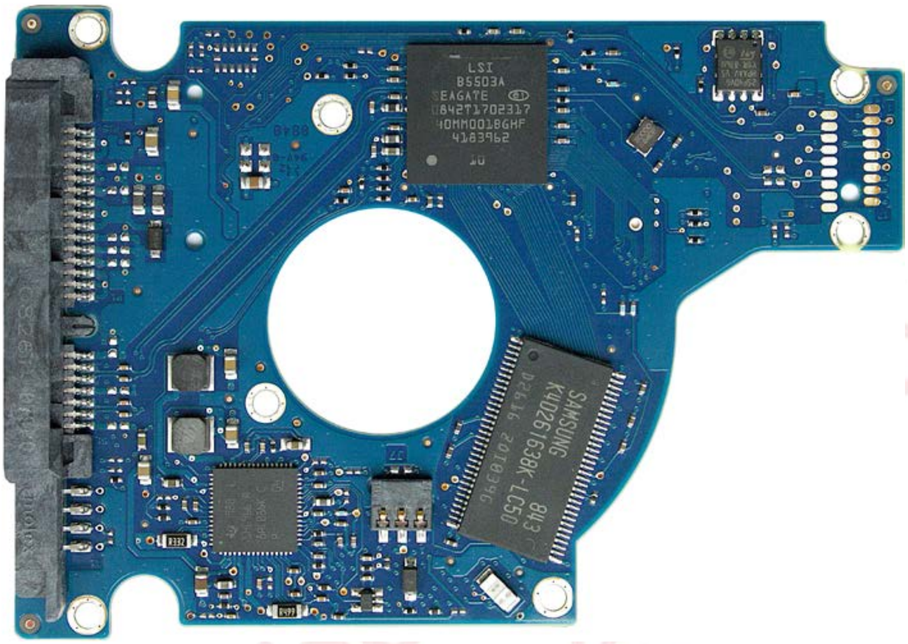


Fig. 14.3. Board view from the components side.

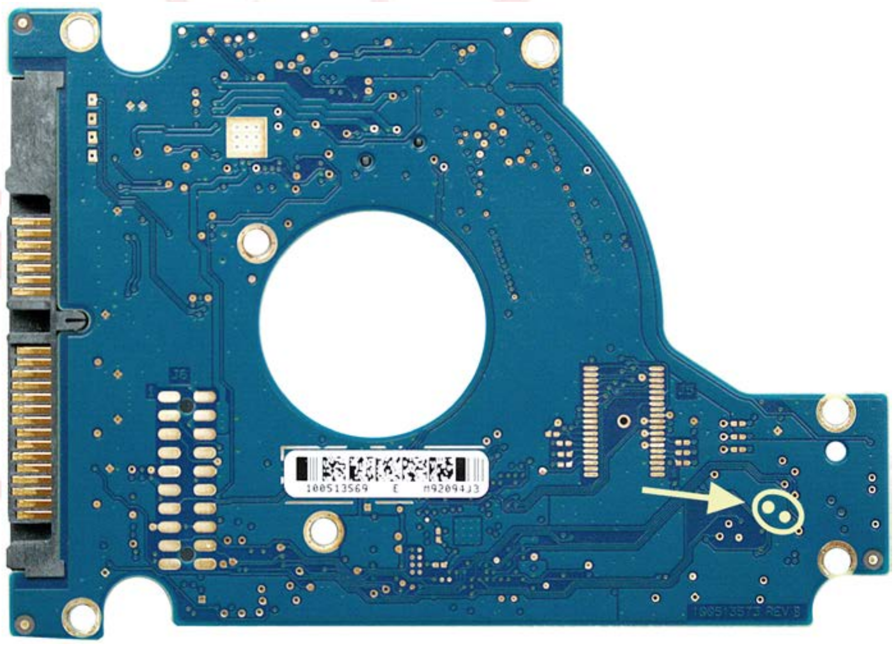


Fig. 14.4. Board view from the external side. Short connection points are highlighted in yellow.

15. Appendix 7. Functional purpose of modules in Seagate Arch F3 HDD

00 = SA defect list
01 = Drive Information File
02 = Performance Parameter File
03 = P-List
04 = SAP (Servo Adaptives Parameters)
05 = Manufacturing Information File
06 = RAP (Read Adaptive Parameters)
07 = CAP (Controller Adaptive Parameters)
08 = unknown purpose
09 = SMART config (filled with a pattern)
0A = SMART Frame
0B = unknown purpose, filled with a pattern
0C = Self Scan Log
0D = unknown purpose, filled with a pattern
0E = DIC (Data Integrity Check), HEAD 0
0F = DIC Dummy File
13 = disk registry (configuration settings, Security)
15 = offline surface self-test file (DOS)
16 = unknown purpose
17 = unknown purpose, filled with a pattern
19 = unknown purpose, some log
1A = unknown purpose, filled with a pattern
1B = unknown purpose, some log
1C = unknown purpose, filled with a pattern
1D = overlay 0
1E = overlay 1
1F = unknown purpose, filled with a pattern
22 = surface self-test file, head 0
23 = surface self-test file, head 1
24 = surface self-test file, head 2
25 = surface self-test file, head 3
26 = surface self-test file, head 4
27 = surface self-test file, head 5
28 = surface self-test file, head 6
29 = surface self-test file, head 7
2A = Saved Mode Pages – HDD settings
2B = RW operations, translator
2C = DIC (Data Integrity Check), HEAD 1, pattern
2D = DIC (Data Integrity Check), HEAD 2, pattern
2E = DIC (Data Integrity Check), HEAD 3, pattern
2F = DIC (Data Integrity Check), HEAD 4, pattern
30 = DIC (Data Integrity Check), HEAD 5, pattern
31 = DIC (Data Integrity Check), HEAD 6, pattern
32 = DIC (Data Integrity Check), HEAD 7, pattern
33 = FDE service file
34 = packed CONGEN structure descriptor
35 = SMART config (filled with a pattern)

16. Appendix 8. Correspondence between the service data modules and System Files in Barracuda 7200.12 (Pharaoh) HDD

```
FILE_3_01A_0 = ~0001.rpm
FILE_3_019_0 = ~0002.rpm
FILE_3_01B_0 = ~0003.rpm
FILE_3_03F_0 = ~0004.rpm
FILE_3_300_0 = ~0005.rpm
FILE_3_001_0 = ~0006.rpm
FILE_3_208_0 = ~0007.rpm
FILE_3_31B_0 = ~0008.rpm
FILE_3_133_0 = ~0009.rpm
FILE_3_134_0 = ~000A.rpm
FILE_3_319_0 = ~000C.rpm
FILE_3_30A_0 = ~0013.rpm
FILE_3_306_0 = ~0015.rpm
FILE_3_115_0 = ~0019.rpm
FILE_3_131_0 = ~001A.rpm
FILE_3_301_0 = ~001B.rpm
FILE_3_110_0 = ~001C.rpm
FILE_3_100_0 = ~001D.rpm
FILE_3_101_0 = ~001E.rpm
FILE_3_093_0 = ~002A.rpm
FILE_3_028_0 = ~002B.rpm
FILE_3_32C_0 = ~0034.rpm
FILE_3_135_0 = ~0035.rpm
```

E.g.: FILE_3_01A_0 = ~0001.rpm
Here: service information module = ~0001.rpm, System File = FILE_3_01A_0, Volume = 3, File ID (FID) = 01A, copy = 0.

System Files are requested using the File ID from Volumes. The utility can access System Files using the 'View and edit HDD resources' tool.

In addition, it has been found that ROM = FILE_A_32A_0, the information allows accessing its content on a drive that does not reach readiness but enters the terminal command mode.

Furthermore, the described utility version also displays in the modules viewing dialog the information about the descriptors of module files, and consequently about their identifiers (provided that the initialization has been successful).

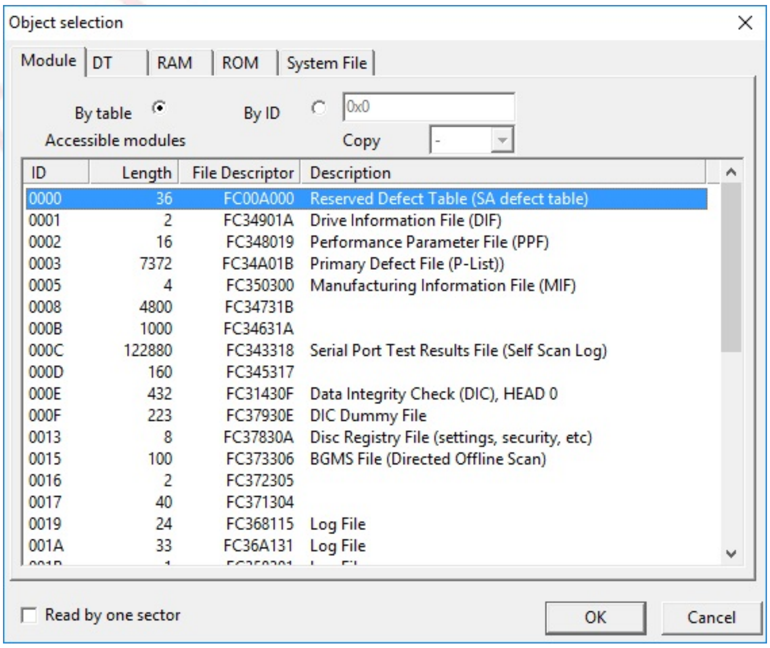


Fig. 16.1.