

" Утилита "PC-Seagate Uxx series".

Ver. 1.2.

Содержание

1. Назначение.....	1
2. Состав семейств	2
3. Работа с утилитами через IDE.....	2
3.1. Служебная информация	2
3.1.1. Работа с ПЗУ.....	2
3.1.2. Работа со служебной зоной.....	3
3.2. Паспорт диска.....	5
3.3. Таблица S.M.A.R.T.....	6
3.4. Таблица дефектов.....	6
4. Работа через терминал.....	6
4.1. Служебное меню терминала.....	6
4.2. Используемые команды терминала.....	6
4.2.1. Получение модуля или Flash-ПЗУ из накопителя.....	7
4.2.2. Запись модуля или Flash-ПЗУ в накопитель.....	7
4.2.3. Таблица команд.....	8
4.2.3.1. Оп-Line команды.....	8
4.2.3.2. Форматы откликов.....	8
4.2.3.3. Расшифровка битов health.....	9
4.2.3.4. Общие команды.....	9
4.2.3.5. Уровень T (уровень 0).....	9
4.2.3.6. Уровень 1.....	10
4.2.3.7. Уровень 2.....	11
4.2.3.8. Уровень 9.....	11
5. Коды ошибок HDD.....	11

1. Назначение.

Утилита предназначена для работы с накопителями Seagate семейств U4, U8, V8, U10, V10, U6. Большинство функций, реализованных данной утилитой, действуют через ATA интерфейс, и, соответственно, для их использования необходимо, чтобы накопитель выходил в готовность. Накопитель, не выходящий в готовность, необходимо предварительно протестировать, используя электротехнические методы. При необходимости надо предпринять действия по восстановлению электроники, замене БМГ, и т. д. После восстановления аппаратных неисправностей следует предпринять диагностику через терминальные команды накопителя. Для этого в утилите предусмотрен встроенный терминал, а в документации приведен список диагностических команд с кратким описанием. Кроме того, в терминал утилиты встроены функции манипуляции ключевыми модулями и Flash-ПЗУ.

2. Состав семейств

Семейство	Модель	Емкость, ГБт.	Кол-во дисков	Кол-во гол.	Максимальный LBA
U4	ST32112A	2,1	1	1	4,225,536
	ST34311A	4,3	1	2	8,452,080
	ST36421A	6,4	2	3	12,596,850
	ST38421A	8,4	2	4	16,498,944
U8, V8	ST34313A	4,3	1	1	8,452,080
	ST38410A	8,6	1	2	16,841,664
	ST313021A	13,0	2	3	25,434,228
	ST317221A	17,2	2	4	33,683,328
U10, V10	ST310212A	10,2	1	2	20,005,650
	ST315323A	15,3	2	3	30,008,475
	ST320423A	20,4	2	4	40,011,300
U6	ST320410A	20	1	1	39,102,336
	ST340810A	40	1	2	78,165,360
	ST360020A	60	2	3	117,231,408
	ST380020A	80	2	4	156,301,488

3. Работа с утилитами через IDE.

Для работы через IDE предназначены пункты меню “Служебная информация”, “Паспорт диска”, “Таблица S.M.A.R.T.”, “Таблица дефектов”.

“Служебная информация”. При выборе выводится меню, содержащее следующие пункты:

*Работа с ПЗУ
Работа со служебной зоной
Останов шпинделя*

“Паспорт диска”. Выводит диалог редактирования паспортной информации. А именно:

- количество логических цилиндров
- количество логических головок
- количество логических секторов
- LBA
- имя модели
- флаг поддержки HPA
- флаг поддержки Security
- флаг поддержки HPA
- флаг поддержки S.M.A.R.T. Self Test
- флаг поддержки S.M.A.R.T. Error Logging

“Таблица S.M.A.R.T.”. При выборе выводится меню, содержащее следующие пункты:

*Просмотреть таблицу S.M.A.R.T.
Сброс S.M.A.R.T. параметров*

“Таблица дефектов”. При выборе выводится меню, содержащее следующие пункты:

*Очистить G-List
Очистить P-List
Пересчёт транслятора*

3.1. Служебная информация

3.1.1. Работа с ПЗУ.

Работа с ПЗУ позволяет считать и записать Flash-ПЗУ.

Чтение ПЗУ - осуществляет считывание содержимого ПЗУ в файл с расширением *.bin или *.rom. При выборе этой операции необходимо указать формат, в котором будет сохранено ПЗУ и имя файла без расширения. Возможны два формата сохранения ПЗУ: внутренний формат команды накопителя работы с ПЗУ

(сырой формат, данные сохраняются так, как они расположены в памяти, в формате утилит обновления) и формат программатора (в данном формате ПЗУ накопителя считывается так, как если бы вы читали его программатором). Считанный файл помещается в текущий подкаталог PC3000. Данная операция корректно обрабатывается накопителями серий U8, U10.

ВНИМАНИЕ! Накопители серии U4 при запросе на выполнение этой функции возвращают ошибку, накопители же U6, вследствие ошибки в микропрограмме, корректно возвращают только половину ПЗУ, в связи с чем функция чтения ПЗУ для семейства U6 в утилите отключена.

Операцию чтения ПЗУ из данного пункта меню можно выполнять только на выходящем в готовность по IDE интерфейсу накопителя. Отдельно для платы электроники эта операция не выполняется.

Запись ПЗУ. Осуществляет запись Flash-ПЗУ из файла с расширением *.bin или *.rom. При выборе этой операции необходимо указать формат и имя файла образа ПЗУ без расширения. Выбор файлов осуществляется из текущего подкаталога PC3000. Данная операция корректно обрабатывается накопителями семейств U8, U10, U6.

После записи Flash-ПЗУ накопитель через некоторое время, необходимое для репрограммирования Flash-ПЗУ, должен выдать в СОМ порт и регистры состояния и ошибки IDE результат выполнения операции и осуществить рестарт микропрограммы.

3.1.2. Работа со служебной зоной.

Данный пункт меню содержит следующие подпункты :

*Проверка структуры служебной информации
Чтение модулей
Запись модулей
Выбор копии для чтения
Подсистема безопасности*

Проверка структуры служебной информации. Позволяет оценить состояние служебной информации Вашего накопителя. Выводимый отчёт выглядит примерно следующим образом (приведена верхняя часть отчёта для накопителя семейства U6) - табл.2

Таблица 2.

Таблица модулей					
ID	CharID	File	RD	IDNT	C_SUM
01	OCCT	key 01	√	√	√

где:

ID – идентификатор модуля – число, присутствующее в теле модуля по смещению 0Ah (не все модули содержат таковой идентификатор, кроме того, не все из перечисленных модулей используются накопителем и потому не инициализированы).

CharID – внутреннее символьное имя модуля. Содержится не во всех модулях.

File – имя модуля в соответствии с номером ID (так называемым “ключом”, “key”). Таблицы соответствия ключа функциям модуля приведены ниже (Табл. 3, 4, 5).

RD – читаемость модуля

IDNT – идентификация. Для проверки используется сравнение запрошенного идентификатора с идентификатором в теле модуля (см. комментарий к ID)

C_SUM – корректность контрольной суммы. Здесь можно сказать, что у идентифицированных модулей KC должна быть верной.

Таблица 3. key files U4

key	name	ID	raw ID	size	примечание
00	rom		F_AT	\$100	имидж ПЗУ, отличается от того, что получаем на программаторе
01	rwf	RW00	RW00	\$011	зонное распределение
02	at stuff	ATST	нет	\$003	шаблон паспорта
03	smart	SMRT	нет	\$0FF	S.M.A.R.T.
04	cspt	CSPT	CSPT	\$004	
05	ovly_cct	CT_C	CT_C	\$018	

06	ovly act	CT A	CT A	\$018	
07	fls	FLS	нет	\$004	
08	udefbak	DEF0	DEF0	\$020	копия userdef (1 шт)
09	userdef	DEF0	DEF0	\$020	3 копии
0A	resvdef	RES0	RES0	\$002	
0B	diag	DIAG	нет	\$018	
0C	sysvar1	SYS1	SYS1	\$002	
0D	sysvar2	SYS2	SYS2	\$002	
0E	skptrk	SKIP	нет	\$001	
0F	flashct	F CT	нет	\$100	
10	flashat	F AT	нет	\$100	

Таблица 4. key files U8, U10

key	name	ID	raw ID	size	примечание
00	rom		F_AT	\$100	имидж ПЗУ, отличается от того, что получаем на программаторе
01	rwf	RW00	RW00	\$013	зонное распределение
02	at stuff	ATST	нет	\$003	шаблон паспорта
03	smart	SMRT	нет	\$006	S.M.A.R.T.
04	cspt	CSPT	CSPT	\$004	
05	ovly cct	OCCT	OCCT	\$018	
06	ovly act	OACT	OACT	\$018	
07	ovly cn1	OCN1	OCN1	\$018	
08	udefbak	DEF4	DEF4	\$044	копия userdef (1 шт)
09	userdef	DEF4	DEF4	\$044	4 копии
0A	resvdef	RES4	RES4	\$002	
0B	ovly an1	OAN1	OAN1	\$018	
0C	sysvar1	SYS1	SYS1	\$002	
0D	sysvar2	SYS2	SYS2	\$002	
0E	skptrk	SKIP	нет	\$001	
0F	flashct	F CT	нет	\$100	
10	flashat	F AT	нет	\$100	
11	ovly at1	OAT1	OAT1	\$018	
12	ovly6	OVL6	нет	\$018	
13	ovly7	OVL7	нет	\$018	
14	ovly8	OVL8	нет	\$018	
15	ovly9	OVL9	нет	\$018	
16	ovlyA	OVLA	нет	\$018	
17	security	SCRT	нет	\$001	модуль паролей
18	vendorsp	VEND	нет	\$064	информация от OEM поставщиков

Таблица 3. key files U6

key	name	ID	raw ID	size	примечание
00	rom		F_AT	\$100	имидж ПЗУ, отличается от того, который получаем на программаторе
01	ovly cct	OCCT	OCCT	\$080	
02	ovly act	OACT	OACT	\$080	
03	ovly xx	OCN1	нет	\$080	
04	defrsv	RSV0	RSV0	\$002	
05	deffty	FTY0	FTY0	\$080	P-List
06	defusr	USR0	USR0	\$020	G-List
07	deflzt	LZT0	LZT0	\$00A	транслятор
08	rwf	RW00	RW00	\$013	зонное распределение
09	sysvar1	SYS1	SYS1	\$005	
0A	sysvar2	SYS2	SYS2	\$002	
0B	cspt	CSPT	CSPT	\$004	
0C	vbpi	VBPI	VBPI	\$00A	

0D	flsh_at	F AT	???	\$100	по АТА не читает
0E	flsh_ct	F CT	???	\$100	по АТА не читает
0F	at_stuff	ATST	нет	\$001	шаблон паспорта
10	security	SCRT	нет	\$001	модуль паролей
11	vendorsp	VEND	нет	\$064	информация от OEM поставщиков
12	smart01	SATT	нет	\$001	(smart_attribute) SMART, атрибуты
13	smart02	STHR	нет	\$001	(smart_threshold) SMART, thresholds
14	smart03	SDIR	нет	\$001	(smard_directory)
15	smart04	SERR	нет	\$001	(smart_errorlog)
16	smart05	SCMP	нет	\$005	(smart_comprehensive_log)
17	smart06	SSLF	нет	\$001	(smart_self_test_log) SMART, self test log
18	smart07	SCRT	нет	\$014	(smart_critical_event_log) по АТА не читает
19	smart08	SHLT	нет	\$065	(smart_health_log)
1A	smart09	SDRV	нет	\$001	(smart_drive_vendor_log)
1B	smart10	SHST	нет	\$200	(smart_host_specific_log) по АТА не читает
1C	wr_prot	WRPT	нет	\$001	(write_protect)
1D	con_gen	CNGN	нет	\$002	
1E		SLST	нет	\$028	(skip_cylinder_list) список пропущенных цилиндров
1F		2TST	нет	\$001	
20		6TST	нет	\$001	
21		SCID	нет	\$001	
22		SCIY	нет	\$001	

Чтение модулей. Позволяет прочитать с поверхности дисков перечисленные выше модули. Модули считываются в подкаталоги "SGMOD_U4", "SGMOD_U8", "SGMODU10", "SGMOD_U6" (в соответствии с выбранной моделью) каталога PC3000 в файлы с расширением grm. Не все модули используются в разных моделях, и при их чтении возможны ошибки.

Запись модулей. Накопители серии Uxx позволяют записать не все прочитанные модули. Это связано в частности с тем, что часть модулей генерируется самим накопителем и не может быть изменена извне. При записи модулей Вы можете произвести пересчет их КС. Не рекомендуется производить пересчет КС для всех модулей подряд, т.к. в некоторых модулях она не предусмотрена.

Выбор копии для чтения. Позволяет выбрать, с какой копией модулей будет манипулировать утилита. Возможны варианты: копия 0, любая копия (чтение в стиле предыдущих версий, управление копиями осуществляет сам накопитель), выбрать (можно задать номер копии от 0 до 15).

ВНИМАНИЕ! для накопителей семейства U6 данная функция не реализована в связи с ошибкой в микропрограмме, неверно осуществляющей чтения большинства модулей при выборе, отличном от "любая копия".

Подсистема безопасности. Выводит меню, содержащее пункты

Просмотр информации о паролях
Очистка паролей

Просмотр информации о паролях. Позволяет просмотреть USER-пароль и текущий уровень безопасности.

Очистка паролей. Позволяет сбросить USER-пароль и задать уровень безопасности high.

3.2. Паспорт диска.

Как уже отмечалось выше, данный пункт позволяет модифицировать элементы паспортной информации накопителя. Модификация производится по IDE через так называемый SetStuff интерфейс. Следовательно, для работы этого пункта необходим выход в готовность накопителя по IDE. В случае, когда накопитель не выходит в готовность по причине некорректности модуля шаблона паспорта (ошибка "STUFF NOT READABLE", выводимая в терминал), необходимо применить команду терминала "инициализация паспорта" (см. ниже), после чего либо переписать модуль AT_STUFF, либо осуществить прямое редактирование параметров паспорта. В связи с тем, что назначение не всех элементов расширенной части паспорта нам известно, рекомендуется именно перезаписать модуль от соответствующей модели, после чего произвести необходимые вам правки.

3.3. Таблица S.M.A.R.T.

Здесь производится соответственно просмотр и сброс в значения по умолчанию данных S.M.A.R.T. Манипуляции производятся по IDE интерфейсу, поэтому для их выполнения необходим выход накопителя в готовность по IDE. В случае, если накопитель не выходит в готовность по IDE по причине ошибок в S.M.A.R.T. подсистеме, необходимо выполнить из терминала команду “инициализация S.M.A.R.T.” (см. ниже).

3.4. Таблица дефектов.

Данный пункт меню реализует манипуляции по IDE интерфейсу, такие, как очистка G-List, очистка P-List, пересчет транслятора. Следовательно, для их функционирования необходим выход накопителя в готовность по IDE. В случае, если накопитель не выходит в готовность, эквивалентные действия можно произвести из терминала, используя группу команд управления дефектами.

4. Работа через терминал.

Данный пункт меню реализует стандартный терминал утилиты pc-seagat, расширенный некоторыми дополнительными функциями, такими, как трансляция ESC-последовательностей управления терминалом Seagate (построение графиков в текстовом режиме, управление цветом символов и т.д.), считывание и запись “ключевых” модулей в потоковом режиме через COM порт и т.д.

Для работы с накопителем, в частности, для работы с модулями и ПЗУ используются его терминальные команды. Команды меню утилиты являются подчиненными.

Для работы необходимо перейти в командный режим. Для этого нужно нажать комбинацию клавиш [Ctrl][Z], после чего накопитель выдаст приглашение “T>”. Затем можно вводить собственно команды накопителя и повторить операцию.

4.1. Службное меню терминала.

Службное меню терминала может быть вызвано по кнопке [F1]. Используя его, можно не только вызывать соответствующие команды, но и узнать состояние терминала (COM порт, скорость и пр.) Приведем ниже пояснения к его пунктам:

“(F2) Выбор COM порта (COM1)” – здесь можно выбрать активный COM порт, а также увидеть (в правых круглых скобках) текущий выбранный порт.

“(F3) Очистка окна” – очистка окна терминала

“(F4) Скорость терминала (9600)” – здесь можно выбрать скорость работы терминала (9600, 19200, 38400, 57600, 115200) и узнать текущую скорость работы.

“(F5) Лог” – здесь можно включить/выключить запись вывода терминала в лог-файл, а также посмотреть, ведется ли сейчас запись и в какой файл. В случае, если запись ведется, справа в круглых скобках будет выведено имя файла.

“(F6) Отсечка” – если данный пункт включен, строки длиннее 80 символов обрезаются, если - нет – следующие за 80-м символы переносятся на новую строку.

“(F7) Файл” – содержит пункты “Принять файл” и “Отправить файл”. См. описание к командам “u”, “d”, “f”.

“(F10) Закрыть терминал”. Завершение работы с терминалом и возврат в основное меню утилиты (IDE режим)

4.2. Используемые команды терминала.

ВНИМАНИЕ! Команды, используемые в ТМОС накопителей, чувствительны к регистру символов.

Вначале подробнее опишем некоторые особенно часто используемые команды.

“Vxxxx” – установить скорость работы COM порта накопителя (Vaudrate),

где xxxx – определенным образом заданная скорость работы COM порта в бит/с (см. таблицу). Сразу после ввода этой команды нужно переключить терминал утилиты на соответствующую скорость (кнопка [F4]). Не все скорости поддерживаются различными накопителями, в частности, 115200 поддерживается U8 и не поддерживается U4. Кроме того, необходимо учесть, что чем выше скорость, тем больше вероятность ошибки при передаче данных через COM интерфейс. Проверить корректность переключения можно любой командой, например, “.” (если при этом появится строка статуса, значит, скорость установлена верно).

“**ухх,уу**” – считать модуль или ПЗУ с накопителя (**upload**), где **хх** – ID модуля в шестнадцатеричном виде (см. табл. 3,4,5); кроме того, ID “00” соответствует работе с Flash-ПЗУ. **уу** – требуемая копия модуля (тоже в шестнадцатеричном виде), например “01”, “02”, “03” и т.д., причём значение копии “ff” – любая считавшаяся копия.

ВНИМАНИЕ! При чтении ПЗУ эта команда возвращает его образ во внутреннем формате, отличном от того, какой получается при чтении программатором. Кроме того, накопители U6 возвращают ПЗУ неверно (см выше).

“**dxх**” – записать модуль в накопитель (**download**), где **хх** – шестнадцатеричный ID модуля. **ВНИМАНИЕ!** Эта команда не предназначена для записи Flash-ПЗУ. Введя эту команду, необходимо дождаться готовности накопителя принять модуль. Для этого можно использовать команду “.” (как только накопитель перестанет отвечать на неё, можно приступать к собственно записи).

“**fxххх,уууу**” – запись Flash-ПЗУ (**flash**), где “**хххх**” и “**уууу**” – внутренние параметры накопителя – ASW и Fallback ASW – коды алгоритмов записи. Их можно просмотреть, используя команду “показать информацию о ПЗУ” (см. ниже). Рекомендуется для записи использовать команду без параметров. В этом случае алгоритм запишет Flash согласно алгоритма по умолчанию.

ВНИМАНИЕ! При ошибке записи накопитель войдет в состояние ожидания и будет выдавать в терминал сообщение об ошибке. В этом случае будет необходимо перезаписать ПЗУ, используя программатор.

В случае же корректной записи через некоторое время накопитель выдаст диагностическое сообщение и раскрутит шпиндель.

После любой операции передачи данных, проведенной на скорости более 9600 бит/с, накопитель устанавливает скорость своего СОМ порта в 9600. В случае успешного завершения операции скорость терминала утилиты автоматически устанавливается в 9600. Если все операции приема и передачи проводить на скорости 9600 бит/с, то дополнительно переключать скорость накопителя и терминала не требуется.

4.2.1. Получение модуля или Flash-ПЗУ из накопителя.

Перед операцией получения модуля или ПЗУ можно установить скорость получения данных (команда накопителя “**Vхххх**” и меню скорости работы терминала).

Процесс считывания выполняется в два этапа. После подачи команды “**ухх,уу**” указанный модуль считывается в буферное ОЗУ накопителя. По завершении этого процесса на экран терминала будет выведен символ псевдографики “**||**”. После этого необходимо в меню “Файл” (кнопка [F7]) выбрать пункт “Принять файл”. Считывание производится в текущий каталог PC3000 в файл с расширением **dmp**. При этом в окно терминала будут выводиться номера блоков, возвращаемых накопителем, и их контрольные суммы (сначала КС, вычисленная утилитой на блоке модуля, затем, в скобках, КС, переданная накопителем для проверки).

Если модуль получен успешно, в окно терминала выводятся контрольные суммы всего модуля, подсчитанные утилитой и самим накопителем. Они должны совпадать.

Необходимо учесть, что по СОМ интерфейсу накопитель возвращает не все модули.

Если при передаче блока через СОМ порт произошла ошибка КС (несовпадение КС на экране, мигает курсор терминала и ввод символов невозможен), то прием данных утилитой прерывается. Сама передача данных накопителем оказывается незавершенной. В этом случае необходимо вывести накопитель из цикла ожидания. Это можно сделать либо переключив питание накопителя (в этом случае необходимо вручную переключить скорость терминала на 9600 бит/с), либо дочитав оставшиеся блоки во временный файл.

- В случае переключения питания необходимо дождаться окончания процесса инициализации, перевести накопитель в командный режим (см. выше).
- Для того, чтобы произвести дочитывание во временный файл, в терминал вводится комбинация клавиш [Ctrl-F] (ASCII 06h). В случае, если накопитель ответил “**||**”, мы можем собственно дочитать остаток модуля, выбрав в меню “Файл” (кнопка [F7]) пункт “Принять файл”. Естественно, блок, в котором произошла ошибка, будет потерян.

Далее необходимо повторить процедуру чтения модуля на той же или меньшей скорости передачи данных.

4.2.2. Запись модуля или Flash-ПЗУ в накопитель.

ВНИМАНИЕ! Данная команда оперирует с образом ПЗУ, отличным от получаемого программатором.

Перед операцией записи модуля или ПЗУ можно установить скорость получения данных (команда накопителя "Вхххх" и меню скорости работы терминала).

После ввода соответствующей команды ("dхх"-для записи модуля и "fхххх,уууу"- для записи Flash-ПЗУ) и выхода накопителя в готовность к передаче данных (см. выше описание к команде "dхх") необходимо в меню "Файл" (кнопка [F7]) выбрать пункт "Отправить файл". По мере записи будет выводиться КС текущего блока.

После передачи данных модуля необходимо подождать некоторое время, необходимое накопителю для операции записи. Завершив запись, он выдаст либо "Goodcopy=хх" и раскрутит шпиндель, либо сообщение об ошибке.

Если при передаче накопителем будет обнаружена ошибка, он перейдет в режим ожидания (при этом, через некоторое время "подвисания" терминала – 5секунд – курсор начнет снова мигать, но ввод команд невозможен). В этом случае необходимо переключить питание накопителя и после его инициализации продолжить работу.

ВНИМАНИЕ! В случае сбоя при записи ПЗУ может потребоваться его перезапись на программаторе.

4.2.3. Таблица команд.

В данном разделе приведены команды операционной системы накопителя – известной как MOS в фирменной документации или ТМОС в нашей документации предыдущих версий.

4.2.3.1. On-Line команды

[Ctrl]+[C]	Firmware Reset
[Ctrl]+[D] или [Ctrl]+[N]	Переключатель степени подробности вывода отчета о действиях микропрограммы
[Ctrl]+[L]	Вывод информационной строки, включающей версию микропрограммы.
[Ctrl]+[O] или [Ctrl]+[I]	Advance Servo Tracing – вывод подробных сведений о PCHS IO
[Ctrl]+[T]	Перезапуск тестов на текущем уровне.
[Ctrl]+[U]	Вывод дампа AT Stuff
[Ctrl]+[Z]	Переход в командный режим.
.	Отобразить статус работы накопителя.
;	Отобразить статус микропрограммы.
Esc	Прервать выполнение

4.2.3.2. Форматы откликов.

Команда ".".

Формат возвращаемых данных:

Pgm=aa Trk=bbbb(cccc).d.eee Zn=f Err=gg ErCt=hhhh Hlth=iiii Chlth=jjjj Ready

Pgm=aa	Активная программа. 00 – диагностический монитор. 50 – программа обработки команд АТА интерфейса.
Trk=bbbb(cccc).d.eee	Текущий активный логический цилиндр (физический цилиндр), головка, сектор.
Zn=f	Зона
Err=gg	Код ошибки текущей операции
ErCt=hhhh	Количество ошибок с последнего Reset-а накопителя или последней команды сброса лога ошибок
Hlth=iiii	Аккумулярованное состояние health битов (4 цифры)
CHlth=jjjj	Текущее состояние health битов (4 цифры)
Ready	Статус накопителя. Может быть Ready или Ntrdy

Пример:

Pgm=50 Trk=5F61(5F61).0.022 Zn=6 Err=00 ErCt=0000 Hlth=0000 CHlth=0000 Ready

Команда ";".

Формат возвращаемых данных:

Age=xx Type=xx MxCyl=xxxx MxHd=x MxSct=xxx Bsz=xx TCode=xxxx

Age=aa	Текущий уровень
Type=bb	Текущий тип накопителя
MxCyl=cccc	Максимальное количество цилиндров для данного накопителя, hex
MxHd=d	Максимальное количество головок для данного накопителя, hex
MxSct=eee	Максимальное количество секторов для данного накопителя, hex
BSz=ffff	Размер единичного буфера, hex
Tcode=gggg	Код теста.

Пример:

Age=50 Type=E0 MxCyl=D993 MxHd=0 MxSct=300 BSz=0800 TCode=0000

4.2.3.3. Расшифровка битов health.

Байт health имеет следующий формат (два слова в hex представлении) : X X X X . Y Y Y Y.

Первые 4 тетрады – текущее состояние health, следующие 4 тетрады – аккумулированное состояние health. Т.е. X X X X – текущее состояние health, Y Y Y Y – накапливаемое состояние health (принудительно его можно сбросить командой **H** на уровне **T**).

номер бита	если бит установлен в 1, то
15	Чрезмерное количество пропусков или Alt-замен
14	Аппаратная ошибка
13	Серьезная R/W ошибка
12	Error Rate Health
11	Offtrack ошибка
10	Ошибка позиционера
9	Серво ошибка
8	Ошибка вращения
7	внутреннее использование
6	внутреннее использование
5	внутреннее использование
4	внутреннее использование
3	R/W предупреждение
2	Невозможно произвести Assign по пропуску или Alt-замене.
1	Предупреждение тока мотора
0	Ошибка вращения при позиционировании

4.2.3.4. Общие команды.

/x	Переход на уровень x
----	----------------------

4.2.3.5. Уровень T (уровень 0).

Vxx	Установить скорость работы COM порта HDD xx – код скорости V1152 - 115200 V625 - 62500 V576 - 57600 V192 - 19200 V96 - 9600 V48 - 4800 V1 - показать список поддерживаемых скоростей
Dx,y,z	Показать CERT логи, начиная с "x". Если указано "y" – показывать только записи с кодом ошибки "y". Если "y" = AA разрешить параметр "z". "z" – Специальная функция. Если "z" = 40 – быстрый вывод
F	Управление паспортом накопителя

	<p>редактирование идет в командной строке SetStuff-> формат вводимых данных ASCIххуууу хх – номер слова в паспорте, соответствующий выбранному параметру. уууу – данные, передаваемые в параметр. Данные должны вводиться в шестнадцатеричном формате, пословно, без пробелов, в порядке расположения в паспорте. Управление параметрами идёт отдельно : на каждый параметр одна команда F. Примеры: "ASCI013FFF" – установить количество логических цилиндров в 0x3FFF "ASCI030016" – установить количество логических головок в 0x16 "ASCI060063" – установить количество логических секторов в 0x63 "ASCI3C45670123" – установить MaxLBA в 0x01234567 "ASCI8345670123" – установить ограничитель MaxLBA в 0x01234567 "ASCIFE" – инициализация паспорта значениями по умолчанию</p>
Gx	Читать сектор из трека VENDOR data. "x" – номер сектора.
Hx,y,z	<p>Показать/сбросить биты health "x" = любой символ – показать текущий health иначе - показать аккумулярованные значения health "y" = не используется "z" = 22 – произвести сброс health</p>
Nx	Установить уровень накопителя (Age) в "x". (Пр: N50 – перевод HDD в уровень 50 – режим работы по АТА)
R	Считать системный сектор (System Sector) в ОЗУ из системной области поверхности (System Sector в том числе содержит адаптивы)
Tx	Запустить заводской тест "x"
Wx,y,z	<p>Сохранить системный сектор (System Sector) из ОЗУ в системную область поверхности (System Sector в том числе содержит адаптивы) Если Age <> 60 – параметры не используются иначе: "x", "y" – не используются, "z" должно быть равно 22 для сброса адаптивов.</p>
Yx	<p>Установить тип накопителя (управление количеством головок). "x" = E0 – одна головка, E1 – две головки, E2 – три головки.... после команды "Y" для сохранения изменений должна быть выполнена команда "W"</p>
d	записать модуль в накопитель (d ownload)
fxxxx,yuuu	запись Flash
r	отобразить версию CERT
u	считать модуль или ПЗУ с накопителя (u pload)
#	Ввести серийный номер HAD

4.2.3.6. Уровень 1.

M	<p>Показать информацию о ПЗУ. Показывает код Flash ROM, код производителя, пользовательский код алгоритма записи, аварийный код алгоритма записи. После выполнения перезагружает</p>
---	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

	микропрограмму.
Nx	Управление SMART и, частично, дефект листами "x" = 0 – переключение SAMRT On/Off 1, 4 – сброс SMART 5 – показать атрибуты SMART 6 – показать SMART Thresholds 7 – показать G-List 8 – показать лог критических событий 9 – показать P-List
\$	установить информацию о PCB и обновить Flash
#	показать информацию о PCB

4.2.3.7. Уровень 2.

U	Раскрутить шпиндель.
Z	Остановить шпиндель
x	Показать зонное распределение

4.2.3.8. Уровень 9.

Fxx,y	Добавка/удаление дефектов. Fxx,A - добавить дефект в G-List (reassign) Fxx,B - добавить дефект в G-List (pending – candidate) F0,C - очистить G-List Fxx,D,yy - добавить yy дефектов начиная с сектора xx в P-List F0,E - инициализировать таблицы дефектов F0,D3 - пересчёт транслятора F0,FF - перезагрузка накопителя
-------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

5. Коды ошибок HDD.

При обнаружении ошибки накопитель выводит ее код и краткую дополнительную информацию в терминал.
Пример: "Code - FE Track 19A1.3.06E Sns E00 Rty F7FF.FF.80FF Rtf 1C00" – ошибка "FE".

Приведем расшифровку кодов ошибок:

00h	No error
03h	Bad ROM CRC
10h	RRO fail ZAP code
11h	Spin error
12h	Oscillatory wavy PES
13h	Offtrack
14h	Write fault
15h	Velocity PES harmonics
16h	Seek timeout
17h	'false' AMDET (mis-timed)
18h	Bad burst error code
19h	Bad grey code
1ah	Early sync code
1bh	Missed am
1ch	Failed the servo defect screen threshold
1dh	Time out waiting for servo ready (approx. 4sec)
1eh	Target generator sector error
1fh	Physical sector error
20h	RWF init error
21h	RWF file incompatible with Current ROM code
30h	Data miscompare

31h	Data miscompare on read longs
36h	Pending block encountered during a write
37h	Pending block encountered and read successfully
38h	Pending block encountered and read failed
40h	Id not found
41h	Long ECC error correction code
42h	Missed servo sector mark
43h	Dataecc error
44h	Fifo over/underrun
45h	Track ID miscompare error
46h	Track ID read TO code
47h	Missed data sync byte (1 st data fragment)
48h	Missed data sync byte (after splitted fragment)
49h	Thermal asperity code
4ah	Write integrity error
4bh	Error log full
4ch	Controller stopped without a reason
4dh	Sd counter not the expected value
4eh	Block release timeout error
4fh	Servo assist error
50h	Skipped sector
51h	Alternated sector
52h	Pad and fill defects
53h	Pad/fill is alt dst, reall source
54h	Error during scratchfill
55h	Bad alt dest ID info
56h	Unable to find valid alt dest
57h	Unable to delete alt dest
58h	Too many defects per head
59h	Too many defects per track
5ah	Too many defects per drive
5bh	Defect table full
5ch	Slips overflowed
5dh	Invalid defect entry
5eh	Padding of data region of zapped burst
60h	No spares available
73h	Unable to read defect sectors code
61h	Write blocked w/o a reason
62h	Write gate came on early
63h	Write gate went off late
64h	GC thermal sensor fault detected
65h	Shock sensor fault detected
66h	Scratch ram protection error
70h	Bad disk resident cert/diag code
71h	Unable to read system sectors code
72h	Unable to read packwriter auto-stamp
83h	Invalid command
8dh	Cylinder filled in
8eh	Cylinder skipped out
8fh	Max cylinder skips has been reached
90h	Zapped burst
91h	2 consecutive zapped bursts
92h	Too many zapped bursts on a track
93h	Unable to zap - 2 consecutive bad bursts
94h	Unable to zap - too many bad bursts on a track
95h	Unable to zap bad burst

97h	Couldn't find burst
98h	Timeout waiting for r/w
A0h	Preamplifier miscommunication
C1h	Wrong major/minor version error
C2h	Wrong engr version error
C3h	File size mismatch error
C4h	DLE checksum = 0 error
C5h	AutoDLE upload error
C6h	Wrong overlay flag data in byte 8,9
C7h	Read1Cp checksum=0 error
C8h	Exceed max key defined for DLE
C9h	Error during reading reserved
Cah	Error comparing Signature
Cbh	Defect map sequence error
Fbh	Defect unspared
Fch	Command aborted (internal only)
Fdh	R/W command rejected
Feh	Invalid address passed to Niwot

ООО НПП «АСЕ»
ТОЛЬКО ДЛЯ ОФИЦИАЛЬНЫХ ПОЛЬЗОВАТЕЛЕЙ