

**Working
Draft**

**T13
1701DT-N
Revision 1
August 16, 2004**

Information Technology - SMART Command Transport (SCT)

This is an internal technical report of T13, a Technical Committee of Accredited Standards Committee NCITS. As such, this is not a completed technical report and has not been approved. The contents may be modified by the T13 Technical Committee. The contents are actively being modified by T13. This technical report is made available for review and comment only.

Permission is granted to members of NCITS, its technical committees, and their associated task groups to reproduce this technical report for the purposes of NCITS standardization activities without further permission, provided this notice is included. All other rights are reserved. Any commercial or for-profit replication or republication is prohibited.

ASC T13 Technical Editor:

Curtis E. Stevens
Western Digital Technologies Inc.
20511 Lake Forest Dr.
Lake Forest, Ca. 92630

Tel: 949-672-7933
E-Mail: Curtis.Stevens@wdc.com

Reference number
ANSI TR INCITS.*** - 200x
Printed August, 16, 2004 5:51PM

T13/1701DT Revision 1

Other Points of Contact:

T13 Chair
Dan Colegrove
Hitachi Global Storage Technology Inc.
2505 Anthem Village Dr. Ste. E-221
Henderson, NV, 89052
Phone: 702-614-6119
Fax: 702-614-7955
E-Mail: Daniel.colegrove@HitachiGST.com

T13 Vice-Chair
Jim Hatfield
Seagate Technology
389 Disc Drive
Longmont, CO 80503
Tel: 720-684-2120
Fax: 720-684-2711
E-mail: James.C.Hatfield@seagate.com

INCITS Secretariat
Administrator Standards Processing
1250 Eye Street, NW Suite 200
Washington, DC 20005
Tel: 202-737-8888
Fax: 202-638-4922
Email: INCITS@ITIC.ORG

T13 Reflector

See the T13 Web Site at <http://www.t13.org> for reflector information.

T13 Web Site

<http://www.t13.org>

T13 Anonymous FTP Site

<ftp.t13.org>

T13 mailings

Global Engineering
15 Inverness Way East
Englewood, CO 80112-5704
Tel: 303-792-2181 or 800-854-7179
Fax: 303-792-2192

Revision History		
Rev	Date	Description
0	June 14, 2004	Initial Release
1	August 16, 2004	<ol style="list-style-type: none">1. Changed from CHS registers to LBA Low/Mid/High2. Updated formatting3. Changed notation from 0x to xxxxh4. Created an overview section5. Added a section describing resets6. Added a section describing functionality when various commands are supported/not supported vs enabled/disabled7. Added General Purpose Logging command samples8. Added Extended Error Codes9. Updated Glossary to include TF DATA and Block Data10. This document does not document the "Super Command"11. Clarified the temperature sensor to be vendor specific

American National Standard
for Information Systems —

SMART Command Transport (SCT)

Secretariat
Information Technology Industry Council

Approved mm dd yy

American National Standards Institute, Inc.

Abstract

This technical report describes a method for transporting commands to an ATA device using SMART log pages.

NCITS's Technical Report Series

This Technical Report is one of a series produced by the American National Standards Committee, NCITS, Information Technology. The secretariat for NCITS is held by Information Technology Industry Council (ITI), 1250 Eye Street, NW Suite 200, Washington DC 20005. As a by-product of the standards development process and the resources of knowledge devoted to it, NCITS from time to time produces Technical Reports. Such Technical Reports are not standards, nor are they intended to be used as such.

NCITS Technical Reports are produced in some cases to disseminate the technical and logical concepts reflected in standards already published or under development. In other cases, they derive from studies in areas where it is found premature to develop a standard due to still changing technology, or inappropriate to develop a rigorous standard due to the existence of a number of viable options, the choice of which depends on the users particular requirements. These Technical Reports, thus, provide guidelines, the use of which may result in greater consistency and coherence of information processing systems.

When the draft Technical Report is completed, the Technical Committee approval process is the same as for a draft standard. Processing by NCITS is also similar to that for a draft standard.

CAUTION: The developers of this Technical Report have requested that the holder's of patents that may be required for the implementation of this Technical Report, disclose such patents to the publisher. However, neither the developers nor the publisher have undertaken a patent search in order to identify which, if any, patents may apply to this Technical Report.

As of the date of this Technical Report and the following calls for the identification of patents that may be required for the implementation of the Technical Report, no such claims have been made. No further patent search is conducted by the developer or the publisher in respect to any Technical Report it Processes. No representation is made or implied that licenses are not required to avoid infringement in the use of this Technical Report.

Published by
American National Standards Institute
11 West 42nd Street, New York, New York 10036

Copyright 200x by American National Standards Institute
All rights reserved.

Contents	Page
FOREWORD	IV
INTRODUCTION	VI
1 SCOPE.....	1
2 DEFINITIONS AND ABBREVIATIONS.....	1
2.1 TF DATA	1
2.2 BLOCK DATA.....	1
3 OVERVIEW.....	1
4 COMMAND TRANSPORT	2
4.1 CAPABILITY IDENTIFICATION.....	3
4.2 COMMAND TRANSPORT	3
4.3 DATA TRANSFER	6
4.4 SCT STATUS REQUEST	7
4.5 HOW TO USE SCT PASS-THROUGH COMMANDS	11
4.5.1 <i>Issuing an SCT pass-thru command to the drive</i>	11
4.6 DRIVE ACCESS METHODS.....	13
4.6.1 <i>Logical Block Address</i>	13
4.7 SCT COMMAND NESTING AND INTERMINGLING WITH STANDARD COMMANDS	13
4.8 RESETS	13
4.9 SMART AND GENERAL PURPOSE LOGGING SUPPORT.....	14
5 SCT COMMAND SET.....	14
5.1 LONG SECTOR ACCESS COMMAND.....	15
5.2 LBA SEGMENT ACCESS COMMAND.....	16
5.3 ERROR RECOVERY CONTROL COMMAND	16
5.4 FEATURE CONTROL COMMAND.....	17
5.5 SCT DATA TABLE COMMAND	18

Tables	Page
TABLE 1– IDENTIFY DEVICE WORD	3
TABLE 2– ISSUE SCT COMMAND USING SMART.....	4
TABLE 3– ISSUE SCT COMMAND USING GPL.....	4
TABLE 4 - SUCCESSFUL SCT COMMAND RESPONSE	4
TABLE 5- SCT COMMAND ERROR RESPONSE.....	5
TABLE 6 – EXTENDED ERROR CODES.....	6
TABLE 7– READ\WRITE SCT DATA USING SMART	7
TABLE 8– READ\WRITE SCT DATA USING GPL	7
TABLE 9– SCT STATUS REQUEST USING SMART	8
TABLE 10– SCT STATUS REQUEST USING GPL	8
TABLE 11 – FORMAT OF SCT STATUS RESPONSE.....	9
TABLE 12 – SCT COMMAND USING SMART	11
TABLE 13–SCT COMMAND USING GPL.....	11
TABLE 14– TRANSFER DATA TO HOST USING SMART.....	12
TABLE 15– TRANSFER DATA TO HOST USING GPL.....	12
TABLE 16– TRANSFER DATA TO THE DRIVE USING SMART	13
TABLE 17– TRANSFER DATA TO DRIVE USING GPL	13
TABLE 18 - KEY SECTOR FORMAT	14
TABLE 19 – SCT ACTION CODES	15
TABLE 20 – LONG SECTOR FORMAT.....	15
TABLE 21 – FEATURE CODE LIST	18
TABLE 22 – SCT DATA TABLES (BY TABLE IDENTIFIER).....	19
TABLE 23 – FAN CONTROL TEMPERATURE HISTORY TABLE FORMAT.....	20
TABLE 24 – HDA TEMPERATURE HISTORY TABLE FORMAT	21

Figures	Page
FIGURE 1– COMMAND DECODE.....	2

Foreword

(This foreword is not part of American National Standard INCITS.***-200x.)

This technical report documents a method for passing commands that are not currently documented in the ATA/ATAPI series of standard to a device using log pages. These log pages can be accessed as a part of SMART, or they may be accessed as a part of the READ/WRITE LOG EXT commands. The commands that are executed by this mechanism include:

1. Long Sector Access (Read/Write Long)
2. LBA Segment Access (Fill sectors or copy sectors)
3. Error Recovery Control (Limits the time for error recovery in READ and WRITE commands)
4. Feature Control (overrides write cache enable/disable)
5. SCT Data Table command (returns HDA Temperature and Fan Ctrl Temperature history)

This technical report was developed by the ATA ad hoc working group of T13 during 2004. The approval process started in 2004.

Requests for interpretation, suggestions for improvement and addenda, or defect reports are welcome. They should be sent to the NCITS Secretariat, Information Technology Industry Council, 1250 Eye Street, NW, Suite 200, Washington, DC 20005-3922.

This technical report was processed and approved for submittal to ANSI by Accredited Standards Committee on Information Processing Systems, INCITS. Committee approval of the technical does not necessarily imply that all committee members voted for approval. At the time it approved this technical, the NCITS Committee had the following members:

[Editors Note: INCITS officers and members go here]

Technical committee T13 on ATA Storage Interfaces, which reviewed this standard, had the following members:

[Editors Note: T13 officers and members go here]

Other ad hoc participants were:

[Editors Note: Table of on-member participants goes here]

Introduction

This technical report encompasses the following:

Clause 1 describes the scope.

Clause 2 provides definitions, abbreviations, and conventions used within this technical report.

Clause 3 provides an overview

Clause 4 describes the transport mechanism.

Clause 5 describes the SCT command set

American National Standard
for Information Systems —

Information Technology —
SMART Command Transport

1 Scope

This technical report assumes that the reader is familiar with the ATA/ATAPI standard, which include SMART, READ LOG EXT and WRITE LOG EXT.

ATA devices are increasingly used in bridged environments where the Parallel ATA or Serial ATA interface is bridged to a different kind of bus. Sometimes this bridging requires that commands be processed or translated before passing them on to the ATA device. If the bridge is not designed to pass a specific command, then there is no way to issue that command to an ATA device. This technical report defines a way to pass commands to an ATA device using SMART or extended logs. By passing the commands through a log write, the bridge sees a write log command. The ATA device can then parse the log entry and execute the intended command.

2 Definitions and abbreviations

2.1 TF Data

TF Data is the number of error correction and detection bytes returned on read and write commands

2.2 Block Data

Block Data is the data returned by an SCT command.

3 Overview

The basis of this technical report for issuing commands to a drive is log pages E0h and E1h. These log pages are used as follows:

	Log Page E0	Log Page E1
Send log page	Issue Command	Send data
Retrieve log Page	Return Status	Receive data

There are 2 ways to access the log pages: using SMART READ/WRITE LOG and READ/WRITE LOG EXT. Both sets of commands access the same log pages and provide the same capabilities. They are also used in the same way: A command is issued, data is transferred (if necessary), and status is retrieved.

The commands that are defined in SCT are subject to requirements documented in ATA/ATAPI-7. This means that if security is enabled and a password has not been issued to unlock that device that media access commands will fail.

Figure 1 is a sample flowchart that shows how to process SCT log page requests using SMART:

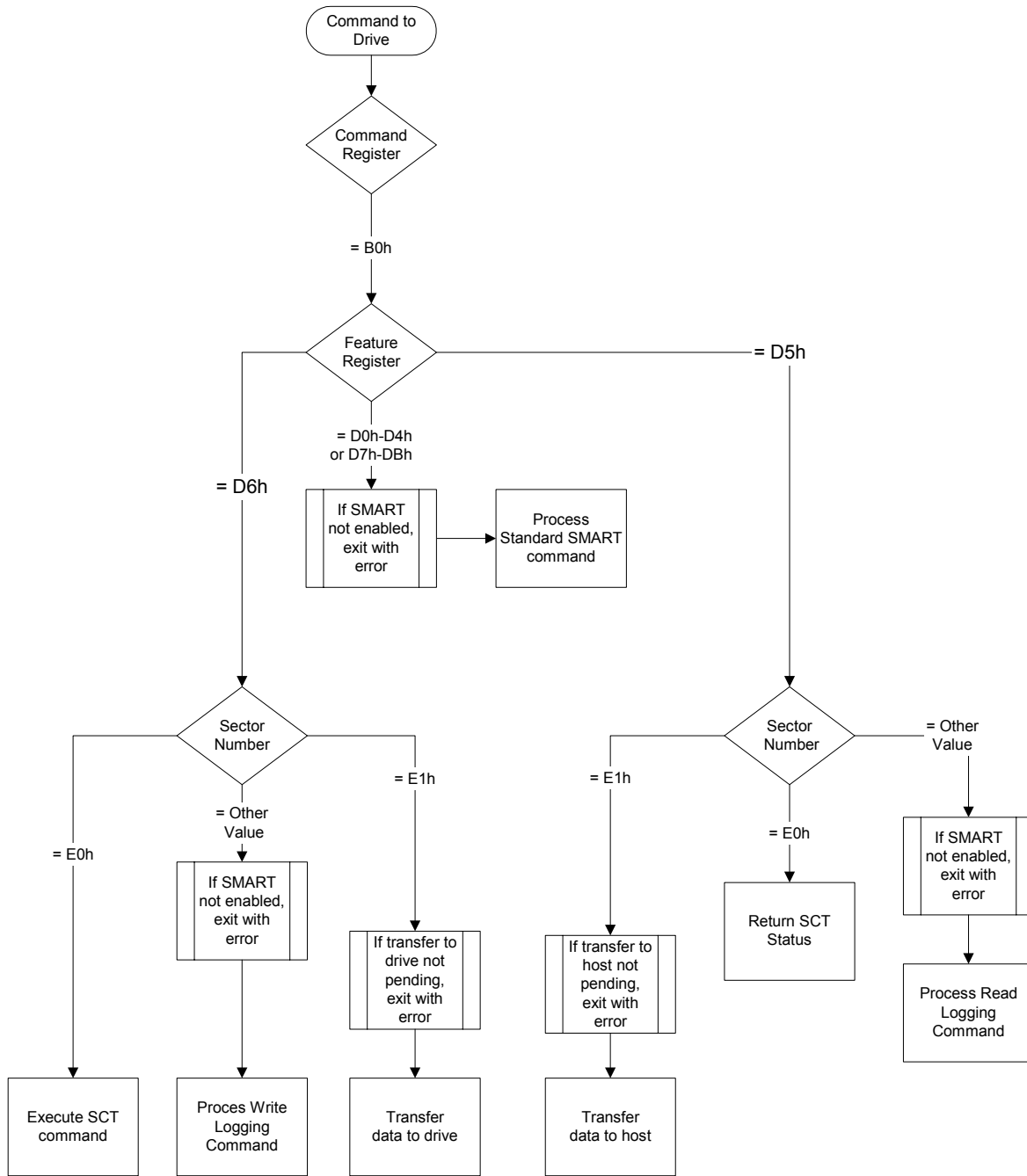


Figure 1– Command Decode

4 Command Transport

There are several phases involved in the issuing and executing of SCT commands. These phases are:

1. Capability Identification
2. Command Transport

3. Data Transfer
4. Status

Capability Identification is performed by issuing IDENTIFY DEVICE and checking to make sure the command is supported. Command Transport occurs when a 512 byte data packet is created and then written to SMART or extended log page E0h. The 512 byte data packet contains one or more commands as documented later in this report. If the command was a data transfer command then the data can be transferred by reading or writing log page E1h. If the data is larger than a single log page, the page is read or written multiple times. Finally, command status can be read at anytime by reading log page E0h. If the command involves data transfer, the host will need to check status before data is transferred to ensure that the drive is ready, as well as when the command is complete, to confirm that the data was transferred successfully. When the command is complete, the host may need to check status a third time to determine if the command succeeded, failed, or partially succeeded.

4.1 Capability Identification

The ATA IDENTIFY DEVICE command is used to determine if SCT is enabled and which SCT Action Codes are supported.

Table 1– IDENTIFY DEVICE Word

Word	O/M	F/V	Description	
206	M		SCT Command set support.	
		X	15 - 6	Reserved
		F	5	SCT Data Table (AC5) supported
		F	4	Features Control command (AC4) supported
		F	3	Error Recovery Control (AC3) supported
		F	2	LBA Segment Access (AC2) supported
		F	1	Long Sector Access (AC1) supported
		F	0	SCT command set supported (includes SCT status)

4.2 Command Transport

The command transport uses log sectors to pass-thru commands as well as inquire about status and control data flow. Two log pages have been allocated to support sending commands, receiving data, and checking status. Log page E0h is used for control and status. Log page E1h is used for reading or writing data. The command process is started by sending a “key” sector to log page E0h (the Control Port). The key sector informs the drive of the Action to be taken, any special mode of operation required for that action, and any other action-specific parameters. The log pages can be accessed by SMART READ/WRITE LOG (See Table 2), or by the General Purpose Logging commands (GPL) READ LOG EXT and WRITE LOG EXT (See Table 3). The command successful response (See Table 4) and the error response (See Table 5) is the same for both the SMART and the GPL methods of issuing command.

Table 2– Issue SCT Command Using SMART

Register	Value
Features	D6h (SMART Write Log subcommand code)
Sector Count	01h (Must be used for all pass-thru commands)
LBA Low	E0h (Must be used for all pass-thru commands)
LBA Mid	4Fh (Smart Enable Code)
LBA High	C2h (Smart Enable Code)
Command	B0h (SMART)

Table 3– Issue SCT Command Using GPL

Register	Value
Features	Current Previous Reserved Reserved
Sector Count	Current Previous 01h 1 sector for pass-through commands 00h
LBA Low	Current Previous E0h Reserved
LBA Mid	Current Previous 00h There is no offset when commands are issued 00h
LBA High	Current Previous Reserved Reserved
Command	3Fh (WRITE LOG EXT)

Table 4 - Successful SCT Command Response

Register	Value
Error	00h
Sector Count	Depends on command (LSB)
LBA Low	Depends on command (MSB)
LBA Mid	Low 8 bits of number of sectors available
LBA High	High 8 bits of number of sectors available
Status	50h

Table 5- SCT Command Error Response

Register	Value
Error	04h
Sector Count	Extended Error code, LSB (See Table 6)
LBA Low	Extended Error code, MSB (See Table 6)
LBA Mid	Low 8 bits of Number of sectors of data available - up to and including the failed sector
LBA High	High 8 bits of Number of sectors of data available - up to and including the failed sector
Status	51h

Table 6 – Extended Error Codes

Error Code	Definition
0000h	Command complete without error
0001h	Invalid Function code in LBA Segment Access command
0002h	Input LBA out of range
0003h	Request sector count overflow. The number of sectors requested to transfer (Sector Count register) in the read or write log command is larger than required by SCT command
0004h	Invalid Function code in Error Recovery command
0005h	Invalid Selection code in Error recovery command
0006h	Host read command timer is less than minimum value
0007h	Host write command timer is less than minimum value
0008h	Background SCT command was aborted interrupting host command.
0009h	Background SCT command was terminated because of unrecoverable servo error
000Ah	Invalid Function code in Long Sector access command
000Bh	SCT data transfer command (SMART Read log or SMART write log) was issued without first issuing SCT control command
000Ch	Invalid Function code in Feature Control command
000Dh	Invalid Feature code in Feature Control command
000Eh	Invalid New State value in Feature Control command
000Fh	Invalid Option Flags value in Feature Control command
0010h	Invalid SCT Action code
0011h	Invalid Table Id (table not supported)
0012h-BFFFh	.Reserved
C000h-FFEFh	Vendor specific
FFF0h-FFFEh	Reserved
FFFFh	SCT command executing in background

4.3 Data Transfer

Once an SCT command has been issued, status can be checked and data can be transferred. Data transfer uses log page E1h. Up to 255 sectors of data can be transferred at a time. If the SCT Command requires more than 255 sectors of data transfer, it can be written or read in 255 sector increments. Table 7 shows how to setup the registers for data transfer using SMART. Table 8 shows how to setup the registers for data transfer using GPL.

Table 7– Read\Write SCT Data using SMART

Register	Value
Features	D5h\D6h (Read data\Write data)
Sector Count	Count (Number of sectors to be transferred)
LBA Low	E1h
LBA Mid	4Fh (SMART enable code)
LBA High	C2h (SMART enable code)
Command	B0h (SMART)

Table 8– Read\Write SCT Data using GPL

Register	Value
Features	Current Previous Reserved Reserved
Sector Count	Current Previous Count (7:0) Count (15:8)
LBA Low	Current Previous E1h Reserved
LBA Mid	Current Previous 00h There is no offset when data is retrieved 00h
LBA High	Current Previous Reserved Reserved
Command	2Fh (READ LOG EXT)/3Fh (WRITE LOG EXT)

4.4 SCT Status Request

Once an SCT command has been issued, a status is reported in the ATA registers. This status indicates that the command was accepted or that an error occurred. This return does not indicate successful completion of the SCT actions. Some commands can take several minutes or even hours to execute. In this case, the host can determine execution progress by requesting SCT status. Some commands may need setup time before they are ready to receive data. SCT status is also used to determine when the device is read to receive data.

Log page E0 contains the status information. Reading log page E0h retrieves the status information. The SCT status may be acquired any time the task file is not busy. This command will not change the power state of the drive, nor terminate any background activity, including any SCT command in progress. Table 9 shows how to setup the registers for retrieving status using SMART. Table 10 shows how to setup the registers for retrieving status using GPL. Table 11 shows the format of the status return information.

Table 9– SCT Status Request Using SMART

Register	Value
Features	D5h (SMART Read Log subcommand code)
Sector Count	01h (Must for all pass-thru commands)
LBA Low	E0h (Must for all pass-thru commands)
LBA Mid	4Fh (SMART enable code)
LBA High	C2h (SMART enable code)
Command	B0h (SMART op code)

Table 10– SCT Status Request Using GPL

Register		Value
Features	Current	Reserved
	Previous	Reserved
Sector Count	Current	01h 1 sector for pass-through status
	Previous	00h
LBA Low	Current	E1h
	Previous	Reserved
LBA Mid	Current	00h There is no offset when is received
	Previous	00h
LBA High	Current	Reserved
	Previous	Reserved
Command		2Fh (READ LOG EXT)

Table 11 – Format of SCT Status Response

Word	Size	Field Name	Description
0	1	Format Ver	0001h - Status Response format version number
1	1	SCT Version	Manufacturers implementation version number MSB = major version (Function change) LSB = minor version (non-functional change/correction)
2	1	SCT Spec.	0001h - Highest level of SCT specification supported
3	2	Status Flags	Bit 0: Drive Zeroed. If this bit is set to 1, an LBA Segment Accesses command write to all LBA's of the drive completed without error. This bit shall be cleared to 0 when any user LBA is written. It is also cleared if the capacity of the drive is change via HPA or DCO. This bit is persevered through a power cycle. Bits 1-31: Reserved
5	1	Drive State	0 = Idle 1 = Stand-by 2 = Sleep 3 = DST executing in background 4 = SMART Off-line Data Collection executing in background 5 = SCT command executing in background
6	1	reserved	00h
7	1	Extended Error Code	Error of last SCT command. FFFFh if SCT command executing in background (See Table 6).
8	1	Action Code	Action code of last SCT command
9	1	Function Code	Function Code of last SCT command
10	10	reserved	Reserved for later use
20	4	LBA	Current LBA of SCT command executing in background
24	76	reserved	00h
100	1	HDA Temp	Current drive HDA temperature in degrees Celsius
101	1	Max Temp	Maximum HDA temperature in degrees Celsius seen this power cycle
102	1	Life Max Temp	Maximum HDA temperature in degrees Celsius seen for the life of the drive

Word	Size	Field Name	Description
103	1	Fan Ctrl Temp	<p>FFh (255) = Indicates invalid Fan Ctrl Temp value</p> <p>0F0h (240) = Over range (drive operating above allowable temperature limits)</p> <p>0D0h (208) = drive operating at max allowable temperature limits. (Typical max allowable value is 55C)</p> <p>The following values are decimal:</p> <p>5 = 5C under the drive operating allowable temperature 10 = 10C under the drive operating allowable temperature 15 = 15C under the drive operating allowable temperature 20 = 20C under the drive operating allowable temperature 25 = 25C under the drive operating allowable temperature 30 = 30C under the drive operating allowable temperature 35 = 35C under the drive operating allowable temperature 40 = 40C under the drive operating allowable temperature 45 = 45C under the drive operating allowable temperature 50 = 50C under the drive operating allowable temperature 55 = 55C under the drive operating allowable temperature 60 = 60C under the drive operating allowable temperature 65 = 65C under the drive operating allowable temperature</p> <p>For example, if max allowable temperature is 55C and the current HDA Temp is 37C, Fan Ctrl Temp is computed by:</p> <p>$55 - 37 = 18$. Round up to next 5C increment.</p> <p>Yields: 20C or 14h</p> <p>The fan control ranges are product specific values selected to give optimal control of a fan that will provide airflow to the drive. The main purpose of having a value for every five degree Celsius delta is to allow the system to change the RPM of the fan at a finer resolution. This is to minimize the fan acoustic level changes going from one RPM level to the next.</p>
104-240	136	reserved	Shall be cleared to zero
241-255	16	Vendor Specific	Vendor Specific

4.5 How to use SCT Pass-Through Commands

SCT commands are referred to as “pass-through” commands because they piggy-back on standard ATA commands. In this case, SMART Read Log and SMART Write Log commands or READ LOG EXT and WRITE LOG EXT. As viewed through the eyes of a standard ATA protocol analyzer, the SCT command will be seen as data being transferred by these commands; whereas from the perspective of a disk drive that implements this commands set, this “data” would be interpreted as either an SCT command request, an SCT command response, SCT status or SCT command data.

4.5.1 Issuing an SCT pass-thru command to the drive

4.5.1.1 Step 1

Build the key sector in a host buffer for the appropriate action and parameters

4.5.1.2 Step 2

Issue the SCT command (Table 12 or Table 13), and send the key sector to the drive.

Table 12 – SCT Command Using SMART

Register	Value
Features	D6h (SMART Write Log subcommand code)
Sector Count	01h (Must always be 01h)
LBA Low	E0h (Command port)
LBA Mid	4Fh (SMART enable code)
LBA High	C2h (SMART enable code)
Command	B0h (SMART)

Table 13–SCT Command Using GPL

Register	Value
Features	Current Previous Reserved Reserved
Sector Count	Current Previous 01h 1 sector for pass-through commands 00h
LBA Low	Current Previous E0h Reserved
LBA Mid	Current Previous 00h There is no offset when commands are issued 00h
LBA High	Current Previous Reserved Reserved
Command	3Fh (WRITE LOG EXT)

The device should respond with a 50h status (see Table 4). If there is a command abort (Status = 51h and Error = 04h), then either the key sector format is invalid, the task file contains an invalid value or the command encountered an execution error. Check Sector Count, LBA Low registers for error code (see Table 5 and Table 6). If the command was a “write” command, the command is terminated and will not accept any data; skip step 3. However, if the command was a “read” command, there maybe partial output available.(For example, on a sector read command, the data up to and including the sector in error, will be available.) In this case, the host may choose to proceed to step 3 to get the partial data. In certain cases, the error is not fatal and serves only as a warning.

If the status is 50h, then check the LBA Mid and LBA High registers. If the values are 0, then the command is complete and terminated without error. If the values are greater than 0, then proceed to step 3.

4.5.1.3 Step 3

Now use the Read Log Sector (Table 14) to transfer data from the drive to the host and use Write Log Sector (Table 16) command to transfer data from the host to the drive. The Sector Count register must be set to the number of sectors to be transferred. The transfer request must be in the range of 1 sector up to the total number of sectors not yet transferred. The number of sectors available will have been posted in the LBA Mid and LBA High registers in the previous step. If the requested number of sectors is larger than the number of the sectors available, the drive will post an error. If the value is less than the number of sectors available, the Host may repeat Step 3 until all sectors have been transferred. The drive will auto advance the sector pointer by the number of sectors transferred, and report in the LBA Mid and LBA High registers the number of sectors remaining to be transferred (if both registers are zero then the command is done). In other words, the host has complete control over the number of sectors to transfer at a time. Note, if the number of sectors to be transferred is greater or equal to FFFFh, the drive will post FFFFh in the LBA Mid and High registers and the value will remain at FFFFh until the number of sectors remaining drops below FFFFh. The exact number to be transferred is reported by the SCT Status command.

4.5.1.4 Step 4

For key commands that require transfer of data to the drive (such as a write command), typically the command is not complete until the last block of data has been transferred to the drive. Upon receiving the last block of data, the drive will perform the specified operation (in this case, write the data to the disk). However, in the case of very large amounts of data, such as write all, some data may be processed (written to the disk) prior to receiving all of the data from the host. Table 14, Table 15, Table 16, and Table 17 show sample commands for transferring data.

Table 14– Transfer Data to Host Using SMART

Register	Value
Features	D5h (SMART Read Log subcommand code)
Sector Count	Count (Number of sectors to transfer)
LBA Low	E1h (Data port)
LBA Mid	4Fh (SMART enable code)
LBA High	C2h (SMART enable code)
Command	B0h (SMART)

Table 15– Transfer Data to Host Using GPL

Register		Value
Features	Current	Reserved
	Previous	Reserved
Sector Count	Current	Count (7:0) Number of sectors to transfer
	Previous	Count (15:8)
LBA Low	Current	E1h (Data Port)
	Previous	Reserved
LBA Mid	Current	00h There is no offset when commands are issued
	Previous	00h
LBA High	Current	Reserved
	Previous	Reserved
Command		2Fh (READ LOG EXT)

Table 16– Transfer Data to the Drive USING SMART

Register	Value
Features	D6h (SMART Write Log subcommand code)
Sector Count	Count (Number of sectors to transfer)
LBA Low	E1h (Data port)
LBA Mid	4Fh (SMART enable code)
LBA High	C2h (SMART enable code)
Command	B0h (SMART)

Table 17– Transfer Data to Drive Using GPL

Register	Value
Features	Current Previous Reserved Reserved
Sector Count	Current Previous Count (7:0) Number of sectors to transfer Count (15:8)
LBA Low	Current Previous E1h (Data Port) Reserved
LBA Mid	Current Previous 00h There is no offset when commands are issued 00h
LBA High	Current Previous Reserved Reserved
Command	3Fh (WRITE LOG EXT)

4.6 Drive Access Methods

Standard ATA commands employ either LBA or Logical CHS access methods; SCT commands only support 48-bit LBA access.

4.6.1 Logical Block Address

For LBA access all user sectors on the drive are numbered in a one-dimensional sequence from 0 to the maximum number of user sectors minus one. ATA Commands support 28-bits of LBA addressing and ATA Extended commands support 48-bits of LBA addressing. All SCT commands support 48-bits of LBA address. In this method, all defective cylinders, heads and sectors are mapped out by defect management, rendering them inaccessible.

4.7 SCT Command Nesting and intermingling with Standard commands

In general, standard ATA commands can be intermingled with SCT Commands but SCT commands can not be nested. SCT commands that do not require a follow-on data transfer operation never have an issue with being intermixed with any ATA commands or each other. SCT commands that do require a data transfer, on the other hand, may not be nested; that is, if a key command that requires a data transfer is issued all data transfer—to or from the host—must complete before another SCT command is issued. In most cases, however, ATA read/write commands may be inserted in between SCT data transfers, that is, between complete SMART Read Log/Write Log commands. Furthermore, any reset (POR, software or hardware) will cause the SCT command to be aborted.

4.8 Resets

If an SCT command is executing, Soft Reset (SRST), Hard Reset (Pin1), and Power-On Reset (POR) all cause the command to be terminated. This could result in partial command execution or data loss. There is no indication once the drive becomes ready that the previous command was terminated.

4.9 SMART and General Purpose Logging Support

If READ LOG EXT or WRITE LOG EXT is not supported, then the commands described in this technical report cannot be issued using the general purpose logging commands.

If SMART is not supported, then the commands described in this technical report cannot be issued using the SMART READ/WRTIE LOG commands.

If SMART is supported, but not enabled, devices that implement this technical report still support SMART READ/WRITE LOG. This deviates from ATA/ATAPI-7 that states an error is returned for SMART READ/WRITE LOG when SMART is disabled.

5 SCT Command Set

An SCT command (a.k.a. a Key Sector) is always 512 bytes long. Table 18 shows the generic format of an SCT command.

Table 18 - Key Sector format

Field	words	Description
Action Code	1	This is the fields that defines the command type and generally specifies the type of data being accessed, such as <i>sector</i> , <i>long sector</i> , etc. or physical action being performed, such as <i>seek</i> .
Function Code	1	This field specifies the type of access and varies by command. For example, this can specify read, write, verify, etc.
Parameter1	Depends on command	Depends on command
Parameter2	Depends on command	Depends on command
.	.	.
.	.	.
.	.	.
Total Words	256	

The supported action codes are defined in Table 19.

Table 19 – SCT Action Codes

Action Code	Block Data	TF Data	Description
0000h	-	-	Reserved
0001h	in\out	Y	Long Sector Access
0002h	in	N	LBA Segment Access
0003h	-	Y	Error Recovery Control
0004h	-	Y	Features Control
0005h	out	N	SCT Data Tables
0006h - BFFFh	-	-	Reserved
C000h - FFFFh	-	-	Vendor specific

5.1 Long Sector Access command

<u>Action Code(1):</u>	0001h
<u>Function Code(1):</u>	1 for read Long 2 for write Long
<u>LBA(4):</u>	Address of long sector (in LSW to MSW order)
<u>Block Transfer:</u>	Output for read Input for write
<u>Task File Output:</u>	Number of ECC/CRC bytes for read and write (reported in Sec Cnt and Sec Num registers)
<u>Description:</u>	Read or write a sector with full CRC and ECC data

The Long Sector format for both read and write is two (512 byte) blocks long. The first block contains the user data; the second block contains the error correction and detections bytes; the remainder of the second block should be all zeros.

Table 20 – Long Sector Format

Field	Size	Description
First Block		
User Data	512	User data as written from host
Second Block		
ECC/CRC Data	Depends on actual sector size	Error correction and detection bytes. The number of bytes is returned as TF Data on both read and write.
Reserved	Remainder of block	All zeros

5.2 LBA Segment Access command

<u>Action Code(1):</u>	0002h
<u>Function Code(1):</u>	1 for repeat write pattern 2 for repeat write sector
<u>Start LBA(4):</u>	First LBA of segment (in LSW to MSW order)
<u>Sector Count(4):</u>	Number of sectors; if zero, do remainder of drive
<u>Pattern(2):</u>	32-bit pattern to be written for entire segment (FC 1 only)
<u>Block Transfer:</u>	One sector input for repeat write sector
<u>Task File Output:</u>	None
<u>Description:</u>	Write LBA segment

This command will begin writing sectors in LBA order until Sector Count sectors have been written. A Sector Count of zero means apply operation from Start LBA until the last user LBA on the drive. (Last LBA is determined by the HPA feature. This command will not write over a hidden partition is set by the Host Protected Area.)

Any command, other than SCT status, issued to the drive while this command is in progress will terminate the command.

Use the SCT Status command to retrieve status information about the current SCT command. Example status information includes: command active or complete, current LBA, and errors. While this command is in progress, the SCT status error code will be FFFFh, and set to 0000h if the command completes without error. It will be less than FFFFh if the command terminated prematurely for any reason.

The Drive Zeroed Flag. Issuing this command with a function code of 1 or 2 and a value of zero for Start LBA and Sector Count will cause all LBA's of the drive to be written with the specified pattern. If command completes without encountering an error or being aborted, then the "Drive Zeroed flag" (bit 0 of the Status Flags in the SCT status. See SCT status) will be set to 1. A sequential write to any LBA on the drive (except another complete write all), will cause the Drive Zeroed flag to be cleared.

5.3 Error Recovery Control command

<u>Action Code(1):</u>	0003h
<u>Function Code(1):</u>	1 to set new value 2 to return current value
<u>Selection Code(1):</u>	1 for Read Command Timer (1 Count = 100ms) 2 for Write Command Timer (1 Count = 100ms)
<u>Set Value(1):</u>	New value to be set (Function Code 1 only)
<u>Block Transfer:</u>	None
<u>Task File Output:</u>	If Task File status is 50h: Current value (Function Code 2 only)
<u>Description:</u>	Set error recovery time

This command can be used to set time limits for read and write error recovery. This may be desirable in a data redundant RAID environment where it is more desirable to have the drive report a data error rather than risk having it being kicked off of the RAID.

The Read Command Timer sets an upper limit to the amount of time the drive's disk task will be operating on a command. Typically amount of the time the drive will be operating on a read command in total but in some cases a read command will require more than one disk operation. Minimum value for this command is TBD. Setting this value to zero will disable Read Command time-out, allowing the drive to perform all available error recovery procedures without time limit.

If the Read Command Timer expires while the drive is performing error recovery, the drive will stop processing the command and report an un-correctable ECC error for the LBA that was causing error recovery to be invoked. Note that the LBA might actually be recoverable given more time for error recovery. At this point the host could reconstruct the data for the failing LBA from the other disk drives, and then issue a write command to the target LBA, allowing the drive to handle the suspect LBA as it best sees fit.

The Write Command Timer sets the upper limit for the amount of time the drive spends recovering from an error while performing a write. The minimum value for this command is TBD. Setting this value to zero will disable Write Command time-out, allowing the drive to perform all available error recovery procedures without a time limit.

The Write Command Timer has the effect of controlling how aggressively the drive will relocate write data when encountering servo defects. A long Write Command Timer allows the drive to use all of its available error recovery procedures for dealing with write errors. A short Write Command Timer will force the drive to relocate sectors that are having difficulty being written sooner rather than later. The relocating of the data needs to occur in order to prevent the write command from exceeding its allotted time.

The typical usage for an ATA or SATA drive is with Write Cache enabled. With write cache enabled, the drive cannot report an error on a write command. This is because the write command that is actually experiencing difficulty was probably finished from the host's perspective many commands ago. This leaves no recourse for the drive other than to relocate any sectors that are having difficulty being written.

Read and Write command timer values are set to default values at power-on but may be altered by a SCT command at any time. These settings are unaffected by software (soft) or hardware (pin 1) reset.

5.4 Feature Control command

<u>Action Code(1):</u>	0004h
<u>Function Code(1):</u>	1 to set new state for selected feature 2 to return current state for selected feature 3 to return current Option Flags for selected feature
<u>Feature Code(1):</u>	Identifier of feature to be acted upon
<u>New State(1):</u>	Feature Code dependent value
<u>Option Flags(1):</u>	All options apply to all Features (except if state otherwise) Bit 0 – Set Permanent (SP). If this bit is set, the state change become the new default state and will be preserved through a power cycle. If cleared, the default set will be restore upon power cycle. Bit 1-15 – Reserved
<u>Block Transfer:</u>	None
<u>Task File Output:</u>	Current feature state (reported in Sec Cnt and Sec Num registers)
<u>Description:</u>	Set new feature state or return current state or Option Flags

Table 21 – Feature Code List

Feature Code	State Definition
0001h	<p>0001h = Allow write cache operation to be determined by ATA Set Features command</p> <p>0002h = Force write cache enabled.</p> <p>0003h = Force write cache disabled</p> <p>If State 0001h is selected, the ATA Set Features command will determined the operation state of write cache per the ATA specification. If States 0002h or 0003h are selected, write cache will be forced into the corresponding operation state, regardless of the current ATA Set Features state. Furthermore, any future use of the ATA Set Features command shall be treated as a NOP, not effect the operation state of write cache and complete normally without reporting an error.</p> <p>In all cases, bit 5 of word 85 in the IDENTIFY DEVICE information will reflect the true operation state of write cache, one indicting enabled and zero indicating disabled.</p> <p>The default is states 01h</p>
0002h	<p>0001h = Enable Write Cache Reordering (use WRPO ordering for writes)</p> <p>0002h = Disable Write Cache Reordering (use FIFO ordering for writes)</p> <p>If State 0002h is selected, disk write scheduling is executed on a first-in-first-out (FIFO) basis, instead of Write Cache Rotational Position Optimization (WRPO) order. If State 0001h is selected, then RPO ordering is used for writes. If write cache is disable, the current Write Cache Reordering state is remembered but has no effect on non-cached writes, which are always written in the order received. The state of Write Cache Reordering has no effect on either NCQ or LCQ queued commands.</p> <p>The default is states 0001h</p>
0003h	<p>Set time Interval between entries for temperature Data Tables.</p> <p>0000h is invalid</p> <p>0001h to FFFFh logging interval in minutes</p> <p>This value always applies equally to Fan Control Temperature History and HDA Temperature History queues. Issuing this command will cause the queues to be reset and any current entries to be lost: Queue Index shall be set to zero and the first queue location shall be set to current Fan Control / HDA Temperature and all remaining queue locations set to FFh</p> <p>Default value is 0001h</p>

5.5 SCT Data Table command

- Action Code(1): 0005h
- Function Code(1): 1 to read data table
- Table Id(1): Identifier of Data Table to be acted upon
- Block Transfer: Number of sectors in table
- Task File Output: none
- Description: Access and control SCT Data Tables

Function Code 1 will return the specified data table (See Table 22).

The Placement, accuracy, and granularity of temperature sensors to support Table 23 and Table 24 are vendors specific.

Table 22 – SCT Data Tables (by Table Identifier)

Table Id	Description
0000h	Invalid
0001h	Fan Control Temperature History Table (in relative degrees C). See Table 23
0002h	HDA Temperature History Table (in absolute degrees C). See Table 24
0003h to CFFFh	Reserved
D000h to FFFFh	Vendor Specific

Table 23 – Fan Control Temperature History Table format

Byte	Size	Field Name	DESCRIPTION
0	2	Format Version	0001h - Data table format version
2	2	Sampling Period	The fan control sampling period in minutes. This is how often the drive actually samples its temperature sensor. 0000h is an invalid sampling period and indicates sampling is disabled.
4	2	Interval	This is the timer interval between entries in the history queue. By default, this value is 1 minute. This value shall be not be less than the Sampling Period.
6	1	Fan Max Op Limit	Fan control maximum operating temperature limit
7	1	Fan Over Limit	Fan Control over range temperature limit
8	22	reserved	Must be zero
30	2	Queue Size	Number of entry locations in history queue
32	2	Queue Index	Last updated entry in queue. Queue Index is zero-based, so Queue Index 0000h is the first location in the buffer (at offset 34) and Queue Index 007Fh is the last location in the buffer (at offset 161). See note 1.
34	128	Queue Buffer	History of the last 128 HDA Temperature values. This is a circular buffer so the newest value will overwrite the oldest value. A value of 00h indicates no value set. All samples are separated by Sampling Period in time. See note 1
162 - 511	350	reserved	Shall be zero

Note 1 - Temperature History is preserved across power cycles, but with the requirement that when the drive powers up, the current entry in the queue will be set FFh, a invalid temp value. This way someone viewing the history could see that the discontinuity in temp result from the drive being turned off. If the drive does not sample HDA temperature during a certain power state—such as Sleep or Standby, then the same should be done when returning to a power state that does sample HDA temperature

Note 2 - When the Temperature history is cleared, for new drives and after changing Logging Interval, the Queue Index shall be set to zero and the first queue location shall be set to current Fan Control / HDA Temperature and all remaining queue locations set to FFh

Table 24 – HDA Temperature History Table format

Byte	Size	Field Name	Description
0	2	Format Version	0001h - Data table format version)
2	2	Sampling Period	The fan control sampling period in minutes. This is how often the drive actually samples its temperature sensor. 0000h is an invalid sampling period and indicates sampling is disabled.
4	2	Interval	This is the timer interval between entries in the history queue. By default, this value is 1 minute. This value shall be not be less than the Sampling Period.
6	24	reserved	Shall be zero
30	2	Queue Size	Number of entry locations in history queue
32	2	Queue Index	Last updated entry in queue. Queue Index is zero-based, so Queue Index 0000h is the first location in the buffer (at offset 34) and Queue Index 007Fh is the last location in the buffer (at offset 161). ee note 1.
34	128	Queue Buffer	History of the last 128 HDA Temperature values This is a circular buffer so the newest value will overwrite the oldest value. A value of 00h indicates no value set. All samples are separated by Sampling Period in time. See note 1
162 - 511	350	reserved	Shall be zero
<p>Note 1 - Temperature History is preserved across power cycles with the requirement that when the drive powers up, that the current entry in the queue shall be set to FFh, an invalid temp value. This way someone viewing the history could see that the discontinuity in temp result from the drive being turned off.</p>			