To: T10 Technical Committee
From: Robert Sheffield, Intel (robert.l.sheffield@intel.com)
Date: 16 June 2005
Subject: 05-247r0 SAT: Add 16-byte CDBs and PIO modes

Revision history

Revision 0 (16 June 2005) - First revision

Related documents

SAT-r04 - SCSI / ATA Translation revision 4

Overview

Reflector traffic prompted the SAT WG to discuss the need to define SAT translations for block commands using the 16-byte CDB format, such as READ CAPACITY (16), and others. The reasoning is that because SAT defines translations involving ATA devices implementing the 48-bit Address feature set, an ATA device may be installed with a capacity that cannot be fully addressed using the 32-bit LBA field provided in the SCSI 10 and 12 byte CDB read/write type commands. READ CAPACITY (16), for example, must be used to determine the capacity of a non-packet device with a capacity greater than 2 Tebibytes accessed through a SATL. This proposal is to add text defining the translations for the following SCSI commands:

a) READ (16) 88h
b) READ CAPACITY(16) 9Eh
c) SYNCHRONIZE CACHE (16) 91h
d) VERIFY (16) 8Fh
e) WRITE (16) 8Ah
f) WRITE AND VERIFY (12) AEh
g) WRITE AND VERIFY (16) 8Eh

These translations may involve use of the following ATA non-packet device commands:

a) FLUSH CACHE
b) FLUSH CACHE EXT
c) READ DMA
d) READ DMA EXT
e) READ DMA QUEUED
f) READ DMA QUEUED EXT
g) READ MULTIPLE
h) READ MULTIPLE EXT
i) READ SECTOR(S)
j) READ SECTOR(S) EXT
k) READ VERIFY SECTOR(S)
l) READ VERIFY SECTOR(S) EXT
m) WRITE DMA
n) WRITE DMA EXT
o) WRITE DMA FUA EXT
p) WRITE DMA QUEUED
q) WRITE DMA QUEUED EXT
r) WRITE DMA QUEUED FUA EXT
s) WRITE MULTIPLE
t) WRITE MULTIPLE EXT
u) WRITE MULTIPLE FUA EXT
v) WRITE SECTOR(S)
w) WRITE SECTOR(S) EXT
x) READ FPDMA QUEUED (SATA IIe)
y) WRITE FPDMA QUEUED (SATA IIe)

This proposal also adds text to set the general rules for translating LBA and transfer length for block storage read/write command translations added to an initial subclause under the block commands subclause, and applies several corrections, clarifications, and editorial changes to text throughout the block commands clause.

Suggested Changes:

Add the following definitions

3.1.1 ATA flush command: .A FLUSH CACHE or FLUSH CACHE EXT command defined in ATA/ATAPI-7.

3.1.2 ATA read command: A non-packet device block read type command which is one of the following: READ DMA, READ DMA EXT, READ DMA QUEUED, READ DMA QUEUED EXT, READ MULTIPLE, READ MULTIPLE EXT, READ SECTOR(S), or READ SECTOR(S) EXT defined in ATA/ATAPI-7; or READ FPDMA QUEUED defined in SATA IIe.

3.1.3 ATA verify command: .A READ VERIFY SECTOR(S) or READ VERIFY SECTOR(S) EXT command defined in ATA/ATAPI-7.

3.1.4 ATA write command: A non-packet device block write type command which is one of the following: WRITE DMA, WRITE DMA EXT, WRITE DMA FUA EXT, WRITE DMA QUEUED, WRITE DMA QUEUED EXT, WRITE DMA QUEUED FUA EXT, WRITE MULTIPLE, WRITE MULTIPLE EXT, WRITE MULTIPLE FUA EXT, WRITE SECTOR(S), or WRITE SECTOR(S) EXT defined in ATA/ATAPI-7; or WRITE FPDMA QUEUED defined in SATA IIe.

3.1.5 ATA write FUA command sequence: .A sequence of commands that writes logical blocks to an attached non-packet device in a way that forces media access and consists of one of the following:

a) a WRITE DMA, WRITE DMA QUEUED, WRITE FPDMA QUEUED, WRITE MULTIPLE, or WRITE SECTOR(S) command followed by a READ VERIFY SECTOR(S) command defined in ATA/ATAPI-7;
b) a WRITE DMA FUA EXT, WRITE DMA QUEUED FUA EXT, or WRITE MULTIPLE FUA EXT command defined in ATA/ATAPI-7, or
c) a WRITE FPDMA QUEUED command defined in SATA IIe with the FUA bit in the Device/Head field set to one.

3.1.6 SCSI read command: A SCSI READ (6), READ (10), READ (12), or READ (16) command defined in SBC-2.

3.1.7 SCSI synchronize cache command: .A SCSI SYNCHRONIZE CACHE(10), or SYNCHRONIZE CACHE (16) command defined in SBC-2

3.1.8 SCSI verify command: .A SCSI VERIFY (10), VERIFY (12), or VERIFY (16) command defined in SBC-2.

3.1.9 SCSI write command: .A SCSI WRITE (6), WRITE (10), WRITE (12), or WRITE (16) command defined in SBC-2.

3.1.10 SCSI write and verify command: .A SCSI WRITE AND VERIFY (10), WRITE AND VERIFY(12), or WRITE AND VERIFY (16) command defined in SBC-2.

Add subclause 9.1 under clause 9 SCSI Block Commands (SBC-2) Mapping as shown below

9.1 Translating LBA and transfer length and ATA command use constraints

ATA commands the SATL may use to implement the functions specified by SCSI block commands depend upon:

a) the value of the LOGICAL BLOCK ADDRESS and TRANSFER LENGTH fields specified in the SCSI CDB, and
b) the capabilities of the attached ATA device and the ATA host within the SATL.

Table 1 identifies constraints in selecting ATA commands to implement SCSI block storage commands.

Table 1 — Read and write type command translation constraints

ATA commands the SATL may use	Prerequisites[a]				
	SCSI CDB	ATA feature sets supported and enabled			
	(TRANSFER LENGTH + LBA) ≤ 2^{28}	48-bit Address[b]	DMA[c]	Overlap[d]	SATA IIe NCQ
FLUSH CACHE	no	no	no	no	no
READ MULTIPLE, READ SECTOR(S), READ VERIFY SECTOR(S), WRITE MULTIPLE, WRITE SECTOR(S)	yes[be]	no	no	no	no
READ DMA, WRITE DMA	yes[be]	no	yes	no	no
READ DMA QUEUED, WRITE DMA QUEUED	yes[be]	no	yes	yes	no
READ DMA EXT, WRITE DMA EXT, WRITE DMA FUA EXT	no	yes	yes	no	no
READ DMA QUEUED EXT, WRITE DMA QUEUED EXT, WRITE DMA QUEUED FUA EXT	no	yes	yes	yes	no
FLUSH CACHE EXT, READ MULTIPLE EXT, READ SECTOR(S) EXT, READ VERIFY SECTOR(S) EXT, WRITE MULTIPLE EXT, WRITE MULTIPLE FUA EXT, WRITE SECTOR(S) EXT	no	yes	no	no	no
READ FPDMA QUEUED, WRITE FPDMA QUEUED	no	no	no	no	yes

[a] An ATA command may be used to implement a SCSI block command only if all the prerequisites in the prerequisite columns for that command marked “yes” are satisfied.

[b] The SATL should not receive a request to access an LBA beyond 2^{28}-1 if the attached non-packet device does not support the 48-bit Address feature set or NCQ (see SATA IIe) (i.e., because the SATL reports a capacity less than 2^{28}).

[c] The DMA prerequisite requires both the SATA host in the SATL and the attached non-packet device to support the DMA capability (i.e. bit 8 of word 49 in the IDENTIFY DEVICE data is set to one and at least one DMA mode is enabled in dword 63 or dword 88 of the IDENTIFY DEVICE data).

[d] See ATA/ATAPI-7 and SATA IIe.

[e] The SATL may transfer the number of bytes requested in the TRANSFER LENGTH field by sending multiple ATA commands, each time incrementing the LBA in the ATA command by the Sector Count transferred.

The SATL may use ATA commands listed in table 1 in the translation of SCSI read (see...), SCSI write (see...), SCSI write and verify (see...), SCSI verify (see...), and SCSI synchronize cache (see...) commands if the prerequisites defined for the command as shown in table 1 are satisfied. The translations for specific SCSI block commands in this clause further constrain the use of the available ATA commands in implementing the translation.

If the logical block address plus the transfer length specified in the SCSI CDB is greater than FFFFFFFh and the attached non-packet device does not support the 48-bit Address feature set, the SATL shall terminate the command with CHECK CONDITION status with the sense key set to ILLEGAL REQUEST and the additional sense code set to LOGICAL BLOCK ADDRESS OUT OF RANGE (see SBC-2).

For SCSI read (see 3.1.6), SCSI verify (see 3.1.8), SCSI write (see 3.1.9), and SCSI write and verify (see 3.1.10) commands received with 6-byte CDBs, if the TRANSFER LENGTH or VERIFICATION LENGTH field is zero, the SATL shall issue ATA commands specifying a Sector Count to operate on 256 logical sectors. If the CDB is 10, 12, or 16 bytes or if the transfer count on a 6-byte CDB is non-zero, the SATL shall transfer or operate on the number of logical sectors specified.

Modify subclause 9.1 FORMAT UNIT command as shown below

9.2 FORMAT UNIT command (4h)

9.2.1 Command summary

The FORMAT UNIT command verifies that all logical block addresses ~~visible~~ available to ~~external~~ SCSI application clients are formatted and may be accessed. ~~All sectors of the visible address space are written to zero~~.

Table 2 — FORMAT UNIT command CDB fields

Field	SATType	Description or reference
OPERATION CODE	E	If no defect list header is provided or a defect list header is provided with the DCRT bit set to one the SATL shall return completion status without issuing any commands to the device with a status of GOOD. If the SATL supports certification of media and a defect list header is provided with the DCRT bit set to zero the SATL shall certify the media as described in 9.2.3
DEFECT LIST FORMAT	E	If the DEFECT LIST FORMAT field is the mandatory format (000b) or the vendor specific format (110b) the defect list length shall be zero (see SBC-2). If the DEFECT LIST FORMAT field is any other value the SATL shall terminate the command with CHECK CONDITION status with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN PARAMETER LIST.
CMPLIST	U	If a CMPLIST is specified the SATL shall terminate the command with a CHECK CONDITION with sense key set to ILLEGAL REQUEST and additional sense code set to INVALID FIELD IN CDB.
FMTDATA	E	If set to 0 no data shall be transferred from the data-out buffer. If set to 1 the FORMAT UNIT parameter list shall be transferred from the client's data out buffer. The SATL may accept a FORMAT UNIT parameter list specifying the IMMED bit and an initialization pattern. The SATL shall ignore any defect list descriptors and any other fields provided in the FORMAT UNIT parameter list (see 9.2.2).
LONGLIST	U	The SATL may implement this field as defined in SBC-2.
FMTPINFO	U	The SATL may implement this field as defined in SBC-2.
CONTROL	I	(see 6.4)

The SATL shall process commands received during the processing of the FORMAT UNIT command as specified in SBC-2.

9.2.2 FORMAT UNIT parameter list

If the FORMAT command CDB specifies a FMTDATA bit of one, the SATL shall accept a FORMAT UNIT parameter list consisting of a short or long defect list header and may accept an initialization pattern descriptor. The SATL shall ignore any defect descriptors provided. Table 3 defines the SATL handling of fields in the FORMAT UNIT defect list header.

Table 3 — SATL defect list header

Field[a]	SATType	Description or reference
FOV	E/U	The SATL shall implement this field as defined in SBC-2.
DPRY	E	The SATL shall ignore this field.
DCRT	E	(see 9.2.3)
STPF	U	The SATL may implement this field as defined in SBC-2.
IP	U	The SATL may implement this field as defined in SBC-2. If supported, the SATL shall write the specified pattern by issuing ~~WRITE SECTOR(S) or WRITE SECTOR(S) EXT~~ ATA write commands (see 3.1.4 and 9.1) to the attached non-packet device.
IMMED	E	The SATL shall implement this field as defined in SBC-2.
DEFECT LIST LENGTH	E	The SATL shall ignore any defect descriptors provided.

[a] If IMMED is one or if FOV is zero or if FOV is one, DCRT is one, and IP is zero, then the SATL may complete the FORMAT UNIT command immediately with SUCCESSFUL status. If IMMED is zero, FOV is one, and either DCRT is zero or IP is one, the SATL may terminate the command with CHECK CONDITION status with the sense key set to ILLEGAL REQUEST with the additional sense code set to INVALID FIELD PARAMETER LIST. Otherwise, the SATL shall issue the required ATA read and ATA write commands to certify and initialize the media as specified by DCRT and IP, and shall then return SUCCESSFUL status if no unrecoverable write errors are encountered.

9.2.3 DCRT bit

If a value of zero is specified and supported the SATL shall issue ~~READ VERIFY SECTOR(S) or READ VERIFY SECTOR(S) EXT~~ ATA verify commands (see 3.1.3) to access every block on the media. If any unrecoverable read errors are encountered the SATL shall issue an ~~WRITE SECTOR(S) or WRITE SECTOR(S) EXT~~ ATA write command (see 3.1.4) to the defective sector to ~~force~~ cause allocation of an alternate. The data written shall be vendor-specific or the data pattern specified by the initialization pattern descriptor if one is provided. After writing the block, the SATL shall again issue an ~~READ VERIFY SECTOR(S) or READ VERIFY SECTOR(S) EXT~~ ATA verify command (see 3.1.3) to the same sector to verify the alternate block is not defective. The process (i.e., verify, write, verify, write, ...) shall repeat until the logical block is verified successfully or the disk reports a fatal error other than an unrecoverable read error (e.g., device fault).

Editor's Note 1: Could add a statement here about the response if a fatal error is detected - or perhaps a reference to the appropriate general error handling subclause.

Modify subclause 9.2 READ (6) command as shown below

9.3 READ (6) command (8h)

9.3.1 Command summary

The READ(6) command is used to request the device to transfer logical blocks of user data to the requester. Different versions of the command support different LBA sizes and different transfer lengths. Data may be read from medium or, data may be read from the device cache if the most recent copy is in the cache and has not been transferred to the medium (see SBC-2)

Table 4 — READ(6) command CDB fields

Field	SATType	Description or reference
OPERATION CODE	I	9.3.2
LOGICAL BLOCK ADDRESS		~~(see 9.3.3)~~9.3.2
TRANSFER LENGTH		The transfer length shall be used to set the ATA sector count ~~(see 9.3.4)~~9.3.2.[a]
CONTROL	I	(see 6.4)

[a] A transfer length of zero specifies to transfer 256 logical blocks from the attached non-packet device to the application client (see SBC-2).

9.3.2 READ operation code translations

~~Table 5 shows the translation for SCSI READ commands to ATA or SATA IIe commands based on the capabilities of the attached non-packet device. This subclause applies to the translation of SCSI READ(6), READ(10), and READ(12).~~

~~**Table 5 — READ command summary**~~

~~Translated to ATA or SATA IIe read commands using the following criterion~~		
~~Is Queuing enabled?~~[a]	~~Are ATA Extended Commands Enabled~~	~~Translated ATA Opcode~~[b]
~~No~~	~~No~~	~~READ DMA~~
~~No~~	~~Yes~~	~~READ DMA EXT~~[c]
~~Yes~~	~~No~~	~~READ DMA QUEUED~~
~~Yes~~	~~Yes~~	~~READ DMA QUEUED EXT~~[d]
~~Yes~~	~~Yes or No~~	~~READ FPDMA QUEUED (See SATA IIe)~~[e]

[a] ~~Refers to either the non-packet device support of the ATA/ATAPI-7 Overlapped feature set or the Native Command Queing (NCQ) feature of SATA IIe.~~
[b] ~~The SATL may attempt READ MULTIPLE, READ MULTIPLE EXT, READ SECTOR(S) or READ SECTOR(S) EXT as a retry operation if the specified ATA or SATA IIe command fails. The selected ATA commands shall have enough bits in the logical sector address to avoid truncation of the LBA supplied in the SCSI CDB.~~
[c] ~~The SATL may substitute READ DMA if the LBAs accessed can be represented in 28 bits.~~
[d] ~~The SATL may substitute READ DMA QUEUED if the LBAs accessed can be represented in 28 bits.~~
[e] ~~The SATL may use these commands only if NCQ is enabled (see SATA IIe).~~

This subclause applies to the translation of SCSI READ(6), READ(10), READ(12), and READ(16) commands.

The SATL shall issue ATA read commands (see 3.1.2) in accordance with the constraints specified in subclause 9.1 to cause the non-packet device to transfer the logical blocks specified in the SCSI read command (see 3.1.6) to the ATA host in the SATL.

The SATL shall transfer the ~~The~~ requested sector(s) ~~shall be returned~~ to the SCSI application client if successfully retrieved from the non-packet device. If the LBA plus the transfer length minus 1 is greater than the ~~maximum~~ highest numbered sector that can be addressed ~~in medium~~ on the non-packet device the SATL device server shall ~~return~~ terminate the command with CHECK CONDITION status with the sense key set to ILLEGAL REQUEST and the additional sense code set to LOGICAL BLOCK ADDRESS OUT OF RANGE (see SBC-2). The SATL device server may transfer a vendor-specific amount of data before terminating the command.

~~9.3.3 LOGICAL BLOCK ADDRESS field~~

~~The SATL shall transfer data blocks starting with the LBA specified.~~

~~9.3.4 TRANSFER LENGTH field~~

~~If the TRANSFER LENGTH field is zero, the SATL shall transfer 256 data blocks.~~

~~If the TRANSFER LENGTH is not zero, the SATL shall transfer the specified number of sectors from the device to the application client.~~

Modify subclause 9.3 READ (10) command as shown below

9.4 READ (10) command (28h)

9.4.1 Command summary

The SATL shall process the READ(10) command the same as the READ(6) command (see 9.3.2). with the additional fields in the CDB implemented as described in ~~the following paragraphs~~ table 6 and subclause 9.4.2.

Table 6 — READ(10) command CDB fields

Field	SATType	Description or reference
OPERATION CODE	I	(see 9.4.2)
RDPROTECT	U	See SBC-2
DPO (disable page out)	U	Ignored
FUA (force unit access)	E	(see 9.4.2)
FUA_NV (force unit access nonvolatile cache)	U	The SATL shall not implement FUA_NV bit.
LOGICAL BLOCK ADDRESS	I	~~(see 9.3.3)~~(see 9.3.2)
GROUP NUMBER	U	The SATL may implement this field as defined in SBC-2.
TRANSFER LENGTH	I/E	The transfer length shall be used to set the ATA sector count ~~(see 9.4.3)~~(see 9.3.2).[a] The SATL shall issue as many ATA commands as needed to satisfy the transfer length specified by the READ (10) command.
CONTROL	I	(see 6.4)

[a] A transfer length of 0 indicates that a data transfer shall not take place. If the transfer length is not zero, the SATL shall transfer the number of sectors specified from the device to the application client.

9.4.2 ~~READ(10,12)~~ READ (10), READ (12) and READ (16) OPERATION CODE and FUA

The SATL may support FUA. If the SATL does not support FUA and FUA is set to one, the SATL shall terminate the command with CHECK CONDITION status with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB. If FUA is set to one and the attached device supports NCQ the SATL shall issue a SATA IIe READ FPDMA QUEUED command with the FUA bit in the Device/Head field set to one. If FUA is set to one and the attached device does not support NCQ, the SATL shall issue an ATA READ VERIFY or READ VERIFY EXT command followed by an ATA read command as specified in subclause 9.3.2. If FUA is zero the SATL shall issue a read command as specified in subclause 9.3.2.

~~9.4.3 TRANSFER LENGTH field~~

~~If the TRANSFER LENGTH field is zero, the SATL shall not transfer any data blocks to the application client.~~

~~If the TRANSFER LENGTH is not zero, the SATL shall transfer the specified number of sectors from the device to the application client.~~

Modify subclause 9.4 READ (12) command as shown below

9.5 READ (12) command (A8h)

9.5.1 Command summary

The READ(12) command is used to request the device to transfer logical blocks of user data to the requester. Different versions of the command support different LBA sizes and different transfer lengths. Data may be read from medium or, data may be read from the device cache if the most recent copy is in the cache and has not been transferred to the medium (see SBC-2)

Table 7 — READ(12) command CDB fields

Field	SATType	Description or reference
OPERATION CODE	I	(see 9.4.2)
RDPROTECT	U	See SBC-2
DPO (disable page out)	U	Ignored
FUA (force unit access)	E	~~The SATL shall implement FUA if the attached device supports NCQ.~~ (see 9.4.2)
FUA_NV (force unit access nonvolatile cache)	U	The SATL shall not implement FUA_NV bit.
LOGICAL BLOCK ADDRESS	I	~~(see 9.3.3)~~(see 9.3.2)
GROUP NUMBER	U	The SATL may implement this field as defined in SBC-2.
TRANSFER LENGTH	I/E	The transfer length shall be used to set the ATA sector count ~~(see 9.4.3)~~(see 9.3.2).[a] The SATL shall issue as many ATA commands as needed to satisfy the transfer length specified by the READ (12) command.
CONTROL	I	(see 6.4)

[a] A transfer length of 0 indicates that a data transfer shall not take place. If the transfer length is not zero, the SATL shall transfer the number of sectors specified from the device to the application client.

~~9.5.2 TRANSFER LENGTH field~~

~~The SATL shall transfer the number of sectors specified in the TRANSFER LENGTH field to the application client. If the TRANSFER LENGTH field is greater than 0xFFFF the SATL device server shall terminate the command with CHECK CONDITION status with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.~~

~~NOTE 1 - This is because the ATA command set supports a maximum of a 16-bit SECTOR COUNT field.~~

Add subclause 9.6 READ (16) command as shown below

9.6 READ (16) command (88h)

9.6.1 Command summary

The READ (16) command is used to request the device to transfer logical blocks of user data to the requester. Different versions of the command support different LBA sizes and different transfer lengths. Data may be read from medium or, data may be read from the device cache if the most recent copy is in the cache and has not been transferred to the medium (see SBC-2)

Table 8 — READ(16) command CDB fields

Field	SATType	Description or reference
OPERATION CODE	I	(see 9.4.2)
RDPROTECT	U	See SBC-2
DPO (disable page out)	U	Ignored
FUA (force unit access)	E	(see 9.4.2)
FUA_NV (force unit access nonvolatile cache)	U	The SATL shall not implement FUA_NV bit.
LOGICAL BLOCK ADDRESS	I	(see 9.3.2)
GROUP NUMBER	U	The SATL may implement this field as defined in SBC-2.
TRANSFER LENGTH	I/E	The transfer length shall be used to set the ATA sector count (see 9.3.2).[a] The SATL shall issue as many ATA commands as needed to satisfy the transfer length specified by the READ (12) command.
CONTROL	I	(see 6.4)

[a] A transfer length of 0 indicates that a data transfer shall not take place. If the transfer length is not zero, the SATL shall transfer the number of sectors specified from the device to the application client.

Add subclause 9.8 READ CAPACITY (16) command as shown below

9.8 READ CAPACITY (16) command (9Eh)

9.8.1 Command summary

The READ CAPACITY (16) command shall request information about the capacity of the block device being addressed.

Table 9 — READ CAPACITY(16) command CDB fields

Field	SATType	Description or reference
OPERATION CODE	E	Use ATA IDENTIFY DEVICE information at power-on or link initialization to compute the maximum user addressable medium capacity. A SATL may use part of the capacity of the attached non-packet device to store persistent state information for management of the emulated SCSI target device.
LOGICAL BLOCK ADDRESS	U	The LOGICAL BLOCK ADDRESS field works in conjunction with the PMI bit described below. Since this requires vendor specific information, this operation shall not be supported and shall be set to zero. If the LOGICAL BLOCK ADDRESS is not 0000000h the SATL device server shall terminate the command with CHECK CONDITION STATUS with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.
PMI (partial medium indicator)	U	The operation specified by this bit shall be ignored. The SATL shall return data for the LBA following the LBA specified in logical block address. If the PMI is not zero the SATL device server shall terminate the command with CHECK CONDITION STATUS with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.
CONTROL	I	(see 6.4)

9.8.2 READ CAPACITY data

The SATL shall return READ CAPACITY data as defined by SBC-2.

Table 10 — READ CAPACITY data

Field	SATType	Description or reference
RETURNED LOGICAL BLOCK ADDRESS	I	The SATL shall return a value no larger than one less than the value from the ATA IDENTIFY DEVICE data TOTALUSERSECTORS field (words 60/61) retrieved from the device. The SATL may report less than the addressable medium capacity reported in the ATA IDENTIFY DEVICE information.The bytes shall be swapped to match SCSI big-endian format.
BLOCK LENGTH IN BYTES	I	This value should be set to 512 bytes. The SATL may operate in a vendor-specific manner with a non-packet device that reports a logical sector size greater than 256 words in IDENTIFY DEVICE words 117-118 (see ATA/ATAPI-7)
RTO_EN	U	See SBC-2
PROT_EN	U	See SBC-2

Modify subclause 9.8 SYNCHRONIZE CACHE (10) command as shown below

9.11 SYNCHRONIZE CACHE (10) command (35h)

9.11.1 Command summary

The SYNCHRONIZE CACHE(10) command is used to flush the most recent data values in the device cache to physical medium. Unlike in SCSI, ATA does not provide a way to specify a particular LBA to start flushing the device cache.

Table 11 — SYNCHRONIZE CACHE(10) command CDB fields

Field	SATType	Description or reference
OPERATION CODE	I	~~Translated into the FLUSH CACHE command (E7h) or FLUSH CACHE EXT command (EAh).~~ The SATL shall issue an ATA flush command (see 3.1.1) in accordance with the constraints described in 9.1
SYNC_NV	U	
IMMED	U	The SATL shall ignore this bit.
LOGICAL BLOCK ADDRESS	U	The SATL shall ignore this field and shall process this command as though this field contained a value of zero.
GROUP NUMBER	U	The SATL may implement this field as defined in SBC-2.
NUMBER OF BLOCKS	U	The SATL shall ignore this field and shall process this command as though this field contained a value of zero (i.e., synchronize all logical blocks starting with the one specified in the LOGICAL BLOCK ADDRESS field to the last logical block on the medium).
CONTROL	I	(see 6.4)

~~Editor's Note 2: There is a proposal being developed that will describe general rules for selecting among ATA 24-bit commands, ATA 48-bit commands, and SATA-II FPDMA commands depending on the CDB received and the capabilities of the attached device. This proposal will cover selection of ATA/SATA commands involved in emulation of SYNCHRONIZE CACHE.~~

Add subclause 9.12 SYNCHRONIZE CACHE (16) command as shown below

9.12 SYNCHRONIZE CACHE (16) command (91h)

9.12.1 Command summary

Editor's Note 3: Here we have a bit of a conundrum. The FLUSH CACHE (EXT) command(s) always flush the entire cache - so the LBA in the SYNCHONIZE CACHE (10/16) command is meaningless to the SATL. The FLUSH CACHE EXT command differs from the FLUSH CACHE command only in that the EXT version can report a larger LBA (up to 48 bits) in case of a media error during execution of the command. SBC-2 defines how to report the LBA of a media error (in the INFORMATION field of sense data), however this capability is only defined for unrecovered ***read*** errors, ***not for write errors***. There is no expectation that a block-storage device (e.g., a SATL) would ever return an unrecovered LBA in the sense data for a SYNCHRONIZE CACHE command. One could argue, therefore, that there is never a reason for a SATL to issue the FLUSH CACHE EXT command. SYNCHRONIZE CACHE (16) is supported because SCSI application clients might expect it. This subclause allows either FLUSH CACHE or FLUSH CACHE EXT to be issued if the attached ATA device supports the 48-bit Address feature.

The SYNCHRONIZE CACHE(16) command is used to flush the most recent data values in the device cache to physical medium. Unlike in SCSI, ATA does not provide a way to specify a particular LBA to start flushing the device cache.

Table 12 — SYNCHRONIZE CACHE(10) command CDB fields

Field	**SATType**	**Description or reference**
OPERATION CODE	I	The SATL shall issue an ATA flush command (see 3.1.1) in accordance with the constraints described in 9.1
SYNC_NV	U	
IMMED	U	The SATL shall ignore this bit.
LOGICAL BLOCK ADDRESS	U	The SATL shall ignore this field and shall process this command as though this field contained a value of zero.
GROUP NUMBER	U	The SATL may implement this field as defined in SBC-2.
NUMBER OF BLOCKS	U	The SATL shall ignore this field and shall process this command as though this field contained a value of zero (i.e., synchronize all logical blocks starting with the one specified in the LOGICAL BLOCK ADDRESS field to the last logical block on the medium).
CONTROL	I	(see 6.4)

Remove subclause 9.9 VERIFY (6) command (not defined in SBC-2)

~~9.9 VERIFY (6) command (13h)~~

~~The VERIFY(6) command is not defined for direct-attach storage devices (disk). The SATL shall terminate the command with CHECK CONDITION STATUS with the sense key set to INVALID REQUEST and the additional sense code set to INVALID COMMAND OPERATION CODE.~~

Modify subclause 9.8 VERIFY (10) command as shown below

9.14 VERIFY (10) command (2Fh)

9.14.1 Command summary

The VERIFY(10) command is used to verify data on medium which includes user data and protection data. This SCSI command is directly translated into ATA read verify sectors command or the extended version of that command.

Table 13 — VERIFY(10) command CDB fields

Field	SATType	Description or reference
OPERATION CODE	I	The SATL shall issue an ATA ~~READ VERIFY SECTOR(S)~~ verify command (see 3.1.3) in accordance with the constraints described in 9.1~~, or if the device supports 48 bit mode, the SATL shall issue an ATA READ VERIFY EXTENDED command~~.
VRPROTECT	U	See SBC-2
DPO	U	Ignored
BYTCHK	U	If the application client specifies a value ~~of one~~ other than zero in this field, the SATL device server shall terminate the command with CHECK CONDITION STATUS with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.
LOGICAL BLOCK ADDRESS	U	(see 9.1) ~~If the LOGICAL BLOCK ADDRESS is larger than what can be accommodated using 28 bits and the device does not support 48 bit mode, the SATL shall return a CHECK CONDITION with sense key set to ILLEGAL REQUEST and additional sense code set to LOGICAL BLOCK ADDRESS OUT OF RANGE.~~
GROUP NUMBER	U	The SATL may implement this field as defined in SBC-2.
VERIFICATION LENGTH	U	The verification length shall be used to set the ATA sector count (see 9.1).
CONTROL	I	(see 6.4)

~~9.14.2 Miscellaneous notes~~

~~The SATL LBA mapping algorithm shall assign the LSB of the LBA field from the SCSI CDB, i.e. byte 5, to LBALow in the ATA FIS, byte 4 from SCSI CDB to LBAMid, byte 3 from SCSI CDB to LBAHigh. If the device supports 48-bit addressing byte 2 of the SCSI CDB shall be assigned to LBALowExp.~~

~~The SectorCount in the ATA FIS shall be derived from the LSB, i.e. byte 8, of the VERIFICATION LENGTH of the SCSI CDB. If the device supports 48-bit addressing, the SectorCountExp is assigned byte 7 of the SCSI CDB, which is the MSB of the VERIFICATION LENGTH field.~~

~~Commands the SATL issues to the non-packet device to process this command shall not be queued.~~

~~NOTE 2 - A performance degradation may be expected when a command such as VERIFY is issued as it must run in a single thread.~~

Modify subclause 9.9 VERIFY (12) command as shown below

9.15 VERIFY (12) command (AFh)

~~The VERIFY(12) command is not defined for SAT. The SATL shall terminate the command with CHECK CONDITION STATUS with the sense key set to INVALID REQUEST and the additional sense code set to INVALID COMMAND OPERATION CODE.~~

9.15.1 Command summary

The VERIFY(12) command is used to verify data on medium which includes user data and protection data. This SCSI command is directly translated into ATA read verify sectors command or the extended version of that command.

Table 14 — VERIFY(12) command CDB fields

Field	**SATType**	**Description or reference**
OPERATION CODE	I	The SATL shall issue an ATA verify command (see 3.1.3) in accordance with the constraints described in 9.1.
VRPROTECT	U	See SBC-2
DPO	U	Ignored
BYTCHK	U	If the application client specifies a value other than zero in this field, the SATL device server shall terminate the command with CHECK CONDITION status with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.
LOGICAL BLOCK ADDRESS	U	(see 9.1)
GROUP NUMBER	U	The SATL may implement this field as defined in SBC-2.
VERIFICATION LENGTH	U	The verification length shall be used to set the ATA sector count (see 9.1).
CONTROL	I	(see 6.4)

Add subclause 9.16 VERIFY (16) command as shown below

9.16 VERIFY (16) command (AFh)

9.16.1 Command summary

The VERIFY(16) command is used to verify data on medium which includes user data and protection data. This SCSI command is directly translated into ATA read verify sectors command or the extended version of that command.

Table 15 — VERIFY(16) command CDB fields

Field	SATType	Description or reference
OPERATION CODE	I	The SATL shall issue an ATA verify command (see 3.1.3) in accordance with the constraints described in 9.1.
VRPROTECT	U	See SBC-2
DPO	U	Ignored
BYTCHK	U	If the application client specifies a value other than zero in this field, the SATL device server shall terminate the command with CHECK CONDITION status with the sense key set to ILLEGAL REQUEST and the additional sense code set to INVALID FIELD IN CDB.
LOGICAL BLOCK ADDRESS	U	(see 9.1)
GROUP NUMBER	U	The SATL may implement this field as defined in SBC-2.
VERIFICATION LENGTH	U	The verification length shall be used to set the ATA sector count (see 9.1).
CONTROL	I	(see 6.4)

Modify subclause 9.12 WRITE (6) command as shown below

9.17 WRITE (6) command (0Ah)

9.17.1 Command summary

The WRITE(6) command is used to request the device to transfer user data to device medium or cache. Different versions of the command support different LBA sizes and or different transfer lengths. Data may be written to medium or the device cache.

Table 16 — WRITE(6) command CDB fields

Field	SATType	Description or reference
OPERATION CODE	I	See 9.17.2.
TRANSFER LENGTH	I	The transfer length shall be used to set the ATA sector count.[a]
LOGICAL BLOCK ADDRESS	I	~~For WRITE(6) commands (0Ah), a 21 bit LBA shall be derived from bytes 3, 2, 1:5 in the CDB, with byte 3 being the LSB and the 5 bits from byte 1 being the MSB.~~ See 9.17.2
CONTROL	I	See 6.4

[a] A transfer length of zero specifies to transfer 256 logical blocks from the application client to the attached non-packet device (see SBC-2).

9.17.2 WRITE command OPERATION CODE translation

~~Table 17 shows the translation for SCSI WRITE(6), WRITE(10), and WRITE(12) commands to ATA or SATA IIe commands based on the capabilities of the attached non-packet device.~~

~~Table 17 — Write command summary~~

~~Translated to ATA or SATA IIe write commands using the following criterion~~			
~~Is Queueing enabled?[a]~~	~~Are ATA Extended Commands enabled~~	~~FUA~~	~~Translated ATA command(s)[b]~~
~~No~~	~~No~~	~~N~~	~~WRITE DMA~~
~~No~~	~~No~~	~~Y~~	~~WRITE DMA, READ VERIFY SECTORS~~
~~No~~	~~Yes~~	~~N~~	~~WRITE DMA EXT[c]~~
~~No~~	~~Yes~~	~~Y~~	~~WRITE DMA FUA EXT~~
~~Yes~~	~~No~~	~~N~~	~~WRITE DMA QUEUED~~
~~Yes~~	~~No~~	~~Y~~	~~WRITE DMA QUEUED, READ VERIFY SECTORS~~
~~Yes~~	~~Yes~~	~~N~~	~~WRITE DMA QUEUED EXT[d]~~
~~Yes~~	~~Yes~~	~~Y~~	~~WRITE DMA QUEUED FUA EXT~~
~~Yes~~	~~Yes or No~~	~~Y or N~~	~~WRITE FPDMA QUEUED (see SATA IIe)[e]~~

~~[a] Refers to either the non-packet device support of the ATA/ATAPI-7 Overlapped feature set or the Native Command Queing (NCQ) feature of SATA IIe.~~

~~[b] The SATL may attempt WRITE MULTIPLE, WRITE MULTIPLE EXT, WRITE SECTOR(S) or WRITE SECTOR(S) EXT as a retry operation if the specified ATA or SATA IIe command fails. The selected ATA commands shall have enough bits in the logical sector address to avoid truncation of the LBA supplied in the SCSI CDB.~~

~~[c] The SATL may substitute WRITE DMA if the LBAs accessed can be represented in 28 bits.~~

~~[d] The SATL may substitute WRITE DMA QUEUED if the LBAs accessed can be represented in 28 bits.~~

~~[e] The SATL may use these commands only if NCQ is enabled (see SATA IIe).~~

This subclause applies to the translation of SCSI WRITE (6), WRITE (10), WRITE (12), and WRITE (16).

The SATL shall transfer the logical blocks specified in the SCSI write command (see 3.1.9) from the SCSI application client and shall issue ATA write commands (see 3.1.4) in accordance with the constraints specified in subclause 9.1 to transfer the specified logical blocks through the ATA host in the SATL to the non-packet device.

Data blocks specified in the LOGICAL BLOCK ADDRESS field shall be transferred to the specified non-packet device, and the device may transfer the data to its cache or medium. ~~CHECK CONDITION shall be reported back if~~ If the LOGICAL BLOCK ADDRESS or the LOGICAL BLOCK ADDRESS plus one less than the TRANSFER LENGTH is greater than the maximum sector that can be addressed in medium the SATL shall terminate the command with CHECK CONDITION status ~~where~~ with the sense key set to ILLEGAL REQUEST and additional sense code set to INVALID FIELD IN CDB.

Modify subclause 9.13 WRITE (10) command as shown below

9.18 WRITE (10) command (2Ah)

9.18.1 Command summary

The WRITE(10) command is used to request the device to transfer user data to device medium or cache. Different versions of the command support different LBA sizes and or different transfer lengths. Data may be written to medium or the device cache.

Table 18 — WRITE(10) command CDB fields

Field	SATType	Description or reference
OPERATION CODE	I	~~See 9.17.2~~See 9.18.2
WRPROTECT	U	See SBC-2
DPO (disable page out)	U	Ignored
FUA (force unit access)	~~U~~ I/E	~~See 9.17.2~~See 9.18.2
FUA_NV (force unit access nonvolatile cache)	U	The SATL shall not implement FUA_NV bit.
LOGICAL BLOCK ADDRESS	I	See 9.18.2 ~~A 32 bit LBA shall be derived from bytes 2 through 5, where byte 5 is the LSB and byte 2 is the MSB.~~
GROUP NUMBER	U	The SATL may implement this field as defined in SBC-2.
TRANSFER LENGTH	I	The transfer length shall be used to set the ATA sector count.[a] ~~The SATL shall issue as many ATA commands as needed to satisfy the transfer length specified by the WRITE (10) command. (see 9.17.2)~~ (see 9.18.2)
CONTROL	I	(see 6.4)

[a] A transfer length of zero indicates that a data transfer shall not take place. If the transfer length is not zero, the SATL shall transfer the number of sectors specified from the device to the application client.

9.18.2 WRITE command OPERATION CODE and FUA translation

This subclause applies to the translation of SCSI WRITE (10), WRITE (12), and WRITE (16).

If the FUA field is zero the SATL shall process this command as described in 9.17.2.

If the FUA bit is one the SATL shall issue an ATA write FUA command sequence (see 3.1.5) to the attached non-packet device in accordance with the constraints described in subclause 9.1.

Modify subclause 9.14 WRITE (12) command as shown below

9.19 WRITE (12) command (AAh)

9.19.1 Command summary

The WRITE(12) command is used to request the device to transfer user data to device medium or cache. Different versions of the command support different LBA sizes and or different transfer lengths. Data may be written to medium or the device cache.

Table 19 — WRITE(12) command CDB fields

Field	SATType	Description or reference
OPERATION CODE	I	~~See 9.17.2~~See 9.18.2
WRPROTECT	U	See SBC-2
DPO (disable page out)	U	Ignored
FUA (force unit access)	~~U~~ I/E	~~See 9.17.2~~See 9.18.2
FUA_NV (force unit access nonvolatile cache)	U	The SATL shall not implement FUA_NV bit.
LOGICAL BLOCK ADDRESS	I	See 9.18.2 ~~A 32 bit LBA shall be derived from bytes 2 through 5, where byte 5 is the LSB and byte 2 is the MSB.~~
GROUP NUMBER	U	The SATL may implement this field as defined in SBC-2.
TRANSFER LENGTH	I	The transfer length shall be used to set the ATA sector count.[a] ~~The SATL shall issue as many ATA commands as needed to satisfy the transfer length specified by the WRITE (12) command. (see 9.17.2)~~ (see 9.18.2)
CONTROL	I	(see 6.4)

[a] A transfer length of zero indicates that a data transfer shall not take place. If the transfer length is not zero, the SATL shall transfer the number of sectors specified from the device to the application client.

Add subclause 9.20 WRITE (16) command as shown below

9.20 WRITE (16) command (8Ah)

9.20.1 Command summary

The WRITE(16) command is used to request the device to transfer user data to device medium or cache. Different versions of the command support different LBA sizes and or different transfer lengths. Data may be written to medium or the device cache.

Table 20 — WRITE(16) command CDB fields

Field	**SATType**	**Description or reference**
OPERATION CODE	I	See 9.18.2
WRPROTECT	U	See SBC-2
DPO (disable page out)	U	Ignored
FUA (force unit access)	I/E	See 9.18.2
FUA_NV (force unit access nonvolatile cache)	U	The SATL shall not implement FUA_NV bit.
LOGICAL BLOCK ADDRESS	I	See 9.18.2
GROUP NUMBER	U	The SATL may implement this field as defined in SBC-2.
TRANSFER LENGTH	I	The transfer length shall be used to set the ATA sector count.[a] (see 9.18.2)
CONTROL	I	(see 6.4)

[a] A transfer length of zero indicates that a data transfer shall not take place. If the transfer length is not zero, the SATL shall transfer the number of sectors specified from the device to the application client.

Modify subclause 9.15 WRITE AND VERIFY (10) command as shown below

9.21 WRITE AND VERIFY(10) command (2Eh)

9.21.1 Command Summary

The WRITE AND VERIFY(10) command is used to transfer application data to medium and then to verify that data was written correctly.

Table 21 — WRITE AND VERIFY(10) command CDB fields

Field	SATType	Description or reference
OPERATION CODE	E	See 9.21.2 ~~The SATL shall issue an ATA write command, and then issue a read verify sector(s) command to the same range of sectors. The WRITE portion of the command translation shall be as specified for the WRITE(10) command (see 9.18).~~ ~~40h for read verify sector command if the device does not support extended commands feature set.~~ ~~42h for READ VERIFY EXTENDED command if the device supports the extended commands feature set.~~
WRPROTECT	U	See SBC-2
DPO (disable page out)	U	Ignored
BYTCHK	U	Ignored. The SATL shall assume a value of zero (i.e., no byte-by-byte comparison is performed).
LOGICAL BLOCK ADDRESS	I	See 9.21.2 ~~The SATL shall translate the LBA as specified for the WRITE(10) command (see 9.18)~~.
GROUP NUMBER	U	The SATL may implement this field as defined in SBC-2.
TRANSFER LENGTH	I	The transfer length shall be used to set the ATA sector count[a] (see 9.21.2).
CONTROL	I	(see 6.4)

[a] A transfer length of zero indicates that a data transfer shall not take place. If the transfer length is not zero, the SATL shall transfer the number of sectors specified from the device to the application client.

9.21.2 WRITE AND VERIFY command OPERATION CODE and FUA translation

This subclause applies to the translation of SCSI WRITE AND VERIFY (10), WRITE WRITE AND VERIFY (12), and WRITE WRITE AND VERIFY (16).

The SATL shall issue an ATA write command (see 3.1.4) followed by an ATA verify command (see 3.1.4) in accordance with the constraints defined in subclause 9.1.

Add subclauses 9.22 WRITE AND VERIFY (12) and 9.23 WRITE AND VERIFY (16) commands as shown

9.22 WRITE AND VERIFY(12) command (AEh)

9.22.1 Command Summary

The WRITE AND VERIFY(12) command is used to transfer application data to medium and then to verify that data was written correctly.

Table 22 — WRITE AND VERIFY(12) command CDB fields

Field	SATType	Description or reference
OPERATION CODE	E	See 9.21.2
WRPROTECT	U	See SBC-2
DPO (disable page out)	U	Ignored
BYTCHK	U	Ignored. The SATL shall assume a value of zero (i.e., no byte-by-byte comparison is performed).
LOGICAL BLOCK ADDRESS	I	See 9.21.2
GROUP NUMBER	U	The SATL may implement this field as defined in SBC-2.
TRANSFER LENGTH	I	The transfer length shall be used to set the ATA sector count[a] (see 9.21.2).
CONTROL	I	(see 6.4)

[a] A transfer length of zero indicates that a data transfer shall not take place. If the transfer length is not zero, the SATL shall transfer the number of sectors specified from the device to the application client.

9.23 WRITE AND VERIFY(16) command (8Eh)

9.23.1 Command Summary

The WRITE AND VERIFY(16) command is used to transfer application data to medium and then to verify that data was written correctly.

Table 23 — WRITE AND VERIFY(16) command CDB fields

Field	SATType	Description or reference
OPERATION CODE	E	See 9.21.2
WRPROTECT	U	See SBC-2
DPO (disable page out)	U	Ignored
BYTCHK	U	Ignored. The SATL shall assume a value of zero (i.e., no byte-by-byte comparison is performed).
LOGICAL BLOCK ADDRESS	I	See 9.21.2
GROUP NUMBER	U	The SATL may implement this field as defined in SBC-2.
TRANSFER LENGTH	I	The transfer length shall be used to set the ATA sector count[a] (see 9.21.2).
CONTROL	I	(see 6.4)

[a] A transfer length of zero indicates that a data transfer shall not take place. If the transfer length is not zero, the SATL shall transfer the number of sectors specified from the device to the application client.