

**Working  
Draft**

**T13  
D1572**

**Revision 2a  
August 24, 2004**

---

## **Information Technology - BIOS Enhanced Disk Drive Services - 3 (EDD-3)**

This is an internal working document of T13, a Technical Committee of Accredited Standards Committee NCITS. As such, this is not a completed standard and has not been approved. The T13 Technical Committee may modify the contents. The contents are actively being modified by T13. This document 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 document 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.

T13 Technical Editor:

Curtis E. Stevens  
Western Digital Corp.  
20511 Lake Forest Dr.  
Lake Forest, Ca. 92630  
USA

Tel: (949) 672-7933  
Email: [Curtis.Stevens@WDC.com](mailto:Curtis.Stevens@WDC.com)

---

Reference number

ANSI NCITS.\*\*\* - 200x

Printed August, 24, 2004 3:10PM

---

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

NCITS Secretariat

Administrator Standards Processing

1250 Eye Street, NW Suite 200

Washington, DC 20005

Tel: 202-737-8888

Fax: 202-638-4922

Email: NCITS@ITIC.ORG

T13 Reflector

See the T13 WEB site for reflector information.

T13 WEB site

[www.t13.org](http://www.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

## DOCUMENT STATUS

Revision 0 - August 14, 2002

Initial revision, document created from ANSI NCITS 363-2002

Added sections for CD/DVD boot

Revision 1 - February 19, 2004

Changed name from EDD-2 to EDD-3 in various places

Filled in CD/DVD boot sections

Added SAS definitions requested by Intel

Revision 1a - March 26, 2004

Updated BIOS functions 1-3F register and function descriptions and formatting

Revision 1b – April 20, 2004

Updated to include changes from April meeting

Revision 2 - June 23, 2004

Incorporated changes from E04138r1

Incorporated changes from E04133r1

Added Definition of AL to INT 13h Fn's 2, 3, 4, 0Ah, and 0Bh

Added Phoenix proposal for reporting HPA and 48 bit usage

Minor edits affecting grammar and capitalization

Changed document reference from FCP-3 to FC-FS

---

American National Standard  
For Information Systems -

## **BIOS Enhanced Disk Drive Services - 3**

Secretariat

**Information Technology Industry Council**

Approved mm dd yy

**American National Standards Institute, Inc.**

### **Abstract**

This standard describes services currently in use on IA-32 and IA-64 compatible architecture personal computer systems. These services are provided by BIOS firmware to support hard disks up to 16 megabytes ( $16 \times 10^{18}$  bytes). This standard also provides BIOS level services for determining the relationship between BIOS device numbers and the physical mass storage devices attached to the personal computer. The services defined in this standard can be applied to mass storage devices with ATA, ATAPI, SCSI, USB, Fibre Channel, 1394, I<sub>2</sub>O, and other interfaces. In addition, this standard defines the following:

1. Formatting required to make a CD or DVD bootable
2. INT 13 services used to access devices <8GB
3. addressing information for Infiniband, Serial ATA, Hypertransport, and PCI Express

# American National Standard

Approval of an American National Standard requires verification by ANSI that the requirements for due process, consensus, and other criteria for approval have been met by the standards developer. Consensus is established when, in the judgment of the ANSI Board of Standards Review, substantial agreement has been reached by directly and materially affected interests. Substantial agreement means much more than a simple majority, but not necessarily unanimity. Consensus requires that all views and objections be considered, and that effort be made towards their resolution.

The use of American National Standards is completely voluntary; their existence does not in any respect preclude anyone, whether he has approved the standards or not, from manufacturing, marketing, purchasing, or using products, processes, or procedures not conforming to the standards.

The American National Standards Institute does not develop standards and will in no circumstances give interpretation on any American National Standard. Moreover, no person shall have the right or authority to issue an interpretation of an American National Standard in the name of the American National Standards Institute. Requests for interpretations should be addressed to the secretariat or sponsor whose name appears on the title page of this standard.

**CAUTION NOTICE:** This American National Standard may be revised or withdrawn at any time. The procedures of the American National Standards Institute require that action be taken periodically to reaffirm, revise, or withdraw this standard. Purchasers of American National Standards may receive current information on all standards by calling or writing the American National Standards Institute.

**CAUTION:** The developers of this standard have requested that holders of patents that may be required for the implementation of the standard, 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 standard.

As of the date of publication of this standard and following calls for the identification of patents that may be required for the implementation of the standard, no such claims have been made.

The developer or the publisher in respect to any standard it processes conducts no further patent search. No representation is made or implied that licenses are not required to avoid infringement in the use of this standard.

Published by

**American National Standards Institute**

**11 West 42nd Street, New York, New York 10036**

Copyright 2nnn by American National Standards Institute

**All rights reserved.**

---



<b>Contents</b>	<b>Page</b>
INTRODUCTION.....	IV
1 SCOPE.....	1
2 NORMATIVE REFERENCES.....	2
2.1 APPROVED REFERENCES.....	2
2.2 REFERENCES UNDER DEVELOPMENT.....	2
2.3 OTHER REFERENCES.....	3
3 KEYWORDS, DEFINITIONS, ABBREVIATIONS, AND CONVENTIONS.....	3
3.1 KEYWORDS.....	3
3.2 DEFINITIONS AND ABBREVIATIONS.....	3
3.3 CONVENTIONS.....	7
4 OVERVIEW.....	8
5 <8GB INT 13H FUNCTIONS (LEGACY FUNCTIONS).....	8
6 >8GB INT 13H FUNCTIONS.....	9
6.1 DATA STRUCTURE.....	10
6.2 REMOVABLE MEDIA.....	10
6.3 INT 13H INTERFACE SUBSETS.....	11
7 CD/DVD BOOT.....	11
7.1 BOOT METHODS.....	14
7.2 CD/DVD FORMAT.....	16
8 INT 13H FUNCTION DEFINITIONS.....	20
8.1 RESET DISK SUBSYSTEM.....	20
8.2 GET STATUS OF LAST OPERATION.....	20
8.3 READ SECTORS INTO MEMORY.....	21
8.4 WRITE SECTORS FROM MEMORY.....	21
8.5 VERIFY SECTORS.....	21
8.6 GET DRIVE PARAMETERS.....	22
8.7 READ LONG.....	22
8.8 WRITE LONG.....	23
8.9 SEEK TO CYLINDER.....	23
8.10 READ BUFFER.....	23
8.11 WRITE BUFFER.....	24
8.12 RECALIBRATE.....	24
8.13 GET DISK TYPE.....	24
8.14 SET MEDIA TYPE.....	24
8.15 PARK HEADS.....	25
8.16 IDENTIFY DEVICE INFO.....	25
8.17 CHECK EXTENSIONS PRESENT.....	25
8.18 EXTENDED READ.....	26
8.19 EXTENDED WRITE.....	26
8.20 VERIFY SECTORS.....	27
8.21 LOCK/UNLOCK MEDIA.....	27
8.22 EJECT REMOVABLE MEDIA.....	27
8.23 EXTENDED SEEK.....	28
8.24 GET DEVICE PARAMETERS.....	29
8.25 GET EXTENDED MEDIA CHANGE STATUS.....	37
8.26 INITIATE DISK EMULATION.....	38
8.27 TERMINATE DISK EMULATION.....	39
8.28 INITIATE DISK EMULATION & BOOT.....	40
8.29 RETURN BOOT CATALOG.....	40
8.30 SET HARDWARE CONFIGURATION.....	41
8.31 SEND PACKET COMMAND.....	41
9 INT 15H REMOVABLE MEDIA EJECT.....	44

<b>Table</b>	<b>Page</b>
TABLE 1 - LEGACY REGISTER DEFINITIONS.....	9
TABLE 2 - LEGACY FUNCTIONS DEFINITIONS.....	9
TABLE 3 - DEVICE ADDRESS PACKET .....	10
TABLE 4 – FLOPPY GEOMETRIES .....	15
TABLE 5 - BOOT VOLUME DESCRIPTOR .....	16
TABLE 6 – VALIDATION ENTRY .....	16
TABLE 7 – INITIAL/DEFAULT ENTRY .....	17
TABLE 8 – SECTION HEADER .....	18
TABLE 9 – SECTION ENTRY .....	19
TABLE 10 – SECTION ENTRY EXTENSION .....	20
TABLE 11 - CHECK EXTENSIONS PRESENT BUFFER .....	26
TABLE 12 - RESULT BUFFER.....	29
TABLE 13 - INTERFACE PATH DEFINITIONS .....	31
TABLE 14 - DEVICE PATH DEFINITIONS.....	32
TABLE 15 - DEVICE PARAMETER TABLE EXTENSION .....	34
TABLE 16 - TRANSLATION TYPE .....	36
TABLE 17 - CD SPECIFICATION PACKET .....	38
TABLE 18 - CD COMMAND PACKET.....	40
TABLE 19 - HARDWARE CONFIGURATION SUB-FUNCTIONS.....	41
TABLE 20 - FORMATTED COMMAND PACKET.....	41
TABLE 21 - FORMATTED PROTOCOL SPECIFIC DATA .....	43
TABLE 22 - OUTPUT PARAMETERS .....	44

<b>Figure</b>	<b>Page</b>
FIGURE 1 - SYSTEM COMPONENT DIAGRAM.....	8
FIGURE 2 – ISO 9660 CD LAYOUT.....	12
FIGURE 3 – BOOT CD OR DVD WITH A SINGLE BOOT IMAGE .....	13
FIGURE 4 – BOOT CD OR DVD WITH MULTIPLE BOOT IMAGES .....	14

## Foreword

(This foreword is not part of American National Standard NCITS.xxx-200x)

DOS accesses its mass storage devices using an INT13 programming interface provided by BIOS firmware to higher-level software. This interface was designed in the early 1980's, upgraded in the late 1980's and upgraded again in the early 1990's. Initially, INT 13 used a cylinder-head-sector interface that was limited to approx. 528MB of disk addressing. Later this same interface was upgraded to support approx. 8.4GB of disk addressing. Finally, extended functions were created that allowed 64-bit disk addressing. At this same time, specifications were created that allowed the CD or DVD ROM drive to boot using the same INT 13 interface that hard drives used. The Disk and CD/DVD functions have never been fully documented in a standards forum. This standard enables the following capabilities:

- CHS addressing with an 8.4GB limitation
- Logical Block Addressing (LBA).
- Removes the requirement of using interrupt 41h/46h to point at the Fixed Disk Parameter Table information.
- Makes location and configuration information available to operating systems that do not use the BIOS to access mass storage devices.
- Use data structures that apply to both IA-32 and IA-64 compatible architecture systems.
- Use data structures that can address media capacities for the next 20 years.
- Boot from CD or DVD.

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 I Street NW, Suite 200, Washington, DC 20005-3922.

This standard was processed and approved for submittal to ANSI by National Committee for Information Technology Standardization (NCITS). Committee approval of this standard does not necessarily imply that all committee members voted for approval. At the time it approved this standard, NCITS had the following members:

[NCITS membership goes here]

Technical Committee T13 on ATA Interfaces, that reviewed this standard, had the following members:

[T13 membership goes here]

## **Introduction**

This standard encompasses the following:

Clause 1 describes the scope.

Clause 2 provides normative references.

Clause 3 provides definitions, abbreviations, and conventions.

Clause 4 is the overview.

Clause 5 is INT 13h functions limited to 8GB if disk addressing.

Clause 6 is INT 13h functions not limited to 8GB of disk addressing.

Clause 7 is CD/DVD Booting

Clause 8 is INT 13 function definitions

Clause 9 is INT 15H Removable Media Eject Functions

American National Standard  
for Information Systems -

## Information Technology - **BIOS Enhanced Disk Drive Services - 3 (EDD-3)**

### 1 Scope

This standard assumes that the reader is familiar with the conventional INT 13h interface, the usage of the BIOS Device Parameter Table, and the basic operation of mass storage devices. This standard describes in detail BIOS functions and data structures that are used as an abstraction layer to allow higher-level applications to access mass storage devices in an interface and command-set independent manner. To comply with this standard, higher-level software shall call the INT functions using the data structures described herein, and system firmware shall provide the INT functions and data structures described herein.

The storage industry has increased the capacity and functionality of many types of mass storage devices. This increase in capacity and functionality has required the development of a BIOS interface. This standard documents the BIOS interface that is supplied by many BIOS vendors. This standard defines solutions to the following INT 13h BIOS-specific issues:

- The INT 13h interface has a limit of 528 megabytes (MB);
- The INT 13h interface allows more than two devices to be attached to a system but has no consistent method for storing the additional configuration parameters;
- The INT 13h interface does not define CHS-independent methods for addressing devices. The methods defined by the INT 13h interface are not device-geometry independent. A different method of address representation and operation is needed;
- Methods of data transfer continue to be added to ATA devices. Capabilities such as, DMA modes, multi-sector data transfers and PIO modes are not reported to the operating system via the INT 13h interface;
- Systems require more than two storage devices, and with this requirement comes the requirement to assign the order in which the devices are to be accessed. The INT 13 interface does not provide this capability;
- The INT 13h interface does not make location and configuration information available to operating systems that do not use the BIOS to access mass storage devices;
- The INT 13h interface does not provide a linkage between the BIOS device assignments on the operating system device letter assignments;
- The INT 13h interface does not use data structures that apply to both IA-32 and IA-64 compatible architecture systems.

## 2 Normative References

The following standards contain provisions that, through reference in the text, constitute provisions of this standard. Many of these standards are referenced because they contain the information necessary for describing a method of accessing a device on the specified interface. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this standard are encouraged to investigate the possibility of applying the most recent editions of the standards listed below.

Copies of the following documents can be obtained from ANSI: Approved ANSI standards, approved and draft international and regional standards (ISO, IEC, CEN/CENELEC, ITUT), and approved and draft foreign standards (including BSI, JIS, and DIN). For further information, contact ANSI Customer Service Department at 212-642-4900 (phone), 212-302-1286 (fax) or via the World Wide Web at <http://www.ansi.org>.

Additional availability contact information is provided below as needed.

### 2.1 Approved References

The following approved ANSI standards and technical reports, approved international and regional standards and technical reports (ISO, IEC, CEN/CENELEC, ITUT), may be obtained from the international and regional organizations who control them.

INCITS 373-2003 *Fibre Channel Framing and Signaling Interface (FC-FS)*

INCITS 361-2002 *AT Attachment with Packet Interface - 6 (ATA/ATAPI-6)*

INCITS 376-2003 *Serial Attached SCSI (SAS)*

INCITS 325-1998 *Serial Bus Protocol - 2 (SBP-2)*

INCITS 330-2003 *Reduced Block Commands (RBC)*

INCITS 351-2001 *SCSI Primary Commands - 2 (SPC-2)*

INCITS 360-2002 *Multi-Media Commands - 3 (MMC-3)*

INCITS 363-2002 *BIOS Enhanced Disk Drive Services - 2 (EDD-2)*

ANSI/IEEE 394:1995 *Extended Unique Identifier, 64-bit (EUI-64)*

IEEE 1394-1995 *IEEE Standard for a High Performance Serial Bus*

IEEE 1394a-2000 *IEEE Standard for a High Performance Serial Bus - Amendment 1*

ISO 9660-1988 *Volume and File Structure of CD-ROM For Information Interchange*

### 2.2 References Under Development

At the time of publication, the following referenced standards were still under development. For information on the current status of the document, or regarding availability, contact the relevant standards body or other organization as indicated.

INCITS 1532D *AT Attachment with Packet Interface - 7 (ATA/ATAPI-7)*

INCITS 1416D *SCSI Primary Commands - 3 (SPC-3)*

INCITS 1545D *Multi-Media Commands - 4 (MMC-4)*

IEEE P1394b *Standard for High Performance Serial Bus (High Speed Supplement)*

INCITS 1467D *Serial Bus Protocol - 3 (SBP-3)*

INCITS 1601D *Serial Attached SCSI - 1.1 (SAS-1.1)*

For more information on the current status of the above documents, contact NCITS. To obtain copies of these documents, contact Global Engineering or NCITS.

## 2.3 Other References

The following standards and specifications were also referenced.

NCITS TR-21 *BIOS Enhanced Disk Drive Technical Report*

BIOS Boot Specification (Compaq, Phoenix and Intel), [www.phoenix.com](http://www.phoenix.com)

EI Torito CD-ROM Boot Specification, [www.phoenix.com](http://www.phoenix.com)

ATAPI Removable Media BIOS Specification, [www.phoenix.com](http://www.phoenix.com)

Universal Serial Bus Revision 1.1, [www.usb.org/developers](http://www.usb.org/developers)

Mass Storage Overview, [www.usb.org/developers](http://www.usb.org/developers)

I<sub>2</sub>O Software Specification v2.0, [www.i2osig.org](http://www.i2osig.org)

## 3 Keywords, Definitions, Abbreviations, and Conventions

### 3.1 Keywords

Several keywords are used to differentiate between different levels of requirements and optionality.

#### 3.1.1 Mandatory

A keyword indicating items to be implemented as defined by this standard.

#### 3.1.2 May

A keyword that indicates flexibility of choice with no implied preference.

#### 3.1.3 Optional

A keyword that describes features that are not required by this standard. However, if any optional feature defined by the standard is implemented, it shall be done in the way defined by the standard. Describing a feature as optional in the text is done to assist the reader.

#### 3.1.4 Reserved

A keyword indicating reserved bits, bytes, words, fields, and code values that are set aside for future standardization. Their use and interpretation may be specified by future extensions to this or other standards. A reserved bit, byte, word, or field shall be set to zero, or in accordance with a future extension to this standard. The recipient shall not check reserved bits, bytes, words, or fields. Receipt of reserved code values in defined fields shall be treated as an error.

#### 3.1.5 Shall

A keyword indicating a mandatory requirement. Designers are required to implement all such mandatory requirements to ensure interoperability with other standard conformant products.

#### 3.1.6 Should

A keyword indicating flexibility of choice with a strongly preferred alternative. Equivalent to the phrase "it is recommended".

### 3.2 Definitions and Abbreviations

For the purposes of this standard, the following definitions apply:

### 3.2.1 ATA

An AT Attachment (also known as IDE) is a storage interface that conforms to an ATA standard.

### 3.2.2 BDA

The BIOS Data Area is an area of reserved memory used by the BIOS and O/S to store data about the system hardware. It is located at memory segment 40h starting with 40h:00h.

### 3.2.3 BIOS

The Basic Input/Output System is the firmware embedded on a chip located on the computer's main board. The BIOS executes POST to test and initialize the system components and then loads the O/S. The BIOS also handles the low-level Input/Output to the various peripheral devices connected to the computer.

### 3.2.4 Byte

A byte is a unit of data that consists of eight bits as described below:

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
-------	-------	-------	-------	-------	-------	-------	-------

### 3.2.5 CF

The Carry Flag is a bit in the Flags Register in the microprocessor.

### 3.2.6 CHS

CHS addressing is a method of addressing the contents of a storage device using logical cylinders (C), logical heads (H), and logical sectors (S). This method of addressing allows a maximum C=16,383, H=16, S=63, resulting in a maximum device capacity of 8.4 gigabytes (16,383 \* 16 \* 63 \* 512 bytes per sector  $\approx$  8.4 gigabytes). . See LBA addressing for another addressing method.

### 3.2.7 Conventional and enhanced

When a word, term, or phrase is modified by the word "conventional" it refers to the legacy style, or method of operation that is limited to addressing ATA devices that have a 528 MB capacity or less. When a word, term, or phrase is modified by the word "enhanced" it means there is a "conventional" and an "enhanced" method of operation. The "enhanced" method is defined by this standard.

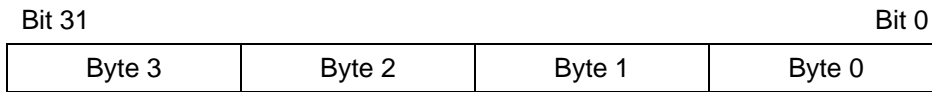
### 3.2.8 DOS

DOS is a disk operating system that uses the system BIOS as a firmware abstraction layer to access system hardware. Examples of DOS based operating systems include MS-DOS®, DR-DOS®, PC-DOS®, Free DOS, Windows® 3.11, and Windows® 95.

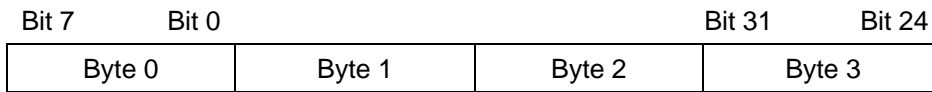
**3.2.9 DWord**

A DWord (Double Word) is a unit of data that consist of four bytes. This data may be represented on paper as a series of bits numbered from 31 to 0. In memory byte 0 of a DWord is stored in the lowest byte address and Byte 3 is stored in the highest byte address.

**On Paper:**



**In Memory:**



**3.2.10 Host**

The Host is the PC that is controlled by the BIOS.

**3.2.11 IA-32 compatible**

IA-32 compatible refers to CPU architectures that are compatible with the Intel Architecture 32-bit wide processor and data bus.

**3.2.12 IA-64 compatible**

IA-64 compatible refers to CPU architectures that are compatible with the Intel Architecture 64-bit wide processor and data bus.

**3.2.13 INT 13h**

A BIOS interrupt service that provides a protocol independent method for addressing floppy, hard drive, and other storage devices.

**3.2.14 INT 40h**

INT 40h is a BIOS interrupt service that provides a protocol independent method for addressing INT 13h devices that have a device number less than or equal to 7Fh.

**3.2.15 IPL Device**

An Initial Program Load Device is any device in the system that may boot and load an O/S. In standard AT machines, this is normally the floppy drive or hard drive.

**3.2.16 LBA**

LBA is a method of addressing a device that involves using a Logical Block Address. For example, this method of addressing allows a maximum sector address of  $2^{28}-1$ , or 137.4 GB of data on an ATA device using 28-bit addressing. See CHS for another address method.

**3.2.17 Logical Address/Geometry**

A logical address or geometry is used to address a device by an application, such as DOS, using the INT 13h interface. INT 13h Fn 8 returns the logical geometry of the device.

**3.2.18 NV Memory**

Non-Volatile memory is memory that retains content even when the power has been removed. The most common type of NV memory on a PC is the CMOS RAM that is used to store system configuration information.

**3.2.19 O/S**

An Operating System is a software abstraction layer that provides services that give applications access to system hardware, in a hardware independent fashion. Examples of these services include memory management, multi-threaded task management, file system management, printer management, and screen management.

**3.2.20 POST**

The Power-On Self-Test is the part of the BIOS that takes control immediately after power is applied to the computer. POST initializes the computer hardware so that an O/S may be loaded.

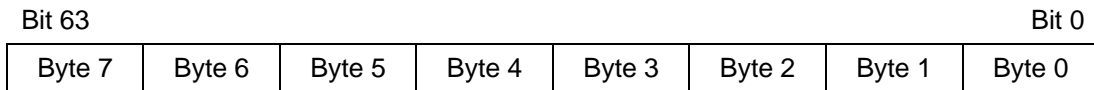
**3.2.21 Protected Mode**

x86 compatible processors have several modes of main memory addressing. One of these modes is called Protected Mode. In this mode all the system memory can be addressed. Another mode is Real Mode. See section 3.2.23 for the definition of Real Mode.

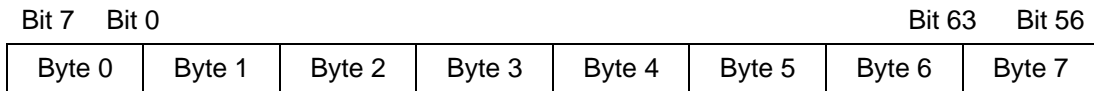
**3.2.22 QWord**

A QWord (Quad Word) is a unit of data that consist of eight bytes. This data may be represented on paper as a series of bits numbered from 63 to 0. In memory byte 0 of a QWord is stored in the lowest byte address and Byte 7 is stored in the highest byte address.

**On Paper:**



**In Memory:**



**3.2.23 Real Mode**

x86 compatible processors have several modes of main memory addressing. One of these modes is called Real Mode. In this mode, systems can only address the first mega-byte of memory. Another mode is called Protected Mode. See section 3.2.21 for the definition of Protected Mode.

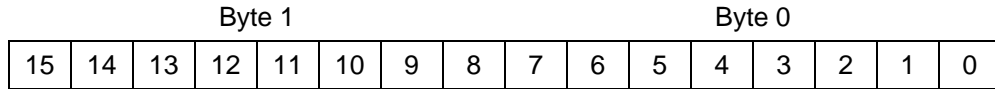
**3.2.24 Standard Floppy Drive**

The Standard Floppy Drive is the generic term to define the currently used 5.25 inch floppy drives and the 3.5 inch floppy diskette drives found in many systems.

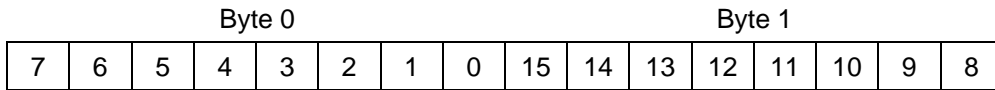
**3.2.25 Word**

A Word is a unit of data that consist of two bytes. This data may be represented on paper as a series of bits numbered from 15 to 0. In memory byte 0 of a Word is stored in the lower byte address and Byte 1 is stored in the higher byte address.

**On Paper:**



**In Memory:**



**3.3 Conventions**

Lowercase is used for words having the normal English meaning. Certain words and terms used in this standard have a specific meaning beyond the normal English meaning. These words and terms are defined either in clause 3 or in the text where they first appear.

The names of abbreviations, commands, fields, and acronyms used as signal names are in all uppercase (e.g., IDENTIFY DEVICE). Fields containing only one bit are usually referred to as the “name” bit instead of the “name” field. (see 3.2.6 for the naming convention used for naming bits.)

Names of device registers begin with a capital letter (e.g., Cylinder Low register).

**3.3.1 Numeric Notation**

Numbers are specified in three different bases throughout this standard: binary (base 2), decimal (base 10), and hexadecimal (base 16). The following notation is used to indicate that base in use:

Base	Notation	Example
Binary	nnnb	1001010b
Decimal	nnn	74
Hexadecimal	nnnh	4Ah

**3.3.1.1 Binary**

Binary numbers use the digits: 0, 1. An example of a binary number is 0101b, this binary number is 5 decimal or 5h hexadecimal.

**3.3.1.2 Decimal**

Decimal numbers use the digits: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9. An example of a decimal number is 1024, this is 01000000000b in binary and 400h in hexadecimal

**3.3.1.3 Hexadecimal**

Hexadecimal numbers use the digits: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F. An example of a hexadecimal number is 1Fh, this is 00011111b in binary and 31 in decimal.

**3.3.2 Register and CF Contents**

The value contained in a register is expressed as “register name = value” (e.g., AH = 01h). The value contained in CF (the Carry Flag) is expressed as “CF = value” (e.g., CF = 1b).

## 4 Overview

In the past, DOS has accessed its mass storage devices using a BIOS provided INT 13h interface. This interface was designed in the early 1980's and upgraded in the late 1980's. The maximum theoretical capacity of this Applications Program Interface (API) shall be 8.4 GB. The INT 13h interface, now known as the legacy INT 13h interface, uses function numbers 1 through 15h and is Cylinder-Head-Sector (CHS) oriented. An extended INT 13h interface has been created. The purpose of these INT 13h extensions shall be to:

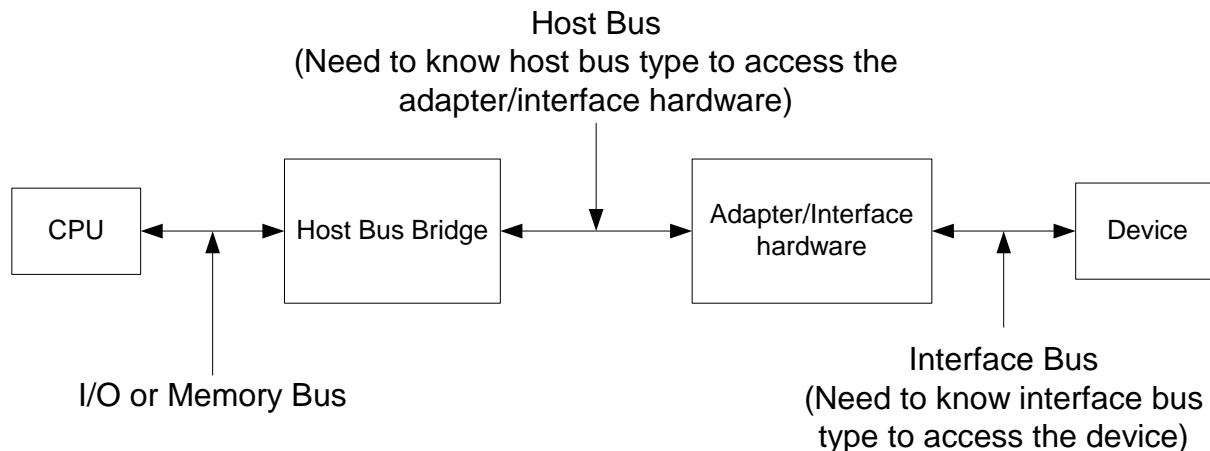
- Replace CHS addressing with Logical Block Addressing (LBA).
- Remove the current requirement of using interrupt 41h/46h to point at the Fixed Disk Parameter Table information.
- Make location and configuration information available to operating systems that do not use the BIOS to access mass storage devices.
- Use data structures that apply to both IA-32 and IA-64 compatible architecture systems.
- Use data structures that can address media capacities for the next 20 years.

Many BIOS, Option ROM, and OS vendors have already implemented the functions defined in this document for ATA and SCSI style devices. This standard builds on NCITS TR-21 BIOS Enhanced Disk Drive Technical Report to enable other mass storage technologies, such as 1394, Fibre Channel, and USB.

DOS and other operating systems, such as Windows™ 98, Windows™ NT, Windows™ 2000, and Windows™ XP, shall gain the capability to consistently provide the same drive letter assignments to the user. The result of this capability shall be that storage devices can be added to an EDD system, and the existing drive letters shall not change.

Data written on media can render the media incompatible with certain drive letters when some drive letter based operating systems are used. Technologies, such as 1394, blur the difference between fixed and removable media. Operating systems shall allow removable media devices to work equally well when named "A:", "B:", "C:", as well as "D:" and other letters.

One of the important aspects of this standard is to allow a BIOS to describe the physical path to a device. The following diagram shows the basic system components that are referenced by this standard.



**Figure 1 - System Component Diagram**

## 5 <8GB INT 13h Functions (Legacy Functions)

The Legacy INT 13h functions can address a maximum of 8.4GB of device space. Sector sizes shall be exactly 512 bytes. The Legacy INT 13h functions pass all parameters in registers using the following template (see Table 1) unless otherwise specified:

Table 1 - Legacy Register Definitions

Register	Description						
AH	INT 13h function number (Range is 00h -15h)						
AL	Number of sectors to transfer						
CH	Low 8 bits of Cylinder						
CL	<table border="1"> <thead> <tr> <th>Bit</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0-5</td> <td>Sector number. This value is 1 based. 0 is an illegal value.</td> </tr> <tr> <td>6-7</td> <td>High order 2 bits of the cylinder</td> </tr> </tbody> </table>	Bit	Description	0-5	Sector number. This value is 1 based. 0 is an illegal value.	6-7	High order 2 bits of the cylinder
Bit	Description						
0-5	Sector number. This value is 1 based. 0 is an illegal value.						
6-7	High order 2 bits of the cylinder						
DH	Head Number						
DL	<table border="1"> <thead> <tr> <th>Bits</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0-6</td> <td>Drive Number</td> </tr> <tr> <td>7</td> <td>Set to 1 for Fixed media Clear to 0 for removable media</td> </tr> </tbody> </table>	Bits	Description	0-6	Drive Number	7	Set to 1 for Fixed media Clear to 0 for removable media
Bits	Description						
0-6	Drive Number						
7	Set to 1 for Fixed media Clear to 0 for removable media						
ES:BX	Buffer pointer. Commands that move data from the device to memory shall use this pointer as the destination for the data. Commands that move data from memory to the device shall use this pointer as the source for the data to be transferred.						

The following legacy functions shown in Table 2 are documented in this standard:

Table 2 - Legacy Functions Definitions

Function	Description
00h	Reset disk subsystem
01h	Get status of last operation
02h	Read Sectors into memory
03h	Write sectors from memory
04h	Verify sector
08h	Get Drive Parameters
0Ah	Read Long
0Bh	Write Long
0Ch	Seek to cylinder
0Eh	Read Buffer
0Fh	Write Buffer
11h	Recalibrate Drive
15h	Get Disk Type
18h	Set Media Type
19h	Park heads
25h	IDENTIFY DEVICE info

## 6 >8GB INT 13h Functions

The extended INT 13h functions are numbered from 41h through 49h, 4E, and 50h. These functions are different from the legacy INT 13h interface in the following ways:

- Register conventions have been changed to support the passing of data structures;
- All media addressing information shall be passed via a buffer, not registers;
- Flags are used to identify optional capabilities.

The caller shall specify a device number in the DL register when calling functions 41h through 48h. This is a logical number that has been assigned to the physical device by system firmware. The physical device addressing information can be retrieved via INT 13h FN 48h. The following registers in IA-32 and IA-64 compatible systems are used: AX, BX, CX, DX, DS, and SI.

## 6.1 Data Structure

The data structure for the INT 13h extensions shall be the device address packet. INT 13h converts addressing information in the device address packet to default parameters appropriate to the media. Table 3 defines the device address packet.

**Table 3 - Device Address Packet**

Offset	Type	Description
0	Byte	Packet size in bytes. The value in this field shall be 16 (10h) or greater. If the packet size is less than 16 the request shall be rejected with CF = 1b and AH = 01h.
1	Byte	Reserved. The value in this field shall be 00h
2	Byte	Number of blocks to transfer. This field shall contain a maximum value of 127 (7Fh). If a any other value is supplied, the request shall be rejected with CF=1b and AH=01h. If this field is set to FFh, then the transfer buffer address shall be found at offset 10h, the number of blocks to transfer shall be found at offset 18h, and the transfer buffer at offset 4 shall be ignored. A block count of 0 means no data shall be transferred.
3	Byte	Reserved. The value in this field shall be 00h.
4	DWord	Address of host transfer buffer. This is the host buffer that Read/Write operations shall use to transfer the data. This is a 32-bit host address of the form Seg:Offset. If this field is set to FFFFh:FFFFh then the address of the transfer buffer shall be found at offset 10h.
8	QWord	<p>Starting logical block address, on the target device, of the data to be transferred. This is a 64-bit unsigned linear address. If the device supports LBA addressing this value should be passed unmodified. If the device does not support LBA addressing the callee shall convert this LBA to a CHS address using the current geometry in the following formula:</p> $LBA = (C_1 * H_0 + H_1) * S_0 + S_1 - 1$ <p>Where:</p> <ul style="list-style-type: none"> <li>C<sub>1</sub> = Selected Cylinder Number</li> <li>H<sub>0</sub> = Number of Heads (Maximum Head Number + 1)</li> <li>H<sub>1</sub> = Selected Head Number</li> <li>S<sub>0</sub> = Maximum Sector Number</li> <li>S<sub>1</sub> = Selected Sector Number</li> </ul> <p>For ATA compatible devices, with less than or equal to 15,482,880 logical sectors, the H<sub>0</sub> and S<sub>0</sub> values are supplied by words 3 and 6 of the data returned as a result of an IDENTIFY DEVICE command.</p>
10h	QWord	64-bit unsigned linear address of the host transfer buffer. This is the host buffer that Read/Write operations shall use to transfer the data if the data at offset 4 is set to FFFFh:FFFFh, or the data at offset 2 is set to FFh.
18h	DWord	Total number of blocks to transfer when the data at offset 2 is set to FFh
1Ch	DWord	Reserved. The value in this field shall be 00h.
<p>NOTE - The options described in table 1 allow a host to use a 7-bit transfer size with a 32 -bit or 64-bit memory address for the transfer buffer. Table 1 also allows a 32-bit transfer size in conjunction with a 64-bit address. A 32-bit transfer size shall not be used in conjunction with a 32-bit memory address.</p>		

## 6.2 Removable Media

The distinction between "removable" disks numbered 00h through 7Fh and "fixed" disks numbered 80h through FFh differs from conventional INT 13h functions. Devices numbered 0 through 7Fh are not changed. They follow conventional INT 13h standards for floppy disk operation. Devices numbered 80h through FFh include traditional fixed disks, and now also include removable media devices that support media change notification as well as

software locking and unlocking capabilities. Functions in this standard support these devices. Return codes defined for the conventional INT 13h interface are vendor specific and may be used. In addition the following return codes support removable media:

Return Code	Description
B0h	Media Not Locked In Device
B1h	Media Locked In Device
B2h	Media Not Removable
B3h	Media In Use
B4h	Lock Count Exceeded
B5h	Valid Eject Request Failed
B6h	Media Present but Read Protected

### 6.3 INT 13h Interface Subsets

It shall be permissible for BIOS to support only certain subsets of the INT 13h extensions. These subsets are defined in this standard. If a subset is supported then all functions within that subset shall be supported. The supported subsets shall be determined via the Check Extensions Present function. If a function is not supported and that function is subsequently invoked; then the function rejects the request with CF = 1b and AH = 01h. There are three subsets defined. At least one of these shall be supported.

NOTE - Conventional INT 13h may be used for media accessing if the Fixed Disk Access Subset is not present.

#### 6.3.1 Fixed Disk Access Subset

These functions support basic access to devices using the device address packet structure as follows:

Function	Code
Check Extensions Present	41h
Extended Read	42h
Extended Write	43h
Verify Sectors	44h
Extended Seek	47h
Get Device Parameters	48h

#### 6.3.2 Device Locking and Ejecting Subset

These functions support software control of media locking and ejecting as follows:

Function	Code
Check Extensions Present	41h
Lock/Unlock Media	45h
Eject Device	46h
Get Device Parameters	48h
Get Extended Disk Change Status	49h
The INT 15h Removable Media Eject Intercept	na

#### 6.3.3 Enhanced Disk Drive (EDD) Support Subset

These functions provide EDD support as follows:

Function	Code
Check Extensions Present	41h
Get Parameters With EDD Extensions	48h

## 7 CD/DVD Boot

CD and DVD media shall be bootable if the media follows the formatting described here. This format is based on ISO-9660, but does not require the media to be ISO-9660 compliant. Boot media shall have an ISO-9660 Boot Volume Descriptor (BVD) at sector 17 relative to the start of the last session on the CD or DVD media. See

section 7.2.1 for a description of the BVD. Other ISO-9660 constructs, such as the “Primary Volume Descriptor” and the “Set Terminator” are not required for compliance to EDD, but are required if the CD or DVD contains material that can be accessed by ISO-9660 compliant drivers. Figure 2 shows a standard ISO-9660 image overview.

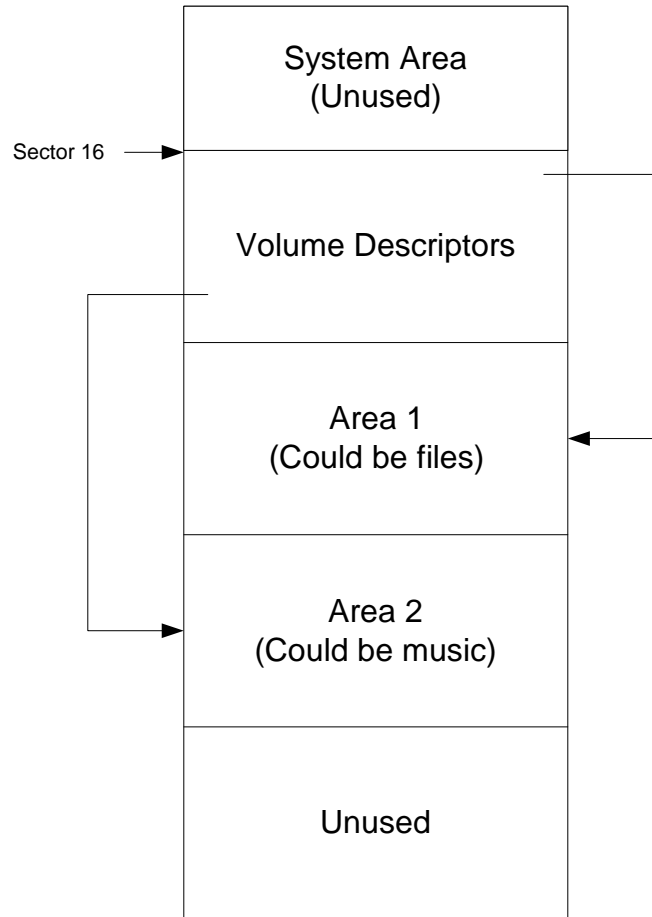


Figure 2 – ISO 9660 CD Layout

Figure 3 Describes a CD or DVD with an Initial/Default Entry but no section or additional boot images. This format is common for CD's or DVD's that are used for installing an operating system.

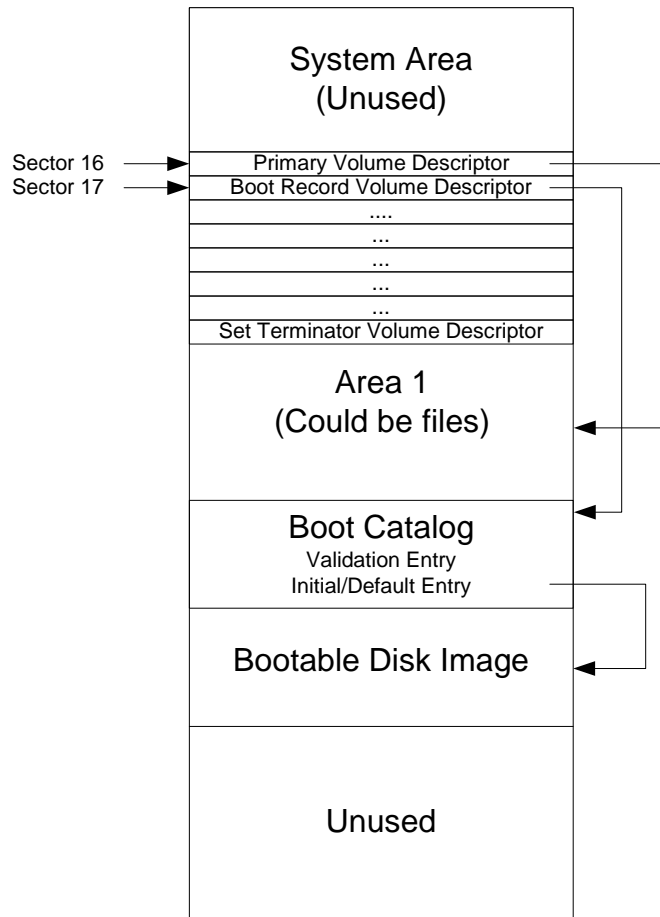


Figure 3 – Boot CD or DVD with a single boot image

Figure 4 illustrates a multi-session CD or DVD with several boot images. This type of CD or DVD may be used in a recovery environment that has several different system types. The Initial/Default entry could be a program that determines the system type and then chooses the appropriate boot image.

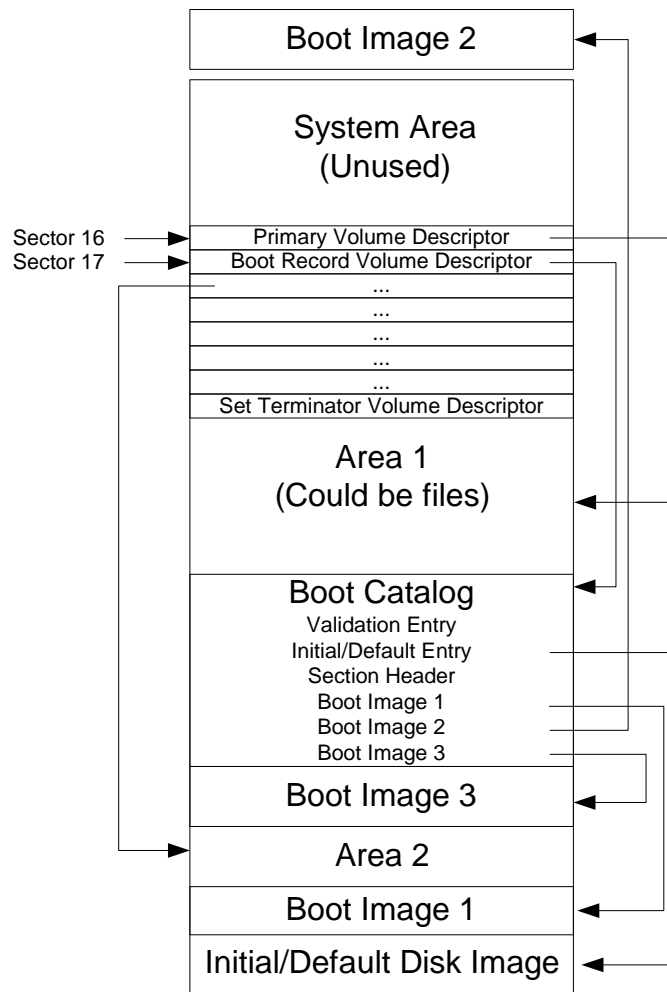


Figure 4 – Boot CD or DVD with multiple boot images

## 7.1 Boot Methods

Bootable CD's and DVD's shall have bootable images that follow one of the following media types.

1. No Emulation Boot
2. 1.22MB Floppy
3. 1.44MB Floppy
4. 2.88MB Floppy
5. Hard Disk

Each of these formats has different requirements for booting. INT 13h Functions 0-3Fh shall operate in the emulated sector size of 512 bytes. Addressing for these functions shall be in 512 byte increments relative to the start of the emulated image. For example, if a boot image starts at sector 100 on a CD, INT 13h sectors 0-3 would reside at sector 100 offset 0, 512, 1024, and 1536 bytes respectively on the CD.

### 7.1.1 Emulation Process

CD's and DVD's use a sector size that is 2KB. Floppy disks and hard drives commonly use a sector size of 512 bytes. In an emulated bootable CD or DVD environment, the system BIOS shall assign a device number to the bootable image. INT 13h Functions 0-3Fh shall continue to operate with a 512-byte sector size. The data that is on the floppy or hard disk drive (HDD) shall be packed four sectors of data to each sector on the CD or DVD.

This packing shall be performed in a linear fashion so that floppy or HDD sectors 0-3 become the first sector in the image on the CD or DVD. Packing shall continue so that floppy or HDD sectors 4-7 are the second sector on the CD or DVD, floppy or HDD sectors 8-11 are the third sector in the image on the CD or DVD. This process is repeated until all the data is on the CD or DVD. INT 13h Functions 0-3Fh shall then unpack the data based on the calling parameters. This method allows software on the CD or DVD to use the system BIOS without taking into account the CD or DVD hardware or sector size. This method also allows a floppy or HDD image to be tested and debugged using a normal floppy or HDD. Once the image on the media is booting and operating properly, the image can then be transferred to the CD or DVD with a high degree of confidence that it will function properly.

INT 13 Functions 41h-49h shall access the CD or DVD using non-emulated sector LBA's in the native sector size of the CD or DVD. INT 13h Function 48h shall return the total number of 2KB sectors on the CD or DVD and shall set the flag that indicates the geometry is invalid. This capability allows software to use generic ISO-9660 or other file system drivers without requiring a CD or DVD hardware specific driver. This capability shall allow the software to boot and run, provided the software uses INT 13 functions exclusively, regardless of the interface bus type. This means that a CD or DVD boot image may be constructed that works the same on ATAPI, Fibre Channel, SCSI, USB, 1394, and other buses. If the system is EDD-3 compliant, the Initial/Default Image could make an INT 13h Function 48h call, determine the interface bus type, and then boot an image with native CD or DVD drivers to perform the requested operation.

**7.1.2 No Emulation Boot**

If an image is marked as No Emulation Boot, the system BIOS shall load the specified number of sectors and jump to the specified segment number to start the boot program executing. Since this is a no emulation boot, the system BIOS shall not provide access to the media through INT 13h Functions 0-3Fh. The system BIOS shall assign the CD-ROM drive a device number. A common industry number is A0h. This is outside the range of normal BIOS hard drive numbers (above 9Ah), but still within INT 13h range limitation.

**7.1.3 Floppy Boot**

When the system attempts to boot a Floppy based image, it shall assign device number 0 to the image. If there is one or more standard floppy drives present in the system, they shall be sequentially numbered by the BIOS starting from 1 and increasing.

EDD-3 supports 3 different floppy formats: 1.2MB, 1.44MB, and 2.88MB. All of these formats can be created on a floppy disk first, tested, and then transferred to the target CD or DVD. See Table 4 for the geometries INT 13h Function 8h shall return for each floppy type.

**Table 4 – Floppy Geometries**

Type	Cylinders	Heads	Sectors
1.2MB	80	2	15
1.44MB	80	2	18
2.88MB	80	2	36

The total capacity of a CD is more than 600MB. Many floppy images may be embedded on one CD or DVD, since floppy images shall be 2.88MB or less. One reason for making a bootable floppy image is to boot the system, load a CD or DVD specific device driver and then provide access to all the media. This capability allows a simple OS, such as DOS, to be used for accessing a self-contained CD. One application for this capability is system recovery or system restore.

**7.1.4 Hard Drive Boot**

When the system attempts to boot a Hard Disk Drive (HDD) based image, it shall assign device number 80h to the image. If there are one or more standard disk devices present in the system, they shall be numbered sequentially starting at 81h and increasing.

The geometry returned by INT 13h Function 08h is based on the partition table embedded in the first sector of the HDD image. The partition type of HDD images shall include cylinder-head-sector (CHS) addresses. This may be

FAT-16 or FAT-32; however, any partition type that supports CHS is acceptable. The partition table shall only have one partition defined.

## 7.2 CD/DVD Format

The system BIOS can boot from a CD or DVD only if the format of the media has been predefined. ISO-9660 is a basic CD-ROM file system that has been incorporated into current CD and DVD file system standards. CD/DVD boot uses ISO 9660 as a framework to provide boot capability. ISO 9660 requires that a set of volume descriptors reside on the media starting at sector 10h. For data CD's, there will usually be a "Primary Volume Descriptor" at sector 10h followed by a "Set Terminator" at sector 11h. The "Primary Volume Descriptor" provides a pointer to the directory structures. ISO 9660 has also defined a "Boot Volume Descriptor" (BVD). BIOSes check the BVS to determine if a CD or DVD is bootable. A CD or DVD shall have a BVS at sector 11h on the media. The "Set Terminator" shall reside at sector 12h or higher. ISO 9660 requires that a "Set Terminator" be present. ISO 9660 formatting need not be present for a CD or DVD to be bootable. The only requirement is that a BVS be present at sector 11h.

### 7.2.1 Boot Volume Descriptor

ISO 9660 does not define the payload for its BVD. EDD-3 provides the necessary formatting to enable bootability. The BIOS shall check the data in the BVD to determine if boot is possible. See Table 5 for the BVD format.

**Table 5 - Boot Volume Descriptor**

Offset	Type	Description
00h	Byte	0 - Boot Record Indicator
01h-05h	Byte	43h 44h 30h 30h 31h. "CD001" - ISO-9660 specification identifier
06h	Byte	1 - Version of this descriptor
07h-26h	Byte	45h 4Ch 20h 54h 4Fh 52h 49h 54h 4Fh 20h 53h 50h 45h 43h 49h 46h 49h 43h 41h 45h 49h 4Fh 4Eh 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h, 00h. "EL TORITO SPECIFICATION" padded to the right with zeros.
27h-46h	Byte	Reserved (shall be 0)
47h-4Ah	DWord	Absolute pointer to the first sector of the boot catalog
4Bh-7FFh	Byte	Reserved (shall be 0)

### 7.2.2 Validation Entry

This first entry in the boot catalog is a validation entry. The validation entry shall be used to determine that a valid boot catalog is on the disk and to determine the manufacturer of the CD. If this entry is valid, the remainder of the boot catalog should be valid. See Table 6 for the format of a validation entry.

**Table 6 – Validation Entry**

Offset	Type	Description								
00h	Byte	01h – header ID								
01h	Byte	Platform ID <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>ID</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>80x86</td> </tr> <tr> <td>1</td> <td>Power PC</td> </tr> <tr> <td>2</td> <td>Mac</td> </tr> </tbody> </table>	ID	Description	0	80x86	1	Power PC	2	Mac
ID	Description									
0	80x86									
1	Power PC									
2	Mac									
02-03h	Word	Reserved								
04h-1Bh	Byte	ID String. This shall indicate the manufacturer or developer of the CD.								
1Ch-1Dh	Word	Checksum Word. This is the 2's complement of all the words in this record. The sum of all the words from 00h-1Eh shall equal zero.								
1Eh	Byte	Key Byte 55h								
1Fh	Byte	Key Byte AAh								

### 7.2.3 Initial/Default Entry

Every boot catalog shall have an Initial/Default entry. This entry shall point to the initial boot image. The default entry is started if the system BIOS does not have the capability to boot from other boot images in the boot catalog. Normally, a system BIOS does not check for additional boot images. If there are multiple boot image possibilities, the Initial/Default entry may be a program that has the appropriate selection mechanism for choosing the proper boot image. See Table 7 for a description of the Initial/Default Entry.

**Table 7 – Initial/Default Entry**

Offset	Type	Description														
00h	Byte	88h – Header ID indicates a bootable image is present. 00h indicates that a non-bootable image is present														
01h	Byte	<p>Boot media type. The boot image shall fall into one of three categories: Floppy, Hard Drive, or None.</p> <table border="1"> <thead> <tr> <th>ID</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>No Emulation</td> </tr> <tr> <td>1</td> <td>1.2MB diskette</td> </tr> <tr> <td>2</td> <td>1.44MB diskette</td> </tr> <tr> <td>3</td> <td>2.88MB diskette</td> </tr> <tr> <td>4</td> <td>Hard Drive</td> </tr> <tr> <td>05h-FFh</td> <td>Reserved</td> </tr> </tbody> </table>	ID	Description	0	No Emulation	1	1.2MB diskette	2	1.44MB diskette	3	2.88MB diskette	4	Hard Drive	05h-FFh	Reserved
ID	Description															
0	No Emulation															
1	1.2MB diskette															
2	1.44MB diskette															
3	2.88MB diskette															
4	Hard Drive															
05h-FFh	Reserved															
02-03h	Word	Load Segment. This shall be the segment address where the first emulated sector of data is loaded into system memory. If this field cleared to zero, the default value of 07C0h shall be used. For flat memory model architectures (such as Motorola), this is the memory address divided by 10h.														
04h	Byte	System type. This shall be a copy of byte 5 (system type) from the partition table found in the boot image.														
05h	Byte	Reserved														
06h-07h	Word	Sector Count. This shall be the number of emulated sectors the system shall transfer from the CD to the Load Segment address in system memory during the boot process. This field will normally be set to 01h for 80x86 systems that are booting a floppy or hard drive image. For No Emulation Booting, this field could potentially ask for 32MB of data to be loaded from the CD or DVD before jumping to the boot address.														
08h-0Bh	DWord	Load Block. This shall be the absolute start sector on the CD or DVD of the image to be used in the emulation.														
0Ch-1Fh	Byte	Reserved														

### 7.2.4 Section Header

If the CD or DVD has boot images in addition to the Initial/Default image, pointers to the various images shall be divided into sections. The identification string in the section header shall be used to determine if the images in that section are available for booting. If the BIOS is not setup to use a specific section as identified by the ID string, then the section shall be ignored. A good example of a section would be to identify a specific language for the boot images. Another example of a section would be to identify a specific model and manufacturer of a PC. This would allow the PC to boot software specific to the system. See Table 8 for a description of the section header.

**Table 8 – Section Header**

Offset	Type	Description
00h	Byte	Header ID <b>ID Description</b> 90 Type is Section Header, there shall be more sections following this one 91 Type is Section Header, this is the final section
01h	Byte	Platform ID <b>ID Description</b> 0 80x86 1 Power PC 2 Mac
02h-03h	Word	Number of section entries. This is the number of Section Entries and Section Entry Extensions following this header. There shall be at least one Section Entry (see 7.2.5 for more information) in each section. If the Header ID is 90h then a Section Header shall follow the last section entry. If the Head ID is 91h then the last section entry shall be the last entry in the boot catalog.
04h-1Fh	Byte	ID String. This identifies the section and serves as the section name.

### 7.2.5 Section Entry

Section entries shall follow Section Headers. The Section Entry identifies a single boot image. This entry shall be the same as the Initial/Default entry with one exception: the reserved bytes at the end of the entry contain selection criteria. The Section Header ID string identifies the section type, and the section type determines how the selection criteria bytes are interpreted. An example would be a section titled "Languages", the 19 bytes could then be language identifiers. If the system language were set to a language, such as French or Japanese, the system may then boot using an image designed for that language. See Table 9 for a description of the Section Entry.

If the 19 bytes provided for selection criteria are insufficient, Section Entry Extensions may be used. There are a possible 30 bytes of information available per extension, see 7.2.6 for more information on Section Entry Extensions.

**Table 9 – Section Entry**

Offset	Type	Description														
00h	Byte	88h – Header ID indicates a bootable image is present. 00h indicates that a non-bootable image is present														
01h	Byte	<p>Boot media type. The boot image shall fall into one of three categories: Floppy, Hard Drive, or None.</p> <table border="1"> <thead> <tr> <th>ID</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>No Emulation</td> </tr> <tr> <td>1</td> <td>1.2MB diskette</td> </tr> <tr> <td>2</td> <td>1.44MB diskette</td> </tr> <tr> <td>3</td> <td>2.88MB diskette</td> </tr> <tr> <td>4</td> <td>Hard Drive</td> </tr> <tr> <td>05h-FFh</td> <td>Reserved</td> </tr> </tbody> </table>	ID	Description	0	No Emulation	1	1.2MB diskette	2	1.44MB diskette	3	2.88MB diskette	4	Hard Drive	05h-FFh	Reserved
ID	Description															
0	No Emulation															
1	1.2MB diskette															
2	1.44MB diskette															
3	2.88MB diskette															
4	Hard Drive															
05h-FFh	Reserved															
02-03h	Word	Load Segment. This shall be the segment address where the first emulated sector of data is loaded into system memory. If this field cleared to zero, the default value of 07C0h shall be used. For flat memory model architectures (such as Motorola), this is the memory address divided by 10h.														
04h	Byte	System type. This shall be a copy of byte 5 (system type) from the partition table found in the boot image.														
05h	Byte	Reserved														
06h-07h	Word	Sector Count. This shall be the number of emulated sectors the system shall transfer from the CD to the Load Segment address in system memory during the boot process. This field will normally be set to 01h for 80x86 systems that are booting a floppy or hard drive image. For No Emulation Booting, this field could potentially ask for 32MB of data to be loaded from the CD or DVD before jumping to the boot address.														
08h-0Bh	DWord	Load Block. This shall be the absolute start sector on the CD or DVD of the image to be used in the emulation.														
0Ch	Byte	<p>Selection Criteria. The selection criteria determines the format for the selection criteria. The following formats are currently assigned:</p> <table border="1"> <thead> <tr> <th>Type</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>00h</td> <td>No selection criteria</td> </tr> <tr> <td>01h</td> <td>Language and version information (IBM format)</td> </tr> <tr> <td>02h-FFh</td> <td>Reserved</td> </tr> </tbody> </table>	Type	Description	00h	No selection criteria	01h	Language and version information (IBM format)	02h-FFh	Reserved						
Type	Description															
00h	No selection criteria															
01h	Language and version information (IBM format)															
02h-FFh	Reserved															
0Dh-1Fh	Byte	Selection Criteria														

## 7.2.6 Section Entry Extension

Section Entry Extensions may follow Section Entries. Section Entry Extensions provide more space for selection criteria when the 13 bytes provided by the Section Entry are insufficient. There may be 1 or more extensions following each Section Entry. This means that a section provides 13 bytes of selection information. A section with one extension provides 43 bytes, and a section with 2 extension provides 73 bytes, etc. See Table 10 for the description of a Section Entry Extension

**Table 10 – Section Entry Extension**

Offset	Type	Description										
00h	Byte	44h – Header ID, indicates a Section Entry Extension										
01h	Byte	Extension continuation information <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>Bit</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>00h-04h</td> <td>Reserved</td> </tr> <tr> <td>05h</td> <td>Shall be cleared to zero if this is the final extension for this section.</td> </tr> <tr> <td>06h-07h</td> <td>Shall be set to one if more extensions follow this extension.</td> </tr> <tr> <td>06h-07h</td> <td>Reserved</td> </tr> </tbody> </table>	Bit	Description	00h-04h	Reserved	05h	Shall be cleared to zero if this is the final extension for this section.	06h-07h	Shall be set to one if more extensions follow this extension.	06h-07h	Reserved
Bit	Description											
00h-04h	Reserved											
05h	Shall be cleared to zero if this is the final extension for this section.											
06h-07h	Shall be set to one if more extensions follow this extension.											
06h-07h	Reserved											
02-1Fh	Byte	Additional selection criteria										

## 8 INT 13h Function Definitions

These function definitions include legacy, extended, and CD/DVD functions. The functions are differentiated by their function numbers and register usage. The extended functions use DS:SI as a pointer to a command packet. The legacy functions use ES:BX as a pointer to their transfer buffer. Functions 0-3Fh are legacy functions. Functions 40h-49h are extended functions. Functions 4Ah-4Dh are CD/DVD boot functions. Functions 4Eh-50h are hardware control functions.

The values in all registers that are not explicitly defined in the following sections shall be preserved at the completion of each function call.

### 8.1 Reset Disk Subsystem

Entry:

AH - 00h  
DL - BIOS device number

Exit:

carry clear - Reset successful  
AH - 0  
carry set - Reset Failed  
AH - error code

This function shall be used to reset all the devices controlled by the INT 13 interface.

### 8.2 Get Status of Last Operation

Entry:

AH - 01h  
DL - BIOS device number

Exit:

carry clear - Last command was successful  
AH - 0  
AL - Status of last INT 13h operation  
carry set - Last command failed  
AH - error code

This function shall be used to return the status of the last INT 13 command executed.

### 8.3 Read Sectors into Memory

Entry:

AH - 02h  
 AL - Number of sectors to read, shall be greater than 0 and less than 128  
 CH - Low order 8 bits of the cylinder number  
 CL - Bits 0-5 specify the sector number, bits 6-7 are the high order 2 bits of the cylinder  
 DH - Head number  
 DL - BIOS device number  
 ES:BX - Pointer to destination buffer in memory

Exit:

carry clear - Read was successful  
 AH - 0  
 AL - Number of sectors read  
 ES:BX - Buffer filled with read data  
 carry set  
 AH - Error code  
 AL - Vendor Specific  
 ES:BX - Pointer to buffer partially filled with read data

This function shall be used to read data from the device into host memory buffer pointed to by ES:BX. If the command transfers some of the data and then returns a failure, carry is set, the buffer pointed to by ES:BX shall contain the partial data transfer, and AH shall contain an error code. When there is a partial data transfer, there is no reliable indication of what part of the data was successfully transferred. Some systems use DMA to return the data in an out-of-order fashion. This function does not have a mechanism to report what data is valid in the case of a partial transfer. In some legacy systems when a partial transfer occurs, AL contains the number of sectors transferred.

### 8.4 Write Sectors From Memory

Entry:

AH - 03h  
 AL - Number of sectors to write, shall be greater than 0 and less than 128  
 CH - Low order 8 bits of the cylinder number  
 CL - Bits 0-5 specify the sector number, bits 6-7 are the high order 2 bits of the cylinder  
 DH - Head number.  
 DL - BIOS device number  
 ES:BX - Pointer to source buffer in memory

Exit:

carry clear - Read was successful  
 AH - 0  
 AL - Number of sectors written  
 carry set  
 AH - Error code  
 AL - Vendor Specific

This function shall be used to transfer data from the host buffer pointed to by ES:BX to the device.

### 8.5 Verify Sectors

Entry:

AH - 04h  
 AL - Number of sectors to verify, shall be greater than 0 and less than 128  
 CH - Low order 8 bits of the cylinder number  
 CL - Bits 0-5 specify the sector number, bits 6-7 are the high order 2 bits of the cylinder  
 DH - Head number.  
 DL - BIOS device number

Exit:

carry clear - Read was successful  
 AH - 0  
 AL - Number of sectors verified  
 carry set  
 AH - Error code  
 AL - Vendor Specific

This function shall be used to check the sectors in the specified range on the device for errors. No data is transferred between the host and device by this command.

## 8.6 Get Drive Parameters

Entry:

AH - 08h  
 DL - BIOS device number

Exit:

carry clear - Get Drive Parameters was successful  
 AH - 00h  
 BL - Vendor Specific  
 CH - Low order 8 bits of the maximum cylinder number  
 CL - Bits 0-5 specify the maximum sector number, bits 6-7 are the high order 2 bits of the maximum cylinder number  
 DH - Maximum head number  
 DL - Total number of INT 13h devices with an INT 13h device number greater than 7Fh  
 ES:DI - Vendor Specific  
 carry set  
 AH - Error code

This function shall be used to find the CHS geometry used by INT 13 functions 2, 3, and 4 to access the drive.

## 8.7 Read Long

Entry:

AH - 0Ah  
 AL - Number of sectors to read, shall be set to 1  
 CH - Low order 8 bits of the cylinder number  
 CL - Bits 0-5 specify the sector number, bits 6-7 are the high order 2 bits of the cylinder  
 DH - Head number.  
 DL - BIOS device number  
 ES:BX - Pointer to destination buffer in memory

Exit:

carry clear - Read was successful  
 AH - 0  
 AL - Number of sectors read  
 ES:BX - Pointer to buffer filled with read data  
 carry set  
 AH - Error code  
 AL - Vendor Specific  
 ES:BX - Pointer to buffer partially filled with read data

This function shall be used to request sectors from the device including additional ECC information. The host shall determine how much data will be returned prior to issuing this command. If the command transfers some of the data and then returns a failure, carry is set, the buffer pointed to by ES:BX shall contain the partial data transfer, and AH shall contain an error code. When there is a partial data transfer, there is no reliable indication of what part of the data was successfully transferred. Some systems use DMA to return the data in an out-of-order fashion. This function does not have a mechanism to report what data is valid in the case of a partial transfer. In some legacy systems when a partial transfer occurs, AL contains the number of sectors transferred.

## 8.8 Write Long

Entry:

AH - 0Bh  
 AL - Number of sectors to write, shall be set to 1  
 CH - Low order 8 bits of the cylinder number  
 CL - Bits 0-5 specify the sector number, bits 6-7 are the high order 2 bits of the cylinder  
 DH - Head number.  
 DL - BIOS device number  
 ES:BX - Pointer to source buffer in memory

Exit:

carry clear - write was successful  
 AH - 0  
 AL - Number of sectors written  
 carry set  
 AH - Error code  
 AL - Vendor Specific

This function shall be used write data to the device including additional ECC information.

## 8.9 Seek to Cylinder

Entry:

AH - 0Ch  
 CH - Low order 8 bits of the cylinder number  
 CL - Bits 0-5 specify the sector number, bits 6-7 are the high order 2 bits of the cylinder  
 DH - Head number.  
 DL - BIOS device number

Exit:

carry clear - seek was successful  
 AH - 0  
 carry set  
 AH - Error code

This function shall be used to move a drive head to the specified cylinder.

## 8.10 Read Buffer

Entry:

AH - 0Eh  
 DL - BIOS device number  
 ES:BX - Pointer to destination buffer in memory

Exit:

carry clear - Read was successful  
 AH - 0  
 carry set  
 AH - Error code

This function shall be used to read 512 bytes of data from the specified device's sector buffer.

## 8.11 Write Buffer

Entry:  
 AH - 0Fh  
 DL - BIOS device number  
 ES:BX - Pointer to source buffer in memory

Exit:  
 carry clear - Write was successful  
 AH - 0  
 carry set  
 AH - Error code

This function shall be used to write 512 bytes of data to the specified device's buffer.

## 8.12 Recalibrate

Entry:  
 AH - 11h  
 DL - BIOS device number

Exit:  
 carry clear - Write was successful  
 AH - 0  
 carry set  
 AH - Error code

This function shall be used move the drive head to sector 0

## 8.13 Get Disk Type

Entry:  
 AH - 15h  
 DL - BIOS device number

Exit:  
 carry clear - Request was successful  
 AH - 03h = Hard Drive present  
 CX:DX - Number of sectors on the media  
 carry set  
 AH - Error code

This function shall be used to find the device type. This function is called by some versions of DOS during the boot process.

## 8.14 Set Media Type

Entry:  
 AH - 18h  
 CH - Low order 8 bits of the maximum cylinder number  
 CL - Bits 0-5 specify the sectors per track, bits 6-7 are the high order 2 bits of the maximum cylinder number  
 DL - BIOS device number

Exit:  
 carry clear - Request was successful  
 AH - 00h - Requested geometry is supported.  
 ES:DI - Pointer to parameter table  
 carry set

AH - Error code

This function shall be used to force the BIOS to use a specific media type. This function is called by some versions of DOS.

### 8.15 Park Heads

Entry:

AH - 19h  
DL - BIOS device number

Exit:

carry clear - Request was successful  
AH - 00h  
carry set  
AH - Error code

This function shall be used to park the device heads.

### 8.16 Identify Device Info.

Entry:

AH - 25h  
DL - BIOS device number  
ES:BX - Pointer to 512 byte buffer for IDENTIFY DEVICE information. See ATA/ATAPI-7 for details.

Exit:

carry clear - Request was successful  
AH - 00h  
ES:BX - Pointer to IDENTIFY DEVICE information  
carry set  
AH - Error code

This function shall be used return IDENTIFY DEVICE information from the device.

### 8.17 Check Extensions Present

Entry:

AH - 41h  
BX - 55AAh  
DL - BIOS device number

Exit:

carry clear  
AH - Version of extensions = 30h  
AL - Internal use only  
BX - AA55h  
CX - Interface support bit map (see Table 11 )  
carry set  
AH - Error code (01h, Invalid Command)

**Table 11 - Check Extensions Present Buffer**

Bit	Description
0	If the content of this bit is one, then the Fixed disk access subset shall be supported.
1	If the content of this bit is one, then the Device locking and ejecting subset shall be supported.
2	This bit shall be set to one to indicate that the Enhanced disk drive support subset is present.
3	If the content of this bit is one, then the 64-bit extensions are present. When the 64-bit extensions are present, the fields starting at offset 10h in the Device Address Packet are valid
4-15	Reserved. The content of these bits shall be zero.

This function shall be used to check for the presence of INT 13h extensions. If CF=1b, the extensions are not supported for the requested device. If CF=0b, BX shall be checked to confirm that it contains the value AA55h indicating that the extensions are present. If BX = AA55h, the value of CX shall be checked to determine what subsets of this interface are supported for the requested device. At least one subset shall be supported. The version of the extensions shall be 30h. This indicates that the INT 13h extensions are compliant with this standard.

### 8.18 Extended Read

Entry:

AH - 42h  
DL - BIOS device number  
DS:SI - Device address packet (See Table 3)

Exit:

carry clear  
AH - 0  
carry set  
AH - Error code

This function shall transfer sectors from the device to memory. In the event of an error, the block count field of the device address packet contains the number of good blocks read before the error occurred.

### 8.19 Extended Write

Entry:

AH - 43h  
AL - 0 or 1, write with verify off; 2, write with verify on  
DL - BIOS device number  
DS:SI - Device address packet (See Table 3)

Exit:

carry clear  
AH - 0  
carry set  
AH - Error code

This function shall transfer sectors from memory to the device. If write with verify is not supported, this function rejects the request with AH = 01h and CF = 1b. Fn 48h shall be used to detect if write with verify is supported. In the event of an error, the block count field of the device address packet contains the number of blocks transferred before the error occurred. The calling software should not assume that the data transferred is validly written to the media. AL also contains the values 00h, 01h, or 02h. This function rejects all other values with AH = 01h and CF = 1b.

## 8.20 Verify Sectors

### Entry:

AH - 44h  
 DL - BIOS device number  
 DS:SI - Device address packet (See Table 3)

### Exit:

carry clear  
 AH - 0  
 carry set  
 AH - Error code

This function verifies sectors without transferring data between the device and system memory. When an error is reported the block count field of the device address packet shall be filled in with the number of blocks verified before the error occurred.

## 8.21 Lock/Unlock Media

### Entry:

AH - 45h  
 AL - 0 - Lock media in device, 1 - Unlock media in device, 2 - Return lock/unlock status, 3h-FFh - Invalid  
 DL - BIOS device number

### Exit:

carry clear  
 AH - 0  
 AL - 1 if device is locked, 0 if not  
 carry set  
 AH - Error code.

This function logically locks or unlocks removable media in a specific device. All removable media devices numbered 80h and above shall implement this function. If a fixed disk (non-removable device) supports the media locking and ejecting subset, this function shall return with success (i.e., AH = 00h and CF = 0b). There shall be support for 255 locks per device. A device shall not be unlocked until all locks to that device have been released with unlock commands. Excess unlock calls shall return with CF = 1b and AH = B0h. If the number of locks supported value is exceeded on a lock request, this function shall reject the request with CF = 1b and AH = B4h. Locking a device without media present shall be a valid operation. On return from a lock or unlock request, AL shall contain the lock state of the media as maintained by the BIOS. This provides for unlock requests when the lock count is greater than zero. In this case, the media shall remain locked. Any physical locking and unlocking of the media shall be implementation dependent, but the caller may operate on the assumption that locked media cannot be removed without an unlock request. After power-on, or a system reset, all devices shall automatically enter an unlocked state.

## 8.22 Eject Removable Media

### Entry:

AH - 46h  
 AL - 0h  
 DL - BIOS device number

### Exit:

carry clear  
 AH - 0  
 carry set  
 AH - Error code

This function shall eject media from the specified device. If a fixed disk (non-removable device) supports the media locking and ejecting interface subset, this function shall always return with "Volume Not Removable" (i.e., CF = 1 and AH = B2h). An attempt to eject media locked in a device shall return with "Media Locked In Device"

(i.e., CF 1 and AH = B1h). This function represents a request to remove media from the selected device. Actual ejection shall be implementation dependent, but system software that issues or observes this function should flush any buffers it is holding. If this function is issued for a device without media the request shall be returned with "No Media In Device" (i.e., CF = 1b and AH = 31h). If this call is issued to an unlocked removable media device that has media present, an INT 15h, Fn 52h (removable media eject) shall be issued to determine if eject removable media may proceed with the ejection request. If INT 15h returns an error, the ejection request shall be rejected. If the ejection request is accepted, followed by an unrecoverable error, this function shall return with "Valid Eject Request Failed" (i.e., CF = 1b and AH = B5h).

### 8.23 Extended Seek

Entry:

AH - 47h  
 DL - BIOS device number  
 DS:SI - Device address packet (See Table 3)

Exit:

carry clear  
 AH - 0  
 carry set  
 AH - Error code

This function allows the host to provide advanced notification that particular data may be requested by the host in a subsequent command. This command shall initiate a seek operation. The seek may not be complete when this function completes.

## 8.24 Get Device Parameters

### Entry:

AH - 48h  
 DL - BIOS device number  
 DS:SI - address of result buffer (See Table 12)

### Exit:

carry clear  
 AH - 0  
 DS:SI - address of result buffer (See Table 12)  
 carry set  
 AH - Error code

This function returns default device parameters. It shall be mandatory regardless of the interface subset that is supported. Table 12 defines the result buffer. On entry the first word of the result buffer shall be the buffer length in bytes.

**Table 12 - Result Buffer**

Offset	Type	Description																				
0	Word	The caller shall set this value to the maximum Result Buffer length, in bytes. If the length of this buffer is less than 30 bytes, this function shall not return the pointer to DPT extension. If the buffer length is 30 or greater on entry, it shall be set to 30 on exit. If the buffer length is between 26 and 29, it shall be set to 26 on exit. If the buffer length is less than 26 on entry an error shall be returned.																				
2	Word	Information Flags. A value of one in a bit indicates that the feature shall be available. A value of zero in a bit indicates the feature shall be not available and shall operate in a manner consistent with the conventional INT 13h interface.  <table border="0"> <thead> <tr> <th>Bit</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>DMA boundary errors are handled transparently</td> </tr> <tr> <td>1</td> <td>The geometry returned in bytes 4-15 shall be valid</td> </tr> <tr> <td>2</td> <td>Media shall be removable. Bits 4-6 are not valid if this bit is cleared to zero</td> </tr> <tr> <td>3</td> <td>Device supports write verify</td> </tr> <tr> <td>4</td> <td>Device has media change notification</td> </tr> <tr> <td>5</td> <td>Media shall be lockable</td> </tr> <tr> <td>6</td> <td>Device geometry shall be set to maximum and no media shall be present when this bit is set to one</td> </tr> <tr> <td>7</td> <td>BIOS calls INT13h FN 50h to access the device</td> </tr> <tr> <td>8-15</td> <td>Reserved</td> </tr> </tbody> </table>	Bit	Description	0	DMA boundary errors are handled transparently	1	The geometry returned in bytes 4-15 shall be valid	2	Media shall be removable. Bits 4-6 are not valid if this bit is cleared to zero	3	Device supports write verify	4	Device has media change notification	5	Media shall be lockable	6	Device geometry shall be set to maximum and no media shall be present when this bit is set to one	7	BIOS calls INT13h FN 50h to access the device	8-15	Reserved
Bit	Description																					
0	DMA boundary errors are handled transparently																					
1	The geometry returned in bytes 4-15 shall be valid																					
2	Media shall be removable. Bits 4-6 are not valid if this bit is cleared to zero																					
3	Device supports write verify																					
4	Device has media change notification																					
5	Media shall be lockable																					
6	Device geometry shall be set to maximum and no media shall be present when this bit is set to one																					
7	BIOS calls INT13h FN 50h to access the device																					
8-15	Reserved																					
4	DWord	Number of default cylinders. The content of this field shall be one greater than the maximum cylinder number. INT 13h Fn 08h shall be used to find the logical number of cylinders.																				
8	DWord	Number of default heads. The content of this field shall be one greater than the maximum head number. INT 13h Fn 08h shall be used to find the logical number of heads.																				
12	DWord	Number of default sectors per track. The content of this field shall be the same as the maximum sector number because sector addresses are 1 based. INT 13h Fn 08h shall be used to find the logical number of sectors per track.																				
16	QWord	Number of sectors. This shall be one greater than the maximum sector number. If this field is greater than 15,482,880 then word 2, bit 1 shall be cleared to zero.																				
24	Word	Number of bytes in a sector.																				
26	DWord	Pointer to the Device Parameter Table Extension (DPTE). This field follows the seg:offset address format. The DPTE shall only be present if INT 13h, Fn 41h, CX register bit 2 is set to one. This field points to a temporary buffer that the BIOS may invalidate on subsequent INT 13h calls. If the length of this result buffer is less than 30, the DPTE shall not be present. This field is only used for INT 13h based systems configured with ATA or ATAPI devices.																				

Offset	Type	Description
30	Word	0BEDDh - Key, indicates presence of Device Path Information
32	Byte	Length of Device Path Information including the key. The content of this byte shall be 2Ch
33	Byte	Reserved. The value in this field shall be 00h.
34	Word	Reserved. The value in this field shall be 0000h.
36	ASCII	Host bus type, 4 bytes. ASCII data shall be left justified and padded with the value 20h PCI PCI Local Bus 50h 43h 49h 20h ISA Legacy 16 bit fixed bus 49h 53h 41h 20h PCI-X PCI-X Bus 50h 43h 49h 58h IBND Infiniband 49h 42h 4Eh 44h XPRS PCI Express 58h 50h 52h 53h HTPT HyperTransport 48h 54h 50h 54h
40	ASCII	Interface type, 8 bytes. ASCII data shall be left justified and padded with the value 20h ATA ATA/ATAPI compliant device using ATA commands 41h 54h 41h 20h 20h 20h 20h 20h ATAPI ATA/ATAPI compliant device using ATAPI commands 41h 54h 41h 50h 49h 20h 20h 20h SCSI SCSI compliant device 53h 43h 53h 49h 20h 20h 20h 20h USB USB Mass Storage compliant device 55h 53h 42h 20h 20h 20h 20h 20h 1394 1394 Mass Storage device 31h 33h 39h 34h 20h 20h 20h 20h FIBRE Fibre Channel 46h 49h 42h 52h 45h 20h 20h 20h I <sub>2</sub> O Intelligent Input/Output 49h 32h 4Fh 20h 20h 20h 20h 20h RAID Redundant Array of Inexpensive Disks (RAID) member 52h 41h 49h 44h 20h 20h 20h 20h SATA Serial ATA 53h 41h 54h 41h 20h 20h 20h 20h SAS Serial Attached SCSI 53h 41h 53h 20h 20h 20h 20h 20h
48	QWord	Interface Path, 8 bytes. See Table 13 for more information
56	Double QWord	Device Path. See Table 14 for more information.
72	Byte	Reserved. The value in this field shall be 00h.
73	Byte	Checksum for Device Path Information includes the 0BEDDh signature. The content of this field shall be the two's complement of the unsigned sum of offset 30 through 72. The unsigned sum of offset 30 through 73 shall be 0.

### 8.24.1 Interface Path

The Interface Path field at offset 48 allows software external to a system BIOS to locate mass storage device interface chips. The format of this field shall be dependent on the Host Bus type, offsets 36 through 39 of the result buffer. The following formats are currently defined:

**Table 13 - Interface Path Definitions**

Host Bus Type	Offset	Type	Definition
ISA	48	Word	16-bit base address
	50	Word	Reserved. The value in this field shall be 0000h
	52	DWord	Reserved. The value in this field shall be 00000000h
PCI	48	Byte	PCI bus number. Values 00h through FEh shall represent a valid PCI bus. Value FFh shall indicate that this field is not used.
	49	Byte	PCI slot number. Values 00h through FEh shall represent a valid PCI slot. Value FFh shall indicate that this field is not used.
	50	Byte	PCI function number. Values 00h through FEh shall represent a valid PCI function. Value FFh shall indicate that this field is not used.
	51	Byte	Channel number. If more than one interface of the same type is in a single Bus, Slot, Function, then the channel number shall identify each interface. If there is only one interface, the content of this field shall be cleared to zero. If there are two interfaces, such as an ATA Primary and Secondary interface, the primary interface shall be zero, and the secondary interface shall be one.  Values 00h through FEh shall represent a valid Channel Number. Value FFh shall indicate that this field is not used
	52	DWord	Reserved. The value in this field shall be 00000000h
PCI-X	48	Byte	PCI-X bus number. Values 00h through FEh shall represent a valid PCI bus. Value FFh shall indicate that this field is not used.
	49	Byte	PCI slot number. Values 00h through FEh shall represent a valid PCI slot. Value FFh shall indicate that this field is not used.
	50	Byte	PCI function number. Values 00h through FEh shall represent a valid PCI function. Value FFh shall indicate that this field is not used.
	51	Byte	Channel number. If more than one interface of the same type is in a single Bus, Slot, Function, then the channel number shall identify each interface. If there is only one interface, the content of this field shall be cleared to zero. If there are two interfaces, such as an ATA Primary and Secondary interface, the primary interface shall be zero, and the secondary interface shall be one.  Values 00h through FEh shall represent a valid Channel Number. Value FFh shall indicate that this field is not used
	52	DWord	Reserved. The value in this field shall be 00000000h
Infiniband	48h	Qword	TBD
PCI Express	48h	Qword	TBD
HyperTransport	48h	Qword	TBD

### 8.24.2 Device Path

The Device Path at offset 56 combined with the Interface Path allows software external to a system BIOS to locate a specific mass storage device. The Device Path field provides a path from an interface to a specific device. The format of the Device Path at offset 40 through 47 is dependent on the Interface type. The following formats are defined in this standard:

**Table 14 - Device Path Definitions**

Interface Type	Offset	Type	Definition
ATA	56	Byte	00h = ATA Device 0, 01h = ATA Device 1
	57	Byte	Reserved. The value in this field shall be 00h.
	58	Word	Reserved. The value in this field shall be 0000h.
	60	DWord	Reserved. The value in this field shall be 00000000h.
	64	QWord	Reserved. The value in this field shall be 0000000000000000h.
ATAPI	56	Byte	00h = ATAPI Device 0, 01h = ATAPI Device 1
	57	Byte	Logical Unit Number
	58	Byte	Reserved. The value in this field shall be 00h.
	59	Byte	Reserved. The value in this field shall be 00h.
	60	DWord	Reserved. The value in this field shall be 00000000h.
	64	QWord	Reserved. The value in this field shall be 0000000000000000h.
SCSI	56	Word	Physical Unit Number/SCSI ID
	58	QWord	Logical Unit Number
	66	Word	Reserved. The value in this field shall be 0000h.
	68	DWord	Reserved. The value in this field shall be 00000000h.
USB	56	QWord	64-bit Serial Number as defined in the USB Mass Storage specifications
	64	QWord	Reserved. The value in this field shall be 0000000000000000h.
1394	56	QWord	64-bit Extended Unique Identifier (EUI-64)
	64	QWord	Reserved. The value in this field shall be 0000000000000000h.
FIBRE	56	QWord	64-bit Worldwide Identifier (WWID)
	64	QWord	Logical Unit Number
I <sub>2</sub> O	56	QWord	64-bit Identity Tag
	64	QWord	Reserved. The value in this field shall be 0000000000000000h.
RAID	56	DWord	RAID array number of which this device is a member
	60	DWord	Reserved. The value in this field shall be 00000000h.
	64	QWord	Reserved. The value in this field shall be 0000000000000000h.

Interface Type	Offset	Type	Definition
SATA	56	Byte	00h = SATA Device 0, 01h = SATA Device 1
	57	Byte	Reserved. The value in this field shall be 00h.
	58	Word	Reserved. The value in this field shall be 0000h.
	60	DWord	Reserved. The value in this field shall be 00000000h.
	64	QWord	Reserved. The value in this field shall be 0000000000000000h.
SAS	56	QWord	64-bit SAS address
	64	QWord	Logical Unit Number

### 8.24.3 Device Parameter Table Extension (DPTE)

The DPTE provides hardware configuration information to applications that bypass INT 13h for addressing an ATA device.

**Table 15 - Device parameter table extension**

Offset	Type	Description
0-1	Word	I/O port base address
2-3	Word	Control port address
4	Byte	Device register bit 0-3     0 bit 4     ATA DEV bit bit 5     1 bit 6     LBA enable (1 = enabled) bit 7     1
5	Byte	BIOS Vendor Specific.
6	Byte	IRQ information bits 0-7    IRQ for this device
7	Byte	Block count for ATA READ/WRITE MULTIPLE commands
8	Byte	DMA information bits 0-3    DMA channel bits 4-7    DMA type
9	Byte	PIO information bits 0-3    PIO type bits 4-7    0
10-11	Word	BIOS selected hardware specific option flags bit 0     PIO accessing enabled bit 1     DMA accessing enabled bit 2     ATA READ/WRITE MULTIPLE accessing enabled bit 3     CHS translation enabled bit 4     LBA translation enabled bit 5     Removable media bit 6     ATAPI device bit 7     32-bit transfer mode bit 8     ATAPI device uses command packet interrupt bits 9-10   Translation type bit 11    Ultra DMA accessing enabled bit 12    HPA Active bit 13    48-bit LBA addressing enabled bits 14-15   Reserved, shall be cleared to zero
12-13	Word	Reserved, shall be cleared to zero
14	Byte	30h, version level of this table.
15	Byte	Checksum, 2's complement of the 8 bit unsigned sum of bytes 0-14

#### 8.24.3.1 Offset 0-1 - I/O port base

This word is the 16-bit address in I/O space of the data register in the ATA Command Block. Any application that provides a proprietary interface to the device may use this base address.

#### 8.24.3.2 Offset 2-3 - control port base

This word is the 16-bit address in I/O space of the device control register. Any application that provides a proprietary interface to the device may use this address.

**8.24.3.3 Offset 4 - head prefix**

The upper four bits of this byte shall be logically ORed with the head number, or upper four bits of the LBA, each time the disk is addressed. It contains the ATA DEV bit and the LBA addressing bits that are preset, and makes these functions transparent to any software using this extension. The LBA addressing bit is set for each disk access and shall not be used to determine the LBA capability of the system. See the LBA translation enabled bit described in section 8.24.3.9.5 for system LBA capability.

**8.24.3.4 Offset 5 - BIOS use only**

BIOS use only.

**8.24.3.5 Offset 6 - IRQ number**

Each ATA channel requires an assigned Interrupt number. This byte identifies which IRQ is used by this device's channel.

**8.24.3.6 Offset 7 - READ/WRITE MULTIPLE command block count**

If the device was configured to use the READ/WRITE MULTIPLE command, then this field shall contain the block size of the transfer, in sectors, used by the BIOS.

**8.24.3.7 Offset 8 - DMA channel/Multiword DMA Type**

If the BIOS has configured the system to perform multiword DMA data transfers in place of PIO transfers, this field shall specify the DMA mode in the upper four bits, as per the definition in ATA-5 or later, and the DMA Channel in the lower four bits. ATA channels that support PCI DMA bus mastering shall set the DMA channel to zero. Note that the DMA Type field does not follow the format of the data returned by the device. The value of the DMA mode shall not be limited to two.

**8.24.3.8 Offset 9 - PIO type**

If the BIOS has configured the system to perform PIO data transfers other than mode 0, this field shall specify the PIO mode as per the definition in ATA-5 or later.

**8.24.3.9 Offset 10-11 - BIOS selected hardware specific option flags**

These bytes specify the current hardware options enabled by the BIOS, a bit for each of the options listed below.

**8.24.3.9.1 Bit 0 - fast PIO**

If the system is configured for a PIO mode greater than 0, this bit shall be set to one and byte 9 (PIO Type) shall be used to configure the system. If this bit is cleared to zero, the PIO-Type field shall be ignored.

**8.24.3.9.2 Bit 1 - fast DMA**

If the system is configured for DMA, this bit shall be set to one and byte 8 (DMA Channel/DMA Type) should be used to configure the system. If this bit and bit 11, section 3.5.9.11, are cleared to zero, then the DMA Channel/DMA Type field shall be ignored.

**8.24.3.9.3 Bit 2 - ATA READ/WRITE MULTIPLE**

If the system is configured for multi-sector transfers, this bit shall be set to one and byte 7 (sector count) specifies the number of sectors used for each data transfer. If block PIO is disabled, ignore the block count field.

**8.24.3.9.4 Bit 3 - CHS translation**

If the device reports more than 1024 cylinders in the IDENTIFY DEVICE command data, this bit shall be set to one. See section 8.24.3.9.10 to determine the method of geometry translation.

**8.24.3.9.5 Bit 4 - LBA translation**

If the system is configured for LBA type addressing, this bit shall be set to one and the Extended INT 13h interface (Fn 41h through 48h) shall pass LBA values directly to the device. The conventional INT 13h interface shall ignore this bit and shall use CHS. LBA-type addressing shall be available on devices with less than 1024 cylinders, and therefore bit 3 (CHS translation) shall be independent from bit 4 (LBA translation).

**8.24.3.9.6 Bit 5 - removable media**

If the device supports removable media, this bit shall be set to one and the extended INT 13h device locking and ejecting subset shall also be supported.

**8.24.3.9.7 Bit 6 - ATAPI device**

If this ATA device implements the PACKET command feature set (ATAPI) as defined in ATA/ATAPI-5, this bit shall be set to one.

**8.24.3.9.8 Bit 7 - 32-bit transfer mode**

If the BIOS has configured the host adapter to perform 32-bit wide data transfers, this bit shall be set to one.

**8.24.3.9.9 Bit 8 - ATAPI device uses command packet interrupt**

If bit 6 is cleared to zero, then this field shall be ignored and shall be zero. If bit 6 is set to one, this bit indicates how the ATAPI devices signals it is ready to receive a packet command. When this bit is set to one, it indicates that the ATAPI device returns an interrupt, and sets DRQ, when it is ready for a packet. When this bit is cleared to zero, it indicates that the ATAPI device sets DRQ, without an interrupt, when it is ready for a packet.

**8.24.3.9.10 Bits 9-10 - translation type**

If bit 3 is cleared to zero then this field shall be ignored and shall be zero. If bit 3 is set to one then this field identifies the geometric translation shown in Table 16 .

**Table 16 - Translation Type**

Bits 9-10	Description
00	Bit-shift translation
01	LBA assisted translation
10	Reserved
11	Vendor specific translation

**8.24.3.9.11 Bit 11 - Ultra DMA**

If the system is configured for Ultra DMA, this bit shall be set to one and byte 8 (DMA Channel/DMA Type) should be used to configure the system. If this bit and bit 1, (Bit 1 - fast DMA, section 6.8.3.9.2) are cleared to zero, then the DMA Channel/DMA Type field shall be ignored.

**8.24.3.9.12 Bit 12 – HPA Active**

The HPA Active bit shall be set to one if the Host Protected Area (HPA) is active. The HPA is active if a volatile or non-volatile SETMAX has been issued with a value other than the NATIVE MAX address. This bit shall be cleared to zero if NATIVE MAX sectors can be accessed using read and/or write commands.

**8.24.3.9.13 Bit 13 – 48-bit LBA addressing enabled**

The 48-bit LBA addressing enabled bit shall be set to one if the INT 13h handler is using 48-bit extended commands to access the device. The 48-bit LBA addressing enabled bit shall be cleared to 0 if the INT 13h handler is not using 48-bit extended commands to access the device.

**8.24.3.9.14 Bits 14-15 - Reserved**

Shall be cleared to zero.

**8.24.3.10 Offset 12-13 - Reserved**

Shall be cleared to zero.

**8.24.3.11 Offset 14 - table revision**

The table version shall be set to 30h indicating compliance with this standard.

**8.24.3.12 Offset 15 - checksum**

This shall be the two's complement of the 8-bit unsigned sum of bytes 0 through 14. Adding bytes 0 through 15 shall in all cases produce an 8-bit result of zero.

**8.25 Get Extended Media Change Status**

Entry:

AH - 49h

DL - BIOS device number

Exit:

carry clear

AH - 00, change-line inactive

carry set

AH - 06, change-line active

This function returns media change status. If it returns with CF = 1b, the media may not have been changed. The media change notification may be activated by unlocking and locking the device door without removing the media. This function corresponds to INT 13h Fn 16h, but explicitly allows any device number to be passed in. If a non-removable device supports the Device Locking and Ejecting interface subset, this function shall return with success, AH = 00h, CF = 0b. This function shall clear the media change notification on exit.

## 8.26 Initiate Disk Emulation

Entry:

AH - 4Ah

AL - 00h

DS:SI - CD Specification Packet (See Table 17)

Exit:

carry clear

AH - 00, emulation request was successful

carry set

AH - Error code, emulation not initiated

**Table 17 - CD Specification Packet**

Offset	Type	Description																						
0	Byte	13h - Packet size in bytes																						
1	Byte	<p>Boot Media Type. This byte shall indicate the type of image to be booted in bits 0-3 and the emulated image requirements in bits 6-7.</p> <table border="1"> <thead> <tr> <th>Bit</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>Value</td> <td>Emulation</td> </tr> <tr> <td>0</td> <td>No Emulation</td> </tr> <tr> <td>1</td> <td>1.2MB Diskette</td> </tr> <tr> <td>0-3</td> <td>2 1.44MB Diskette</td> </tr> <tr> <td>3</td> <td>2.88MB Diskette</td> </tr> <tr> <td>4</td> <td>Hard Disk</td> </tr> <tr> <td>5-0Fh</td> <td>Reserved</td> </tr> <tr> <td>4-5</td> <td>Reserved</td> </tr> <tr> <td>6</td> <td>0 – Emulated image is not driver dependent. 1- Emulated image contains ATAPI drivers for CD/DVD-ROM, bytes 8 and 9 refer to an ATA/ATAPI interface.</td> </tr> <tr> <td>7</td> <td>0 – Emulated image is not driver dependent. 1- Emulated image contains SCSI drivers for CD/DVD-ROM, bytes 8 and 9 refer to a SCSI interface.</td> </tr> </tbody> </table>	Bit	Description	Value	Emulation	0	No Emulation	1	1.2MB Diskette	0-3	2 1.44MB Diskette	3	2.88MB Diskette	4	Hard Disk	5-0Fh	Reserved	4-5	Reserved	6	0 – Emulated image is not driver dependent. 1- Emulated image contains ATAPI drivers for CD/DVD-ROM, bytes 8 and 9 refer to an ATA/ATAPI interface.	7	0 – Emulated image is not driver dependent. 1- Emulated image contains SCSI drivers for CD/DVD-ROM, bytes 8 and 9 refer to a SCSI interface.
Bit	Description																							
Value	Emulation																							
0	No Emulation																							
1	1.2MB Diskette																							
0-3	2 1.44MB Diskette																							
3	2.88MB Diskette																							
4	Hard Disk																							
5-0Fh	Reserved																							
4-5	Reserved																							
6	0 – Emulated image is not driver dependent. 1- Emulated image contains ATAPI drivers for CD/DVD-ROM, bytes 8 and 9 refer to an ATA/ATAPI interface.																							
7	0 – Emulated image is not driver dependent. 1- Emulated image contains SCSI drivers for CD/DVD-ROM, bytes 8 and 9 refer to a SCSI interface.																							
2	Byte	INT 13h device number. This shall be the INT 13h number of the device which shall be started or terminated. This value shall be zero for a floppy image, 80h for a bootable hard disk, and 81h-FFh for a non-bootable or no emulation device.																						
3	Byte	Controller Index. This shall be the controller number of the to which the CD/DVD-ROM drive is attached. If there are several controllers in a system, their number is system dependent.																						
4-7	DWord	Logical Block Address of the disk image to be emulated. This shall be the absolute sector address of the image on the CD or DVD media.																						
8-9	Word	Device Number. SCSI controllers shall use byte 8 as the Logical Unit Number (LUN) and byte 9 as the Physical Unit Number (PUN) of the CD or DVD drive. ATA/ATAPI controllers shall use bit 0 of byte 8 to indicate device 0 or device 1.																						
10-11	Word	User Buffer Segment. If this field is non-zero, the system may use a supplied buffer for caching CD/DVD reads. This buffer shall be at least 3k in size. The buffer begins at segment:0																						
12-13	Word	Load Segment. This field is only used for booting (INT 13h FN 4Ch). If the value is zero, the system shall use 7C0h as the load address and start address for booting. if the value is non-zero, the system shall load the boot image at																						

Offset	Type	Description
		segment:0 and initiate the boot process by jumping to segment:0
14-15	Word	Sector Count. This field is only used for booting (INT 13h FN 4Ch). The sector count is the number of emulated sectors to load, starting at the load segment (see offset 12-13) during the initial boot procedure.
16	Byte	This shall match the value returned in CH when INT 13h FN – 8h is invoked. This field is Bits 0-7 of the cylinder count.
17	Byte	This shall match the value returned in CL when INT 13h FN – 8h is invoked. This field contains the number of sectors per track in bits 0-5 and the high order 2 bits (bits 8 and 9) of the cylinder count
18	Byte	This shall match the value returned in DH when INT 13h FN – 8h is invoked. This field contains the number of heads.

Initiate Disk Emulation uses the CD Specification Packet to create an emulated disk drive. The emulated device shall have the device number supplied in the CD Specification Packet and can be accessed using INT13h functions 0-49h.

## 8.27 Terminate Disk Emulation

Entry:

AH - 4Bh

AL - 00h, return status and terminate emulation, 01h - return status only, do not terminate emulation.

DL - Device number to terminate, 7Fh means terminate all devices.

DS:SI - Empty CD Specification Packet (See Table 17)

Exit:

carry clear

AH - 00

DS:SI - Filled-in CD Specification Packet (See Table 17)

carry set

AH - Error code

Terminate Disk Emulation returns the system to a configuration that does not include the emulated devices. If the CD booted as a floppy disk, and the system floppy drive was pushed to be the second drive, or B:, the system floppy drive shall now become the first drive, or A:. This could have implications for currently loaded drivers and software. Effected drivers and software shall account for the change in this access method. If the CD booted as a hard disk, the system hard disk was pushed to the second drive to allow the emulated hard disk to be the boot device. If the emulation is terminated, the system hard disk shall move to become the first hard disk. This could have implications for currently loaded drivers and software. Effected drivers and software shall account for the change in this access method.

When the system is booted from a CD, software can determine the CD/DVD ROM drive that booted the system by issuing Terminate Disk Emulation with AL=01h. This shall return a completed CD Specification Packet as defined in Table 17

## 8.28 Initiate Disk Emulation & Boot

### Entry:

AH - 4Ch  
 AL - 00h  
 DS:SI - CD Specification Packet (See Table 17)

### Exit:

carry clear

This condition shall not happen. If the function is successful this system shall reboot causing the calling software to be terminated.

carry set

AH - Error code, emulation not initiated, boot failed.

Initiate Disk Emulation uses the CD Specification Packet to create an emulated disk drive. The emulated device shall have the device number supplied in the CD Specification Packet and can be accessed using INT13h functions 0-49h. Once the emulation has been initiated, the system shall be rebooted using the emulated device. This capability allows a CD to have a default selection program which may allow the user to choose from a variety of boot images on the CD. Once the choice is made, the system can be rebooted using the new image.

## 8.29 Return Boot Catalog

### Entry:

AH - 4Dh  
 AL - 00h  
 DS:SI - Pointer to CD Command Packet (See Table 18)

### Exit:

carry clear

AH - 00

carry set

AH - Error code, boot catalog not available

**Table 18 - CD Command Packet**

Offset	Type	Description
0	Byte	8 - Packet size in bytes
1	Byte	Sector count. Number of sectors in the boot catalog to transfer
2-5	DWord	Pointer to the buffer where the boot catalog shall be stored
6-7	Word	Beginning sector to transfer. This number shall be relative to the start of the boot catalog. This value shall be cleared to zero to retrieve the first sector.

Return Boot Catalog is returns the Boot Catalog from the CD or DVD media. See section 7 for a description of the Boot Catalog.

### 8.30 Set Hardware Configuration

Entry:

AH - 4Eh  
 AL - Hardware configuration sub-function (See Table 19)  
 DL - BIOS device number.

Exit:

carry clear  
 AH - 0  
 AL - 0 if command was safe, 1 if other devices are affected  
 carry set  
 AH - Error code

**Table 19 - Hardware Configuration Sub-Functions**

AL	Sub-function description
0h	Enable prefetch
1h	Disable prefetch
2h	Set maximum PIO transfer mode.
3h	Set PIO mode 0.
4h	Return to default PIO transfer mode. Return the system to the PIO mode enabled by the BIOS setup utility.
5h	Enable INT 13h DMA maximum mode. Set the maximum rate allowed by both the host adapter and the device.
6h	Disable INT 13h DMA

The purpose of this function is to allow non-hardware-specific software to configure host adapter and devices for optimal operation. ATA channels may have two devices attached, but this function operates on a single-device basis. This shall be accommodated by the value that is returned in AL. If the host adapter supports the requested sub-function on a device basis, AL shall be cleared to 00h. If the host adapter only supports the setting on an ATA channel basis, AL shall be set to 01h. Once this function has been invoked, all subsequent INT 13h device-access functions shall use the mode specified by this invocation. This means that if "DMA Maximum" is enabled, INT 13h Fn 02h shall read from the device using DMA transfers. The DMA/PIO selections are mutually exclusive. When "DMA Maximum" is enabled, "PIO Maximum" shall be disabled. If the requested mode change is not supported this function shall return with CF = 1b and AH = 01h

### 8.31 Send Packet Command

Entry:

AH - 50h  
 AL - D7h  
 DL - BIOS device Number  
 ES:SI - Pointer to formatted command packet, (see Table 20).

Exit:

carry clear  
 AH - 0  
 carry set  
 AH - 01 - Function not implemented, 80h - Command failed to complete, 97h - Subfunction D7h not supported for this device, C3h - Formatted Command Packet is too short

**Table 20 - Formatted Command Packet**

Offset	Type	Description
0	Word	Length of this record in bytes
2-n	Byte	Formatted protocol specific data

This function defines a service that the system BIOS shall call for sending data to and from a packet oriented device. The BIOS shall provide this service before the OS is loaded. When an operating system takes control of the device controller it takes the hook for this service to provide a seamless transfer of control from the BIOS to the operating system. This service allows several BIOS level services to continue functioning, even after the OS has taken control of the device controller, for example:

- The INT 13h mass storage interface
- Power Management
- Suspend to disk

The BIOS is a single threaded, master device. This means that the BIOS shall not process asynchronous requests from other devices. The BIOS shall send commands to devices and wait for responses. This means that the operating system may take control of the serial interface with no hand-off information from the system BIOS. The operating system shall reconfigure the interface and hook the service described above. The system BIOS may provide INT 13h Fn 50h for the 1394, USB, and any other packet oriented bus.

The format of the packet shall be determined by the requirements of the target bus and is beyond the scope of this document. In the case of USB, the packet format shall be determined by the USB specification. In the case of 1394, IEEE 1394-1995 shall determine the size of a packet with payload information defined in SBP-2.

#### **8.31.1 Packet Sending Service (PSS) For SCSI Command Data Block's (CDB)**

The purpose of this interface is to allow an application to send SCSI CDB's using BIOS INT 13h calls to a device, regardless of the bus on which the device resides. This interface is currently defined for ATAPI, SCSI, 1394, and USB devices. Future buses that use SCSI CDB's may also adopt this mechanism for transporting commands

This interface shall not retry any operation. If an error occurs either in command format, or operation of the device, an error shall be returned by INT 13h. The application may choose to retry a command by issuing it again.

This PSS shall provide transport specific wrappers and modify the CDB where necessary with bus specific information. If the secondary bus is a hot-plug bus such as 1394 or USB, the PSS shall insert device addresses or EUI-64 as necessary. In the event that a device address changes due to a bus reset or re-enumeration, the PSS shall connect with the device without generating an error.

### 8.31.1.1 Formatted Protocol Specific Data

Table 21 defines the format of the data passed in ES:SI (See 8.31) to INT 13h Fn 50h.

**Table 21 - Formatted Protocol Specific Data**

Offset	Type	Description
0	Word	Packet length in bytes including this word
2	Byte	Information flags Bits 6-7 = Data direction. 00 = No data exchanged with device (command only) 01 = Receive data from device 10 = Send data to device 11 = Reset Interface. Bits 0-5 = Reserved.
3	Byte	Bytes in command packet, or zero if no command is to be sent.
4	DWord	Pointer to command packet. This field follows the seg:offset address format. This field is ignored if offset 3 is zero.
8	DWord	Number of bytes to transfer following command complete. Shall be a multiple of the sector size of the media, as returned by Function 48h. If the data direction bits in Flags do not indicate a send or receive data command, then this field shall be ignored. Otherwise, if this value shall be zero, then a 0-byte transfer shall occur.
10h	DWord	Pointer to the start of the data stream to transfer. This field follows the seg:offset address format.
14	Word	Access timeout. Maximum time in milliseconds to wait for command start. Command start includes sending the command to the devices and waiting for the first data byte to transfer. If this field contains the value of 0000h, an 80h error shall result
16	Word	Sector timeout increment. The PSS shall divide the amount of data transferred by the sector size, round up, multiply by this value, and add to the Access Timeout (offset 14), the maximum result is limited to FFFFh. This shall be the total timeout for the specified CDB. This value shall be ignored for command-only and bus-reset calls.

### 8.31.1.2 Output Parameters

The following output parameters, see Table 22, match the parameters defined in 8.31. No other return parameters are necessary for this transport.

**Table 22 - Output Parameters**

OUT	Description														
AL	Undefined														
AH	<table border="1"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>00h</td> <td>No Error</td> </tr> <tr> <td>01h</td> <td>AX=50D7h not implemented</td> </tr> <tr> <td>80h</td> <td>A condition has occurred which prevented successful command completion. This may be due to a busy device, a timeout, a bus reset, short packet, or other event. The host should issue a Request-Sense-type command in the required protocol.</td> </tr> <tr> <td>97h</td> <td>There is no support for the indicated logical drive, although Function 50D7h is recognized.</td> </tr> <tr> <td>C3h</td> <td>PSSP is too short.</td> </tr> <tr> <td>Remaining Values</td> <td>Reserved</td> </tr> </tbody> </table>	Value	Description	00h	No Error	01h	AX=50D7h not implemented	80h	A condition has occurred which prevented successful command completion. This may be due to a busy device, a timeout, a bus reset, short packet, or other event. The host should issue a Request-Sense-type command in the required protocol.	97h	There is no support for the indicated logical drive, although Function 50D7h is recognized.	C3h	PSSP is too short.	Remaining Values	Reserved
Value	Description														
00h	No Error														
01h	AX=50D7h not implemented														
80h	A condition has occurred which prevented successful command completion. This may be due to a busy device, a timeout, a bus reset, short packet, or other event. The host should issue a Request-Sense-type command in the required protocol.														
97h	There is no support for the indicated logical drive, although Function 50D7h is recognized.														
C3h	PSSP is too short.														
Remaining Values	Reserved														
All other registers	Preserved														
Carry flag	0 if AH is cleared to zero 1 if AH is 1through FFh.														

## 9 INT 15h Removable Media Eject

Entry:

AH - 52h  
DL - BIOS device number

Exit:

carry clear  
AH - 0, ejection may proceed  
carry set  
AH - error code, B1h or B3h, ejection is rejected

This function shall be called by the BIOS in response to a software request (INT 13h, AH=46h, Eject Device) to eject media from a removable media device.

A user may press an eject button or use a software command to request that a particular media be ejected. By default the INT 15h handler returns with ejection accepted status. A disk cache program may hook this INT15h call and return acceptance or rejection based on the state of its buffers for this disk. This function may also be used by operating system software as a media change request.