

**Universal Serial Bus  
Mass Storage Class  
Specification Overview**

Revision 1.4  
February 19, 2010

## Document Status

Revision History			
Rev	Date	Author	Description
1.0	October 22, 1998		1) Initial release
1.1	June 28, 2000		1) Update
1.2	June 23, 2003		1) Update list of specs 2) Restrict CBI to full-speed floppies only
1.3	September 5, 2008	John Geldman	1) Updated related documents & subclass references, 2) added MSC-Lock & IEEE 1667 subclasses, 3) added Vendor Specific subclass & protocol, 4) updated protocol list, added section on Request Codes
1.4rc1	June 23, 2009	John Geldman Curtis E. Stevens	1) Corrected references to obsolete SubClass Codes in Table 2.1 2) Adds USB Attached SCSI Protocol Reference 3) Adds Protocol UASP subgroup approved for usage in T10 UAS work 4) Updated list of contributors 5) Performed spelling and grammar check 6) Updated formatting for consistency.
1.4rc2	August 5, 2009	Curtis E. Stevens	1) Updated with letter ballot comment corrections 2) Added list of previous contributors 3) Added USB-2 as "core Spec"
1.4rc3	February 19, 2010	Curtis E, Stevens	1) Changed the introduction to the contributors list to match USB-IF requirements.

The authors of this specification would like to thank the people from the Promoter companies, the people who participated in the UASP Specification technical working group, and those throughout the industry who contributed to the development of this specification. The following people have contributed to the Mass Storage Overview Class Specification.

<b>MSC Specification Overview Contributors</b>		
<b>Name</b>	<b>E-Mail</b>	<b>Company</b>
Amit Nanda		Cypress Semiconductor
Hans van Antwerpen		Cypress Semiconductor
Chuck Trefts	chuck.trefts@ellisis.com	Ellisis
Mario Pasquali	mario.pasquali@ellisis.com	Ellisis
Yuji Oishi	bigstone.gm@hscjpn.co.jp	Hagiwara Sys-Com
Steve McGowan	steve.mcgowan@intel.com	Intel
Boris Dinkevich		Jungo
Joel Silverman	joel@k-micro.us	Kawasaki Microelectronics
Daniel Jacobs	daniel.jacobs@lecroy.com	Lecroy
John Geldman	jgeldman@lexar.com	Lexar Media
Jason R. Oliver		MCCI
Cristian Chis	cchis@mcci.com	MCCI
John Garney	john.garney@mcci.com	MCCI
Jim Bovee	jibove@microsoft.com	Microsoft
Kiran Bangalore	kiran.bangalore@microsoft.com	Microsoft
Eugene Gryazin	eugene.gryazin@nokia.com	Nokia
Matthew Stephens	mstephens@plxtech.com	PLX/Oxford
Shigekatsu Tateno	stateno@plxtech.com	PLX/Oxford
Thomas Friend	tfriend@plxtech.com	PLX/Oxford
Dave Landsman	Dave.Landsman@sandisk.com	SanDisk
Martin Furuholm	martin.r.furuholm@seagate.com	Seagate Technology
Tony Priborsky	Tony.Priborsky@seagate.com	Seagate Technology
Cindy Lee		SMSC
Shannon Cash		SMSC
Gideon Intrater	Gideon.Intrater@symwave.com	Symwave
Biao Jia	Biao.Jia@symwave.com	Symwave
Rudy Liang	Rudy.Liang@symwave.com	Symwave
Christopher Thomas	chris.thomas@symwave.com	Symwave
Grant Ley	g-ley@ti.com	Texas Instruments
Ed Beeman		USB-IF
Paul E. Berg		USB-IF
Curtis E. Stevens	Curtis.Stevens@wdc.com	Western Digital Technologies

<b>Contributors to Previous Versions</b>	
<b>Name</b>	<b>Company</b>
Shing F. Lin	Adaptec
Bill Stanley	Adaptec
Masayuki Kitagawa	Alpha Peripherals
Bill Russell	Canon
Dave Gilbert	Cypress Semiconductor
Steve Kolokowsky	Cypress Semiconductor
David Luke	Cypress Semiconductor
Eric Luttmann	Cypress Semiconductor
David Cho	Genesys Logic
Sean S. Cho	Genesys Logic
Hikomichi Oribe	Hagiwara Sys-Com Co.
David L. Jolley	Iomega Corporation
Mike Poulsen	Iomega Corporation
James Quigley	Iomega Corporation
Darrell Redford	Iomega Corporation
Jim Sandman	Iomega Corporation
Martin Furuhjelm	Lexar Media
John Geldman	Lexar Media
Pat LaVarre	Lexar Media
Alex Afshar	Matsushita Semiconductor
Nathan Obr	Microsoft Corporation
Glen Slick	Microsoft Corporation
Mark Williams	Microsoft Corporation
Al Rickey	Phoenix Technologies
Johan Craeybeckk	PIMC/Philips
Jan Matejica	PIMC/Philips
Peter S'Heeren	PIMC/Philips
Yoni Shternhell	SanDisk
Trenton Henry	SMSC
Shigenori Tagami	Sony Corporation
Frits Vanderlinden	Sun Microsystems
Curtis E. Stevens	Western Digital
Masahiro Ito	Yamagata Fujitsu
Toyoko Shimizu	Y-E Data, Inc
Jim Blackson	Y-E Data, Inc

Copyright © 2009, USB Implementers Forum, Inc.

All rights reserved.

A LICENSE IS HEREBY GRANTED TO REPRODUCE THIS SPECIFICATION FOR INTERNAL USE ONLY. NO OTHER LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, IS GRANTED OR INTENDED HEREBY.

USB-IF AND THE AUTHORS OF THIS SPECIFICATION EXPRESSLY DISCLAIM ALL LIABILITY FOR INFRINGEMENT OF INTELLECTUAL PROPERTY RIGHTS, RELATING TO IMPLEMENTATION OF INFORMATION IN THIS SPECIFICATION. USB-IF AND THE AUTHORS OF THIS SPECIFICATION ALSO DO NOT WARRANT OR REPRESENT THAT SUCH IMPLEMENTATION(S) WILL NOT INFRINGE THE INTELLECTUAL PROPERTY RIGHTS OF OTHERS.

THIS SPECIFICATION IS PROVIDED "AS IS" AND WITH NO WARRANTIES, EXPRESS OR IMPLIED, STATUTORY OR OTHERWISE. ALL WARRANTIES ARE EXPRESSLY DISCLAIMED. NO WARRANTY OF MERCHANTABILITY, NO WARRANTY OF NON-INFRINGEMENT, NO WARRANTY OF FITNESS FOR ANY PARTICULAR PURPOSE, AND NO WARRANTY ARISING OUT OF ANY PROPOSAL, SPECIFICATION, OR SAMPLE.

IN NO EVENT WILL USB-IF OR USB-IF MEMBERS BE LIABLE TO ANOTHER FOR THE COST OF PROCURING SUBSTITUTE GOODS OR SERVICES, LOST PROFITS, LOSS OF USE, LOSS OF DATA OR ANY INCIDENTAL, CONSEQUENTIAL, INDIRECT, OR SPECIAL DAMAGES, WHETHER UNDER CONTRACT, TORT, WARRANTY, OR OTHERWISE, ARISING IN ANY WAY OUT OF THE USE OF THIS SPECIFICATION, WHETHER OR NOT SUCH PARTY HAD ADVANCE NOTICE OF THE POSSIBILITY OF SUCH DAMAGES.



**Table of Contents**

	Page
Document Status .....	ii
Table of Contents .....	vii
List of Tables .....	viii
1 Introduction .....	1
1.1 Overview .....	1
1.2 Specification Relationships .....	1
1.3 Purpose .....	1
1.4 Terms and Abbreviations .....	1
1.5 Related Documents .....	2
2 Subclass Codes .....	3
3 Protocol Codes .....	4
4 Request Codes .....	5
5 Class Specific Descriptor Codes .....	6

## List of Tables

	Page
Table 1 - SubClass Codes Mapped to Command Block Specifications.....	3
Table 2 - Mass Storage Transport Protocol.....	4
Table 3 - Mass Storage Request Codes.....	5
Table 4 - Mass Storage Class Specific Descriptors.....	6

## 1 Introduction

### 1.1 Overview

This document gives an overview of the USB Mass Storage Class specifications. How mass storage devices behave on the USB bus is the subject of this and other USB Mass Storage Class specifications. In addition to this Overview specification, several other USB Mass Storage Class specifications are supported by the USB Mass Storage Class Working Group (CWG). The titles of these specifications are:

*USB Mass Storage Class Control/Bulk/Interrupt (CBI) Transport*

*USB Mass Storage Class Bulk-Only (BBB) Transport*

*USB Mass Storage Class UFI (UFI) Command Specification*

*USB Mass Storage Class Bootability Specification*

*USB Mass Storage Class Compliance Test Specification*

*USB Lockable Storage Devices Feature Specification (LSD FS)*

*USB Mass Storage Class USB Attached SCSI Protocol (UASP)*

The USB Mass Storage Class Control/Bulk/Interrupt (CBI) Transport specification is approved for use only with full-speed floppy disk drives. CBI shall not be used in high-speed capable devices, or in devices other than floppy disk drives. CBI shall not be used in devices that implement LSDFS. Usage of CBI for any new design is discouraged.

NOTE 1 — The Compliance Test specifications are still under development, and are not yet publicly available.

### 1.2 Specification Relationships

This is a normative document for USB Mass Storage Class devices. If there is a code assignment conflict between this document and another USB document, then the assignments in this document override the other document.

The CBI and Bulk-Only transport protocol specifications are each intended to be stand-alone documents for the USB Mass Storage class, enabling development of a USB Mass Storage compliant device. A device manufacturer may choose to implement both CBI and Bulk-Only, but shall follow each specification as applicable.

Booting an operating system from a USB Mass Storage Class device requires no special considerations with regard to Mass Storage Class support. Either CBI or Bulk-Only devices may be bootable. Bootability may, however, require other considerations such as particular types of media formatting, etc. Such considerations are hardware- or operating system dependent, and are beyond the scope of the Mass Storage Class specifications.

### 1.3 Purpose

The purpose of this document is to provide an overview of all the specifications that describe how Mass Storage devices behave on the USB bus. Section 1.1 gives the rules for using the different USB Mass Storage class specifications.

Note that these rules can change. As other companies with different USB Mass Storage Class device projects in mind join the USB Mass Storage Class CWG, other specifications may be developed by the CWG and added to the set of specifications that fully describe how a Mass Storage Class device behaves on the USB bus. If and when that happens, the USB Mass Storage CWG will reconsider the rules specified in section 1.1 of this document.

### 1.4 Terms and Abbreviations

**May:** A keyword that indicates an option.

**Shall:** A keyword that indicates a requirement.

## 1.5 Related Documents

USB Mass Storage specifications use the command sets from several existing protocols. The command blocks of these command sets are placed in a USB wrapper which follows USB protocol. The following specifications are referenced by the USB Mass Storage specifications:

- a) Universal Serial Bus Specification revision 2.0 (core USB), available at <http://www.usb.org>
- b) Reduced Block Commands (RBC), INCITS 330:2000, available at <http://www.t10.org>
- c) Multi-Media Command Set 5 (MMC-5), T10/1675-D available at <http://www.t10.org>
- d) Standard Protocol for Authentication in Host Attachments of Transient Storage Devices (IEEE 1667) available at [www.ieee1667.com](http://www.ieee1667.com)
- e) USB Attached SCSI (UAS) T10/2095-D Available at <http://www.t10.org>

## 2 Subclass Codes

The Interface Descriptor of a USB Mass Storage Class device includes a *bInterfaceSubClass* field. This field denotes the industry-standard command set transported by a Mass Storage Class interface. The value of the *bInterfaceSubClass* field shall be set to one of the Subclass codes as shown in table 1.

The Subclass code values used in the *bInterfaceSubClass* field specify the industry-standard specification that defines transport command sets transported by the interface; these Subclass codes do not specify a type of storage device (such as a CD-ROM or floppy disk drive).

**Table 1 — SubClass Codes Mapped to Command Block Specifications**

Subclass	Command Block Specification	Comment
00h	SCSI command set not reported	De facto use
01h	RBC	Allocated by USB-IF for RBC. RBC is defined outside of USB.
02h	MMC-5 (ATAPI)	Allocated by USB-IF for MMC-5. MMC-5 is defined outside of USB.
03h	Obsolete	Was QIC-157
04h	UFI	Specifies how to interface Floppy Disk Drives to USB
05h	Obsolete	Was SFF-8070i
06h	SCSI transparent command set	Allocated by USB-IF for SCSI. SCSI standards are defined outside of USB.
07h	LSD FS	LSDFS specifies how host has to negotiate access before trying SCSI
08h	IEEE 1667	Allocated by USB-IF for IEEE 1667. IEEE 1667 is defined outside of USB.
09h - FEh	Reserved	Reserved
FFh	Specific to device vendor	De facto use

### 3 Protocol Codes

The Interface Descriptor of a USB Mass Storage Class device includes a *bInterfaceProtocol* field. This field denotes the transport protocol used by this interface.

**Table 2 — Mass Storage Transport Protocol**

<i>bInterfaceProtocol</i>	Protocol Implementation	Comment
00h	CBI (with command completion interrupt)	USB Mass Storage Class Control/Bulk/Interrupt (CBI) Transport
01h	CBI (with no command completion interrupt)	USB Mass Storage Class Control/Bulk/Interrupt (CBI) Transport
02h	Obsolete	
03h-4Fh	Reserved	Reserved
50h	BBB	USB Mass Storage Class Bulk-Only (BBB) Transport
51h - 61h	Reserved	Reserved
62h	UAS	Allocated by USB-IF for UAS. UAS is defined outside of USB.
63h-FEh	Reserved	Reserved
FFh	Specific to device vendor	De facto use

The USB Mass Storage Class Control/Bulk/Interrupt (CBI) Transport specification (Protocol codes 0x00 and 0x01) is approved for use only with full-speed floppy disk drives. CBI shall not be used in high-speed capable devices, or in devices other than floppy disk drives. Usage of CBI for *any* new design is discouraged.

## 4 Request Codes

Core USB specifies that a USB control Request addressed to  $wIndex = bInterfaceNumber$  of a USB Mass Storage Class device interface includes a  $bRequest$  field.

The meaning of the  $bRequest$  code is specific to the device vendor when the  $bmRequestType$ .Type is Vendor, but the meaning of the  $bRequest$  code is specific to the interface class when the  $bmRequestType$ .Type is Class.

**Table 3 — Mass Storage Request Codes**

<b><i>bRequest</i></b>	<b>Name</b>	<b>Comment</b>
00h	Accept Device-Specific Command (ADSC)	Assigned in context by USB Mass Storage Class Control/Bulk/Interrupt (CBI) Transport, also aliases core USB request 00h Get Status.
01h - 0Dh	Reserved	Aliases of core USB bRequest codes
0Eh - FBh	Reserved	Reserved
FCh	Get Requests	Assigned by Lockable Storage Devices Feature Specification
FDh	Put Requests	Assigned by Lockable Storage Devices Feature Specification
FEh	Get Max LUN (GML)	Assigned by USB Mass Storage Class Bulk-Only (BBB) Transport
FFh	Bulk-Only Mass Storage Reset (BOMSR)	Assigned by USB Mass Storage Class Bulk-Only (BBB) Transport

## 5 Class Specific Descriptor Codes

Some Mass Storage subclass and protocol specifications require class specific descriptors. Table 4 lists the class specific descriptors used by the Mass Storage Class.

**Table 4 — Mass Storage Class Specific Descriptors**

<b>Descriptor Code</b>	<b>Name</b>	<b>Comment</b>
00h - 23h		Not defined by Mass Storage
24h	Pipe Usage Class Specific Descriptor	Allocated by USB-IF for UAS. UAS is defined outside of USB.
25h - FFh		Not defined by Mass Storage