Errata for "Universal Serial Bus Class Definitions for Communications Devices Revision 1.2 November 16, 2007" as of November 3, 2010

Issue A

Related to: 1.3 Related documents

Issue: NCM is missing in list of related documents.

Resolution: Correction as follows.

Reference	Title	Revision
[USB2.0]	Universal Serial Bus Specification (also referred to as the USB Specification).	2.0
[USBCCS1.0]	Universal Serial Bus Common Class Specification	1.0
[USBPSTN1.2]	USB CDC Subclass Specification for PSTN Devices	1.2
[USBISDN1.2]	USB CDC Subclass Specification for ISDN Devices	1.2
[USBECM1.2]	USB CDC Subclass Specification for Ethernet Devices	1.2
[USBNCM1.0]	Universal Serial Bus Communications Class Subclass Specifications for Network Control Model Devices	1.0
[USBATM1.2]	USB CDC Subclass Specification for ATM Devices	1.2
[USBWMC1.1]	USB CDC Subclass Specification for Wireless Mobile Communications Devices	1.1
[USBEEM1.0]	USB CDC Subclass Specification for Ethernet Emulation Devices	1.0
[ISO3166]	ISO 3166:1993, Codes for the representation of the names of countries. http://www.iso.org	

Field Code Changed

Issue B

Related to: Table 4 Class Subclass Code

Issue: Missing Subclass Code for NCM.

Resolution: Correction as follows.

Table 4 Class Subclass Code

Code	Subclass	Reference
00h	RESERVED	
01h	Direct Line Control Model	[USBPSTN1.2]
02h	Abstract Control Model	[USBPSTN1.2]
03h	Telephone Control Model	[USBPSTN1.2]
04h	Multi-Channel Control Model	[USBISDN1.2]
05h	CAPI Control Model	[USBISDN1.2]
06h	Ethernet Networking Control Model	[USBECM1.2]
07h	ATM Networking Control Model	[USBATM1.2]
08h	Wireless Handset Control Model	[USBWMC1.1]
09h	Device Management	[USBWMC1.1]
0Ah	Mobile Direct Line Model	[USBWMC1.1]
0Bh	OBEX	[USBWMC1.1]
0Ch	Ethernet Emulation Model	[USBEEM1.0]
0Dh	Network Control Model	[USBNCM1.0]
0Eh-7Fh	RESERVED (future use)	
80-FEh	RESERVED (vendor specific)	

Deleted: 0Dh

Issue C

Related to: Table 7: Data Interface Class Protocol Codes

Issue: Missing NCM Data Class Protocol code ntb.

Resolution: Correction as follows.

Table 7: Data Interface Class Protocol Codes

Protocol Code	Reference document	Description
00h	USB specification	No class specific protocol required
01h	[USBNCM1.0]	Network Transfer Block
02h – 2Fh	None	RESERVED (future use)
30h	I.430	Physical interface protocol for ISDN BRI
31h	ISO/IEC 3309-1993	HDLC
32h	None	Transparent
33h – 4Fh	None	RESERVED (future use)
50h	Q.921M	Management protocol for Q.921 data link protocol
51h	Q.921	Data link protocol for Q.931
52h	Q921TM	TEI-multiplexor for Q.921 data link protocol
53h – 8Fh	None	RESERVED (future use)
90h	V.42bis	Data compression procedures
91h	Q.931/Euro- ISDN	Euro-ISDN protocol control
92h	V.120	V.24 rate adaptation to ISDN
93h	CAPI2.0	CAPI Commands
94h - FCh	None	RESERVED (future use)
FDh	None	Host based driver. Note: This protocol code should only be used in messages between host and device to identify the host driver portion of a protocol stack.
FEh	CDC Specification	The protocol(s) are described using a Protocol Unit Functional Descriptors on Communications Class Interface
FFh	USB Specification	Vendor-specific

Deleted: 01h

Issue D

Related to: Table 13: bDescriptor SubType in Communications Class Functional Descriptors

Issue: NCM Functional Descriptor missing.

Resolution: Correction as follows.

Table 13: bDescriptor SubType in Communications Class Functional Descriptors

Descriptor subtype	Functional description
00h	Header Functional Descriptor, which marks the beginning of the concatenated set of functional descriptors for the interface.
01h	Call Management Functional Descriptor.
02h	Abstract Control Management Functional Descriptor.
03h	Direct Line Management Functional Descriptor.
04h	Telephone Ringer Functional Descriptor.
05h	Telephone Call and Line State Reporting Capabilities Functional Descriptor.
06h	Union Functional Descriptor
07h	Country Selection Functional Descriptor
08h	Telephone Operational Modes Functional Descriptor
09h	USB Terminal Functional Descriptor
0Ah	Network Channel Terminal Descriptor
0Bh	Protocol Unit Functional Descriptor
0Ch	Extension Unit Functional Descriptor
0Dh	Multi-Channel Management Functional Descriptor
0Eh	CAPI Control Management Functional Descriptor
0Fh	Ethernet Networking Functional Descriptor
10h	ATM Networking Functional Descriptor
11h	Wireless Handset Control Model Functional Descriptor
12h	Mobile Direct Line Model Functional Descriptor
13h	MDLM Detail Functional Descriptor
14h	Device Management Model Functional Descriptor
15h	OBEX Functional Descriptor

16h	Command Set Functional Descriptor
17h	Command Set Detail Functional Descriptor
18h	Telephone Control Model Functional Descriptor
19h	OBEX Service Identifier Functional Descriptor
1Ah	NCM Functional Descriptor
1Bh-7Fh	RESERVED (future use)
80h-FEh	RESERVED (vendor specific)

Deleted: 1Ah

Issue E

Related to: Table 19: Class Specific Request Codes

Issue:

Resolution: Correction as follows.

Table 19: Class-Specific Request Codes

Request Code	Value	reference
SEND_ENCAPSULATED_COMMAND	00h	6.2.1
GET_ENCAPSULATED_RESPONSE	01h	6.2.2
SET_COMM_FEATURE	02h	[USBPSTN1.2]
GET_COMM_FEATURE	03h	[USBPSTN1.2]
CLEAR_COMM_FEATURE	04h	[USBPSTN1.2]
RESERVED (future use)	05h-0Fh	
SET_AUX_LINE_STATE	10h	[USBPSTN1.2]
SET_HOOK_STATE	11h	[USBPSTN1.2]
PULSE_SETUP	12h	[USBPSTN1.2]
SEND_PULSE	13h	[USBPSTN1.2]
SET_PULSE_TIME	14h	[USBPSTN1.2]
RING_AUX_JACK	15h	[USBPSTN1.2]
RESERVED (future use)	16h-1Fh	
SET_LINE_CODING	20h	[USBPSTN1.2]
GET_LINE_CODING	21h	[USBPSTN1.2]
SET_CONTROL_LINE_STATE	22h	[USBPSTN1.2]
SEND_BREAK	23h	[USBPSTN1.2]
RESERVED (future use)	24h-2Fh	
SET_RINGER_PARMS	30h	[USBPSTN1.2]
GET_RINGER_PARMS	31h	[USBPSTN1.2]
SET_OPERATION_PARMS	32h	[USBPSTN1.2]
GET_OPERATION_PARMS	33h	[USBPSTN1.2]
SET_LINE_PARMS	34h	[USBPSTN1.2]
GET_LINE_PARMS	35h	[USBPSTN1.2]

DIAL_DIGITS	36h	[USBPSTN1.2]
SET_UNIT_PARAMETER	37h	[USBISDN1.2]
GET_UNIT_PARAMETER	38h	[USBISDN1.2]
CLEAR_UNIT_PARAMETER	39h	[USBISDN1.2]
GET_PROFILE	3Ah	[USBISDN1.2]
RESERVED (future use)	3Bh-3Fh	
SET_ETHERNET_MULTICAST_FILTERS	40h	[USBECM1.2]
SET_ETHERNET_POWER_MANAGEMENT_PATTERN_FILTER	41h	[USBECM1.2]
GET_ETHERNET_POWER_MANAGEMENT_PATTERN_FILTER	42h	[USBECM1.2]
SET_ETHERNET_PACKET_FILTER	43h	[USBECM1.2]
GET_ETHERNET_STATISTIC	44h	[USBECM1.2]
RESERVED (future use)	45h-4Fh	
SET_ATM_DATA_FORMAT	50h	[USBATM1.2]
GET_ATM_DEVICE_STATISTICS	51h	[USBATM1.2]
SET_ATM_DEFAULT_VC	52h	[USBATM1.2]
GET_ATM_VC_STATISTICS	53h	[USBATM1.2]
RESERVED (future use)	54h-5Fh	
MDLM Semantic-Model specific Requests	60h – 7Fh	[USBWMC1.2]
GET_NTB_PARAMETERS	80h	[USBNCM1.0]
GET_NET_ADDRESS	81h	[USBNCM1.0]
SET_NET_ADDRESS	82h	[USBNCM1.0]
GET_NTB_FORMAT	83h	[USBNCM1.0]
SET_NTB_FORMAT	84h	[USBNCM1.0]
GET_NTB_INPUT_SIZE	85h	[USBNCM1.0]
SET_NTB_INPUT_SIZE	86h	[USBNCM1.0]
GET_MAX_DATAGRAM_SIZE	87h	[USBNCM1.0]
SET_MAX_DATAGRAM_SIZE	88h	[USBNCM1.0]
GET_CRC_MODE	89h	[USBNCM1.0]
SET_CRC_MODE	8Ah	[USBNCM1.0]
RESERVED (future use)	8Bh-FFh	

Deleted: 80h

Issue F

Related to: *6.3.3 ConnectionSpeedChange*

Issue: The CDC specification has been written to be for mobile devices so many of the authors and the potential readers are using the 3gpp definitions. Hence the CDC 1.1 specification had a redefinition of downstream and upstream (E.g the opposite to the core spec) to be more 3gpp like. This redefinition was lost in the cleanup work of CDC 1.2. To not build in an area where most people will make mistakes we should try to clarify the intention as follows

Resolution: Correction as follows.

1.4 Terms and Abbreviations

Downlink   direction from mobile network to the mobile device

Uplink   direction from mobile device to mobile network

6.3.3 ConnectionSpeedChange

This notification allows the device to inform the host-networking driver that a change in either the uplink or the downlink bit rate of the connection has occurred.

bmRequestType	bNotification	wValue	wIndex	wLength	Data
10100001B	CONNECTION_SPEED _CHANGE	Zero	Interface	8	Connection Speed Change Data Structure

The data phase for this notification contains a data structure with two 32 bit unsigned integers. The two values are the downlink bit rate, followed immediately by the uplink bit rate. (Table 21)

To assure that the host networking driver can always report the correct link speed, the device must send a *ConnectionSpeedChange* notification immediately after every *NetworkConnection* notification is sent. This normally occurs when the physical layer makes or loses a connection, but additionally appears implicitly after the device is reset.

Table 21: ConnectionSpeedChange Data Structure

Offset	Field	Size	Value	Description
0	DLBitRate	4	Number	Contains the downlink bit rate, in bits per second, as sent on the IN pipe
4	ULBitRate	4	Number	Contains the uplink bit rate, in bits per second, as sent on the OUT pipe

Note that uplink/downlink is the 3gpp definition, which is not the same as the upstream/downstream in the core specification.

Deleted: upstream
Deleted: downstream
Deleted: upstream
Deleted: downstream
Deleted: USBitRate
Deleted: upstream
Deleted: DSBitRate
Deleted: downstream
Deleted: is