

# Universal Serial Bus Device Class Definition for Video Devices: Stream Based Payload

Revision 1.5  
August 9, 2012

### **Contributors**

Abdul R. Ismail	Intel Corp.
Anand Ganesh	Microsoft Corp.
Anshuman Saxena	Texas Instruments
Bertrand Lee	Microsoft Corp.
David Goll	Microsoft Corp.
Eric Luttmann	Cypress Semiconductor Corp.
Geraud Mudry	Logitech Inc.
Hiro Kobayashi	Microsoft Corp.
Jean-Michel Chardon	Logitech Inc.
Jeff Zhu	Microsoft Corp.
Olivier Lechenne	Logitech Inc.
Remy Zimmermann	Logitech Inc.
Stephen Cooper	Microsoft
Richard Webb	Microsoft
Maribel Figuera	Microsoft
Chandrashekhara Rao	Logitech Inc.

**Copyright © 2012, 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.

All product names are trademarks, registered trademarks, or service marks of their respective owners.

Please send comments via electronic mail to [<video-chair>@usb.org](mailto:<video-chair>@usb.org)

## Revision History

Version	Date	Description
1.1	June 1 <sup>st</sup> , 2005	Initial release
1.5	June 7, 2012	Allow payload timestamps as optional

## Table of Contents

1	Introduction .....	1
1.1	Purpose .....	1
1.2	Scope .....	1
1.3	Related Documents .....	1
2	Video Class-Specific Information .....	2
2.1	Payload Header .....	2
2.2	Payload Data .....	3
2.2.1	Packet-Oriented Payload Data.....	3
2.2.2	Byte-Oriented Payload Data.....	4
3	Payload-Specific Information.....	4
3.1	Descriptors .....	4
3.1.1	Stream Based Format Descriptor .....	4
3.2	Video Samples.....	5

## **List of Tables**

Table 2-1 Header Definition for Stream Based Format	2
Table 3-1 Stream Based Format Descriptor	4

## **1 Introduction**

### **1.1 Purpose**

This document defines a general extensibility mechanism by which vendors can support Stream Based payload formats not defined by the *USB Device Class Definition for Video Devices* standard payload format specifications. A Stream Based payload format is any format where the video data is transferred as a Byte- or Packet-oriented stream. If the packetization scheme allows for multiplexed streams, then multiple video, as well as non-video, streams may be transported over a single endpoint.

### **1.2 Scope**

The payload format and associated header information is fully specified by this document. This includes:

- Payload Header
- Payload Data

### **1.3 Related Documents**

*USB Specification* Revision 2.0, April 27, 2000, [www.usb.org](http://www.usb.org)

*USB Device Class Definition for Video Devices*, [www.usb.org](http://www.usb.org)

*ISO/IEC International Standard 11172*; "Coding of moving pictures and associated audio for digital storage media up to about 1,5Mbits/s", November 1993.

*ISO/IEC International Standard 13818*; "Generic coding of moving pictures and associated audio information", November 1994.

*Pr ETS 300 468*, Digital broadcasting systems for television, sound and data services – Specification for service information (SI) in digital video broadcasting (DVB) systems

## 2 Video Class-Specific Information

### 2.1 Payload Header

The following is a description of USB VC Header definition for Stream Based format.

**Table 2-1 Header Definition for Stream Based Format**

HLE	Header Length							
BFH [0]	EOH	ERR	STI	RES	SCR	PTS	EOF	FID
PTS	PTS [7:0]							
	PTS [15:8]							
	PTS [23:16]							
	PTS [31:24]							
SCR	SCR [7:0]							
	SCR [15:8]							
	SCR [23:16]							
	SCR [31:24]							
	SCR [39:32]							
	SCR [47:40]							

#### HLE

*Size: 1 byte, Value: unit number in bytes*

The header length field specifies the length of the header, in bytes.

#### BFH[0]

*Size: 1 byte, Value: bit field*

*FID: Frame ID*

When the D0 bit of the **bmFramingInfo** field of the Video Probe and Commit Control is set, this field is used to indicate codec-specific segments, such that the value will remain constant throughout a codec-specific segment, then toggle at the beginning of the next segment. Otherwise, this field is ignored and shall be set to zero. For detailed information, see section 4.3.1.1 “Video Probe and Commit Controls” in the *USB Device Class Definition for Video Devices* specification.

*EOF: End of Frame*

When the D1 bit of the **bmFramingInfo** field of the Video Probe and Commit Control is set, this field is used to indicate the end of a codec-specific segment. Otherwise, this field is ignored and shall be set to zero. For detailed information, see section 4.3.1.1 “Video Probe and Commit Controls” in the *USB Device Class Definition for Video Devices* specification.

*PTS: Presentation Time Stamp*

This bit, when set, indicates the presence of a PTS field.



*SCR: Source Clock Reference*

This bit, when set, indicates the presence of a SCR field.

*RES: Reserved.*

This bit shall be set to zero.

*STI: Still Image*

This bit shall be set to zero.

*ERR: Error Bit*

This bit, when set, indicates an error in the streaming device.

*EOH: End of Header*

This bit shall be set to 1.

**Presentation time stamp (PTS) field**

The PTS field is present when the PTS bit is set in the BFH[0] field. See Section 2.4.3.3 “Video and Still Image Payload Headers” in the “USB Device Class Definition for Video Devices” specification.

**Source clock reference (SCR) field**

The SCR field is present when the SCR bit is set in the BFH[0] field. See Section 2.4.3.3 “Video and Still Image Payload Headers” in the “USB Device Class Definition for Video Devices” specification.

## **2.2 Payload Data**

Payload Data is either *Packet-Oriented* or *Byte-Oriented*. When **dwPacketLength** (see Table 3-1) is non-zero, the stream is Packet-Oriented and shall be interpreted as a sequence of format-specific packets. When **dwPacketLength** is zero, the stream is Byte-Oriented and shall be interpreted as a sequence of bytes, without regard to format-specific boundaries (if any) within the stream. A stream consisting of variable length packets is considered to be a Byte-Oriented stream. Out-of-band boundary indicators can be applied to both Byte-Oriented and Packet-Oriented streams via the FID and EOF bits of the Payload Header (see section 2.1, “Payload Header” of this specification for additional details).

### **2.2.1 Packet-Oriented Payload Data**

Packet-Oriented streams have additional constraints, governed by the relationship between the value of **dwPacketLength** and the size of the Payload Data portion of a Payload Transfer.

When **dwPacketLength** is greater than the Payload Data size, the start of a format-specific packet must coincide with a new payload transfer (i.e. the first byte of a format-specific packet shall be the first byte after the Payload Header in a Payload Transfer). The final byte of a format-specific packet may fall anywhere within the Payload Data.

When **dwPacketLength** is less than or equal to the Payload Data size, the Payload Data portion of a Payload Transfer shall contain an integral number of format-specific packets. If, due to timing considerations, no format-specific packets are available over a period of one or more (micro)frames, those (micro)frames shall be empty.

### 2.2.2 Byte-Oriented Payload Data

Byte-Oriented streams do not have any Payload Data boundary constraints. Payload Transfers can be of variable length within the limit set by **dwMaxPayloadTransferSize** of the Video Probe and Commit Control. A sequence of one or more variable length packets within a Byte-Oriented stream can be indicated via the FID and EOF bits of the Payload Header. For additional details, see section 4.3.1.1 of the *USB Device Class Definition for Video Devices* specification.

## 3 Payload-Specific Information

### 3.1 Descriptors

#### 3.1.1 Stream Based Format Descriptor

The Stream Based Format Descriptor defines the characteristics of the Stream Based payload. A USB IN or OUT endpoint, and the interface it belongs to, supports one or more format definitions.

Stream Based Format Descriptor is defined in Table 3-1.

**Table 3-1 Stream Based Format Descriptor**

Offset	Field	Size	Value	Description
0	<b>bLength</b>	1	Number	Size of this Descriptor, in bytes: 24
1	<b>bDescriptorType</b>	1	Constant	CS_INTERFACE Descriptor type
2	<b>bDescriptorSubtype</b>	1	Constant	VS_FORMAT_STREAM_BASED Descriptor subtype
3	<b>bFormatIndex</b>	1	Number	Index of this Format Descriptor
4	<b>guidFormat</b>	16	GUID	Globally Unique Identifier used to identify stream-encoding format
20	<b>dwPacketLength</b>	4	Number	If non-zero, indicates a format-specific packet size in a Packet-Oriented stream. If zero, indicates that the format-specific data is either Byte-Oriented, or consists of variable size format-specific packets. See section 2.2, “Payload Data” of this specification for more details.

### **3.2 Video Samples**

Since this specification addresses Stream Based Format over USB Video Class, Video sample information, such as aspect ratio, picture position, and so on, are not included in this Stream Based format specification. This information is encapsulated within the format-specific stream.

For example, this information is described as a profile in ISO/IEC 11172 for MPEG-1 SS streams.