Universal Serial Bus Device Class Definition for Video Devices: Uncompressed Payload

Revision 1.1

June 1, 2005

Contributors

Abdul R. Ismail	Intel Corp.
Akihiro Tanabe	Canon Inc.
Anand Ganesh	Microsoft Corp.
Andy Hodgson	STMicroelectronics
Anshuman Saxena	Texas Instruments
Bertrand Lee	Microsoft Corp.
Charng Lee	Sunplus Technology Co., Ltd
David Goll	Microsoft Corp.
Eric Luttmann	Cypress Semiconductor Corp.
Fernando Urbina	Apple Computer Inc.
Geert Knapen	Philips Electronics
Geraud Mudry	Logitech Inc.
Hiro Kobayashi	Microsoft Corp.
Jean-Michel Chardon	Logitech Inc.
Jeff Zhu	Microsoft Corp.
Ken-ichiro Ayaki	Fujifilm
Mitsuo Niida	Canon Inc.
Nobuo Kuchiki	Sanyo Electric Co., Ltd
Olivier Lechenne	Logitech Inc.
Paul Thacker	STMicroelectronics
Remy Zimmermann	Logitech Inc.
Shinichi Hatae	Canon Inc.
Steve Miller	STMicroelectronics
Tachio Ono	Canon Inc.
Takashi Sato	Philips Electronics
Yoichi Hirata	Matsushita Electric Industrial Co., Ltd

INTELLECTUAL PROPERTY DISCLAIMER

THIS SPECIFICATION IS PROVIDED "AS IS" WITH NO WARRANTIES WHATSOEVER INCLUDING ANY WARRANTY OF MERCHANTABILITY, FITNESS FOR ANY PARTICULAR PURPOSE, OR ANY WARRANTY OTHERWISE ARISING OUT OF ANY PROPOSAL, SPECIFICATION, OR SAMPLE.

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

AUTHORS OF THIS SPECIFICATION DISCLAIM ALL LIABILITY, INCLUDING LIABILITY FOR INFRINGEMENT OF PROPRIETARY RIGHTS, RELATING TO IMPLEMENTATION OF INFORMATION IN THIS SPECIFICATION. AUTHORS OF THIS SPECIFICATION ALSO DO NOT WARRANT OR REPRESENT THAT SUCH IMPLEMENTATION(S) WILL NOT INFRINGE SUCH RIGHTS.

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

Revision History

Version	Date	Description
1.0	September 4, 2003	Initial release
1.1	June 1st, 2005	Removed restrictions on use of Aspect Ratio fields Updated Table 3-1 to remove the Display Mode bits from the bmInterlaceFlags field Flag added to distinguish between fixed and dynamic frame rate devices (RR0043) Corrected the Frame Descriptor length Update SCR/PTS field. Deprecated the field **dwMaxVideoFrameBufferSize** in Table 3-2. (RR0064)

Table of Contents

List of Tables

List of Figures

1 Introduction

1.1 Purpose

This document defines uncompressed payload formats for devices that are compliant with the *USB Device Class Definition for Video Devices* document.

1.2 Scope

The payload format and associated header information are fully specified in this document. This includes:

- USB Video Class stream header
- List of supported formats
- List of format identifiers

1.3 Related Documents

USB Specification Revision 2.0, April 27, 2000, www.usb.org
USB Device Class Definition for Video Devices www.usb.org
http://www.fourcc.org The [almost definitive] FourCC definition list

1.4 Terms and Abbreviations

This section defines terms and abbreviations used throughout this document.

Term	Description
Macropixel	Block consists of one or more packed pixel(s) that contain all the component information.
Packed	Pixel components are transmitted in interleaved order.
Planar	Pixel components are transmitted in separate successive planes.
YUV	Luminance and chrominance components

2 Video Class-Specific Information

2.1 Compression Class

This specification defines uncompressed streams in YUV color spaces. Each frame is independently sent by the device to the host.

2.2 Compression Formats

The following compression formats are supported.

Table 2-1 Compression Formats

Format	GUID
YUY2	32595559-0000-0010-8000-00AA00389B71
NV12	3231564E-0000-0010-8000-00AA00389B71

2.3 Format Constraints

The vertical and horizontal dimensions of the image are constrained by the color components subsampling; image size must be a multiple of macropixel block size. No padding is allowed. The USB Video Class supports one packed 4:2:2 YUV format (YUY2) and one planar 4:2:0 YUV format (NV12).

2.4 Stream Header

The following is a description of the packet header format for uncompressed streams.

Table 2-2 Stream Header Format for Uncompressed Streams

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]							

Header length field

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

Bit field header field

FID: Frame Identifier

This bit toggles at each frame start boundary and stays constant for the rest of the frame.

EOF: End of Frame
This bit indicates the end of a video frame and is set in the last video sample belonging to a frame.
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.
Set to 0.
STI: Still Image
This bit, when set, identifies a video sample as belonging to a still image.
ERR: Error Bit
This bit, when set, indicates an error in the device streaming.
EOH: End of Header
This bit, when set, indicates the end of the BFH fields.

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.

3 Payload-Specific Information

The Color Matching descriptor is mandatory for uncompressed video formats. For detailed information, see section "Color Matching Descriptor" in *Universal Serial Bus Device Class Definition for Video Devices* documentation.

3.1 Descriptors

This section provides detailed information about the following descriptors:

- Uncompressed Video Format Descriptor
- Uncompressed Frame Descriptor

3.1.1 Uncompressed Video Format Descriptor

The Uncompressed Video Format descriptor defines the characteristics of a specific video stream. It is used for formats that carry uncompressed video information, including all YUV variants.

A Terminal corresponding to a USB IN or OUT endpoint, and the interface it belongs to, supports one or more format definitions. To select a particular format, host software sends control requests to the corresponding interface.

The **bFormatIndex** field contains the one-based index of this format descriptor, and is used by requests from the host to set and get the current video format.

The **guidFormat** field uniquely identifies the video data format that shall be used when communicating with this interface at the corresponding format index. For a video source function, the host software will deploy the corresponding video format decoder (if necessary) based on the format specified in this field.

The **bAspectRatioX** and **bAspectRatioY** fields specify the X and Y dimensions of the picture aspect ratio respectively for video field (interlaced) data. For example, **bAspectRatioX** will be 16 and **bAspectRatioY** will be 9 for a 16:9 display.

An Uncompressed Video Format Descriptor is followed by one or more Uncompressed Video Frame Descriptor(s); each Video Frame Descriptor conveys information specific to a frame size supported for the format.

An Uncompressed Video format descriptor identifies the following.

Table 3-1 Uncompressed Video Format Descriptor

Offset	Field	Size	Value	Description
0	**bLength**	1	Number	Size of this descriptor in bytes: 27
1	**bDescriptorType**	1	Constant	CS_INTERFACE descriptor type
2	**bDescriptorSubtype**	1	Constant	VS_FORMAT_UNCOMPRESSED descriptor subtype
3	**bFormatIndex**	1	Number	Index of this format descriptor

4	**bNumFrameDescriptors**	1	Number	Number of frame descriptors following that correspond to this format
5	**guidFormat**	16	GUID	Globally Unique Identifier used to identify stream-encoding format
21	**bBitsPerPixel**	1	Number	Number of bits per pixel used to specify color in the decoded video frame
22	**bDefaultFrameIndex**	1	Number	Optimum Frame Index (used to select resolution) for this stream
23	**bAspectRatioX**	1	Number	The X dimension of the picture aspect ratio.
24	**bAspectRatioY**	1	Number	The Y dimension of the picture aspect ratio.
25	**bmInterlaceFlags**	1	Bitmap	Specifies interlace information. If the scanning mode control in the Camera Terminal is supported for this stream, this field shall reflect the field format used in interlaced mode. (Top field in PAL is field 1, top field in NTSC is field 2.): D0: Interlaced stream or variable. 1 = Yes D1: Fields per frame. 0= 2 fields, 1 = 1 field D2: Field 1 first. 1 = Yes D3: Reserved D5..4: Field pattern 00 = Field 1 only 01 = Field 2 only 10 = Regular pattern of fields 1 and 2 11 = Random pattern of fields 1 and 2 D7..6: Reserved. Do not use.
26	**bCopyProtect**	1	Boolean	Specifies whether duplication of the video stream is restricted: 0: No restrictions 1: Restrict duplication

3.1.2 Uncompressed Frame Descriptor

Uncompressed Video Frame descriptors (or Frame descriptors for short) are used to describe the decoded video and still-image frame dimensions and other frame-specific characteristics supported by a particular stream. One or more Frame descriptors follow the Uncompressed

Video Format descriptor they correspond to. The Frame descriptor is also used to determine the range of frame intervals supported for the frame size specified.

The Uncompressed Video Frame descriptor is used only for video formats for which the Uncompressed Video Format descriptor applies (see section 3.1.1, "Uncompressed Video Format Descriptor").

The **bFrameIndex** field contains the one-based index of this frame descriptor, and is used by requests from the host to set and get the current frame index for the format in use. This index is one-based for each corresponding format descriptor supported by the device.

The range of frame intervals supported can be either a continuous range or a discrete set of values. For a continuous range, **dwMinFrameInterva**l, **dwMaxFrameInterval** and **dwFrameIntervalStep** indicate the limits and granularity of the range. For discrete values, the **dwFrameInterval(x)** fields indicate the range of frame intervals (and therefore frame rates) supported at this frame size. The frame interval is the average display time of a single decoded video frame in 100ns units.

A Frame descriptor identifies the following.

Table 3-2 Uncompressed Video Frame Descriptors

Offset	Field	Size	Value	Description
0	**bLength**	1	Number	Size of this descriptor in bytes when bFrameIntervalType is 0: 38 Size of this descriptor in bytes when bFrameIntervalType > 0: 26+(4*n)
1	**bDescriptorType**	1	Constant	CS_INTERFACE descriptor type
2	**bDescriptorSubtype**	1	Constant	VS_FRAME_UNCOMPRESSED descriptor subtype
3	**bFrameIndex**	1	Number	Index of this frame descriptor
4	**bmCapabilities**	1	Number	D0: Still image supported Specifies whether still images are supported at this frame setting. This is only applicable for VS interfaces with an IN video endpoint using Still Image Capture Method 1, and should be set to 0 in all other cases. D1: Fixed frame-rate Specifies whether the device provides a fixed frame rate on a stream associated with this frame descriptor.

				Set to 1 if fixed rate is enabled; otherwise, set to 0. D7..2: Reserved, set to 0.
5	**wWidth**	2	Number	Width of decoded bitmap frame in pixels
7	**wHeight**	2	Number	Height of decoded bitmap frame in pixels
9	**dwMinBitRate**	4	Number	Specifies the minimum bit rate at the longest frame interval in units of bps at which the data can be transmitted.
13	**dwMaxBitRate**	4	Number	Specifies the maximum bit rate at the shortest frame interval in units of bps at which the data can be transmitted.
17	**dwMaxVideoFrameBufferSize**	4	Number	Use of this field has been deprecated. Specifies the maximum number of bytes that the compressor will produce for a video frame or still image. The **dwMaxVideoFrameSize** field of the Video Probe and Commit control replaces this descriptor field. A value for this field shall be chosen for compatibility with host software that implements an earlier version of this specification.
21	**dwDefaultFrameInterval**	4	Number	Specifies the frame interval the device would like to indicate for use as a default. This must be a valid frame interval described in the fields below.
25	**bFrameIntervalType**	1	Number	Indicates how the frame interval can be programmed: 0: Continuous frame interval 1..255: The number of discrete frame intervals supported (n)
26…				See the following frame interval tables.

Table 3-3 Continuous Frame Intervals

Offset	Field	Size	Value	Description
26	**dwMinFrameInterval**	4	Number	Shortest frame interval supported (at highest frame rate), in 100 ns units.

30	**dwMaxFrameInterval**	4	Number	Longest frame interval supported (at lowest frame rate), in 100 ns units.
34	**dwFrameIntervalStep**	4	Number	Indicates granularity of frame interval range, in 100 ns units.

Table 3-4 Discrete Frame Intervals

Offset	Field	Size	Value	Description
26	**dwFrameInterval**(1)	4	Number	Shortest frame interval supported (at highest frame rate), in 100 ns units.
…	…	…	…	…
26+(4*n)-4	**dwFrameInterval**(n)	4	Number	Longest frame interval supported (at lowest frame rate), in 100 ns units.

3.2 Video Samples

Each Uncompressed frame is considered a single video sample. A video sample is made up of one or more *payload transfers* (as defined in the USB Device Class Specification for Video Devices).

For an isochronous pipe, each (micro) frame will contain a single payload transfer. Each payload transfer will consist of a payload header immediately followed by payload data in one or more data transactions (up to 3 data transactions for high speed high bandwidth endpoints).

For a bulk pipe, the first bulk data packet of each payload transfer shall contain a payload header at the beginning of the packet, followed by payload data, extending through additional bulk data transactions as needed.

3.3 Uncompressed Payload Information

The following paragraphs describe constraints on the payload transfers.

3.3.1 Planar Formats

Transfers of planar payloads have no data alignment restrictions.

3.3.2 Packed Formats

Transfers of a packed payload format must be aligned on macro-pixel boundaries.

4 Examples

4.1 Isochronous Transfer IN

The following example shows the relationship between Video Samples, Payload Transfers and the token and data packets when receiving isochronous transfers from the device. This example shows high-speed, high-bandwidth transfers, but this is only illustrative and not a requirement of the Uncompressed payload format. The actual video sample size and bandwidth usage will vary according to the requirements of the device.

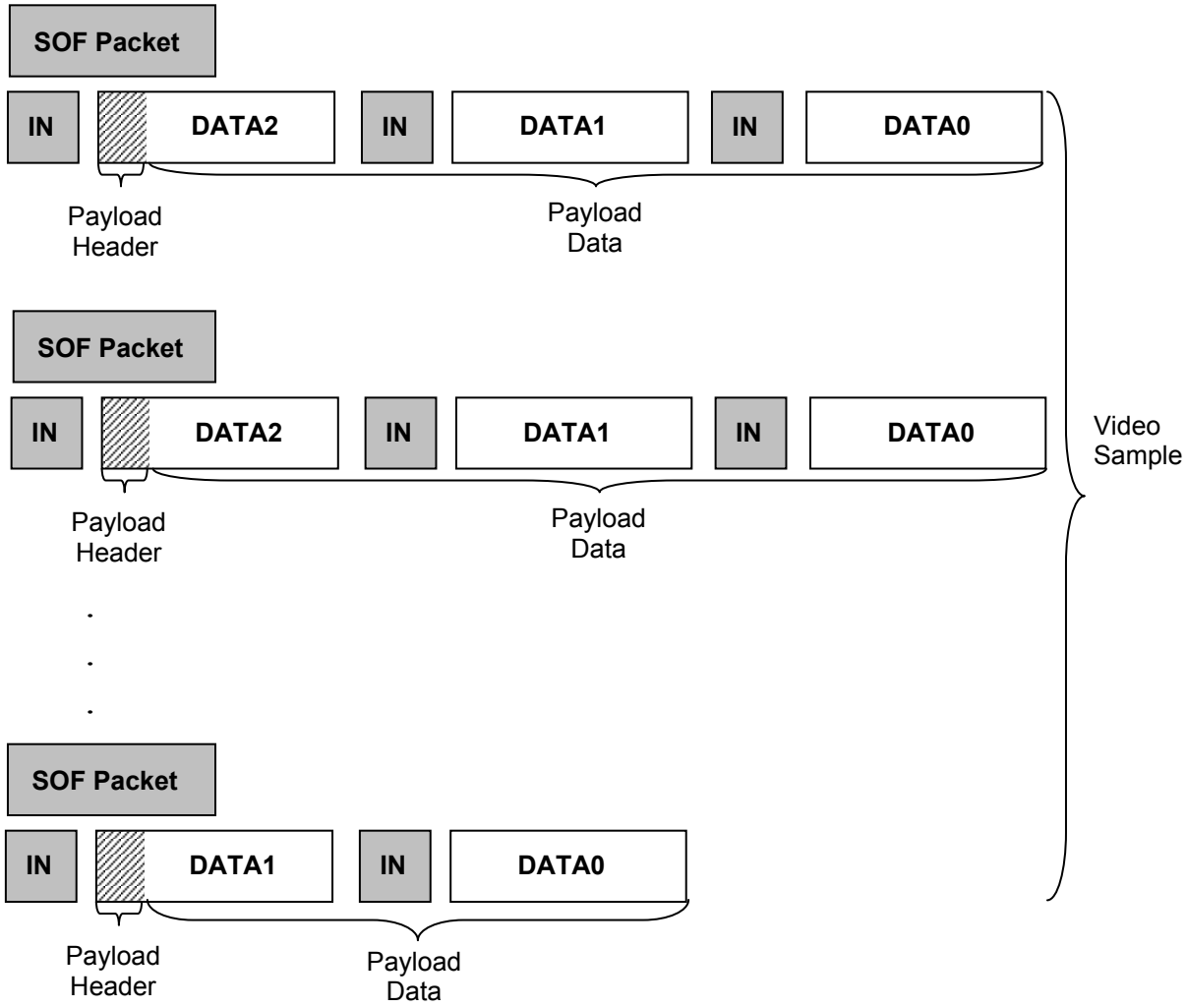

Figure 4-1 Example Uncompressed Isochronous Transfer, IN Endpoint

4.2 Isochronous Transfer OUT

The following example shows the relationship between Video Samples, Payload Transfers and the token and data packets when sending isochronous transfers to the device. This example shows high-speed, high-bandwidth transfers, but this is only illustrative and not a requirement of the Uncompressed payload format. The actual video sample size and bandwidth usage will vary according to the requirements of the device.

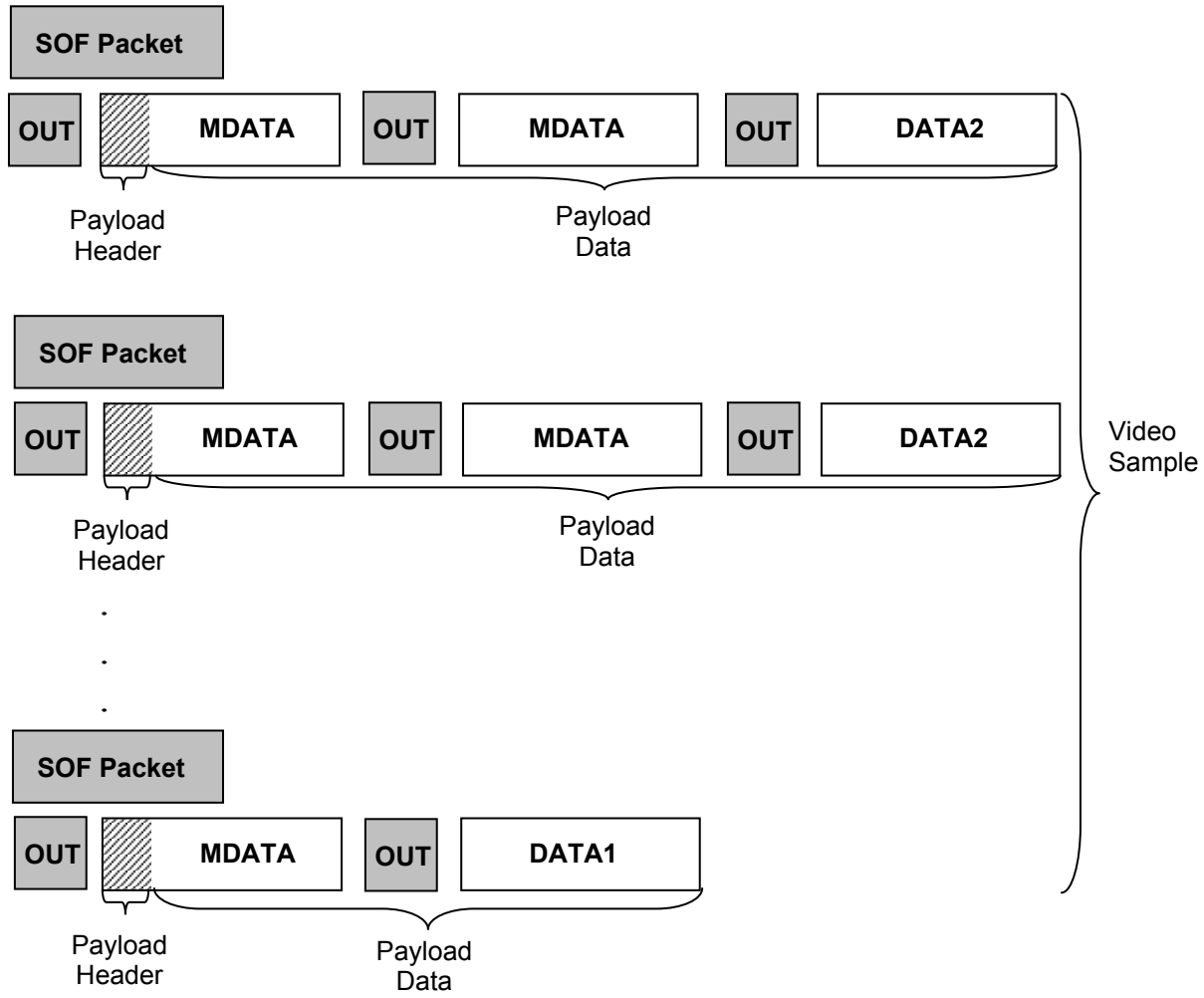

Figure 4-2 Example Uncompressed Isochronous Transfer, OUT Endpoint

4.3 Bulk Transfer IN

The following example shows the relationship between Video Samples, Payload Transfers and the token and data packets when receiving bulk transfers from a device. Handshake packets are not shown for the sake of clarity.

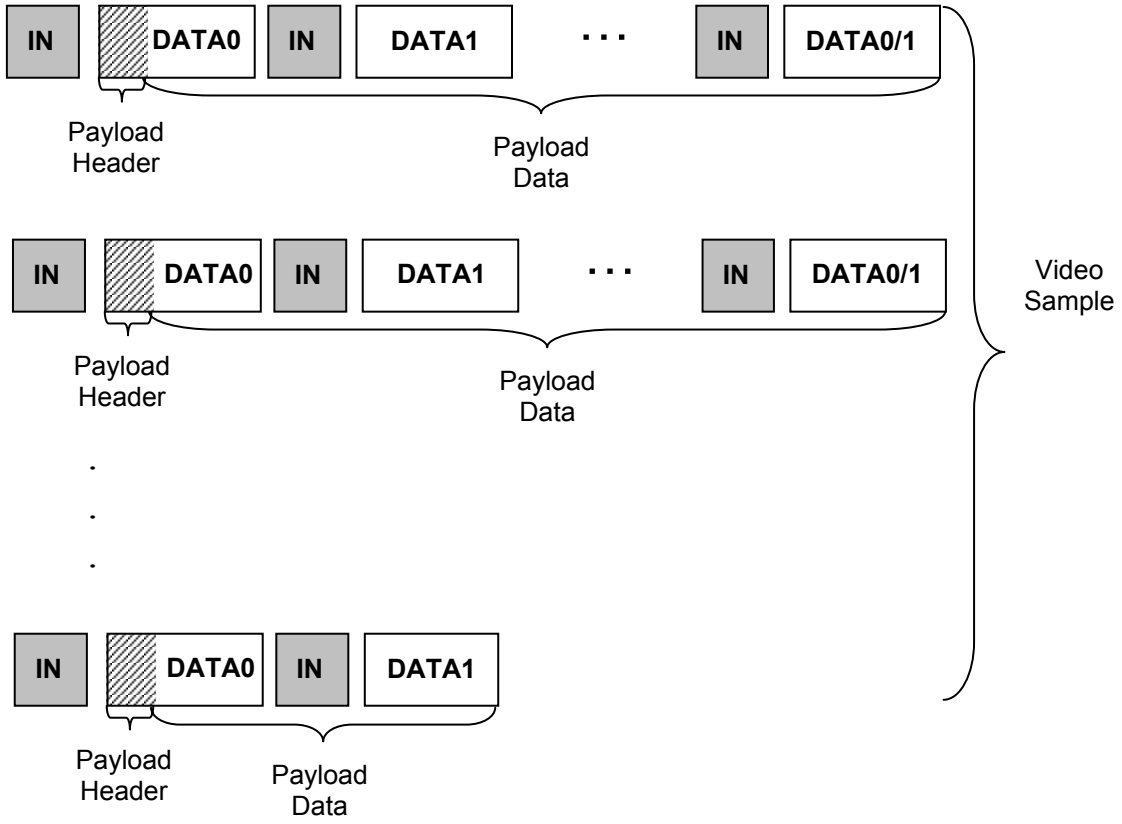

Figure 4-3 Example Uncompressed Bulk Transfer, IN Endpoint

4.4 Bulk Transfer OUT

The following example shows the relationship between Video Samples, Payload Transfers and the token and data packets when sending bulk transfers to the device. Handshake packets are not shown for the sake of clarity.

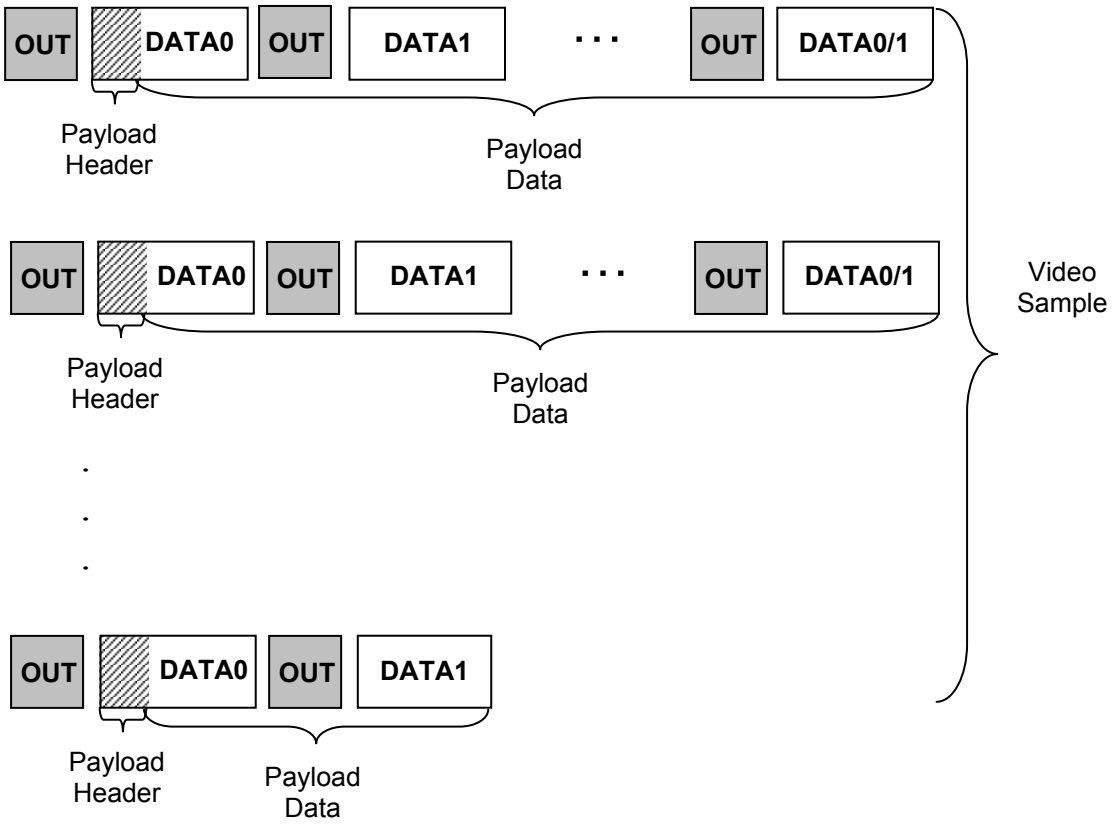

Figure 4-4 Example Uncompressed Bulk Transfer, OUT Endpoint