

Universal Serial Bus Device Class Definition for Video Devices: Motion-JPEG Payload

Revision 1.5

July 25, 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. |

**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

Revision History

| Version | Date | Description |
|---------|-----------------------------|--|
| 1.0 | September 4, 2003 | Initial release |
| 1.1 | June 1 st , 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 (RR0045) Removed Terms and Abbreviations section. Update of SCR/PTS fields. Deprecation of field dwMaxVideoFrameBufferSize in Table 3-2. (RR0064) |
| 1.5 | July 19, 2012 | No Changes |

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 | Compression Class | 2 |
| 2.2 | Stream Header | 2 |
| 3 | Payload-Specific Information..... | 4 |
| 3.1 | Descriptors | 4 |
| 3.1.1 | MJPEG Video Format Descriptor | 4 |
| 3.1.2 | MJPEG Video Frame Descriptors | 6 |
| 3.2 | Video Samples..... | 8 |
| 3.3 | MJPEG Payload Information | 8 |
| 4 | Examples | 10 |
| 4.1 | Isochronous Transfer IN..... | 10 |
| 4.2 | Isochronous Transfer OUT..... | 11 |
| 4.3 | Bulk Transfer IN | 12 |
| 4.4 | Bulk Transfer OUT | 13 |

List of Tables

| | |
|--|---|
| Table 2-1 Stream Header Format for the Motion-JPEG | 2 |
| Table 3-1 Motion-JPEG Video Format Descriptor | 5 |
| Table 3-2 Motion-JPEG Video Frame Descriptor | 6 |
| Table 3-3 Continuous Frame Intervals | 8 |
| Table 3-4 Discrete Frame Intervals | 8 |

List of Figures

| | |
|---|----|
| Figure 4-1 Example MJPEG Isochronous Transfer, IN Endpoint | 10 |
| Figure 4-2 Example MJPEG Isochronous Transfer, OUT Endpoint | 11 |
| Figure 4-3 Example MJPEG Bulk Transfer, IN Endpoint | 12 |
| Figure 4-4 Example MJPEG Bulk Transfer, OUT Endpoint | 13 |

1 Introduction

1.1 Purpose

This document defines the Motion-JPEG payload format 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
- Payload-specific header

1.3 Related Documents

USB Specification Revision 3.0, November 12, 2008, www.usb.org

USB Specification Revision 2.0, April 27, 2000, www.usb.org

USB Device Class Definition for Video Devices, www.usb.org

ISO/IEC 10918-1 / ITU-T Recommendation T.81 information technology – Digital compression and coding of continuous-tone still images - Requirements and guide-lines.

2 Video Class Specific Information

2.1 Compression Class

The Joint Photographic Experts Group (JPEG) standard defines a family of compression algorithms for continuous-tone, still images. This still image compression standard can be applied to video by compressing each frame of video as an independent still image and then transmitting them in series. Video that has been coded in this fashion is defined as a Motion JPEG.

2.2 Stream Header

The following is description of header for the Motion-JPEG format.

Table 2-1 Stream Header Format for the Motion-JPEG

| | | | | | | | | |
|---------|---------------|-----|-----|-----|-----|-----|-----|-----|
| 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 that belongs 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 that belongs 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 MJPEG format payloads. 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:

- MJPEG Video Format Descriptor
- MJPEG Frame Descriptor

3.1.1 MJPEG Video Format Descriptor

The MJPEG Video Format Descriptor defines the characteristics of a specific video stream. It is used for formats that carry MJPEG video information, including all YUV/RGB 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 **bDescriptorSubtype** field uniquely identifies the video data format that should 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 respectively of the picture aspect ratio for video field (interlaced) data. For example, **bAspectRatioX** will be 16 and **bAspectRatioY** will be 9 for a 16:9 display.

The **bmFlags** field holds information about the video data stream characteristics. **FixedSizeSamples** indicates whether all video samples are the same size.

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

A MJPEG Video Format Descriptor identifies the following.

Table 3-1 Motion-JPEG Video Format Descriptor

| Offset | Field | Size | Value | Description |
|--------|-----------------------------|------|----------|--|
| 0 | bLength | 1 | Number | Size of this Descriptor, in bytes: 11 |
| 1 | bDescriptorType | 1 | Constant | CS_INTERFACE Descriptor type. |
| 2 | bDescriptorSubtype | 1 | Constant | VS_FORMAT_MJPEG 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 | bmFlags | 1 | Number | Specifies characteristics of this format D0: FixedSizeSamples. 1 = Yes All other bits are reserved for future use and shall be reset to zero. |
| 6 | bDefaultFrameIndex | 1 | Number | Optimum Frame Index (used to select resolution) for this stream |
| 7 | bAspectRatioX | 1 | Number | The X dimension of the picture aspect ratio. |
| 8 | bAspectRatioY | 1 | Number | The Y dimension of the picture aspect ratio. |
| 9 | bmInterlaceFlags | 1 | Bitmap | Specifies interlace information. If the scanning mode control in the Camera Terminal is supported for this stream, this field should 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. |

| | | | | |
|----|---------------------|---|---------|---|
| 10 | bCopyProtect | 1 | Boolean | Specifies if duplication of the video stream should be restricted: 0: No restrictions 1: Restrict duplication |
|----|---------------------|---|---------|---|

3.1.2 MJPEG Video Frame Descriptors

MJPEG Video Frame Descriptors (or simply Frame Descriptors) 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 MJPEG Video Format Descriptor they correspond to. The Frame Descriptor is also used to determine the range of frame intervals that are supported for the specified frame size.

The MJPEG Video Frame Descriptor is used only for video formats for which the MJPEG Video Format Descriptor applies (see section 3.1.1, "MJPEG 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, **dwMinFrameInterval**, **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 Motion-JPEG Video Frame Descriptor

| 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_MJPEG 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 |

| | | | | |
|----|----------------------------------|---|--------|--|
| | | | | <p>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.</p> <p>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.</p> <p>D7..2: Reserved, set to 0.</p> |
| 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 default compression quality and longest frame interval in Units of bps at which the data can be transmitted. |
| 13 | dwMaxBitRate | 4 | Number | Specifies the maximum bit rate at default compression quality and shortest frame interval in Units of bps at which the data can be transmitted. |
| 17 | dwMaxVideoFrameBufferSize | 4 | Number | <p>Use of this field has been deprecated.</p> <p>Specifies the maximum number of bytes for a video (or still image) frame the compressor will produce.</p> <p>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.</p> |
| 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 100ns units. |
| 30 | dwMaxFrameInterval | 4 | Number | Longest frame interval supported (at lowest frame rate), in 100ns units. |
| 34 | dwFrameIntervalStep | 4 | Number | Indicates granularity of frame interval range, in 100ns 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 100ns units. |
| ... | ... | ... | ... | ... |
| 26+(4*n)-4 | dwFrameInterval(n) | 4 | Number | Longest frame interval supported (at lowest frame rate), in 100ns units. |

3.2 Video Samples

Each MJPEG 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 MJPEG Payload Information

Each frame of MJPEG payload is coded by JPEG compression and preceded with a header containing required and optional definitions for compression parameters such as quantization tables and Huffman coding tables. The required and optional parameters are identified with "markers" and comprise a marker segment.

The structure of each frame is as follows.

- **SOI** (Start of Image, 0xFFD8) – **required**
- **APPn** (Application Marker, 0xFFEn) – **optional**, unless interlaced video is used, in which case APP0 segment with "AVI1" marker and Field ID info is required.
- **DRI** (Define Restart Interval, 0xFFDD) – **optional**
- **DQT** (Define Quantization Table, 0xFFDB) – **required**
- **DHT** (Define Huffman Table, 0xFFC4) – **optional**, if not specified, standard tables as specified in JPEG standard (ISO 10918-1) section K.3.3 are used.
- **SOF0** (Start of Frame, 0xFFC0)- **required**. All other SOFn segments are not supported.
- **SOS** (Start of Scan, 0xFFDA) – **required**
- **Encoded Image Data** – **required**
- **RSTn** (Restart count, 0xFFDn) – **optional**
- **EOI** (End of Image, 0xFFD9) - **required**

The following is required for the image data:

- Color encoding - YCbCr
- Bits per pixel - 8 per color component (before filtering/subsampling)
- Subsampling - 422
- Baseline sequential DCT (SOF0)
- All key frames

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 MJPEG payload format. The actual video sample size and bandwidth usage will vary according to the requirements of the device.

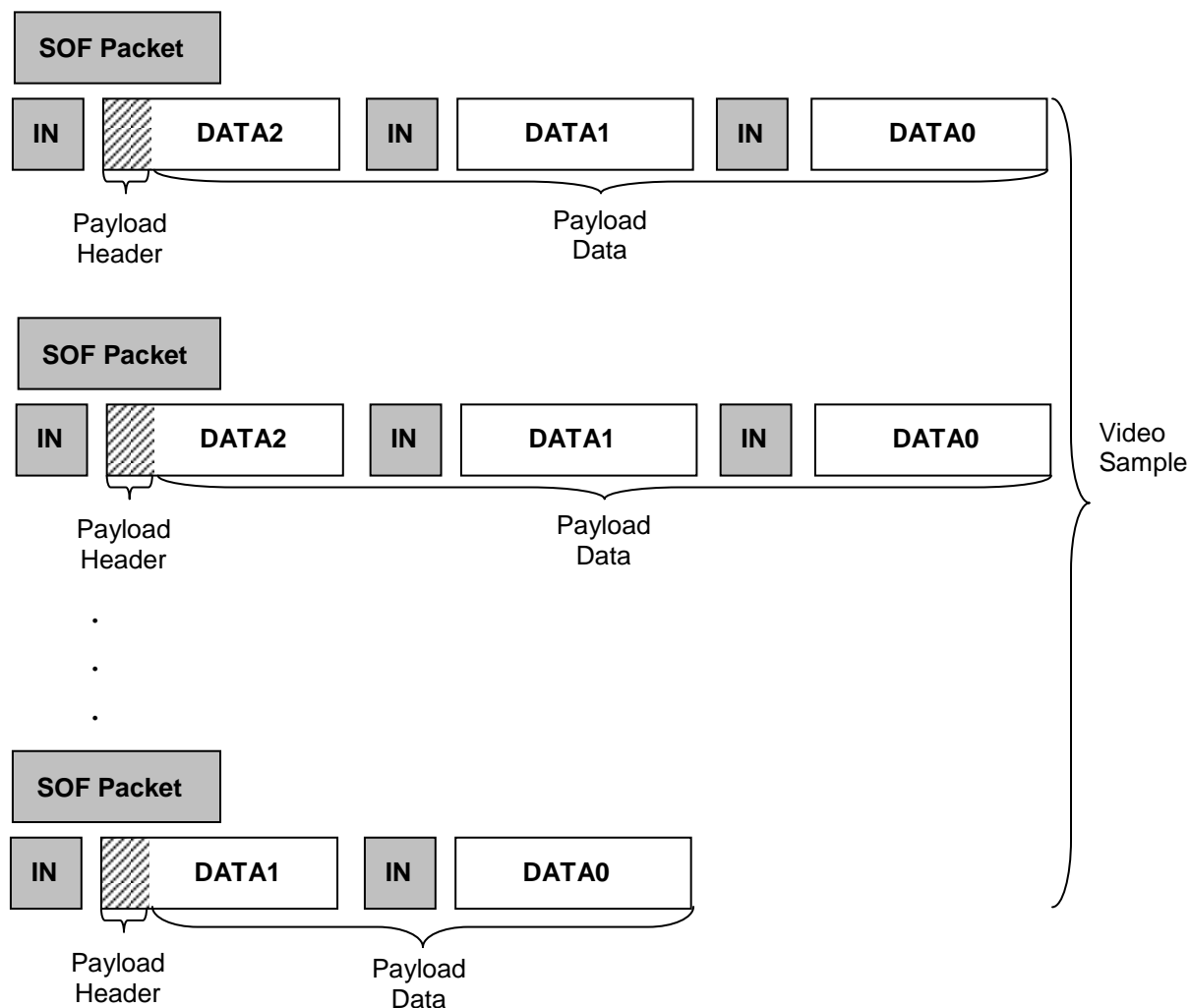


Figure 4-1 Example MJPEG Isochronous Transfer, IN Endpoint

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 MJPEG payload format. The actual video sample size and bandwidth usage will vary according to the requirements of the device.

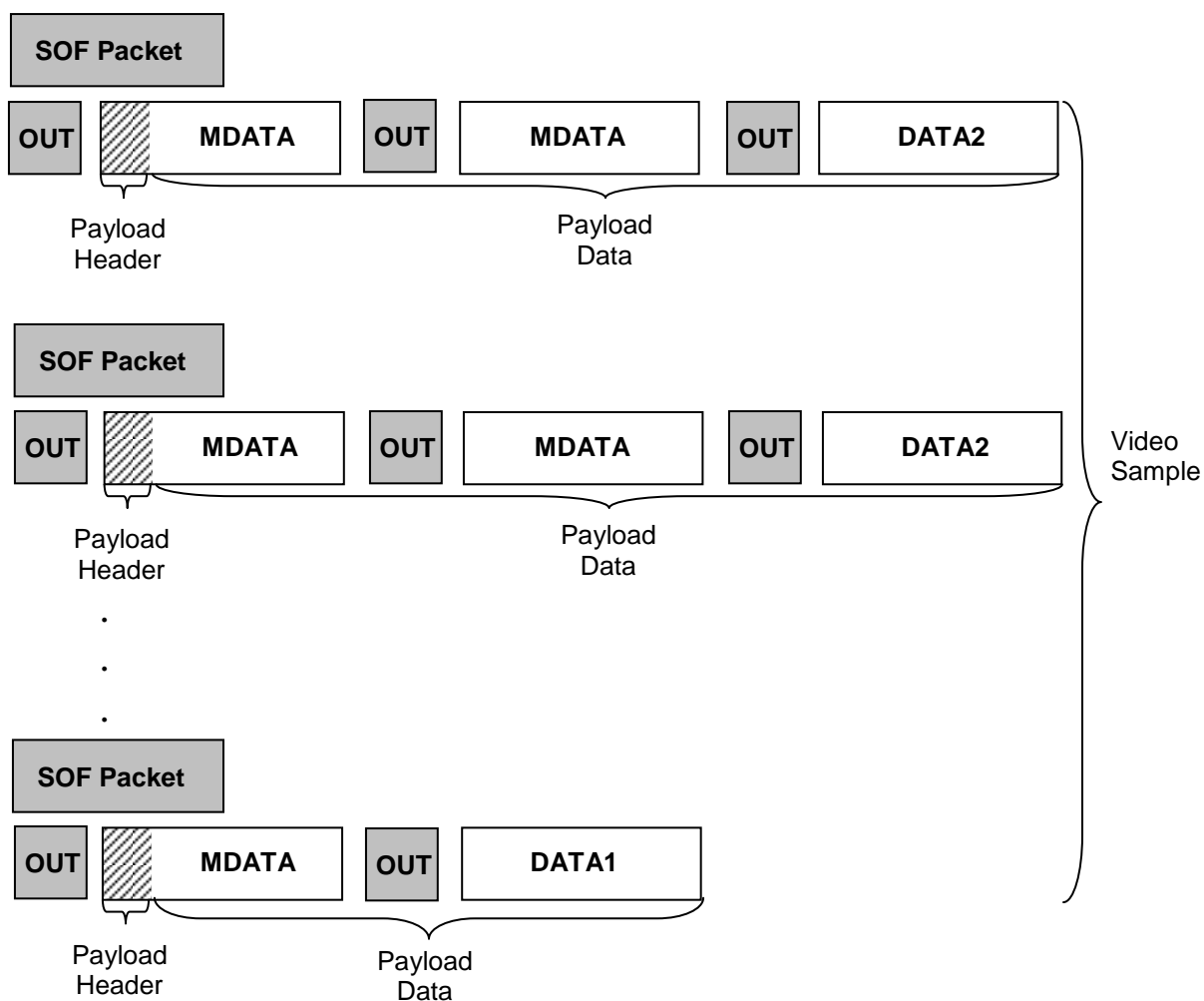


Figure 4-2 Example MJPEG Isochronous Transfer, OUT Endpoint

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.

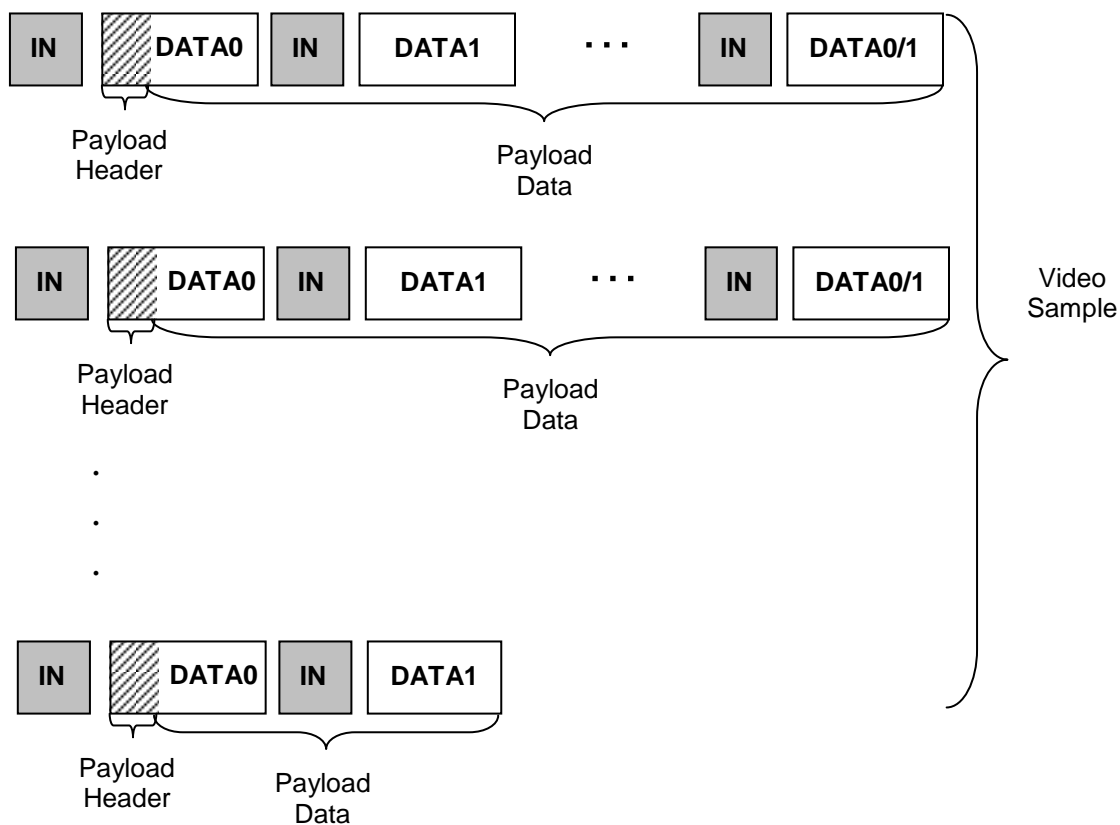


Figure 4-3 Example MJPEG Bulk Transfer, IN Endpoint

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.

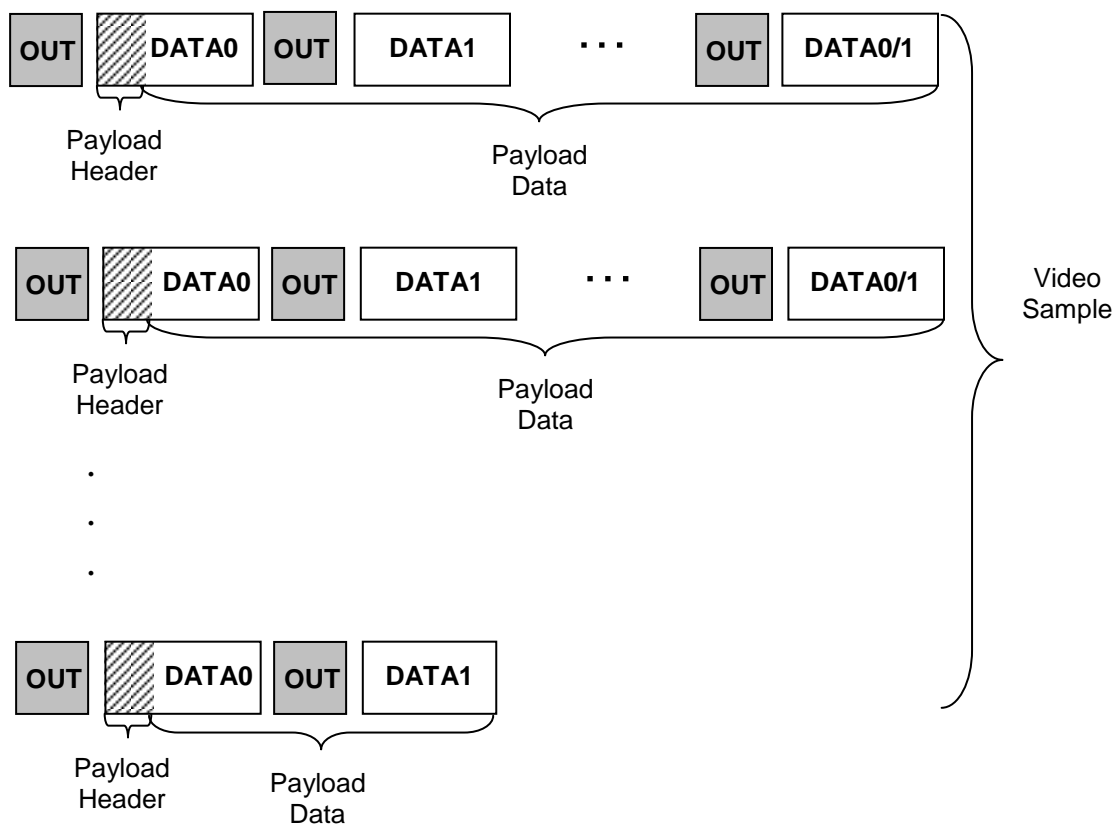


Figure 4-4 Example MJPEG Bulk Transfer, OUT Endpoint