# Universal Serial Bus Device Class Definition for Video Devices: MPEG-2 TS Payload

Revision 1.5

August 9, 2012

## Contributors

| | |
|---|---|
| Abdul R. Ismail | Intel Corp. |
| Allison Hicks | Texas Instruments |
| 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. |

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 1st, 2005 | Address packet jitter. Updates to chapter 2, paragraph 3.1.1 and section 3.2 (RR0024)<br>Latency optimizations for Stream-based formats (RR0041)<br>Address identification of the Stride Data in the MPEG-2 TS Payload (RR0062)<br>Removed "**Terms and Abbreviations**" section |
| 1.5 | July 25, 2012 | No Changes |

# Table of Contents

# List of Tables

# List of Figures

# 1 Introduction

## 1.1 Purpose

This document defines the MPEG-2 TS (Transport Stream) 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 is fully specified by this document. This includes:

- USB Video Class stream header
- Payload-specific header
- Payload format

## 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
*MPEG-2 TS Packets definitions shall comply with ISO/IEC 13818 series*
*ISO/IEC 13818-1:* Information technology -- Generic coding of moving pictures and associated audio information: Part 1: Systems
*ISO/IEC 13818-2:* Information technology - Generic coding of moving pictures and associated audio information: Part 2: Video
*ISO/IEC 13818-3:* Information technology - Generic coding of moving picture and associated audio information: Part 3: Audio
*ISO/IEC 13818-9:* Information technology - Generic coding of moving pictures and associated audio information: Part 9: Extension for real time interface for systems decoders
*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 Compression Class

The ISO/IEC JTC1/SC29 WG11 (also referred to as the MPEG committee) has defined the MPEG-2 standard (ISO/IEC 13818); this document describes a packetization scheme to transport MPEG-2 TS streams over USB.

ISO/IEC standards terminology is used throughout this specification; the reader should consult the original references listed in section 1.3, "Related Documents", for the definitions of these terms.

## 2.2 Stream Header

This section describes the stream header used for the MPEG-2 TS payload format.

USB Video Class header definition for MPEG-2 TS format.

**Table 2-1 Header Definition for MPEG-2 TS Format**

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| HLE | Header Length | | | | | | | |
| BFH [0] | EOH | ERR | STI | RES | SCR | PTS | EOF | FID |

**HLE**

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

The header length field specifies the length of the header, in bytes. This field shall be set to 2.

**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 Universal Serial Bus Device Class Definition for Video Devices specification.

*PTS: Presentation Time Stamp*

This bit shall be set to zero.

*SCR: Source Clock Reference*
This bit shall be set to zero.

*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.

## 2.3 Payload Data

The payload data block consists of one or more MPEG-2 TS packets (TSPs). Each TSP may be accompanied by additional data, depending upon device and application requirements. One such use of this additional data is to carry Application Packet Timing (APT) information. See section 2.4 "Application Packet Timing Information" for a description of APT. Without additional data, the payload data block length shall be an integral multiple of 188 bytes. With additional data, the payload data block length shall be an integral multiple of 188+n bytes, where n is derived from the *stride* information in the MPEG-2 TS Format Descriptor. See section 3.1.1 "MPEG-2 TS Format Descriptor" for a description of the *stride* mechanism.

For isochronous transfer, empty (micro)frames are permitted if, due to timing considerations, payload data is not ready to be transmitted for that (micro)frame. Payload transfers consisting of only a payload header are prohibited. The total payload transfer length (payload header and payload data, combined) must remain within the constraints of the maximum packet size the endpoint is configured to send.

For bulk transfer, the transfer length is limited only by implementation limitations and desired latency characteristics.

## 2.4 Application Packet Timing Information

Some decoder implementations can cope with packet jitter introduced during packet transfer over High-Speed USB endpoints (±125 μs), while others cannot. Also some systems may require a jitter-free stream, independent of its decoder capability. Therefore, this specification defines a method for including Application Packet Timing (APT) information with each MPEG-2 TS packet as it is transferred over a High-Speed (HS) USB endpoint. The APT information supports the removal of packet jitter in the stream. The APT method is not supported over Full-Speed (FS) USB endpoints.

A USB host shall support bi-directional transfer of MPEG-2 TS packets, with or without APT information, at the device's discretion. Devices are not required to support the APT method. For OTG scenarios, if one device requires APT information, both devices must support the APT method in order to interoperate. Devices indicate whether or not to use the APT method via their format descriptors, as defined in section 3.1.1 "MPEG-2 TS Format Descriptor". If a device is able to operate both with and without APT information, it shall provide at least two format descriptors (one for APT, and one for non-APT).

In the APT method, each MPEG-2 TS packet shall be preceded by APT information, defined as follows:

**Table 2-2 Application Packet Timing Definition for MPEG-2 TS Format**

| | | | |
|---|---|---|---|
| APT [0] | Reserved [31:25] | Microframe_count [24:12] | Microframe_offset [11:0] |

*APT[0]*
*Reserved:*
This field is 7 bits long and shall be ignored.

*Microframe_count:*
This field is 13 bits long and counts 125 μs cycles (USB microframes). The value in this field shall wrap around to zero after reaching 7999.

*Microframe_offset:*
This field is 12 bits long and counts 27 MHz clock ticks. The value in this field shall be reset to zero at each microframe boundary or after reaching 3374.

The *Microframe_count* and *Microframe_offset* fields collectively indicate the time when the corresponding packet was delivered from the application to the UVC (USB Video Class) layer at the source.

The APT method assumes microframe phase synchronization between the source and the sink UVC layers. There shall be a fixed offset between the *microframe_count* value at the sink and the one at the source and it shall remain constant during the course of the stream.

The sink UVC layer shall store each MPEG-2 TS packet in a buffer until its associated APT value matches the local (*microframe_count* : *microframe_offset*) value plus an implementation-dependent constant value.

The initialization of the local (*microframe_count* : *microframe_offset*) value and the constant value is implementation-dependent. These values shall be initialized so that buffer overflow and underflow will not occur during the course of the stream.

# 3 Payload-Specific Information

## 3.1 Descriptors

### 3.1.1 MPEG-2 TS Format Descriptor

The MPEG-2 TS Format Descriptor defines the characteristics of a specific MPEG-2 TS stream. A Terminal corresponding to a USB IN or OUT endpoint, and the interface it belongs to, supports one or more format definitions.

MPEG-2 TS Format Descriptors have no accompanying Frame Descriptors. An MPEG-2 TS Format Descriptor includes a mechanism that allows the addition of 0 or more bytes of extra information at the beginning, end, or both beginning and end, of each 188-byte TS packet. This is generically referred to as *stride*. The format of the stride data is application and device dependent, but is identified via a Globally Unique Identifier (GUID). This specification defines one such format, for the carriage of Application Packet Timing (APT). See section 2.4 "Application Packet Timing Information" for a description of APT.

MPEG-2 TS Format Descriptor is defined in Table 3-1.

**Table 3-1 MPEG-2 TS Format Descriptor**

| Offset | Field | Size | Value | Description |
|---|---|---|---|---|
| 0 | **bLength** | 1 | Number | Size of this Descriptor, in bytes: 23 |
| 1 | **bDescriptorType** | 1 | Constant | CS_INTERFACE Descriptor type |
| 2 | **bDescriptorSubtype** | 1 | Constant | VS_FORMAT_MPEG2TS Descriptor subtype |
| 3 | **bFormatIndex** | 1 | Number | Index of this Format Descriptor |
| 4 | **bDataOffset** | 1 | Number | Offset to TSP packet within MPEG-2 TS transport stride, in bytes. |
| 5 | **bPacketLength** | 1 | Number | Length of TSP packet, in bytes (typically 188). |
| 6 | **bStrideLength** | 1 | Number | Length of MPEG-2 TS transport stride. |
| 7 | **guidStrideFormat** | 16 | GUID | A Globally Unique Identifier indicating the format of the stride data (if any). Set to zeros if there is no Stride Data, or if the Stride Data is to be ignored by the application. |

#### 3.1.1.1 MPEG-2 TS Format Descriptor without Stride

If there is no stride data associated with the TS packets, **bDataOffset**, **bPacketLength**, **bStrideLength**, and **guidStrideFormat** shall have values as shown in Table 3-2.

**Table 3-2 MPEG-2 TS Format Descriptor without Stride**

| Field | Value |
|---|---|
| **bDataOffset** | 0 |

| | |
|---|---|
| **bPacketLength** | 188 |
| **bStrideLength** | 188 |
| **guidStrideFormat** | 00000000-0000-0000-0000-000000000000 |

#### 3.1.1.2 MPEG-2 TS Format Descriptor with APT

If the stride data associated with the TS packets is for the carriage of APT information, **bDataOffset**, **bPacketLength**, **bStrideLength**, and **guidStrideFormat** shall have values as shown in Table 3-3.

**Table 3-3 MPEG-2 TS Format Descriptor with APT**

| Field | Value |
|---|---|
| **bDataOffset** | 4 |
| **bPacketLength** | 188 |
| **bStrideLength** | 192 |
| **guidStrideFormat** | AE73111F-B352-4E3E-8B4E-CE827BAAE8EE |

#### 3.1.1.3 MPEG-2 TS Format Descriptor with Application-Specific Stride Data

If the stride data associated with the TS packets is application-specific, **bDataOffset**, **bPacketLength**, **bStrideLength**, and **guidStrideFormat** shall have values as shown in Table 3-4.

**Table 3-4 MPEG-2 TS Format Descriptor with Application-Specific Stride**

| Field | Value |
|---|---|
| **bDataOffset** | Offset to TSP packet within MPEG-2 TS transport stride, in bytes. |
| **bPacketLength** | Length of TSP packet, in bytes (typically 188). |
| **bStrideLength** | Length of MPEG-2 TS transport stride (typically **bDataOffset** + **bPacketLength**, assuming the TS packet data follows the extra data). |
| **guidStrideFormat** | A Globally Unique Identifier indicating the format of the stride data. The value of **guidStrideFormat** is mutually agreed between the application and the device. The GUID should be generated using any of the publicly available GUID generators.<br><br>Note: If stride data is included, but is to be ignored by the application or device, this value shall be: 00000000-0000-0000-0000-000000000000 |

### 3.2 Video Samples

The scope of this specification is based on the ISO/IEC 13818 series. Video sample information, such as aspect ratio, picture position, quantization of audio sampling, number of audio channels, and so on, are described as a profile in the ISO/IEC 13818 series. Therefore, this information is not included in this MPEG-2 TS payload specification.

# 4 Examples

## 4.1 Isochronous Transfer IN

The following example shows the relationship between Payload Transfers, 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 MPEG-2 TS payload format. The actual bandwidth usage will vary according to the requirements of the device. The base transmission rate (one TSP(188Bytes)/payload) in MPEG-2 TS is 12.032 Mbps.

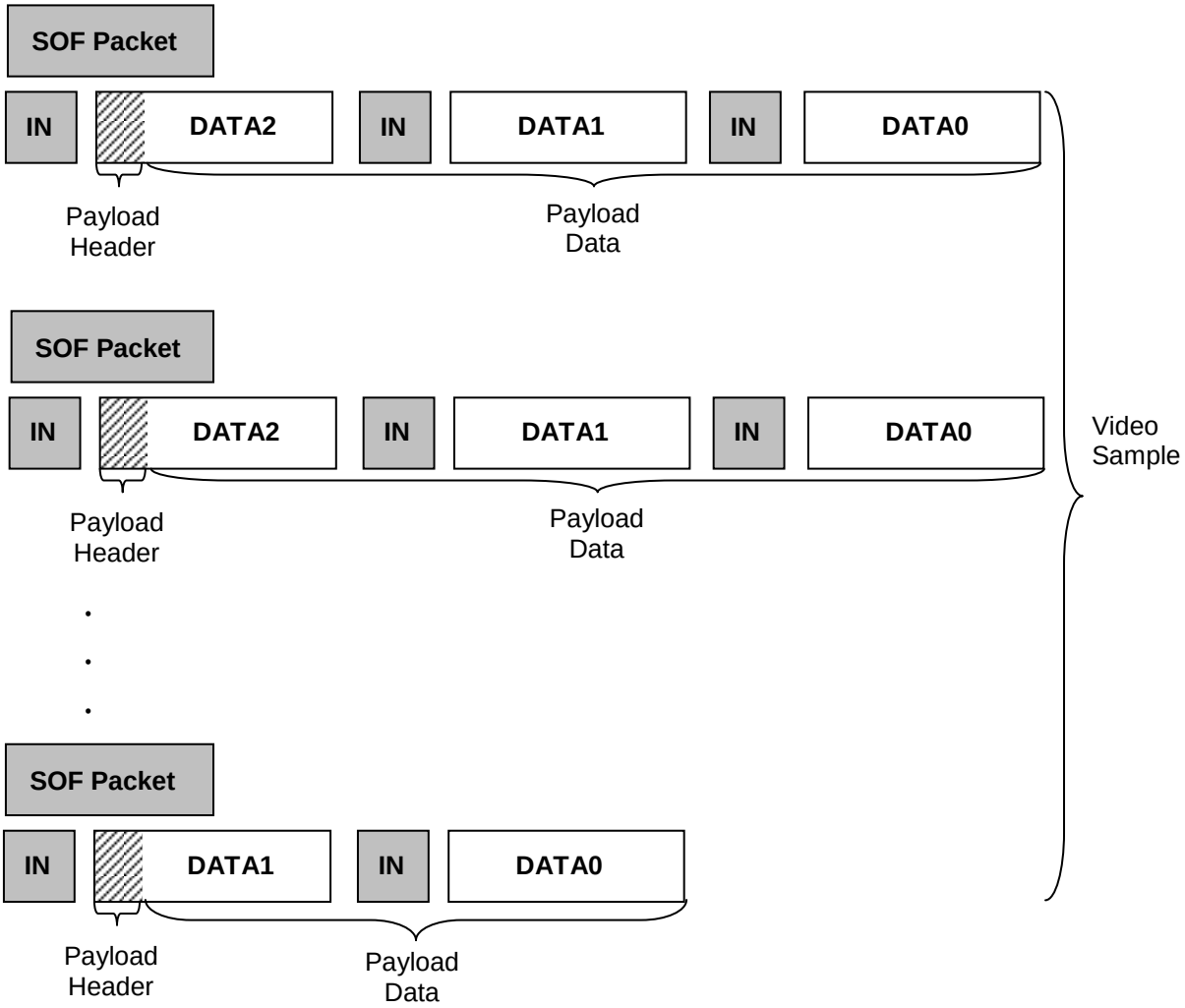


**Figure 4-1 Example MPEG-2 TS Isochronous Transfer, IN Endpoint**

## 4.2 Isochronous Transfer OUT

The following example shows the relationship between Payload Transfers, 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 MPEG-2 TS payload format. The actual bandwidth usage will vary according to the requirements of the device.

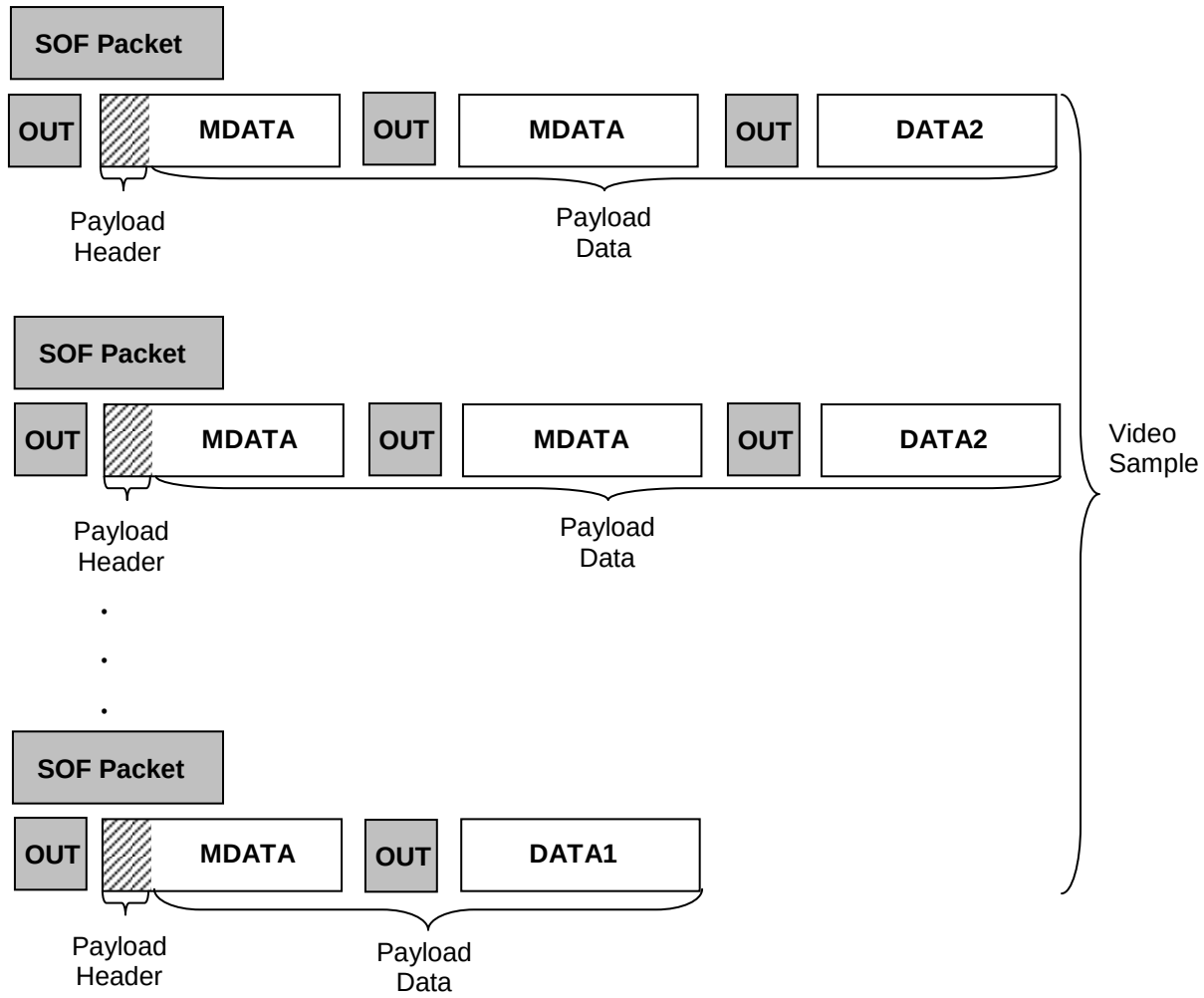


**Figure 4-2 Example MPEG-2 TS Isochronous Transfer, OUT Endpoint**

### 4.3 Bulk Transfer IN

The following example shows the relationship between Payload Transfers, the token and data packets of the MPEG-2 TS payload format when receiving bulk transfers from a device. Handshake packets are not shown for the sake of clarity.

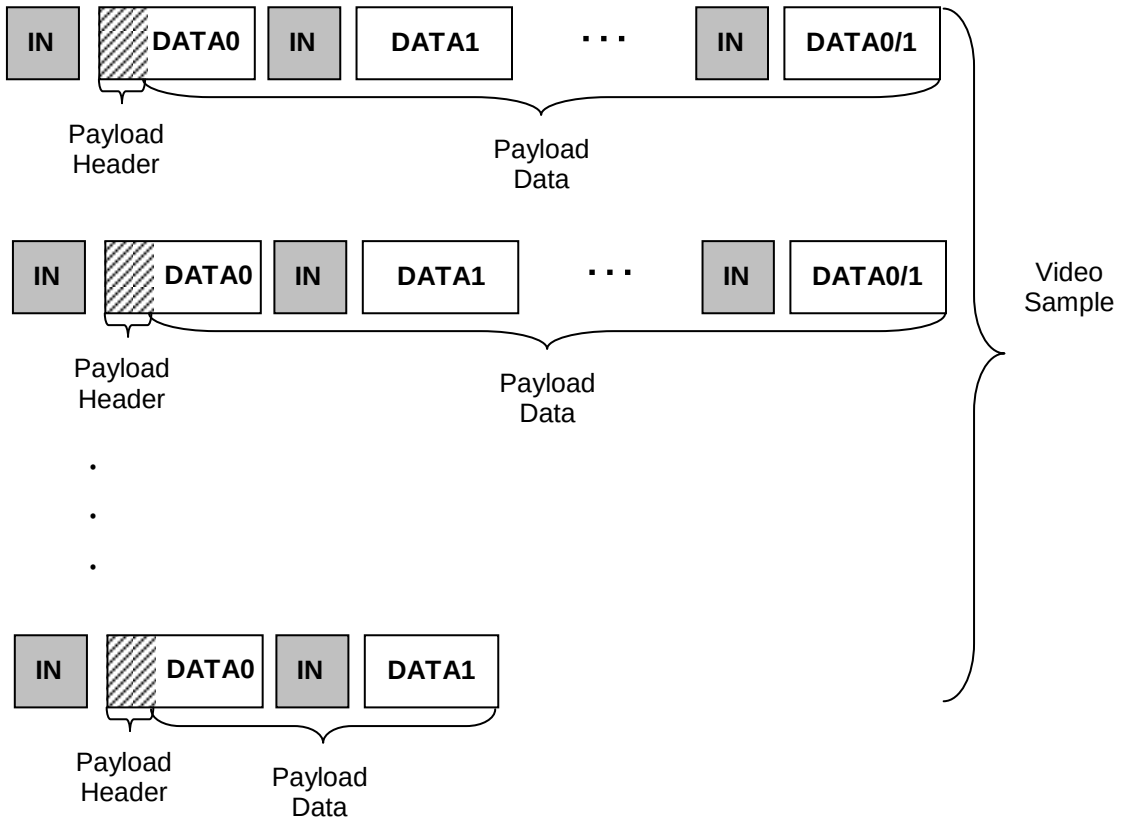


**Figure 4-3 Example MPEG-2 TS Bulk Transfer, IN Endpoint**

### 4.4 Bulk Transfer OUT

The following example shows the relationship between Payload Transfers, the token and data packets of the MPEG-2 TS payload format when sending bulk transfers to the device. Handshake packets are not shown for the sake of clarity.

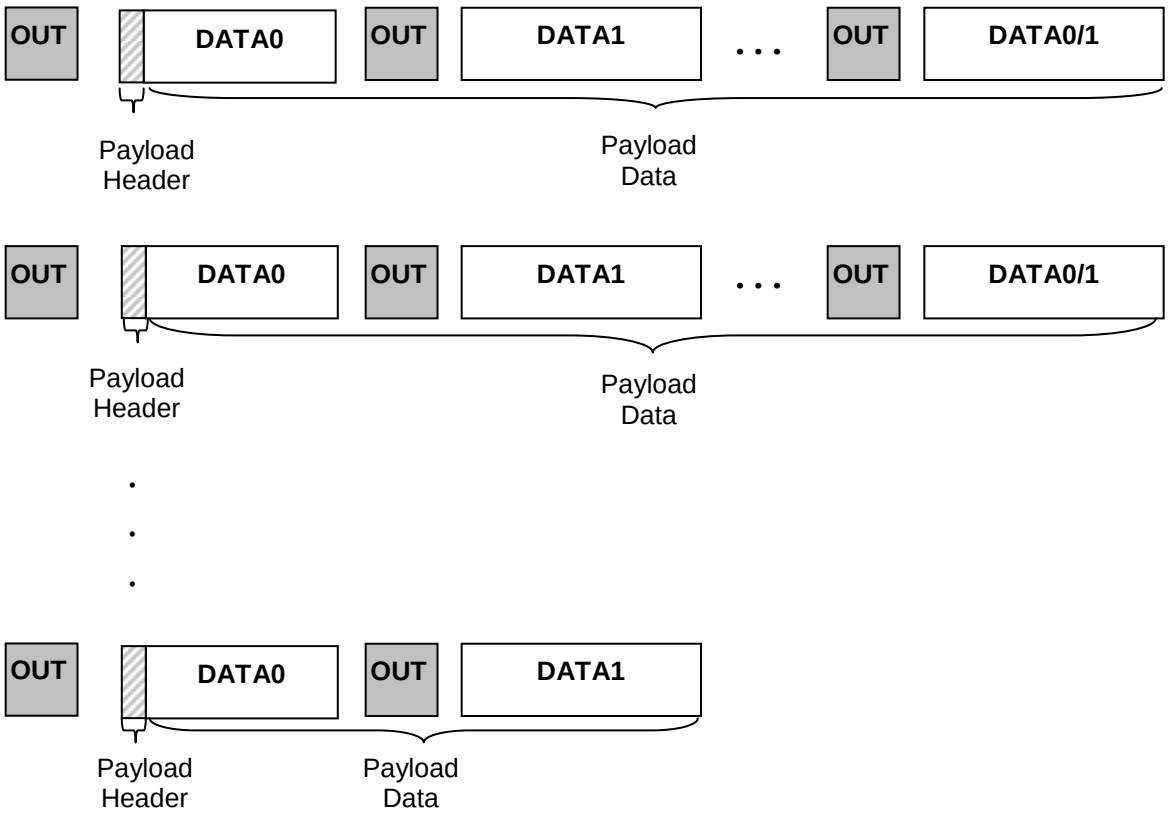


**Figure 4-4 Example MPEG-2 TS Bulk Transfer, OUT Endpoint**