

ICS 号: ICS 33.160.25
中国标准文献分类号: M74

世界超高清视频产业联盟标准

T/UWA 005.2-1-2025

高动态范围（HDR）视频技术 第 2-1 部分:

应用指南 系统集成

High Dynamic Range Video Technology Part 2-1:

Application Guide System Integration

(V1.1)

(征求意见稿)

2025-xx-xx 发布

2025-xx-xx 实施

世界超高清视频产业联盟

目 次

前 言	I
1. 范围	1
2. 规范性引用文件	1
3. 术语和定义	1
4. 缩略语	2
5. 概述	2
6. HDR Vivid 的 ES 流封装	2
7. HDR Vivid MP4 文件封装	7
8. HDR Vivid HLS 流封装	9
9. HDR Vivid DASH 封装	10
10. HDR Vivid DVB TS 流封装要求	10
11. 终端应用要求	12
12. 元数据生成	29
13. 内容保护	32

前 言

本文件按照 GB/T 1.1—2020《标准化工作导则 第1部分：标准化文件的结构和起草规则》给出的规则起草。

本文件是 UWA 005《高动态范围（HDR）视频技术》的第2-1部分。UWA 005《高动态范围（HDR）视频技术》包含以下部分：

- 第1部分：元数据及适配
- 第2-1部分：应用指南 系统集成
- 第2-2部分：后期制作要求及流程
- 第2-3部分：应用指南 环境光自适应
- 第3-1部分：技术要求和测试方法 显示设备
- 第3-2部分：技术要求和测试方法 便携式显示设备
- 第3-3部分：技术要求和测试方法 播放设备
- 第3-4部分：技术要求和测试方法 播放软件
- 第3-5部分：技术要求和测试方法 实时编码设备
- 第3-6部分：技术要求和测试方法 播放软件用设备
- 第3-7部分：技术要求和测试方法 投影显示设备

本文件代替 T/UWA 005.2-1-2022《高动态范围（HDR）视频技术第2-1部分：应用指南 系统集成》，与 T/UWA 005.2-1-2022 相比，除结构调整和编辑性改动外，主要技术变化如下：

- a) 更新HDR VIVID视频流程处理框图（5）
- b) 细化ES流封装的总体要求（6.1）；
- c) 增加HEVC/H.265静态元数据封装要求（6.2）
- d) 增加GPMI元数据传输要求（11.4）
- e) “后期制作的应用说明”章节更名为“元数据生成”（12）

请注意本文件的某些内容可能涉及专利。本文件的发布机构不承担识别这些专利的责任。

本文件由世界超高清视频产业联盟提出并归口。

本文件主要起草单位：

本文件主要起草人：

高动态范围（HDR）视频技术 第2-1部分：应用指南 系统集成

1.范围

本文件规定了支持T/UWA005.1-2022规定的HDR系统集成总体内容，包括制作，传输和接收和解码。

本文件适用于广播电视、数字电影、网络电视、网络视频、视频监控、实时通信、数字存储媒体等视频应用中高动态范围视频信号的采集、制作、播出、分发、接收、显示等处理。

2.规范性引用文件

下列文件中的内容通过文中的规范性引用而构成本文件必不可少的条款。其中，注日期的引用文件，仅该日期对应的版本适用于本文件。不注日期的引用文件，其最新版本（包括所有的修改单）适用于本文件。

GB/T 17975.1 信息技术 运动图像及其伴音信息的通用编码 第1部分：系统

GB/T 33475.2 信息技术 高效多媒体编码 第2部分：视频

T/UWA 005.1-2024 高动态范围（HDR）视频技术 第1部分：元数据及适配

GY/T 368 先进高效视频编码

T/SUCA 001.1 通用多媒体接口规范 第1部分：架构

ISO/IEC 14496-12:2022 信息技术—视听对象编码—第12部分：ISO基础媒体文件格式（Information technology — Coding of audio-visual objects—Part 12: ISO base media file format）

ISO/IEC 23009-1 信息技术—HTTP上的动态自适应码流—第1部分：媒体呈现描述和片段格式（Information technology — Dynamic adaptive streaming over HTTP (DASH) — Part 1: Media presentation description and segment formats）

ITU-R BT.2020 超高清电视系统节目制作和国际交换的参数数值（Parameter values for ultra-high definition television systems for production and international programme exchange）

ITU-T H.274 用于编码视频位流的多功能补充增强信息消息（Versatile supplemental enhancement information messages for coded video bitstreams）

ITU-T Rec. T.35对于非标准化设备ITU-T代码分配程序（Procedure for the allocation of ITU-T defined codes for non-standard facilities）

IETF RFC 8216 基于HTTP的自适应码率流媒体传输（HTTP Live Streaming）

3.术语和定义

下列术语和定义适用于本文件。

3.1 HDR Vivid

T/UWA 005.1规定的HDR技术规范，及配套衍生技术的代称。

3.2 HDR Vivid 播放设备

能够进行视频解码并依据 T/UWA 005.1-2021完成元数据和相应图像处理，最后通过数字视频接口进行输出的设备。

3.3 HDR Vivid 显示设备

够依据 T/UWA 005.1-2021完成元数据和相应图像处理，并进行图像显示的设备。

3.4 AVS2

GB/T 33475.2规定的视频编码技术。

3.5 AVS3

GY/T 368规定的视频编码技术。

3.6 通用多媒体接口 general purpose multimedia interface (GPMI)

T/SUCA 001.1规定的接口。

4. 缩略语

下列缩略语适用于本文件。

DASH 基于HTTP的动态自适应流媒体 (Dynamic Adaptive Streaming over HTTP)

DCCD 设备综合能力描述 (Device Comprehensive Capability Description)

DIP 描述性信息报文 (Descriptive Information Packet)

ES 基本数据流 (Elemental Stream)

EOTF 电光转换函数 (Electro-Optical Transfer Function)

HDR 高动态范围 (High Dynamic Range)

HLG 基于混合对数伽马量化方法 (Hybrid Log-Gamma)

HLS 基于HTTP的自适应码率流媒体传输 (HTTP Live Streaming)

OETF 光电转换函数 (Opto-Electrical Transfer Function)

ISOBMFF ISO基础媒体文件格式 (ISO Base Media File Format)

PQ 基于人眼特性的感知量化方法 (Perceptual Quantizer)

TS 传输数据流 (Transport Stream)

XAVC 扩展先进视频编码 (Extended Advanced Video Coding)

5. 概述

HDR Vivid视频流包含符合 T/UWA 005.1规定的动态元数据流，解码器需要解析该HDR Vivid视频流中的每一帧动态元数据信息，终端需要使用动态元数据信息，根据 T/UWA 005.1做相应的处理，如下图所示：



图1 HDR Vivid视频流处理框图

HDR元数据生成按第12章的要求。HDR元数据封装按第6章至第10章的要求。HDR元数据终端应用按第11章的要求。

6. HDR Vivid 的 ES 流封装

6.1.总体要求

视频中的每一帧均应携带动态元数据，并且使用该帧的动态元数据对该帧进行显示适配。

如果发生动态元数据丢失情况，宜采用丢失前最后一帧有效动态元数据，直到重新获得有效的动态元数据。

动态元数据传输包括两种模式，即统计信息模式和曲线参数模式。

当色调映射标识位(`tone_mapping_enable_mode_flag`)为0，不传输动态元数据中的色调映射曲线参数，即不传输T/UWA 005.1-2024中7.4.7~7.4.27定义的元数据。这种元数据传输模式称为统计信息模式。

当色调映射标识位为1，传输包括T/UWA 005.1-2024中7.4.1~7.4.6中的元数据，也包括T/UWA 005.1-2024中7.4.7~7.4.27中的元数据。这种动态元数据传输模式称为曲线参数模式。

6.2.AVC/H.264、HEVC/H.265 以及 VVC/H.266 ES 流封装

HDR Vivid静态元数据封装要求如下。HDR Vivid ES流打包时每一个IDR帧应包含该视频序列对应的静态元数据。静态元数据应封装在AVC/H.264、HEVC/H.265以及VVC/H.266 ES的`content_light_level_info()`和`mastering_display_colour_volume()`，其语法描述见ITU-T Rec.H.274。

HDR Vivid 动态元数据封装要求如下。

HDR Vivid 每一帧的动态元数据在 AVC/H.264、HEVC/H.265 以及 VVC/H.266 ES 码流中的封装应符合 ITU-T Rec. T.35 (02/2000) 及以下要求：

a) 动态元数据封装在`user_data_registered_itu_t_t35(payloadSize)`中；

b) `user_data_registered_itu_t_t35(payloadSize)`中的`itu_t_t35_country_code`为0x26，`itu_t_t35_country_code`表示国家码，0x26表示为中国；

c) `itu_t_t35_payload()`中的`terminal_provide_code`为0x0004，`terminal_provide_oriented_code`为0x0005。其中`terminal_provide_code`表示为组织或者机构码，0x0004为CUVA，`terminal_provide_oriented_code`为应用码，0x0005为HDR Vivid的版本1.0，版本定义见表5。

ES 封装的语法见表 1：

表1 ES流动态元数据封装语法结构

伪代码	描述符
<code>user_data_registered_itu_t_t35(payloadSize) {</code>	
<code>itu_t_t35_country_code</code>	0x26
<code>if(itu_t_t35_country_code != 0xFF){</code>	
<code>i = 1</code>	
<code>}</code>	
<code>else {</code>	
<code>itu_t_t35_country_code_extension_byte</code>	b(8)
<code>i = 2</code>	
<code>}</code>	
<code>do {</code>	
<code>itu_t_t35_payload_byte</code>	b(8)
<code>i++</code>	
<code>} while(i < payloadSize)</code>	

}	
---	--

其中itu_t_t35_payload_byte中的数据如下表所示

表2 T35 payload语法结构

itu_t_t35_payload(){	描述符
terminal_provide_code	0x0004
terminal_provide_oriented_code	u(16)
dynamic_metadata()	
}	

6.3.AVS2 以及 AVS3 ES 流封装

HDR Vivid AVS2流和HDR Vivid AVS3流每一帧的ES流中每一帧的HDR Vivid动态元数据封装在extension_data(i)的hdr_dynamic_metadata_extension()中，且hdr_dynamic_metadata_extension()中的第一个extension_id为0x5（4位），第二个hdr_dynamic_metadata_type为0x5（4位），hdr_dynamic_metadata_extension()语法结构见表3：

表3 hdr_dynamic_metadata_extension语法结构

高动态范围图像元数据扩展定义	描述符
hdr_dynamic_metadata_extension(){	
extension_id	0x5
hdr_dynamic_metadata_type	0x5
while(next_bits(24) != '0000 0000 0000 0000 0000 0001'){	
extension_data_byte	u(8)
}	
next_start_code()	
}	

extension_data_byte（）中的语法结构见表4：

表4 extension_data_byte语法结构

动态元数据定义	描述符
extension_data_byte(){	
itu_t_t35_country_code	0x26
terminal_provide_code	0x0004
terminal_provide_oriented_code	u(16)
system_start_code	u(8)
if(system_start_code==0x01){	
num_windows=1	
for(w = 0; w < num_windows; w++) {	
minimum_maxrgb_pq[w]	u(12)
marker_bit	f(1)
average_maxrgb_pq[w]	u(12)

marker_bit	f(1)
variance_maxrgb_pq[w]	u(12)
marker_bit	f(1)
maximum_maxrgb_pq[w]	u(12)
marker_bit	f(1)
}	
for(w = 0; w < num_windows; w++) {	
tone_mapping_enable_mode_flag[w]	u(1)
if(tone_mapping_enable_mode_flag[w]==1){	
tone_mapping_param_enable_num[w]	u(1)
tone_mapping_param_enable_num[w]++	
for(i=0; i < tone_mapping_param_enable_num[w]; i++){	
targeted_system_display_maximum_luminance_pq[i][w]	u(12)
base_enable_flag[i][w]	u(1)
marker_bit	f(1)
if(base_enable_flag[i][w]){	

表4 extension_data_byte数据表（续）

动态元数据定义	描述符
base_param_m_p[i][w]	u(14)
base_param_m_m[i][w]	u(6)
marker_bit	f(1)
base_param_m_a[i][w]	u(10)
base_param_m_b[i][w]	u(10)
marker_bit	f(1)
base_param_m_n[i][w]	u(6)
base_param_K1[i][w]	u(2)
base_param_K2[i][w]	u(2)
base_param_K3[i][w]	u(4)
base_param_Delta_enable_mode[i][w]	u(3)
marker_bit	f(1)
base_param_enable_Delta[i][w]	u(7)
}	
3Spline_enable_flag[i][w]	u(1)
if(3Spline_enable_flag[i][w]){	
3Spline_enable_num[i][w]	u(1)
3Spline_enable_num[i][w]++;	
for(j = 0; j < 3Spline_enable_num[i][w]; j++) {	
3Spline_TH_enable_mode[j][i][w]	u(2)
if((3Spline_TH_mode[j][i][w]==0) (3Spline_TH_mode[j][i][w]==2)){	
3Spline_TH_enable_MB[j][i][w]	f(8)
}	

marker_bit	f(1)
3Spline_TH_enable[j][i][w]	f(12)
marker_bit	f(1)
3Spline_TH_enable_Delta1 [j][i][w]	f(10)
3Spline_TH_enable_Delta2 [j][i][w]	f(10)
marker_bit	f(1)
3Spline_enable_Strength[j][i][w]	f(8)
}	
}	
}	
}	
}	
color_saturation_mapping_enable_flag[w]	u(1)
if(color_saturation_mapping_enable_flag[w]) {	
color_saturation_enable_num[w]	u(3)
for(i = 0; i < color_saturation_enable_num [w]; i++) {	
color_saturation_enable_gain[i][w]	u(8)
marker_bit	f(1)
}	
}	
}	
}	
stuffing_bit	'1'
while(!byte_aligned())	
stuffing_bit	'0'
}	

extension_data_byte () 中的terminal_provide_cod和terminal_provide_oriented_code分别对应6.2中itu_t_t35_payload () 结 构 中 描 述 的 terminal_provide_code 及 terminal_provide_oriented_code 。 terminal_provide_code为0x0004（16位）， terminal_provide_oriented_code（16位）为当前版本号。

extension_data_byte () 中的marker_bit是为了不应出现从任意字节对齐位置开始的21个以上连续的“0”。**extension_data_byte ()** 中其他语法元素请参考 T/UWA 005.1-2021。

6.4.版本后向兼容

HDR Vivid目前保留了4个版本。每个版本可以生成一套独立的动态元数据，封装在同一个ES流中，即一个ES流中可以携带多版本的动态元数据，并且多版本可以是不连续的版本号。例如一个ES流可以同时传版本1和版本2的动态元数据；也可以同时传输传版本1、版本3和版本4的动态元数据。itu_t_t35_payload () 中的terminal_provide_oriented_code 码字用来标识版本。终端设备宜提取支持的最高版本的动态元数据进行后处理。版本号和标识的映射关系见表5。

注：如果终端不能识别HDR Vivid某个版本，可忽略该版本；如果全部不能识别，可不作HDR Vivid处理。

表5 版本号与标识的映射关系

HDR Vivid 版本号	itu_t_t35_payload() 中的terminal_provide_oriented_code码字
1.0	0x0005
2.0	0x0006
3.0	0x0007
4.0	0x0008

7.HDR Vivid MP4 文件封装

HDR Vivid MP4文件结构应符合ISO/IEC 14496-12:2022。增加HDR Vivid视频码流格式语法盒(Box)的方法如下:

a) 在保留原本MP4所有结构的基础上,应在视频轨的Sample Description Box(stsd)内的VisualSampleEntry Box中扩展添加一种新的描述HDR Vivid视频码流格式的Box: CUVV Configuration Box。

b) 可修改VisualSampleEntry Box 中的 compressorname描述字段为“HDR Vivid video”

注: 应用程序主要通过 cuvv box 来识别是否是 HDR Vivid video。

CUVV Configuration Box的语法结构见表6。

表6 CUVV Configuration Box语法结构

class CUVVConfigurationBox extends Box('cuvv')	描述符
{	
unsigned int (16) cuva_version_map;	u (16)
unsigned int (16) terminal_provide_code;	0x0004
unsigned int (16) terminal_provide_oriented_code;	u (16)
const unsigned int (32)[4] reserved = 0;	u (128)
}	

CUVV Configuration Box的语法盒类型(four-character-code)为‘cuvv’。

内部结构结构中cuva_version_map(16位)描述了HDR Vivid视频的版本号信息。每一位表示一个版本号,一共可以表示16个版本号,其中高位表示高版本号,低位表示低版本号。

注: 例如cuva_version_map为0x0009该码流有版本4码流和版本1码流(9为1001)。

另外两位描述符分别对应6.2中itu_t_t35_payload()结构中描述的terminal_provide_code及terminal_provide_oriented_code。terminal_provide_code为0x0004(16位),terminal_provide_oriented_code(16位)为当前码流中包含的最高版本对应的值。

注: 例如当cuva_version_map为0x0009时,当前码流包含的最高版本为版本4,根据6.4节中表6.4可知版本4对应的terminal_provide_oriented_code应为0x0008。

在MP4文件中添加cuvv的具体方法为在moov->trak(video)->mdia->minf->stbl->stsd内的VisualSampleEntry Box中,添加一个cuvv Box。添加后的MP4文件结构见图2。

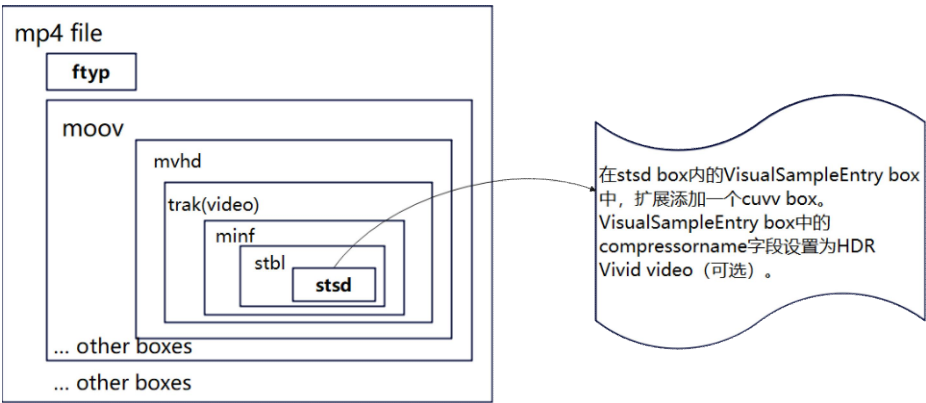


图 2：添加语法盒类型 ‘cuvv’ 后的 MP4 文件结构

VisualSampleEntry描述见ISO/IEC 14496-12(2015-12-15)的12.1.3.2，如表7所示：

表7 VisualSampleEntry描述表

class VisualSampleEntry(codingname) extends SampleEntry (codingname){	描述符
unsigned int(16) pre_defined = 0;	
const unsigned int(16) reserved = 0;	
unsigned int(32)[3] pre_defined = 0;	
unsigned int(16) width;	
unsigned int(16) height;	
template unsigned int(32) horizresolution = 0x00480000; // 72 dpi	
template unsigned int(32) vertresolution = 0x00480000; // 72 dpi	
const unsigned int(32) reserved = 0;	
template unsigned int(16) frame_count = 1;	
string[32] compressorname;	
template unsigned int(16) depth = 0x0018;	
int(16) pre_defined = -1;	
// other boxes from derived specifications	
CleanApertureBox clap; // optional	
PixelAspectRatioBox pasp; // optional	
}	

将新增的CUVV Configuration Box嵌入 “other boxes from derived specifications”位置，在其它扩展 box之后添加CUVV Configuration Box。

示例：在H.265的MP4文件中，VisualSampleEntry为'hvc1'或'hev1'，已经存在hvcC box用于描述视频编码码流信息，CUVA MP4文件则是在hvcC box后面添加一个新的box: cuvv Box; 其余结构除了compressorname 描述修改为“CUVA HDR Video”之外，全部保持不变。

注：HDR Vivid MP4文件示例见图3：

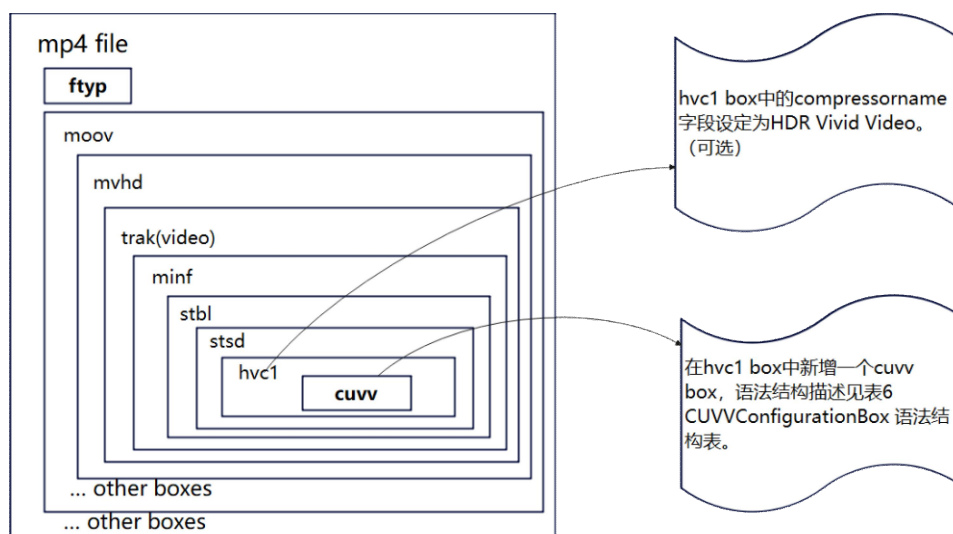


图3 HDR Vivid MP4文件描述图

8.HDR Vivid HLS 流封装

8.1.总体要求

HDR Vivid HLS流的播放列表文件（playlist file）除了增加m3u8文件关于HDR Vivid视频信息外，其他均符合RFC 8216的规定。

8.2.HLS 流封装

HLS流中使用#EXT-X-STREAM-INF或者EXT-X-I-STREAM-INF标识视频或者音频的属性，CUVA HLS流的封装在保持原m3u8描述的基础上，对于每一路视频，新增一条#EXT-X-STREAM-INF或者EXT-X-I-STREAM-INF描述信息，其中CODECS标签内原来描述视频信息的部分更改为CUVA描述符。CUVA描述符的格式定义为：[CUVAHDR_video].[CUVA_Version_map]，其中CUVAHDR_video标识表示的是HDR Vivid码流。

HLS流为mp4或者TS封装时，CUVAHDR_video值都为cuvv。CUVA_Version_map表示CUVA的版本号信息，CUVA_Version_map中的每一位表示一个版本号，其中高位表示高版本号，低位表示低版本号。

注：例如CUVA_Version_ma为1101表示该码流有版本4码流，版本3码流和版本1码流。其它的音视频属性信息还是直接使用原来的描述。

在m3u8的#EXT-X-STREAM-INF标签增加VIDEO-RANGE属性。该属性值是一个枚举字符串，有效的字符串有SDR、HLG和PQ。

注：VIDEO-RANGE描述项参照了《HTTP Live Streaming 2nd Edition draft-pantos-hls-rfc8216bis-16》。

HLS M3u8 描述示例：

/*新增的两条不同码率音视频属性描述*/

```
#EXT-X-STREAM-INF:BANDWIDTH=5120000,VIDEO-RANGE=PQ,CODECS="cuvv.1101,mp4a.40.29",FRAME-RATE=50.000,RESOLUTION=3840x2160,AUDIO="aac"
```

low/video.m3u8

```
#EXT-X-STREAM-INF:BANDWIDTH=7680000,VIDEO-RANGE=PQ,CODECS="cuvv.1101,mp4a.40.29",FRAME-RATE=50.000,RESOLUTION=3840x2160,AUDIO="aac"
```

high/video.m3u8

/*原始的两条不同码率音视频属性描述*/

```
#EXT-X-STREAM-INF:BANDWIDTH=5120000,VIDEO-RANGE=PQ, CODECS="hvc1.2.4.L153.b0,
mp4a.40.29",FRAME-RATE=50.000,RESOLUTION=3840x2160,AUDIO="aac"
low/video.m3u8
#EXT-X-STREAM-INF:BANDWIDTH=7680000,VIDEO-RANGE=PQ, CODECS="hvc1.2.4.L153.b0,
mp4a.40.29",FRAME-RATE=50.000,RESOLUTION=3840x2160,AUDIO="aac"
high/video.m3u8
```

9.HDR Vivid DASH 封装

9.1.总体要求

HDR Vivid DASH流的mpd文件除了在Representation字段中的codecs标签新增HDR Vivid视频信息外，其他均符合ISO/IEC 23009-1要求。

9.2.DASH 流封装

DASH流的mpd文件的Representation字段标识关于视频或者音频的属性，包含了帧率，分辨率，码率，编码器，带宽等信息。

HDR Vivid DASH流的封装在保持原Representation描述的基础上，对于每一路视频，新增一条Representation描述信息，其中codecs标签内原来描述视频信息的部分更改为HDR Vivid描述符。HDR Vivid描述符的格式定义为：[CUVAHDR_video].[CUVA_Version]，其中HDR Vivid_video标识表示HDR Vivid码流。

当DASH流为mp4或者TS封装时，HDR VividR_video都为cuvv。CUVA_Version_map表示HDR Vivid的版本号信息，CUVA_Version_map中的每一位表示一个版本号，其中高位表示高版本号，低位表示低版本号。

注：CUVA_Version_map为1110表示该码流有版本4、版本3和版本1码流。DASH 音视频描述分离，所以codecs内只需描述视频信息，新增的Representation也仅在每个视频轨上新增一条即可。

HDR Vivid DASH码流mpd文件描述示例如下：

```
<Representation id="bbb_30fps_1024x576_2500k" codecs="cuvv.110" bandwidth="3134488"/> //新增
<Representation id="bbb_30fps_1280x720_4000k" codecs="cuvv.110" bandwidth="4952892"/> //新增
<Representation id="bbb_30fps_1024x576_2500k" codecs=" hvc1.2.4.L153.b0" bandwidth="3134488"/>
<Representation id="bbb_30fps_1280x720_4000k" codecs=" hvc1.2.4.L153.b0" bandwidth="4952892"/>
```

10.HDR Vivid DVB TS 流封装要求

10.1.总体要求

HDR Vivid DVB TS流除了增加component_descriptor中关于CUVA码流的描述，其他语法均符合ETSI EN 300 468 V1.16.1 (2019-08)要求，具体修改的表为PMT表。

当节目含有HDR服务，而元数据缺失，应进行容错处理。

10.2.TS 流的封装

在保持component_descriptor原始描述的语法基础上，新增加以下字段来指定使用的HDR Vivid的动态元数据。除了新增的两个描述符信息registration_descriptor()和CUVV_video_stream_descriptor()，其它结构不变。

10.2.1.registration_descriptor()

在PMT表的视频描述信息中，新增一个注册描述符registration_descriptor()，内部字段定义见表8:

表8 registration_descriptor注册描述符表

registration_descriptor ()	描述符
{	
unsigned int (8) descriptor_tag ;	0x05
unsigned int (8) descriptor_length;	0x04
unsigned int (32) Format_identifier ;	‘cuvv’
}	

descriptor_tag 为 0x05，为注册描述符 TAG。

descriptor_length 为 0x04，为‘cuvv’描述符长度。

Format_identifier 值必须为 0x63757676(‘cuvv’)值,这个值可以用于快速甄别 TS 码流是否为 HDR Vivid 码流。

10.2.2.CUVV_video_stream_descriptor()

在 PMT 表的视频描述信息中，新增一个用户自定义描述符 CUVV_video_stream_descriptor()，可以用于描述 HDR Vivid 版本信息等，内部字段定义见表 9：

表9 用户自定义描述符表

CUVV_video_stream_descriptor()	描述符
{	
unsigned int (8) descriptor_tag;	0xF3
unsigned int (8) descriptor_length;	0x0A
unsigned int (32) cuvv_tag ;	‘cuvv’
unsigned int (16) cuva_version_map;	0x0005(版本号信息)
unsigned int (16) terminal_provide_code;	0x0004
unsigned int (16) terminal_provide_oriented_code;	0x0007(最高版本号)
}	

descriptor_tag 使用用户自定义 0xF3。

descriptor_length 为 10。

cuvv_tag 值为 0x63757676(‘cuvv’)，主要用户 0xF3 用户自定义 tag 冲突时，进一步标识该描述符内容为 CUVV 描述信息。

CUVA_Version_map 表示 HDR Vivid 的版本号信息，总共 16 位，CUVA_Version_map 中的每一位表示一个版本号，其中高位表示高版本号，低位表示低版本号。

注：CUVA_Version_map 为 0x0005(5 对应 0101)表示该码流有版本 3 码流和版本 1 码流。

最后两位描述符分别对应 6.2 中 itu_t_t35_payload () 结构中描述的 terminal_provide_code 及 terminal_provide_oriented_code。terminal_provide_code 为 0x0004(16 位)，terminal_provide_oriented_code

（16 位），为当前码流中包含的最高版本对应的值。

注：当 cuva_version_map 为 0x0005 时，当前码流包含的最高版本为版本 3。根据 6.4 节中表 5 可知版本 3 对应的 terminal_provide_oriented_code 应为 0x0007。

10.3.TS 流的丢包处理

HDR Vivid DVB TS流出现无法同步或者丢包3个以上，应重新同步。

11.终端应用要求

11.1.终端系统组成

11.1.1.功能模块

HDR Vivid终端部分包含视频接收解码和视频内容呈现两部分功能，其框图见图4。

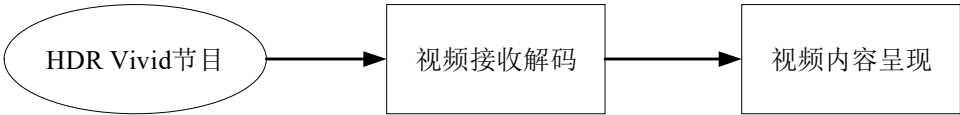


图4 HDR Vivid终端功能框图

视频接收解码通过播放设备实现，一般主要指带有网口、同轴接口、USB等输入接口并带有视频解码能力的设备，包括机顶盒，播放器，或带解码能力的显示设备等。

视频内容呈现通过显示设备实现，一般主要指电视，投影仪，显示器，笔记本电脑，手机等具备图像呈现功能的设备。根据其对于HDR Vivid的支持能力，显示设备的类别见表10。

表10 显示设备分类

设备类型	说明
HDR Vivid 显示设备	符合 HDR Vivid 规范，能够接收、处理 HDR Vivid 元数据并显示视频信号。
HDR 显示设备	仅能接收、解码、显示 BT.2100 定义的 PQ-HDR 或 HLG-HDR 视频信号及静态元数据。
SDR 显示设备	不支持 BT.2100 定义的 HDR 信号，仅能接收、解码、显示 BT.2020 或 BT.709 或 BT.601 定义的 SDR 视频信号。

11.1.2.应用方式

终端系统可以有以下两种应用方式：

a) 一体机应用方式

显示设备直接从视频信号分发传输网络接收HDR Vivid视频信号，在显示设备内部完成解码、播放、显示呈现全过程。

b) 播放器应用方式

显示设备通过播放设备间接从视频节目分发传输网络接收HDR Vivid视频信号，在播放设备内进行HDR Vivid视频信号的解码播放，显示设备进行显示呈现。播放设备与显示设备通过数字视频接口联接及传输HDR Vivid视频信号。

播放设备仅需要支持播放器应用方式。显示设备应支持一体机方式或播放器方式。

11.2.HDR Vivid 信号终端解码与呈现

11.2.1.基本要求

播放设备需要至少提供一种适配模式，在该模式下仅进行基于本规范要求的处理和其他必要的图像

处理，以确保输出的图像和信息满足本标准的要求。

显示设备需要至少提供一种适配模式，能够按照11.2.2的要求进行图像处理，以确保HDR Vivid视频图像能够按照本标准要求的方式进行呈现。

11.2.2.播放器应用方式的适配模式

当HDR Vivid的解码和显示由播放设备与显示设备共同完成时，二者有两种适配模式：

a) 接收端适配模式

系指播放设备将HDR Vivid信号及动态元数据送给显示设备，由显示设备按照 T/UWA 005.1要求进行色彩动态范围转换过程和色彩校正过程处理的适配模式。接收端适配模式下，当显示设备接收到的视频图像出现不同传输特性曲线、不同色域或色彩空间转换矩阵等信息变化时，画面应当保持稳定，不出现视觉可见的闪烁、黑屏、色彩断层、曝光亮度溢出等现象。

b) 监视器适配模式

系指播放设备根据显示设备提供的动态范围信息（EDID）按照 T/UWA 005.1要求完成色彩动态范围转换过程和色彩校正过程处理，并将处理后的图像传送给显示设备的适配模式。这种模式下，显示设备接收到播放设备发送的图像信号后，不再进行色彩动态范围转换过程和色彩校正过程处理，只将该图像信号进行传输特性曲线转换和其他必要处理后进行显示。

当HDR Vivid显示设备具备数字视频输入接口且该接口支持HDR Vivid格式时，应支持至少一种适配模式。HDR Vivid播放设备需要完全支持上述两种适配模式。HDR Vivid播放设备应该优先以接收端适配模式连接HDR Vivid显示设备；同时可提供菜单供用户自行选择适配模式。

11.2.3.PQ HDR 信号的互联适配

当HDR Vivid播放设备接收到类型为PQ HDR的HDR Vivid信号时，需要根据对接的显示设备类型进行信号格式的转换。具体方式如下表所示：

表 11显示设备的PQ信号格式转换表

显示设备类型	HDR Vivid显示设备		HDR显示设备	SDR显示设备
	监视器适配模式	接收端适配模式		
播放设备输出信号	经处理的PQ HDR信号和HDR Vivid VS-IF信息帧	PQ HDR信号和HDR Vivid动态元数据VS-EMDS信息帧	经处理的PQ HDR信号及必要的元数据信息。	经处理转换为SDR信号，输出色域优先级依次为ITU-R BT.2020、ITU-R BT.709、ITU-R BT.601

11.2.4.HLG HDR 信号的互联适配

当HDR Vivid播放设备接收到类型为HLG HDR的HDR Vivid信号时，需要根据对接的显示设备类型进行信号格式的转换。具体方式如下表所示：

表 12显示设备的HLG信号格式转换表

显示设备类型	HDR Vivid显示设备		HDR显示设备	SDR显示设备
	监视器适配模式	接收端适配模式		
播放设备输出信号	经处理的PQ HDR信	HLG HDR信号和	1) HLG HDR (PQ	经处理转换为SDR信

	号和HDR Vivid VS-IF信息帧	HDR Vivid动态元数据VS-EMDS信息帧（如果视频信号中存在HDR Vivid动态元数据 ^{注1} ）	HDR ^{注2} ）信号及静态元数据（如果视频信号中包含静态元数据） 2）或经处理的PQ HDR信号	号，输出色域优先级依次为ITU-R BT.2020、ITU-R BT.709、ITU-R BT.601
--	----------------------	--	--	---

注1:HDR Vivid对于HLG HDR信号是否需携带动态元数据不做强制要求。

注2:部分HDR显示设备不能支持HLG HDR格式，只能支持PQ HDR格式，这种情况下，播放设备应将HLG HDR信号转为PQ HDR信号送给HDR显示设备显示。

11.3.HDMI 动态元数据传输要求

11.3.1.通则

HDR Vivid播放设备与HDR Vivid显示设备通过HDMI进行连接和图像数据传输时，应当按照本章节要求进行动态元数据传输。

基于HDMI传输的HDR Vivid信号，信号的电-光传输特性由VS-IF或VS-EMDS中transfer_character字段确定，信号的色域为ITU-R BT.2020。

当HDR Vivid播放设备与HDR Vivid显示设备建立连接后，播放设备通过查询显示设备的EDID中符合11.3.2规定的VS-VDB数据块来了解显示设备对HDR Vivid标准的支持能力。

当显示设备的VS-VDB表明能够支持HDR Vivid接收端适配模式时，播放设备应将HDR Vivid的动态元数据封装在符合11.3.4规定的VS-EMDS信息帧中，配合HDR Vivid视频图像发送给显示设备。

当显示设备的VS-VDB表明能够支持HDR Vivid监视器适配模式时，播放设备应根据显示设备VS-VDB中提供的最大显示亮度和最小显示亮度信息，将HDR Vivid的视频内容根据 T/UWA 005.1-2024 第9章的规定进行适配处理后发送给显示设备，并同时向显示设备发送11.3.3规定的VS-IF信息。

11.3.2.VS-VDB 信息定义

11.3.2.1.语法

HDR Vivid显示设备在HDMI EDID中采用VS-VDB数据块来标识是否具备HDR Vivid动态元数据接收能力。HDR Vivid VS-VDB数据块信息语法见表13：

表 13 VS-VDB数据块信息语法表

字节\比特	7	6	5	4	3	2	1	0
0	Tag Code (0x07)			Length (0x1B)				
1	Extended Tag Code=0x01 (VSVDB)							
2	IEEE OUI/CID (0x03)							
3	IEEE OUI/CID (0x75)							
4	IEEE OUI/CID (0x04)							
5	system_start_code[7...0]							
6	version_code[3...0]				0 (保留)	0 (保留)	0 (保留)	0 (保留)
7	display_maximum_luminance [7...0]							
8	display_maximum_luminance [15...8]							
9	display_maximum_luminance [23...16]							
10	display_maximum_luminance [31...24]							

11	display_minimum_luminance [7...0]		
12	display_minimum_luminance [15...8]		
13	monitor_mode_support	rx_mode_support	0（保留）
14~27	0（保留）		

11.3.2.2.语义

系统起始码 system_start_code

8位无符号整数。表示终端支持的HDR Vivid版本号,当前版本赋值为1.0。

如果当前版本号高于码流中接收到的动态元数据所包含的system_start_code，则按照HDR Vivid VS-EMDS包数据结构将动态元数据进行打包；否则在当前发送端设备上根据T/UWA 005.1进行显示适配处理，然后将进行过适配处理之后的数据发送到接收端。

系统版本码 version_code

4位无符号整数。表示终端的CUVA系统版本号，当前版本赋值为1.0。发送端按照接收端的版本发送相应格式的动态元数据信息。

设备最高显示亮度 display_maximum_luminance

32位无符号整数，表示显示设备在D65色温下可以显示的最大亮度，单位是0.0001cd/m²。当码字为0x00000001时，表示0.0001cd/m²。MaxDisplay最终取值为display_maximum_luminance /10000。当显示设备支持监视器适配模式时，显示设备需要为display_maximum_luminance设置适当数值，以便播放设备使用该信息进行图像处理；当显示设备不支持监视器适配模式时，应该将display_maximum_luminance设置为0。

设备最小显示亮度 display_minimum_luminance

16位无符号整数，表示显示设备在D65色温下可以显示的最小亮度，单位是0.0001cd/m²。当码字为0x0001时，表示0.0001cd/m²。MinDisplay最终取值为display_minimum_luminance /10000。当显示设备支持监视器适配模式时，显示设备需要为display_minimum_luminance设置适当数值，以便播放设备使用该信息进行图像处理；当显示设备不支持监视器适配模式时，应该将display_minimum_luminance设置为0。

监视器适配模式支持标识monitor_mode_support

1位无符号整数。表示显示终端对监视器适配模式的支持能力，monitor_mode_support=1表示显示终端支持监视器适配模式，monitor_mode_support=0表示显示终端不支持监视器适配模式。

接收端适配模式支持标识rx_mode_support

1位无符号整数。表示显示终端对接收端适配模式的支持能力，rx_mode_support=1表示显示终端支持接收端适配模式，rx_mode_support=0表示显示终端不支持接收端适配模式。

11.3.3.VS-IF 信息定义

11.3.3.1.语法

在HDR Vivid 监视器适配模式下，采用HDMI中VS-IF信息帧从播放设备向显示设备传输静态元数据。VS-IF信息帧语法如下表所示：

表14 VS-IF信息帧语法表

Byte\Bit	7	6	5	4	3	2	1	0
HB0	VSIF Type Code =0x81							
HB1	VSIF Version=0x01							

HB2	0	0	0	Payload Length=27				
PB00	checksum							
PB01	IEEE OUI/CID （0x03）							
PB02	IEEE OUI/CID （0x75）							
PB03	IEEE OUI/CID （0x04）							
PB04	system_start_code[7...0]							
PB05	version_code[3...0]				Monitor_m ode_enable =0x01	transfer_ character=0	0（保留）	0（保留）
PB06~PB27	0（Reserved）							

11.3.3.2.语义

监视器适配模式支持标识monitor_mode_enable

1位无符号整数。本版本规范要求monitor_mode_enable固定为1，表示此时的HDR Vivid信号的互联适配处理方式为监视器适配模式。

图像传输特性 transfer_character

1位无符号整数。表示图像的传输特性，本版本规范要求transfer_character固定为0，表示监视器适配模式下，图像传输特性为ST2084 EOTF曲线。

11.3.4.VS-EMDS 信息定义

在HDR Vivid 接收端适配模式下，采用HDMI中VS-EMDS信息帧从播放设备向显示设备传输动态元数据。

11.3.4.1.语法

HDR Vivid的VS-EMDS信息语法如下表所示：

表15 VS-EMDS信息语法表

Byte\ Bit	7	6	5	4	3	2	1	0
HB0	0	1	1	1	1	1	1	1
HB1	1 (First)	0 (Last)	0 (保留)	0(保留)	0 (保留)	0 (保留)	0 (保留)	0 (保留)
HB2	0 (Sequence_Index)							
PB0	1 (New)	0 (End)	1 (DS_Type)		0 (AFR)	1 (VFR)	1 (Sync)	0 (保留)

PB01 0 (保留)

PB02 0
(Organization_I
D)

PB03 0x00
(Data_Set_Tag_
MSB)

PB04 0x02
(Data_Set_Tag_

Byte\ Bit	7	6	5	4	3	2	1	0
	LSB)							
PB05	0x00 (Data_Set_Leng th_MSB)							
PB06	0x38 (Data_Set_Leng th_LSB)							
PB07	IEEE OUI/CID (03)							
PB08	IEEE OUI/CID (75)							
PB09	IEEE OUI/CID (04)							
PB10	system_start_cod e[7...0]							
PB11	version_code[3... 0]		minimu m_max rgb_pq [11...8]					
PB12	minimum_maxrg b_pq [7...0]							
PB13	0 (保留)	0 (保留)	0 (保留)	0(保留)	average_maxrgb_pq [11...8]			
PB14	average_maxrgb_ pq [7...0]							
PB15	0 (保留)	0 (保留)	0 (保留)	0(保留)	variance_maxrgb_pq [11...8]			
PB16	variance_maxrgb _pq [7...0]							
PB17	0 (保留)	0 (保留)	0 (保留)	0(保留)	maximum_maxrgb_pq [11...8]			
PB18	maximum_maxrg b_pq [7...0]							
PB19	0 (保留)	0 (保留)	0 (保留)	0(保留)	targeted_system_display_maximum_luminance_pq[11...8]			
PB20	targeted_system_ display_maximu m_luminance_ pq[7...0]							
PB21	Transfer_Cha racter	base_ enable flag[0]	base_param_m_p[13...8]					
PB22	base_param_m_p [7...0]							

Byte\ Bit	7	6	5	4	3	2	1	0
PB23	0（保留）	0（保留）	base_param_m_m [5...0]					
PB24	0（保留）	0（保留）	0（保留）	0(保留)	0（保留）	0（保留）	base_param_m_a[9,8]	
PB25	base_param_m_a [7...0]							
PB26	0（保留）	0（保留）	0（保留）	0(保留)	0（保留）	0（保留）	base_param_m_b[9...8]	
PB27	base_param_m_b [7...0]							
HB0	0	1	1	1	1	1	1	1
HB1	0（First）	0（Last）	0（保留）	0(保留)	0（保留）	0（保留）	0（保留）	0（保留）
HB2		1（Sequence_Index）						
PB0	0（保留）	0（保留）	base_param_m_n[5...0]					
PB1	0（保留）	0（保留）	base_param_K1[1..0]		base_param_K2[1..0]		base_param_K3[1..0]	
PB2	0（保留）	0（保留）	0（保留）	0(保留)	0（保留）	base_param_Delta_enable_mode[2..0]		
PB3	0（保留）	base_param_enable_Delta[6...0]						
PB4	0（保留）	0（保留）	0（保留）	3Spline_enable_num		3Spline_enable_flag [0]	3Spline_TH_enable_mode0[1, 0]	
PB5	3Spline_TH_enab le_MB0[7...0]							
PB6	0（保留）	0（保留）	0（保留）	0(保留)	3Spline_TH_enable0[11...8]			
PB7	3Spline_TH_enab le0[7...0]							
PB8	0（保留）	0（保留）	0（保留）	0(保留)	0（保留）	0（保留）	3Spline_TH_enable_0Delta1[9,8]	
PB9	3Spline_TH_enab le_0Delta1[7...0]							
PB10	0（保留）	0（保留）	0（保留）	0(保留)	0（保留）	0（保留）	3Spline_TH_enable_0Delta2[9,8]	
PB11	3Spline_TH_enab le_0Delta2[7...0]							
PB12	3Spline_enable_S trength0[7...0]							
PB13	0（保留）	0（保留）	0（保留）	0(保留)	0（保留）	0（保留）	3Spline_TH_enable_model[1, 0]	
PB14	3Spline_TH_enab le_MB1[7...0]							
PB15	0（保留）	0（保留）	0（保留）	0(保留)	3Spline_TH_enable1[11...8]			

Byte\ Bit	7	6	5	4	3	2	1	0
-----------	---	---	---	---	---	---	---	---

PB16 3Spline_TH_enab
le1[7...0]

PB17	0 (保留)	0 (保留)	0 (保留)	0 (保留)	0 (保留)	0 (保留)	3Spline_TH_enable_1Delta1[9,8]	
------	--------	--------	--------	--------	--------	--------	--------------------------------	--

PB18 3Spline_TH_enab
le_1Delta1[7...0]

PB19	0 (保留)	0 (保留)	0 (保留)	0 (保留)	0 (保留)	0 (保留)	3Spline_TH_enable_1Delta2[9,8]	
------	--------	--------	--------	--------	--------	--------	--------------------------------	--

PB20 3Spline_TH_enab
le_1Delta2[7...0]

PB21 3Spline_enable_S
trength1[7...0]

PB22	0 (保留)	0 (保留)	0 (保留)	0 (保留)	0 (保留)	color_saturation_enable_num[2...0]		
------	--------	--------	--------	--------	--------	------------------------------------	--	--

PB23 color_saturation_
enable_gain0[7...
0]

PB24 color_saturation_
enable_gain1[7...
0]

PB25 color_saturation_
enable_gain2[7...
0]

PB26 color_saturation_
enable_gain3[7...
0]

PB27 color_saturation_
enable_gain4[7...
0]

HB0	0	1	1	1	1	1	1	1
HB1	0 (First)	1 (Last)	0 (保留)	0 (保留)	0 (保留)	0 (保留)	0 (保留)	0 (保留)
HB2		2 (Sequence_Index)						

PB0 color_saturation_
enable_gain5[7...
0]

PB1 color_saturation_
enable_gain6[7...
0]

PB2 color_saturation_
enable_gain7[7...
0]

PB3 graphic_

Byte\ Bit	7	6	5	4	3	2	1	0
	source_display_value[7..0]							
PB4	0 (Reservd)							
PB5	max_display_mastering_luminance, MSB							
PB6	max_display_mastering_luminance, LSB							
PB7~PB27	0 (Reservd)							

11.3.4.2. 语义

系统版本码 **version_code**

4位无符号整数。表示系统版本号,当前版本赋值为1。

图像传输特性 **transfer_character**

1位无符号整数。表示图像的传输特性, transfer_character为0, 表示传输特性为ST2084 EOTF曲线; transfer_character为1, 表示传输特性为Hybrid Log-Gamma (HLG) OETF曲线。

图像固定曲线区间参数 **graphic_source_display_value_pq**

8位无符号整数, 表示播放设备传输给显示设备的图形对应的PQ域非线性归一化内容显示亮度码值, 图形内容的色域及传输特性与视频一致, 传输特性由图像传输特性transfer_character确定。graphic_source_display_value_pq为0时, 代表没有图形传输。图像固定曲线源亮度参数graphic_source_value_pq取值为graphic_source_display_value_pq/255。使用参考11.3.4接收端适配模式显示端适配过程的图形处理。

图像主监视器最大亮度 **max_display_mastering_luminance**

16位无符号整数, 合并max_display_mastering_luminance, LSB和max_display_mastering_luminance, MSB获得。其中0x0001表示1cd/m², 0xFFFF表示65535cd/m²。其余参数参考T/UWA 005.1中动态元数据相关语法。

11.3.5. 接收端适配模式显示端适配过程的图形处理

- 接收端接收到 VS-EMDS包之后, 根据HDR Vivid VS-EMDS包数据结构, 从中提取出HDR Vivid元数据, 以及图像固定曲线区间参数;
- 参照 T/UWA 005.1-2024的第9章内容生成中间色调映射曲线P, 包含基础曲线参数P_{tone_mapping}, 包含m_p、m_m、m_n、m_a、m_b、K1、K2、K3; 一次样条曲线参数P_{1spline}, 包含MB[0][0]和TH3[0]; 三次样条曲线参数P_{3spline}, 包含TH1[3Spline_num]、TH2[3Spline_num]、TH3[3Spline_num]、MA[2][3Spline_num]、MB[2][3Spline_num]、MC[2][3Spline_num]、MD[2][3Spline_num];
- 若graphic_source_value_pq=0, 则将最终色调映射曲线S设置为P;
- 否则, 若graphic_source_value_pq>0, 则图形显示的目标亮度为
$$graphic_target_value_pq = \max(\min(graphic_source_value_pq, MaxDisplayPQ \times 0.9), 0.5081)$$
 将S中的一次样条曲线参数S_{1spline}, 包含MB[0][0]和TH3[0]设置为:

MB[0][0] = graphic_target_value_pq / graphic_source_value_pq;

TH3[0] = graphic_source_value_pq;

base_offset=0

将S中的基础曲线参数 $S_{\text{tone_mapping}}$, 包含 s_m_p 、 s_m_m 、 s_m_n 、 s_m_a 、 s_m_b 、 s_K1 、 s_K2 、 s_K3 设置为基础曲线参数 $P_{\text{tone_mapping}}$, 包含 m_p 、 m_m 、 m_n 、 m_a 、 m_b 、 $K1$ 、 $K2$ 、 $K3$;

参照T/UWA 005.1-2024中9.3节, 按照如下步骤生成三次样条曲线参数 $S_{3\text{spline}}$,

步骤1: 设置 $TH1[1] = TH3[0]$, 并计算 $TH2[1] = TH1[1] + B$, $TH3[1] = TH2[1] + C$, 其中 B 默认值为0.15, C 为 $B/2$; 若 $TH3[1] > \text{MaxSource}$, 则 $TH3[1] = \text{MaxSource}$; $TH2[1] = TH1[1] + 2 \times (TH3[1] - TH1[1]) / 3$; 然后通过T/UWA 005.1-2024中9.3.3.1或9.3.3.2生成三次样条曲线参数 $S_{3\text{spline}}$, 并判断该三次样条曲线是否单调递增 (见11.3.6), 若是, 则执行步骤2; 若否, 则执行步骤3;

步骤2: 当 $TH1[3\text{Spline_num}]$ 大于 $TH3[1]$ 时, 将该三次样条曲线参数 $P_{3\text{spline}}$, 包含 $TH1[3\text{Spline_num}]$ 、 $TH2[3\text{Spline_num}]$ 、 $TH3[3\text{Spline_num}]$ 、 $MA[2][3\text{Spline_num}]$ 、 $MB[2][3\text{Spline_num}]$ 、 $MC[2][3\text{Spline_num}]$ 、 $MD[2][3\text{Spline_num}]$ 赋值给三次样条曲线参数 $S_{3\text{spline}}$; 否则, $TH1[2]$ 、 $TH2[2]$ 、 $TH3[2]$ 、以及第2条三次样条参数均为0, 并将metadata中 3spline_num 设为1, 结束三次样条曲线参数生成过程。

步骤3: 设置 $TH3[1] = \text{MaxSource}$; $TH2[1] = TH1[1] + 2 \times (TH3[1] - TH1[1]) / 3$, 生成三次样条曲线参数 $S_{3\text{spline}}$, 并判断此时三次样条曲线是否单调递增 (见11.3.6): 若是, 则执行步骤4; 若否, 则执行步骤5。

步骤4: 用二分查找法在 $[TH1[1] + B + C, \text{MaxSource}]$ 中查找一个最小的 $TH3$, 通过T/UWA 005.1-2024中9.3.3.1或9.3.3.2生成三次样条曲线参数 $S_{3\text{spline}}$, 使曲线单调递增 (见11.3.6)。最多查找10次, 结束三次样条曲线参数生成过程。

步骤5: $TH3[1] = \text{MaxSource}$, $TH2[1] = TH3[1]$,

$MA[0][1] = \text{graphic_target_value_pq}$, $MB[0][1] = (VA2 - MA[0][1]) / (TH2[1] - TH1[1])$, $MC[0][1]$, $MD[0][1]$, $MA[1][1]$, $MB[1][1]$, $MC[1][1]$, $MD[1][1]$ 均为0, 并结束三次样条曲线参数生成过程。

11.3.6. 判断三次样条曲线是否单调递增的过程

通过11.3.5的调节, 可能引起映射曲线的单调性失效。为此播放设备需按照本条方式进行检查和调整。

一个三次样条区间包含两个三次样条曲线, 见公式(1)~公式(2):

$$F(L) = MD[0][n] \times (L - TH1[n])^2 + MC[0][n] \times (L - TH1[n])^2 + MB[0][n] \times (L - TH1[n])^2 + MA[0][n] \cdots (1)$$

其中 L 为区间 $[TH1[n], TH2[n]]$ 中的自变量,

$$F(L) = MD[1][n] \times (L - TH2[n])^2 + MC[1][n] \times (L - TH2[n])^2 + MB[1][n] \times (L - TH2[n])^2 + MA[1][n] \cdots (2)$$

其中 L 为区间 $[TH2[n], TH3[n]]$ 中的自变量, $0 < n \leq 3\text{Spline_num}$, 用于指示三次样条所属区间。

需要同时满足两个三次样条曲线都单调递增, 具体判断条件如下,

a) 对任一样条区间 n , 第一段三次样条曲线单调递增时要满足的条件为:

条件一:

$$TH1[n] \leq (TH1[n] - \frac{MC[0][n]}{3 \times MD[0][n]}) \leq TH2[n] \text{ 且 } MB[0][n] - \frac{MC[0][n]^2}{3 \times MD[0][n]} \geq 0$$

条件二:

$$\begin{aligned} & TH1[n] > (TH1[n] - \frac{MC[0][n]}{3 \times MD[0][n]}) \text{ 且} \\ & \begin{cases} MB[0][n] \geq 0 \\ 3 \times MD[0][n] \times (DTH2)^2 + 2 \times MC[0][n] \times DTH2 + MB[0][n] \geq 0 \end{cases} \end{aligned}$$

条件三:

$$\left(TH1[n] - \frac{MC[0][n]}{3 \times MD[0][n]}\right) > TH2[n] \text{ 且}$$

$$\begin{cases} MB[0][n] \geq 0 \\ 3 \times MD[0][n] \times (DTH2)^2 + 2 \times MC[0][n] \times DTH2 + MB[0][n] \geq 0 \end{cases}$$

其中: $DTH2 = (TH2[n] - TH1[n])$

b)对任一样条区间n, 第二段三次样条曲线单调递增时要满足的条件为:

$$\text{条件一: } TH2[n] \leq \left(TH2[n] - \frac{MC[1][n]}{3 \times MD[1][n]}\right) \leq TH3[n] \text{ 且 } MB[1][n] - \frac{MC[1][n]^2}{3 \times MD[1][n]} \geq 0$$

条件二:

$$TH2[n] > \left(TH2[n] - \frac{MC[1][n]}{3 \times MD[1][n]}\right) \text{ 且}$$

$$\begin{cases} MB[1][n] \geq 0 \\ 3 \times MD[1][n] \times (DTH3)^2 + 2 \times MC[1][n] \times DTH3 + MB[1][n] \geq 0 \end{cases}$$

条件三:

$$\left(TH2[n] - \frac{MC[1][n]}{3 \times MD[1][n]}\right) > TH3[n] \text{ 且}$$

$$\begin{cases} MB[1][n] \geq 0 \\ 3 \times MD[1][n] \times (DTH3)^2 + 2 \times MC[1][n] \times DTH3 + MB[1][n] \geq 0 \end{cases}$$

其中: $DTH3 = (TH3[n] - TH2[n])$ 。

11.4.GPMI 元数据传输要求

11.4.1.通则

HDR Vivid播放设备与HDR Vivid显示设备通过GPMI进行连接和图像数据传输时, 应当按照11.4要求进行元数据传输。

基于GPMI传输的HDR Vivid信号, 信号的电-光传输特性由GPMI视频元数据描述信息报文(Descriptive Information Packet, 简称DIP)中transfer_character字段确定, 信号的色域按照ITU-R BT.2020的要求。

当HDR Vivid播放设备与HDR Vivid显示设备建立连接后, 播放设备通过查询显示设备的GPMI设备综合能力描述(Device Comprehensive Capability Description, 简称DCCD)信息中符合8.3.2规定的数据块来了解显示设备对HDR Vivid的支持能力。

当显示设备的DCCD表明能够支持HDR Vivid监视器适配模式时, 播放设备应根据显示设备DCCD中提供的最大显示亮度和最小显示亮度信息, 将HDR Vivid的视频内容根据T/UWA 005.1-2024第9章的规定进行适配处理后发送给显示设备, 并同时向显示设备发送11.4.3规定的DIP信息。

当显示设备的DCCD表明能够支持HDR Vivid接收端适配模式时, 或者同时支持HDR Vivid接收端适配模式和HDR Vivid监视器适配模式这两种模式时, 播放设备应将HDR Vivid的动态元数据封装在符合11.4.4规定的DIP信息中, 配合HDR Vivid视频图像发送给显示设备。

11.4.2.HDR Vivid 能力描述

11.4.2.1. 语法

HDR Vivid的能力应按照表16中的DCCD数据块描述。其中字节0至字节2, 为GPMI关于动态HDR能力描述所确定的信息与内容。字节3至字节27则是HDR Vivid所需提供的信息。

表 16 DCCD 数据块信息语法

字节\比特	7	6	5	4	3	2	1	0
0	0x1C (HDR Vivid Capabilities)							
1	0 (Reserved)			Version_code (=0x01)				
2	Length (=25)							
3	system_start_code[7...0]							
4	version_code[3..0]				0(保留)	0(保留)	0(保留)	0(保留)
5	display_maximum_luminance [7...0]							
6	display_maximum_luminance [15...8]							
7	display_maximum_luminance [23...16]							
8	display_maximum_luminance [31...24]							
9	display_minimum_luminance [7...0]							
10	display_minimum_luminance [15...8]							
11	monitor_mode_support	rx_mode_support	0（保留）					
12~27	0（保留）							

11.4.2.2. 语义

系统起始码 system_start_code

8位无符号整数。表示终端支持的HDR Vivid版本号,当前版本赋值为1。

如果显示设备DCCD中版本号高于或等于码流中接收到的动态元数据所包含的system_start_code,则可按照8.3.4规定的DIP包数据结构将动态元数据进行打包传输给显示设备;否则应在播放设备上显示适配处理,然后将进行过适配处理之后的图像和按照8.3.3规定的DIP包数据结构将元数据信息发送到显示设备。

系统版本码 version_code

4位无符号整数。表示系统版本号,当前版本赋值为1。

设备最高显示亮度 display_maximum_luminance

32位无符号整数,表示显示设备在D65色温下可以显示的最大亮度,单位是0.0001cd/m²。当码字为0x00000001时,表示0.0001cd/m²。MaxDisplay最终取值为display_maximum_luminance /10000。当显示设备支持监视器适配模式时,显示设备需要为display_maximum_luminance设置适当数值,以便播放设备使用该信息进行图像处理;当显示设备不支持监视器适配模式时,应该将display_maximum_luminance设置为0。

设备最小显示亮度 display_minimum_luminance

16位无符号整数,表示显示设备在D65色温下可以显示的最小亮度,单位是0.0001cd/m²。当码字为0x0001时,表示0.0001cd/m²。MinDisplay最终取值为display_minimum_luminance /10000。当显示设备支持监视器适配模式时,显示设备需要为display_minimum_luminance设置适当数值,以便播放设备使用该信息进行图像处理;当显示设备不支持监视器适配模式时,应该将display_minimum_luminance设置为0。

监视器适配模式支持标识monitor_mode_support

1位无符号整数。表示显示设备对监视器适配模式的支持能力,monitor_mode_support=1表示显示设备支持监视器适配模式,monitor_mode_support=0表示显示设备不支持监视器适配模式。

接收端适配模式支持标识rx_mode_support

1位无符号整数。表示显示设备对接收端适配模式的支持能力,rx_mode_support=1表示显示设备支

持接收端适配模式，rx_mode_support=0表示显示设备不支持接收端适配模式。

11.4.3.静态元数据传输

11.4.3.1.语法

GPMI传输中，在HDR Vivid监视器适配模式下，采用DIP从播放设备向显示设备传输静态元数据。HDR Vivid静态元数据DIP报文应符合表17中的规定。其中HB0至HB3是GPMI所确定的HDR静态元数据DIP报文头信息，CRC0至CRC3则是HDR静态元数据DIP报文的循环校验信息，在本条中均为固定值。

表17 静态元数据DIP报文格式

字节\比特	7	6	5	4	3	2	1	0
HB0	DIP_Type (=0x04)							
HB1	0 (保留)			Index (=0)				
HB2	OUI (= 0001b)			Meta_Type (=0000b)				
HB3	Length (= 0x5)							
DB0	0 (保留)							
DB1	0 (保留)							
DB2	0 (保留)							
DB3	system_start_code[7...0]							
DB4	version_code[3...0]			monitor _mode_ enable (=1)	transfer_ character (=0)	0(保留)		0(保留)
DB5~DB31	0 (保留)							
CRC0	CRC32[31:24]							
CRC1	CRC32[23:16]							
CRC2	CRC32[15:8]							
CRC3	CRC32[7:0]							

11.4.3.2.语义

系统版本码 **version_code**

4位无符号整数。表示系统版本号,当前版本赋值为1。

监视器适配模式支持标识**monitor_mode_enable**

1位无符号整数。本文件中monitor_mode_support固定为1，表示此时的HDR Vivid信号的互联适配处理方式为监视器适配模式。

图像传输特性 **transfer_character**

1位无符号整数。表示图像的传输特性，在使用本条规范传输HDR Vivid静态元数据时，要求transfer_character固定为0，表示监视器适配模式下，图像传输特性为SMPTE ST 2084 EOTF曲线。

循环校验码 **CRC32**

32bit无符号整数，用于校验整个报文从HB0校DB31验内信息的准确性。CRC的计算方法按照8.3.5

的要求。

11.4.4.动态元数据传输

11.4.4.1.语法

GPMI传输中，在HDR Vivid 接收端适配模式下，采用DIP报文从播放设备向显示设备传输动态元数据。HDR Vivid动态元数据DIP报文应符合表19和表20中的规定。其中HB0至HB3是GPMI所确定的HDR动态元数据DIP报文头信息，在本章节中为固定值。CRC0至CRC3则是HDR静态元数据DIP报文的循环校验信息，应根据DIP报文实际内容生成。

由于HDR Vivid动态元数据大于32字节，需要使用2个元数据DIP报文才能完整传输。第1个HDR Vivid动态元数据DIP报文见表18。第2个HDR Vivid动态元数据DIP报文见表20。

表18 第1个动态元数据DIP报文格式

字节\比特	7	6	5	4	3	2	1	0
HB0	DIP_Type (=0x04)							
HB1	0 (Reserved)				Index (=0)			
HB2	OUI (= 0001b)				Meta_Type (=0001b)			
HB3	Length (= 0x20)							
DB0	0 (保留)							
DB1	0 (保留)							
DB2	0 (保留)							
DB3	system_start_code[7...0]							
DB4	version_code[3...0]				minimum_maxrgb_pq [11...8]			
DB5	minimum_maxrgb_pq [7...0]							
DB6	0 (保留)	0 (保留)	0 (保留)	0 (保留)	average_maxrgb_pq [11...8]			
DB7	average_maxrgb_pq [7...0]							
DB8	0 (保留)	0 (保留)	0 (保留)	0 (保留)	variance_maxrgb_pq [11...8]			
DB9	variance_maxrgb_pq [7...0]							

表18（续）

字节\比特	7	6	5	4	3	2	1	0
DB10	0 (保留)	0 (保留)	0 (保留)	0 (保留)	maximum_maxrgb_pq [11...8]			
DB11	maximum_maxrgb_pq [7...0]							
DB12	0 (保留)	0 (保留)	0 (保留)	0 (保留)	targeted_system_display_maximum_luminance_pq[11...8]			
DB13	targeted_system_display_maximum_luminance_pq[7...0]							
DB14	transfer_character	base_enab leflag[0]	base_param_m_p[13...8]					
DB15	base_param_m_p[7...0]							
DB16	0 (保留)	0 (保留)	base_param_m_m [5...0]					
DB17	0 (保留)	0 (保留)	0 (保留)	0 (保留)	0 (保留)	0 (保留)	base_param_m_a[9,8]	
DB18	base_param_m_a[7...0]							

DB19	0 (保留)	0 (保留)	0 (保留)	0 (保留)	0 (保留)	0 (保留)	base_param_m_b[9,8]
DB20	base_param_m_b[7...0]						
DB21	0 (保留)	0 (保留)	base_param_m_n[5...0]				
DB22	0 (保留)	0 (保留)	base_param_K1[1..0]		base_param_K2[1..0]		base_param_K3[1..0]
DB23	0 (保留)	0 (保留)	0 (保留)	0 (保留)	0 (保留)	base_param_Delta_enable_mode[2..0]	
DB24	0 (保留)	base_param_enable_Delta[6...0]					
DB25	0 (保留)	0 (保留)	0 (保留)	3Spline_enable_num		3Spline_enable_flag[0]	3Spline_TH_enable_mode0[1,0]
DB26	3Spline_TH_enable_MB0[7...0]						
DB27	0 (保留)	0 (保留)	0 (保留)	0 (保留)	3Spline_TH_enable0[11...8]		
DB28	3Spline_TH_enable0[7...0]						
DB29	0 (保留)	0 (保留)	0 (保留)	0 (保留)	0 (保留)	0 (保留)	3Spline_TH_enable_0Delta1[9,8]
DB30	3Spline_TH_enable_0Delta1[7...0]						
DB31	0 (保留)	0 (保留)	0 (保留)	0 (保留)	0 (保留)	0 (保留)	3Spline_TH_enable_0Delta2[9,8]
CRC0	CRC[31:24]						
CRC1	CRC[23:16]						
CRC2	CRC[15:8]						
CRC3	CRC[7:0]						

表19 第2个动态元数据DIP报文格式

字节\比特	7	6	5	4	3	2	1	0
HB0	DIP_Type (=0x04)							
HB1	0 (Reserved)				Index (=1)			
HB2	OUI (= 0001b)				Meta_Type (=0001b)			
HB3	Length (= 0x1A)							
DB0	3Spline_TH_enable_0Delta2[7...0]							
DB1	3Spline_enable_Strength0[7:0]							
DB2	0 (保留)	0 (保留)	0 (保留)	0 (保留)	0 (保留)	0 (保留)	3Spline_TH_enable_model1[1,0]	
DB3	3Spline_TH_enable_MB1[7...0]							
DB4	0 (保留)	0 (保留)	0 (保留)	0 (保留)	3Spline_TH_enable1[11...8]			
DB5	3Spline_TH_enable1[7...0]							
DB6	0 (保留)	0 (保留)	0 (保留)	0 (保留)	0 (保留)	0 (保留)	3Spline_TH_enable_1Delta1[9,8]	
DB7	3Spline_TH_enable_1Delta1[7...0]							

DB8	0 (保留)	0 (保留)	0 (保留)	0 (保留)	0 (保留)	0 (保留)	3Spline_TH_enable_1Delta2[9,8]
DB9	3Spline_TH_enable_1Delta2[7...0]						
DB10	3Spline_enable_Strength1[7...0]						
DB11	0 (保留)	0 (保留)	0 (保留)	0 (保留)	0 (保留)	color_saturation_enable_num[2...0]	
DB12	color_saturation_enable_gain0[7...0]						
DB13	color_saturation_enable_gain1[7...0]						
DB14	color_saturation_enable_gain2[7...0]						
DB15	color_saturation_enable_gain3[7...0]						
DB16	color_saturation_enable_gain4[7...0]						
DB17	color_saturation_enable_gain5[7...0]						
DB18	color_saturation_enable_gain6[7...0]						
DB19	color_saturation_enable_gain7[7...0]						
DB20	graphic_source_display_value_pq[7...0]						
DB21	0 (保留)						
DB22	max_display_mastering_luminance,MSB						
DB23	max_display_mastering_luminance,LSB						
DB24	min_display_mastering_luminace,MSB						
DB25	min_display_mastering_luminace,LSB						
DB26	0 (保留)						
DB27	0 (保留)						
DB28	0 (保留)						
DB29	0 (保留)						
DB30	0 (保留)						
DB31	0 (保留)						
CRC0	CRC32[31:24]						
CRC1	CRC32[23:16]						
CRC2	CRC32[15:8]						
CRC3	CRC32[7:0]						

11.4.4.2.语义

系统版本码 **version_code**

4位无符号整数。表示系统版本号,当前版本赋值为1。

图像传输特性 **transfer_character**

1位无符号整数。表示图像的传输特性,transfer_character为0,表示传输特性为SMPTE ST 2084 EOTF曲线;

transfer_character为1,表示传输特性为HLG OETF曲线。

图像固定曲线区间参数 **graphic_source_display_value_pq**

8位无符号整数,表示播放设备传输给显示设备的图形对应的PQ域非线性归一化内容显示亮度码值,图形内容的色域及传输特性与视频一致,传输特性由图像传输特性transfer_character确定。graphic_source_display_value_pq 为0时,代表没有图形传输。图像固定曲线源亮度参数graphic_source_value_pq取值为graphic_source_display_value_pq/255。使用参考8.3.4.3接收端适配模式显

示端适配过程的图形处理。

图像主监视器最大亮度 **max_display_mastering_luminance**

16 位 无 符 号 整 数 ， 合 并 **max_display_mastering_luminance_LSB** 和 **max_display_mastering_luminance_MSB** 获得。其中 0x0001 表示 1nit， 0xFFFF 表示 65535cd/m²。

图像主监视器最小亮度 **min_display_mastering_luminance**

16 位 无 符 号 整 数 ， 合 并 **min_display_mastering_luminance_LSB** 和 **min_display_mastering_luminance_MSB** 获得。其中 0x0001 表示 0.0001cd/m²， 0xFFFF 表示 6.5535cd/m²。

循环校验码 **CRC32**

32bit 无符号整数，用于校验整个报文从 HB0 到 DB31 所有信息的准确性。CRC32 的计算方法按照 8.3.5 的要求。

其余参数见 T/UWA 005.1-2024 的 7.3。

11.4.4.3. 接收端适配模式显示端适配过程的图形处理

在接收端适配模式下，播放设备上为实现音量调节等用户交互功能，可能需要叠加用户菜单等图形内容在 HDR Vivid 视频上传送给显示设备。为避免出现这种叠加的图形内容随动态元数据处理而出现亮度变化，需要采用本条规定的方式来调整。接收端适配模式显示端适配过程的图形处理方式如下：

- a) 接收端接收到动态元数据包之后，根据 HDR Vivid 元数据包数据结构，从中提取出 HDR Vivid 元数据，以及图像固定曲线区间参数；
- b) 参照 11.3.5 进行图形处理。

11.4.4.4. 判断三次样条曲线是否单调递增的过程

参照 11.3.6 判断三次样条曲线是否单调递增的过程。

11.4.5.GPMI DIP 报文循环冗余校验码 CRC32 的计算方式

CRC32 生成校验结果的关键参数如下：

- a) CRC32 生成多项式：

$$g(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$$

- b) 初始值应设为 0xFFFF_FFFFh；
- c) 输入数据应反转（设置为 True）；
- d) 输出数据应反转（设置为 True）；
- e) 输出校验码的异或值采用 0xFFFF_FFFFh。

CRC32 生成电路见图 5，图中 DW 是 32 比特双字。CRC32 生成电路在被校验的所有比特输入后产生校验域 CRC[31:0]。

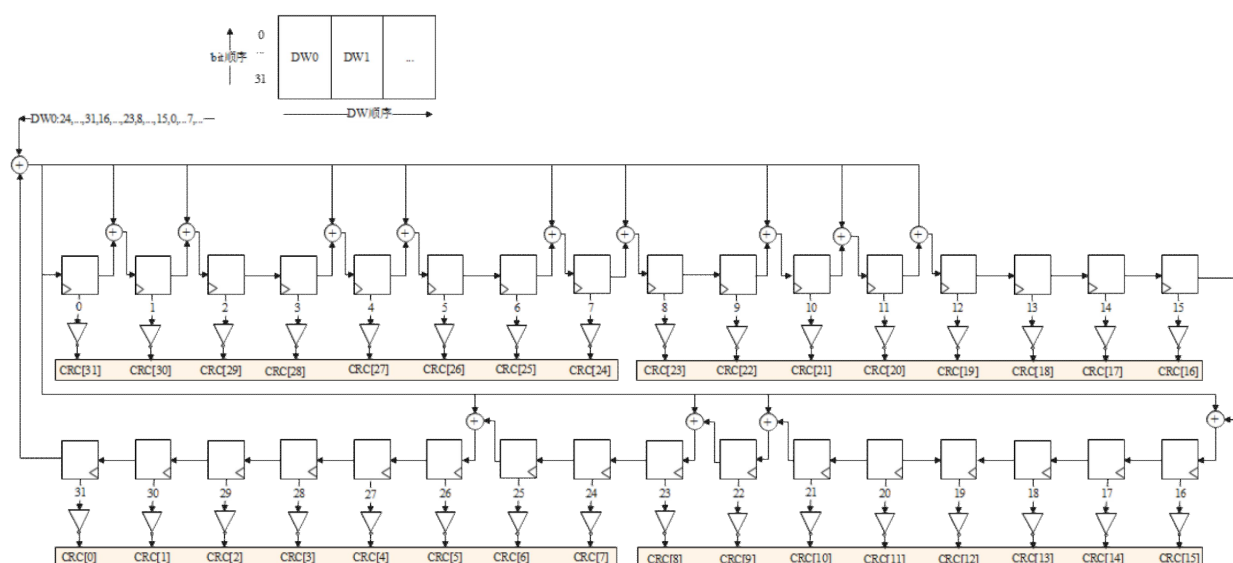


图5 CRC32 生成电路

12.元数据生成

12.1.输入输出

HDR Vivid元数据宜在内容制作环节中生成，包括动态元数据的提取、调节和文件的封装等。

内容制作输入应为ITU-R BT.2020色域。可以是线性域的RGB信号，PQ域的RGB信号，或者PQ域的YUV信号。线性域位深宜不小于16bit，PQ域位深宜不小于12bit。YUV的采样格式宜使用YUV444或者YUV422。

内容制作输出应为PQ域的YUV420、YUV422或者YUV444的10bit位深的视频数据加上动态元数据。若输出目的为直接分发，宜输出指定编码格式的码流文件，将动态元数据嵌在ES码流中。若输出目的为归档，建议输出浅压缩的编码格式，如XAVC。动态元数据宜嵌在每帧视频ES数据中。

12.2.画面质量监控

后期制作需要支持两路显示输出：

——第一路，视频源需要输出到专业的监视器上显示，作为高质量的参考。

——第二路，采用HDR Vivid动态元数据对视频进行后处理（色调映射），并将后处理的结果输出到消费级终端设备显示，如一台500 nit的HDR Vivid电视，作为终端显示器的参考。

12.3.动态元数据生成模式

动态元数据的生成包括两种模式：自动模式和导演模式。

12.3.1.自动模式

自动模式对视频的每一帧提取HDR Vivid动态元数据，进行HDR Vivid实时后处理，将结果输出到显示设备播放。自动模式宜集成 T/UWA 005.1-2024中附录A规定的动态元数据自动提取算法和后处理算法。

自动模式的界面应支持视频源的读入和播放、动态元数据的提取、HDR Vivid后处理以及后处理结果的输出显示。

12.3.2.导演模式

12.3.2.1.概述

导演模式的目的是帮助调色师手动调节每一个场景或者每一帧的动态元数据,达到调色师最满意的效果。基本要求是提供HDR Vivid色调映射曲线的调节控件,并且在调色师调节的过程中可显示当前的色调映射曲线,同时可以基于当前的动态元数据或者曲线进行实时的后处理并输出显示。

导演模式需要集成自动模式的动态元数据提取算法、后处理算法和导演模式曲线调节算法。

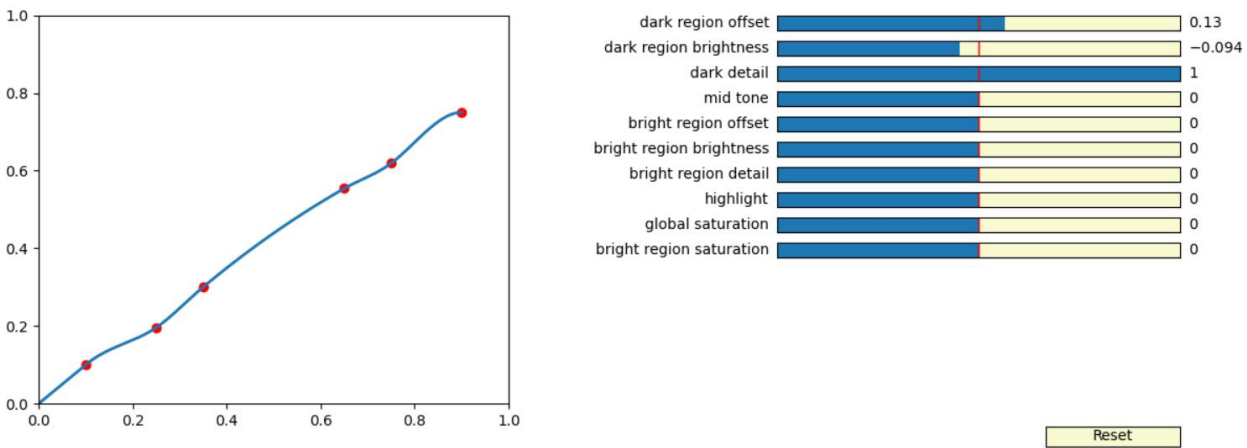


图4 导演模式界面参考

12.3.2.2.导演模式组成

导演模式应符合以下要求:

a) 运行导演模式前,必须先运行自动模式。自动模式提取的动态元数据,和对应的色调映射曲线,应当作为导演模式的参照。假如调色师没有做出任何调节,那么导演模式输出应当等同于自动模式的输出。或者,调色师在调节过程中,恢复默认设置,那么动态元数据和色调映射曲线需要回到自动模式的状态。调色师调节完毕之后,用调色师调节过的动态元数据替代自动模式的动态元数据,并写入到输出文件。

b) 控件的数量、调节范围和调节的功能要符合 T/UWA 005.1。控件数量宜为 10 个,类型宜为滑块,每个控件的调节范围应为-1.0~1.0,默认值是 0 (对应自动模式)。当所有控件的值为 0 时,动态元数据和色调映射曲线应与自动模式一致。

12.3.2.3.曲线锚点

HDR Vivid曲线由6个锚点决定,宜在界面中画出这6个锚点的位置。导演模式的控件通过调节6个锚点的位置,计算HDRVivid曲线并生成对应的动态元数据。这6个锚点的x值从小到大依次就是T/UWA 005.1定义的TH1、TH2、TH3、TH1_HIGH、TH2_HIGH、TH3_HIGH。 纵坐标就是HDR Vivid曲线在这六个位置的y值,表示为TH1_Y、 TH2_Y、 TH3_Y、TH1_HIGH_Y、TH2_HIGH_Y、TH3_HIGH_Y。

导演模式以自动模式为基础,自动模式对应的锚点值表示为TH1_ref、TH2_ref、TH3_ref、TH1_HIGH_ref、TH2_HIGH_ref、 TH3_HIGH_ref、 TH1_Y_ref、 TH2_Y_ref、 TH3_Y_ref、TH1_HIGH_Y_ref、TH2_HIGH_Y_ref、 TH3_HIGH_Y_ref。

曲线调节算法负责把控件的值映射到锚点的位置,进而计算出曲线参数和动态元数据。6个锚点的位置是相互依赖的,曲线调节算法会最大程度的协调6个锚点的位置,力求生成符合T/UWA 005.1的曲线。但是,在某些极端情况下,不存在一条符合T/UWA 005.1的曲线,曲线调节算法会反馈拟合失败,控件、曲线和动态元数据应返回到上一次的正常状态。

12.3.2.4.控件

控件是指在软件中，用于实现特定功能和用户界面交互的可复用软件模块。各控件的功能和对应的锚点位置以及动态元数据如表20所述。

表20 控件描述

编号	控件	描述
1	暗区偏移	<p>调节TH3的值，对应动态元数据里定义的TH3。</p> <p>控件值为0时，TH3的值等于自动模式下的TH3_ref；-1时，TH3取调节范围的最小值；+1时，取调节范围的最大值。调节范围建议采用[TH2_ref,0.45]。</p> <p>小于TH3的区域被定义为暗区。TH1、TH2、TH3这三个锚点确定暗区的三次样条曲线。调节TH3的值会改变暗区三次样条的覆盖范围。主曲线由TH3和TH1_HIGH这两个锚点确定的，当改变TH3时，应重新计算主曲线的参数，包括动态元数据里定义的m_p, m_a, m_b, m_m, m_n, K1, K2, K3。</p>
2	暗区亮度	<p>调节TH1_Y的值，对应动态元数据里定义的MB的值，即曲线开始处一次样条的斜率，会影响图像最暗部分的亮度。</p> <p>控件取值为0时，MB取自动模式下值MB_ref；为-1时，取调节范围的最小值；为+1时，最大值。调节范围建议[0,1]。</p>
3	暗部细节	<p>调节TH2_Y的值，对应动态元数据里定义的strength，影响暗区三次样条曲线的形状，进而影响暗区的细节表现。</p> <p>控件值取0时，strength取自动模式下的值strength_ref；-1时，取调节范围的最小值；+1时，最大值。建议的取值范围是[-0.5,0.5]。</p>
4	中灰亮度	<p>调节TH3_Y的值。TH3_Y定义为中灰亮度，没有直接对应的动态元参数，但是会影响到动态元数据里定义的m_p, m_a, m_b, m_m, m_n, K1, K2, K3。</p> <p>控件值取0时，TH3_Y取自动模式下的值TH3_Y_ref；-1时，取调节范围的最小值；+1时，最大值。调节范围建议采用[TH2_Y_ref,TH1_HIGH_Y_ref]。</p> <p>改变TH3_Y的值，需要重新计算主曲线参数，包括动态元数据里定义的m_p, m_a, m_b, m_m, m_n, K1, K2, K3。假如TH3_Y调到某一位置以后，计算不出合规的CUVA曲线，那么触发联动机制，即在TH3_Y调节的相同方向上，搜索新的TH1_HIGH_Y的值，直到可以计算出合规的CUVA曲线。假如失败了，则说明当前调节位置不合法，退回上一个合法的状态和控件值。</p>
5	亮区偏移	<p>调节TH1_HIGH的值，对应动态元数据里定义的TH1_HIGH。</p> <p>控件值为0时，TH1_HIGH的值等于自动模式下的TH1_HIGH_ref；-1时，TH1_HIGH取调节范围的最小值；+1时，取调节范围的最大值。调节范围建议采用[0.48,TH2_HIGH_ref]。</p> <p>大于TH1_HIGH的部分定义为亮区。TH1_HIGH、TH2_HIGH、TH3_HIGH这三个锚点确定暗区的三次样条曲线。调节TH1_HIGH的值会改变亮区三次样条的覆盖范围。另外，主曲线是由TH3和TH1_HIGH这两个锚点确定的，所以改变TH1_HIGH,需要重新计算主曲线的参数，包括动态元数据里定义的m_p, m_a, m_b, m_m, m_n, K1, K2, K3。</p>
6	亮区亮度	<p>调节TH1_HIGH_Y的值，没有直接对应的动态元参数，但是会影响到动态元数据里定义的m_p, m_a, m_b, m_m, m_n, K1, K2, K3。</p> <p>控件值取0时，TH1_HIGH_Y取自动模式下的值TH1_HIGH_Y_ref；-1时，取调节范围的最小值；+1时，最大值。调节范围建议采用[TH3_Y_ref,TH2_HIGH_Y_ref]。</p> <p>改变TH1_HIGH_Y的值，需要重新计算主曲线参数，包括动态元数据里定义的m_p, m_a, m_b, m_m, m_n, K1, K2, K3。假如TH1_HIGH_Y调到某一位置以后，计算不出合规的CUVA曲线，那么触发联动机制，即在TH1_HIGH_Y调节的相同方向上，搜索新的TH3_Y的值，直到可以计算出合规的CUVA曲线。假如失败了，则说明当前调节位置不合法，退回上一个合法的状态和控件值。</p>

编号	控件	描述
7	亮区 细节	<p>调节的是TH2_HIGH_Y的值，对应动态元数据里定义的strength_HIGH，影响亮区三次样条曲线的形状，进而影响亮区的细节表现。</p> <p>控件值取0时，strength_HIGH取自动模式下的值strength_HIGH_ref；-1时，取调节范围的最小值；+1时，最大值。建议的取值范围是[-0.5,0.5]。</p>
8	高光	<p>调节的是TH3_HIGH的值，对应动态元数据里定义的TH3_HIGH。</p> <p>控件值取0时，TH3_HIGH等于自动模式的值TH3_HIGH_ref；-1时，取调节范围的最小值；+1时，最大值。建议的取值范围是[TH2_HIGH,max_source],max_source表示当前帧的最大亮度的PQ值。</p> <p>TH3_HIGH_Y在导演模式下，一定等于参考显示器的最大亮度，所以大于TH3_HIGH的区域属于过曝区域。这个控件就是用来调节过曝区域的范围，以控制高光的效果。</p>
9	整体 饱和度	<p>调节的是整个亮度范围的饱和度，对应动态元参数里定义的color_saturation_gain[0]。</p> <p>控件取值为0时，对应自动模式下color_saturation_gain[0]的值；-1时，取最小值0；+1时，最大值255。</p>
10	亮区 饱和度	<p>针对高亮区域调节饱和度，对应动态元参数里定义的color_saturation_gain[1]。</p> <p>控件取值为0时，对应自动模式下color_saturation_gain[1]的值；-1时，取最小值0；+1时，最大值255。</p>

13.内容保护

HDR Vivid的视频流可以被正常的数字版权保护系统加密和解密。因其关联的元数据并不能还原或者反向推导出视频流，所以不宜对HDR的元数据进行额外的内容保护，增加系统复杂度。