

# Advanced Media Framework – AV1 Video Encoder

---

## Programming Guide

---

### Disclaimer

The information contained herein is for informational purposes only, and is subject to change without notice. While every precaution has been taken in the preparation of this document, it may contain technical inaccuracies, omissions and typographical errors, and AMD is under no obligation to update or otherwise correct this information.

Advanced Micro Devices, Inc. makes no representations or warranties with respect to the accuracy or completeness of the contents of this document, and assumes no liability of any kind, including the implied warranties of noninfringement, merchantability or fitness for particular purposes, with respect to the operation or use of AMD hardware, software or other products described herein. No license, including implied or arising by estoppel, to any intellectual property rights is granted by this document. Terms and limitations applicable to the purchase or use of AMD's products are as set forth in a signed agreement between the parties or in AMD's Standard Terms and Conditions of Sale.

AMD, the AMD Arrow logo, ATI Radeon™, CrossFireX™, LiquidVR™, TrueAudio™ and combinations thereof are trademarks of Advanced Micro Devices, Inc. Other product names used in this publication are for identification purposes only and may be trademarks of their respective companies.

Windows™, Visual Studio and DirectX are trademark of Microsoft Corp.

---

### Copyright Notice

© 2022 Advanced Micro Devices, Inc. All rights reserved

Notice Regarding Standards. AMD does not provide a license or sublicense to any Intellectual Property Rights relating to any standards, including but not limited to any audio and/or video codec technologies such as MPEG-2, MPEG-4; AVC/H.264; HEVC/H.265; AV1; AAC decode/FFMPEG; AAC encode/FFMPEG; VC-1; and MP3 (collectively, the "Media Technologies"). For clarity, you will pay any royalties due for such third party technologies, which may include the Media Technologies that are owed as a result of AMD providing the Software to you.

### MIT license

Copyright (c) 2022 Advanced Micro Devices, Inc. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

---

## Contents

1. Introduction
  - 1.1 Scope
  - 1.2 Pre-defined Encoder Usages
2. AMF Video Encoder VCN-AV1 Component
  - 2.1 Input Submission and Output Retrieval
  - 2.2 Encode Parameters
    - 2.2.1 Static Properties
    - 2.2.2 Dynamic Properties
    - 2.2.3 Frame Per-Submission Properties
    - 2.2.4 ROI Feature 6
    - 2.2.5 SVC Properties
    - 2.2.6 LTR Properties
3. Sample Applications
  - 3.1 List of Parameters
  - 3.2 Command line example
    - 3.2.1 Transcoding application (TranscodeHW.exe)
    - 3.2.2 D3D application (VCEEncoderD3D.exe)
4. Annex A: Encoding & frame parameters description
  - Table A-1. Encoder parameters
  - Table A-2. Input frame and encoded data parameters
  - Table A-3. Encoder capabilities exposed in AMFCaps interface

## 1 Introduction

---

### 1.1 Scope

This document provides a complete description of the AMD Advanced Media Framework (AMF) Video Encoder Component. This component exposes the AMD Video Compression Engine, which provides hardware accelerated AV1 video encoding functionality.

Figure 1 provides a system overview of the AMF Video Encoder Component.

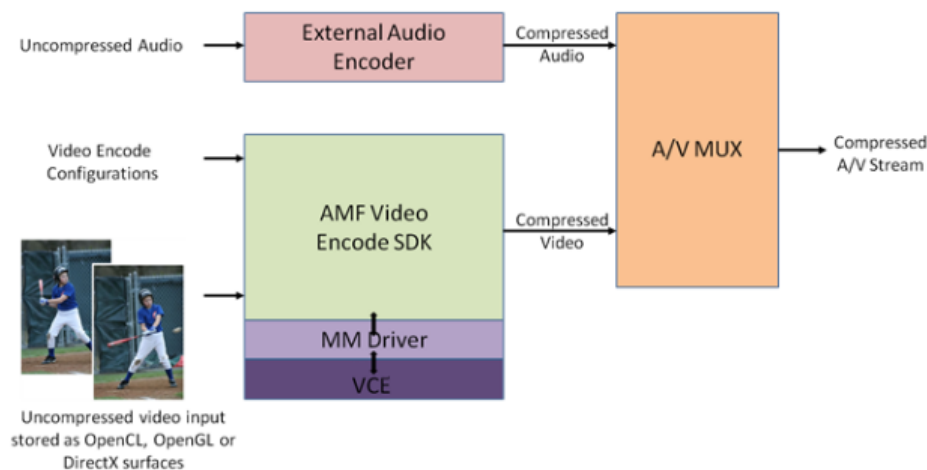


Figure 1 — System overview of the AMF Video Encode SDK

The AMF Video Encoder Component compresses RAW uncompressed video to an AV1 elementary bitstream.

The component does not provide a mechanism to handle audio compression, or stream multiplexing.

The component provides six different sets of pre-defined usages, which provide a convenient way for developers to configure the encoder to match the intended application use case. Advanced developers can also adjust encoding parameters to tailor the behavior to their specific application requirements.

## 1.2 Pre-defined Encoder Usages

The following table provides a brief overview of the encoding usage modes that have been defined:

Usage Mode	Intended use-cases	Comments
Transcoding	Transcoding, video editing	Favor compression efficiency and throughput over latency.
Low Latency	Video collaboration, remote desktop	Optimize for low latency scenarios but allow occasional bitrate overshoots to preserve quality.

Table 1. Encoding usage modes

## 2 AMF Video Encoder VCN-AV1 Component

The AMF Video Encoder AV1 component provides hardware accelerated AV1 encoding using AMD's IP.

To instantiate the AMF Video Encoder component, call the `AMFFactory::CreateComponent` method passing `AMFVideoEncoderHW_AV1` component IDs defined in the `public/include/components/VideoEncoderAV1.h` header.

### 2.1 Input Submission and Output Retrieval

The AMF Video Encoder component accepts `AMFSurface` objects as input and produces `AMFBuffer` objects for output.

### 2.2 Encode Parameters

Annex A provides the detailed description of encoding parameters (i.e., encoder properties) exposed by the Video Encoder AV1 component.

All properties are accessed using the `AMFPropertyStorage` interface of the Encoder object.

#### 2.2.1 Static Properties

Static properties (e.g., profile, tier, level, usage) must be defined before the `Init()` function is called, and will apply until the end of the encoding session.

#### 2.2.2 Dynamic Properties

All dynamic properties have default values. Several properties can be changed subsequently and these changes will be flushed to encoder only before the next `Submit()` call.

#### 2.2.3 Frame Per-Submission Properties

Per submission properties are applied on a per frame basis. They can be set optionally to force a certain behavior (e.g., force frame type to IDR) by updating the properties of the `AMFSurface` object that is passed through the `AMFComponent::Submit()` call.

#### 2.2.4 ROI Feature

Region of importance (ROI) feature provides a way to specify the relative importance of the macroblocks in the video frame. Encoder will further adjust the bits allocation among code blocks based on the importance, on top of the base rate control

decisions. More important blocks will be encoded with relatively better quality.

The ROI map can be attached to the input frame on a per frame basis. Currently, the ROI map can only use system memory. The ROI map includes the importance values of each 64x64 CTB, ranging from 0 to 10, stored in 32bit unsigned format. Refer to SimpleROI sample application for further implementation details.

### 2.2.5 SVC Properties

Scalable Video Coding (SVC) is enabled by setting `AMF_VIDEO_ENCODER_AV1_NUM_TEMPORAL_LAYERS` to a value that is greater than 1. `AMF_VIDEO_ENCODER_AV1_NUM_TEMPORAL_LAYERS` is a dynamic property and can be changed at any time during an encoding session. To ensure proper support, `AMF_VIDEO_ENCODER_AV1_MAX_NUM_TEMPORAL_LAYERS` needs to be set before initializing the encoder to a value that is not smaller than the number of temporal layers. As an example, the maximum number of temporal layers shall be set to 4 if the number of temporal layers will be changed from 3 to 4 in an encoding session. The maximum number of temporal layers supported by the encoder can be queried from the encoder capabilities before initializing the encoder.

To define SVC parameters per layer, the following format must be used:

```
TL<Temporal_Layer_Number>.QL<Quality_Layer_Number>.<Parameter_name>
```

As an example, with two temporal layers, to configure "Target bitrate" for the base/first temporal layer and first quality layer, the following parameter should be used:

```
TL0.QL0.AMF_VIDEO_ENCODER_AV1_TARGET_BITRATE
```

To configure "Target bitrate" for the second temporal layer and first quality layer, the following parameter should be used:

```
TL1.QL0.AMF_VIDEO_ENCODER_AV1_TARGET_BITRATE
```

When setting per layer parameters, the equivalent non-SVC layer parameters should not be set for the encoder otherwise the per layer configuration will be overwritten.

Remark: quality layers are not supported. "QL0" must be used for quality layers.

The framerate for each layer should follow a fixed relationship as the table below:

Layer=2, framerate0:framerate1=1:1, here `framerate0` means the framerate of `layer0` and `framerate1` means the framerate of `layer1`.

```
Layer=3, framerate0:framerate1:framerate2=1:1:2
```

```
Layer=4, framerate0:framerate1:framerate2:framerate3=1:1:2:4
```

### 2.2.6 LTR Properties

LTR (Long Term Reference) is to manually select a reference frame which can be far away to encode current frame. Normally, the encoder selects last frame as reference or a frame at lower layer in the SVC case.

In AV1, maximum of 8 reference frames are supported according to the spec. These 8 reference frames are shared by SVC and LTR. To use LTR, you need to set these properties as Static Properties:

```
AMF_VIDEO_ENCODER_AV1_MAX_LTR_FRAMES, Max number of LTR frames. The maximum value can be queried from  
AMF_VIDEO_ENCODER_AV1_CAP_MAX_NUM_LTR_FRAMES. AMF_VIDEO_ENCODER_AV1_LTR_MODE default =  
AMF_VIDEO_ENCODER_AV1_LTR_MODE_RESET_UNUSED; remove/keep unused LTRs (not specified in property  
AMF_VIDEO_ENCODER_AV1_FORCE_LTR_REFERENCE_BITFIELD)
```

The LTR\_MODE has two options:

```
enum AMF_VIDEO_ENCODER_AV1_LTR_MODE_ENUM  
{
```

```

    AMF_VIDEO_ENCODER_AV1_LTR_MODE_RESET_UNUSED    = 0,
    AMF_VIDEO_ENCODER_AV1_LTR_MODE_KEEP_UNUSED
};

```

Reset\_unused: encoder will discard all other LTR frames stored once a LTR frame is used as reference.

Keep\_unused: encoder will not change other LTR frames stored once any LTR frame is used as reference. When we enable auto LTR mode in PA, this mode will be automatically selected internally and `AMF_VIDEO_ENCODER_AV1_MAX_LTR_FRAMES` will be set to 4 no matter what users set. For details of “auto LTR mode”, please refer to `AMF_Video_PreAnalysis_API` document.

There are two Frame Per-Submission Properties need be set to use LTR:

- `AMF_VIDEO_ENCODER_AV1_MARK_CURRENT_WITH_LTR_INDEX` , Mark current frame with LTR index. -1 means don't save current frame into LTR slots. `0 ~ N` means save current frame into a LTR slot with index of `0 ~ N` . Here N should be  $\leq \text{AMF\_VIDEO\_ENCODER\_AV1\_MAX\_LTR\_FRAMES} - 1$  . When we use SVC encoding, only next base frame can be stored as LTR frame (i.e. only temporal layer number = 0 frames are allowed to be saved into LTR slot.)
- `AMF_VIDEO_ENCODER_AV1_FORCE_LTR_REFERENCE_BITFIELD` , force LTR bit-field. This is a bit-field mask that indicate which LTR slot can be used as reference for current frame. `0b1` means only slot 0 can be used as reference. `0b10` means only slot 1 can be used as reference. `0b100` means only slot 2 can be used as reference..... `0b0` means no LTR frame will be used as reference for current frame hence current frame will select short term reference frame (usually last frame) as reference. When there are multiple bits are enabled, for example: `0b1111` (=decimal 15), that means LTR slots 0,1,2 and 3 are all allowed to be selected as reference. In this case, the closest LTR frame to current frame will be selected. When we encode a key frame or switch frame, all save LTR slots will be cleared.

Referring to a LTR frame not existing in LTR slot will generate an Intra only frame.

## 3 Sample Applications

The AMF Encoder Sample application show how to setup and use the AMF Video Encoder AV1 Component to encode video frames that are loaded from disk or rendered by the DirectX 3D engine.

### 3.1 List of Parameters

Sample applications support almost all visible encoder parameters (except `PictureStructure` , `EndOfSequence` , `EndOfStream` ) and few additional parameters.

Additional parameters of `TranscodeHW` application:

Name	Type
CODEC	string
OUTPUT	string
INPUT	string
WIDTH	int
HEIGHT	int
ADAPTERID	int
ENGINE	string
FRAMES	int
THREADCOUNT	int
PREVIEWMODE	bool

Table 2. Additional miscellaneous parameters of TranscodeHW application

---

**Name:** CODEC

**Values:** AV1 or av1

**Default Values:** AV1

**Description:** Specify codec type.

---

**Name:** OUTPUT

**Values:** File name, relative or absolute path

**Default Value:** NULL

**Description:** Output AV1 file for encoded data.

---

**Name:** INPUT

**Values:** File name, relative or absolute path

**Default Value:** NULL

**Description:** Input file with frames (AVC or HEVC or YUV raw data).

---

**Name:** WIDTH

**Values:** Frame width

**Default Value:** 0

**Description:** Frame width.

---

**Name:** HEIGHT

**Values:** Frame height

**Default Value:** 0

**Description:** Frame height.

---

**Name:** AdapterID

**Values:** Number

**Default Value:** 0

**Description:** Index of GPU adapter.

---

**Name:** ENGINE

Values: DX9 , DX11

Default Value: DX11

Description: Specify Engine type.

---

Name: FRAMES

Values: Number of frames to be encoded

Default Values: 100

Description: Number of frames to render.

---

Name: THREADCOUNT

Values: Number

Default Values: 1

Description: Number of session run ip parallel.

---

Name: PREVIEWMODE

Values: true , false

Default Values: false

Description: Preview Mode .

---

Additional parameters of VCEncoderD3D application:

Name	Category
CODEC	string
OUTPUT	string
RENDER	string
WIDTH	int
HEIGHT	int
FRAMES	int
ADAPTERID	int
WINDOWMODE	bool
FULLSCREEN	bool
QueryInstanceCount	bool
UseInstance	int
FRAMERATE	int

Table 3. Miscellaneous parameters of VCEEncoderD3D application.

---

**Name:** CODEC

**Values:** AV1 or av1

**Default Value:** AV1

**Description:** Specify codec type.

---

**Name:** OUTPUT

**Values:** File name, relative or absolute path

**Default Value:** NULL

**Description:** Output AV1 file for encoded data.

---

**Name:** RENDER

**Values:** DX9 , DX9Ex , DX11 , OpenGL , OpenCL , Host , OpenCLDX9 , OpenCLDX11 , OpenGLDX9 , OpenGLDX11 , OpenCLOpenGLDX9 , OpenCLOpenGLDX11 , HostDX9 , HostDX11 , DX11DX9 , Vulkan

**Default Value:** DX11

**Description:** Specifies render type.

---

**Name:** WIDTH

**Values:** Frame width

**Default Value:** 1280

**Description:** Frame width.

---

**Name:** HEIGHT

**Values:** Frame height

**Default Value:** 720

**Description:** Frame height.

---

**Name:** FRAMES

**Values:** Number of frames to be encoded

**Default Value:** 100

**Description:** Number of frames to render.

---

**Name:** ADAPTERID



Values: Number

Default Value: 0

Description: Index of GPU adapter.

---

Name: WINDOWMODE

Values: true , false

Default Value: false

Description: Shows rendering window for D3D sample application.

---

Name: FULLSCREEN

Values: true , false

Default Value: false

Description: Enables full screen.

---

Name: QueryInstanceCount

Values: true , false

Default Value: false

Description: If the flag is set, the number of independent VCE instances will be queried and printed.

---

Name: UseInstance

Values: 0 ... number of instances - 1

Default Value: Depends on usage

Description: If there are more than one VCE Instances, you can force which instance to use. Valid range is [0.. (number of instances - 1)] .

---

Name: FRAMERATE

Values: Render frame rate

Default Value: 0

Description: Render frame rate.

---

## 3.2 Command line example

### 3.2.1 Transcoding application (TranscodingHW.exe)

```
TranscodeHW.exe -input input.h264 -output out.mp4 -codec AV1 -width 1280 -height 720 -Av1Usage transcoding -AV1RateControlMethod cbr -AV1TargetBitrate 100000
```

This command transcodes H264 elementary stream to AV1 video in .mp4 container. Encoder is created with “Transcoding” usage.

3.2.2 D3D application (VCEncoderD3D.exe)

```
VCEncoderD3D.exe -output VideoSample_1024x768.mp4 -codec AV1 -width 1024 -height 768 -AV1Usage transcoding -
AV1RateControlMethod cbr -AV1TargetBitrate 500000 -frames 400
```

This command encodes 400 frames through D3D renderer and creates an output file with the encoded data. Encoder is created with “Transcoding” usage. Initial configuration sets bitrate to a value of 500 kbits/sec.

4 Annex A: Encoding & frame parameters description

Table A-1. Encoder parameters

Name (Prefix “AMF_VIDEO_ENCODER_AV1_”)	Type
USAGE	amf_int64
PROFILE	amf_int64
LEVEL	amf_int64
MAX_LTR_FRAMES	amf_int64
TILES_PER_FRAME	amf_int64
LTR_MODE	amf_int64
MAX_NUM_REFRAMES	amf_int64
ENCODING_LATENCY_MODE	amf_int64
ALIGNMENT_MODE	amf_int64
PRE_ANALYSIS	amf_bool
MAX_NUM_TEMPORAL_LAYERS	amf_int64
ENABLE_SMART_ACCESS_VIDEO	amf_bool

Table 4. Encoder static parameters

Name: AMF\_VIDEO\_ENCODER\_AV1\_USAGE

Values: AMF\_VIDEO\_ENCODER\_AV1\_USAGE\_ENUM : AMF\_VIDEO\_ENCODER\_AV1\_USAGE\_TRANSCODING , AMF\_VIDEO\_ENCODER\_AV1\_USAGE\_LOW\_LATENCY

Default Value: N/A

Description: Selects the AMF usage (see 1.2)

Name: AMF\_VIDEO\_ENCODER\_AV1\_PROFILE

Values: AMF\_VIDEO\_ENCODER\_AV1\_PROFILE\_ENUM : AMF\_VIDEO\_ENCODER\_AV1\_PROFILE\_MAIN

Default Value: AMF\_VIDEO\_ENCODER\_PROFILE\_MAIN

Description: Selects the AV1 profile

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_LEVEL

**Values:** AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_ENUM : AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_2\_0 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_2\_1 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_2\_2 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_2\_3 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_3\_0 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_3\_1 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_3\_2 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_3\_3 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_4\_0 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_4\_1 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_4\_2 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_4\_3 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_5\_0 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_5\_1 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_5\_2 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_5\_3 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_6\_0 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_6\_1 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_6\_2 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_6\_3 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_7\_0 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_7\_1 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_7\_2 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_7\_3

**Default Value:** AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_6\_2

**Description:** Selects the AV1 Profile Level

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_MAX\_LTR\_FRAMES

**Values:** 0 ... 8

**Default Value:** 0

**Description:** The number of long-term references controlled by the user.

Remarks:

- When == 0 , the encoder can not use LTRs during encoding.
  - When > 0 , the user has control over all LTR.
  - With user control of LTR, Intra-refresh features are not supported.
  - The actual maximum number of LTRs allowed depends on SVC setting and AV1 Level limits, encoding resolution, and DPB size. The DPB size limit impacts the maximum number of LTR allowed.
- 

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_TILES\_PER\_FRAME

**Values:** >=1

**Default Value:** 1

**Description:** Sets the number of tiles per frame

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_LTR\_MODE

**Values:** AMF\_VIDEO\_ENCODER\_AV1\_LTR\_MODE\_ENUM : AMF\_VIDEO\_ENCODER\_AV1\_LTR\_MODE\_RESET\_UNUSED , AMF\_VIDEO\_ENCODER\_AV1\_LTR\_MODE\_KEEP\_UNUSED

**Default Value:** AMF\_VIDEO\_ENCODER\_AV1\_LTR\_MODE\_RESET\_UNUSED

**Description:** Remove/keep unused LTRs not specified inside the LTR reference bitfield.

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_MAX\_NUM\_REFRAMES

**Values:** 0 ... 8

**Default Value:** 1

**Description:** Maximum number of reference frames

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_ENCODING\_LATENCY\_MODE

**Values:** AMF\_VIDEO\_ENCODER\_AV1\_ENCODING\_LATENCY\_MODE\_ENUM : AMF\_VIDEO\_ENCODER\_AV1\_ENCODING\_LATENCY\_MODE\_NONE ,  
AMF\_VIDEO\_ENCODER\_AV1\_ENCODING\_LATENCY\_MODE\_POWER\_SAVING\_REAL\_TIME ,  
AMF\_VIDEO\_ENCODER\_AV1\_ENCODING\_LATENCY\_MODE\_REAL\_TIME , AMF\_VIDEO\_ENCODER\_AV1\_ENCODING\_LATENCY\_MODE\_LOWEST\_LATENCY

**Default Value:** depends on USAGE

**Description:** Choose different mode to balance encoder latency with power consumption

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_ALIGNMENT\_MODE

**Values:** AMF\_VIDEO\_ENCODER\_AV1\_ALIGNMENT\_MODE\_ENUM : AMF\_VIDEO\_ENCODER\_AV1\_ALIGNMENT\_MODE\_64X16\_ONLY ,  
AMF\_VIDEO\_ENCODER\_AV1\_ALIGNMENT\_MODE\_64X16\_1080P\_CODED\_1082 , AMF\_VIDEO\_ENCODER\_AV1\_ALIGNMENT\_MODE\_NO\_RESTRICTIONS

**Default Value:** AMF\_VIDEO\_ENCODER\_AV1\_ALIGNMENT\_MODE\_64X16\_ONLY

**Description:** AV1 alignment Mode.

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_PRE\_ANALYSIS

**Values:** true , false

**Default Value:** false

**Description:** Enables the pre-analysis module. Some features require this to be enabled. Refer to AMF Video PreAnalysis API reference for more details.

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_MAX\_NUM\_TEMPORAL\_LAYERS

**Values:** Maximum number of temporal layers supported <= 4

**Default Value:** 1

**Description:** Sets the maximum number of temporal layers. It shall not be exceeded by the number of temporal layers. The maximum number of temporal layers supported is determined by the corresponding encoder capability.

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_ENABLE\_SMART\_ACCESS\_VIDEO

**Values:** true , false

**Default Value:** false

**Description:** true = enables smart access video feature, which allocate the encoding task smartly on different GPU

---

Name (Prefix "AMF_VIDEO_ENCODER_AV1_")	Type
FRAMESIZE	AMFSize

Table 5. Encoder resolution parameters

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_FRAME\_SIZE

**Values:** Width: 192 – 4096 Height: 128 – 2176

**Default Value:** 0,0

**Description:** Frame width/Height in pixels, maximum value is hardware-specific, should be queried through AMFCaps

Name (Prefix "AMF_VIDEO_ENCODER_AV1_")	Type
TARGET_BITRATE	amf_int64
PEAK_BITRATE	amf_int64
RATE_CONTROL_METHOD	amf_int64
QVBR_QUALITY_LEVEL	amf_int64
RATE_CONTROL_SKIP_FRAME	amf_bool
MIN_Q_INDEX_INTRA	amf_int64
MAX_Q_INDEX_INTRA	amf_int64
MIN_Q_INDEX_INTER	amf_int64
MAX_Q_INDEX_INTER	amf_int64
Q_INDEX_INTRA	amf_int64
Q_INDEX_INTER	amf_int64
FRAMERATE	AMFRate
VBV_BUFFER_SIZE	amf_int64
INITIAL_VBV_BUFFER_FULLNESS	amf_int64
ENFORCE_HRD	amf_bool
RATE_CONTROL_PREENCODE	amf_bool
AQ_MODE	amd_int64
FILLER_DATA	amf_bool
HIGH_MOTION_QUALITY_BOOST	amf_bool

Table 6. Encoder rate-control parameters

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_TARGET\_BITRATE

**Values:** >0

**Default Value:** 20 mbps

**Description:** Sets the target bitrate, bit/s based on use case

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_PEAK\_BITRATE

**Values:** >= TargetBitrate

**Default Value associated with usages:**

- Transcoding: 30 mbps
- Low latency: 20 mbps

**Description:** Sets the peak bitrate, use for HRD model

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_RATE\_CONTROL\_METHOD

**Values:** AMF\_VIDEO\_ENCODER\_AV1\_RATE\_CONTROL\_METHOD\_ENUM : AMF\_VIDEO\_ENCODER\_AV1\_RATE\_CONTROL\_METHOD\_UNKNOWN , AMF\_VIDEO\_ENCODER\_AV1\_RATE\_CONTROL\_METHOD\_CONSTANT\_QP , AMF\_VIDEO\_ENCODER\_AV1\_RATE\_CONTROL\_METHOD\_LATENCY\_CONSTRAINED\_VBR , AMF\_VIDEO\_ENCODER\_AV1\_RATE\_CONTROL\_METHOD\_PEAK\_CONSTRAINED\_VBR , AMF\_VIDEO\_ENCODER\_AV1\_RATE\_CONTROL\_METHOD\_CBR , AMF\_VIDEO\_ENCODER\_AV1\_RATE\_CONTROL\_METHOD\_QUALITY\_VBR , AMF\_VIDEO\_ENCODER\_AV1\_RATE\_CONTROL\_METHOD\_HIGH\_QUALITY\_VBR , AMF\_VIDEO\_ENCODER\_AV1\_RATE\_CONTROL\_METHOD\_HIGH\_QUALITY\_CBR

**Default Value:** AMF\_VIDEO\_ENCODER\_AV1\_RATE\_CONTROL\_METHOD\_PEAK\_CONSTRAINED\_VBR

**Description:** Selects the rate control method:

- CQP – Constrained QP,
- VBR\_LAT - Latency Constrained VBR,
- VBR - Peak Constrained VBR,
- CBR - Constant Bitrate,
- QVBR – Quality VBR,
- HQVBR – High Quality VBR,
- HQCBR – High Quality CBR.

Remarks:

- When SVC encoding is enabled, some rate-control parameters can be configured differently for a particular SVC-layer. An SVC-layer is denoted by an index pair [SVC-Temporal Layer index][SVC-Quality Layer index] . E.g. The bitrate may be configured differently for SVC-layers [0][0] and [1][0] .
  - We restrict all SVC layers to have the same Rate Control method.
  - QVBR, HQVBR and HQCBR are only supported if PreAnalysis is enabled.
  - QVBR, HQVBR and HQCBR target improving subjective quality with the possible loss of objective quality (PSNR SSIM or VMAF).
- 

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_QVBR\_QUALITY\_LEVEL

**Values:** 1 – 51

**Default Value:** 23

**Description:** Sets the quality level for QVBR rate control method.

Remarks:

- Only available for QVBR rate control method.
-

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_RATE\_CONTROL\_SKIP\_FRAME

**Values:** true , false

**Default Value associated with usages:**

- Transcoding: false
- Low latency: true

**Description:** Enables skip frame for rate control

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_MIN\_Q\_INDEX\_INTRA

**Values:** 0 – 255

**Default Value:** 0

**Description:** Sets the minimum Q for Intra frame

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_MAX\_Q\_INDEX\_INTRA

**Values:** 0 – 255

**Default Value:** 51

**Description:** Sets the maximum Q for Intra frame

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_MIN\_Q\_INDEX\_INTER

**Values:** 0 – 255

**Default Value:** 0

**Description:** Sets the minimum Q for inter frame

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_MAX\_Q\_INDEX\_INTER

**Values:** 0 – 255

**Default Value:** 51

**Description:** Sets the maximum Q for inter frame.

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_Q\_INDEX\_INTRA

**Values:** 0 – 255

**Default Value:** 26

**Description:** Sets the constant Q for Intra frames.

**Remarks:** Only available for CQP rate control method.

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_Q\_INDEX\_INTER

**Values:** 0 – 255

**Default Value:** 26

**Description:** Sets the constant Q for Inter frames.

**Remarks:** Only available for CQP rate control method.

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_FRAMERATE

**Values:** 1\*FrameRateDen ... 120\* FrameRateDen

**Default Value:** 30 fps

**Description:** Frame rate numerator/denominator. Input is : AMFRate for example, the code below will set the frame rate to 30000 / 1001 = 29.97 FPS: `pEncoder->SetProperty(AMF_VIDEO_ENCODER_AV1_FRAMERATE, ::AMFConstructRate(30000, 1001));`

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_VBV\_BUFFER\_SIZE

**Values:** >0

**Default Value associated with usages:**

- Transcoding: 20 mbits
- Low latency: 4 mbits

**Description:** Sets the VBV buffer size in bits based on use case, use for HRD model.

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_INITIAL\_VBV\_BUFFER\_FULLNESS

**Values:** 0 – 64 , 0 = 0% , 64 = 100%

**Default Value:** 64

**Description:** Sets the initial VBV buffer fullness, use for HRD model

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_ENFORCE\_HRD

**Values:** true , false

**Default Value:** false

**Description:** Disables/enables constraints on QP variation within a picture to meet HRD requirement(s).

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_RATE\_CONTROL\_PREENCODE

**Values:** true , false

**Default Value:** false

**Description:** Pre-analysis assisted rate control.

---



**Name:** AMF\_VIDEO\_ENCODER\_AV1\_AQ\_MODE

**Values:** AMF\_VIDEO\_ENCODER\_AV1\_AQ\_MODE\_ENUM : AMF\_VIDEO\_ENCODER\_AV1\_AQ\_MODE\_NONE , AMF\_VIDEO\_ENCODER\_AV1\_AQ\_MODE\_CAQ

**Default Value:** AMF\_VIDEO\_ENCODER\_AV1\_AQ\_MODE\_NONE

**Description:** By default, disable Adaptive Quality Mode. This feature can not be used under CQP rate control method.

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_FILLER\_DATA

**Values:** true , false

**Default Value:** false

**Description:** Enable filler data for CBR usage

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_HIGH\_MOTION\_QUALITY\_BOOST

**Values:** true , false

**Default Value:** false

**Description:** Enable high motion quality boost mode to pre-analyze the motion of the video and use this information to improve encoding.

---

Name (Prefix "AMF_VIDEO_ENCODER_AV1_")	Type
MAX_COMPRESSED_FRAME_SIZE	amf_int64
HEADER_INSERTION_MODE	amf_int64
SWITCH_FRAME_INSERTION_MODE	amf_int64
SWITCH_FRAME_INTERVAL	amd_int64
GOP_SIZE	amd_int64
CDEF_MODE	amd_int64
INTRA_REFRESH_MODE	amf_int64
INTRAREFRESH_STRIPES	amf_int64

Table 7. Encoder picture-control parameters

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_MAX\_COMPRESSED\_FRAME\_SIZE

**Values:** 0 – no limit

**Default Value:** 0

**Description:** Maximum compressed frame size in bits that rate control algorithm will try to limit. May still larger than this number in some cases.

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_HEADER\_INSERTION\_MODE

**Values:** AMF\_VIDEO\_ENCODER\_AV1\_HEADER\_INSERTION\_MODE\_ENUM : AMF\_VIDEO\_ENCODER\_AV1\_HEADER\_INSERTION\_MODE\_NONE , AMF\_VIDEO\_ENCODER\_AV1\_HEADER\_INSERTION\_MODE\_GOP\_ALIGNED , AMF\_VIDEO\_ENCODER\_AV1\_HEADER\_INSERTION\_MODE\_KEY\_FRAME\_ALIGNED

**Default Value:** AMF\_VIDEO\_ENCODER\_AV1\_HEADER\_INSERTION\_MODE\_NONE

**Description:** Sets the headers insertion mode

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_SWITCH\_FRAME\_INSERTION\_MODE

**Values:** AMF\_VIDEO\_ENCODER\_AV1\_SWITCH\_FRAME\_INSERTION\_MODE\_ENUM AMF\_VIDEO\_ENCODER\_AV1\_SWITCH\_FRAME\_INSERTION\_MODE\_NONE , AMF\_VIDEO\_ENCODER\_AV1\_SWITCH\_FRAME\_INSERTION\_MODE\_FIXED\_INTERVAL

**Default Value:** depends on USAGE

**Description:** switch frame insertion mode

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_SWITCH\_FRAME\_INTERVAL

**Values:** >0

**Default Value:** depends on USAGE

**Description:** the interval between two inserted switch frames. Valid only when AMF\_VIDEO\_ENCODER\_AV1\_SWITCH\_FRAME\_INSERTION\_MODE is AMF\_VIDEO\_ENCODER\_AV1\_SWITCH\_FRAME\_INSERTION\_MODE\_FIXED\_INTERVAL .

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_GOP\_SIZE

**Values:** >0

**Default Value associated with usages:**

- Transcoding: 30
- Low latency: 300

**Description:** The period to insert key frame in fixed size mode. 0 means only insert the first frame (infinite GOP size)

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_CDEF\_MODE

**Values:** AMF\_VIDEO\_ENCODER\_AV1\_CDEF\_MODE\_ENUM : AMF\_VIDEO\_ENCODER\_AV1\_CDEF\_DISABLE , AMF\_VIDEO\_ENCODER\_AV1\_CDEF\_ENABLE\_DEFAULT

**Default Value:** AMF\_VIDEO\_ENCODER\_AV1\_CDEF\_ENABLE\_DEFAULT

**Description:** Disable/enable the CDEF filter.

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_INTRA\_REFRESH\_MODE

**Values:** AMF\_VIDEO\_ENCODER\_AV1\_INTRA\_REFRESH\_MODE\_ENUM : AMF\_VIDEO\_ENCODER\_AV1\_INTRA\_REFRESH\_MODE\_\_DISABLED , AMF\_VIDEO\_ENCODER\_AV1\_INTRA\_REFRESH\_MODE\_\_GOP\_ALIGNED , AMF\_VIDEO\_ENCODER\_AV1\_INTRA\_REFRESH\_MODE\_\_CONTINUOUS

**Default Value:** AMF\_VIDEO\_ENCODER\_AV1\_INTRA\_REFRESH\_MODE\_\_DISABLED

**Description:** The mode of intra refresh.

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_INTRAREFRESH\_STRIPES

**Values:**  $\geq 1$ , how many stripes in a frame for intra refresh

**Default Value:** N/A

**Description:** Valid only when intra refresh is enabled.

---

Name (Prefix "AMF_VIDEO_ENCODER_AV1_")	Type
QUALITY_PRESET	amf_int64
QUERY_TIMEOUT	amf_int64
EXTRA_DATA	AMFBufferPtr

Table 8. Encoder miscellaneous parameters

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_QUALITY\_PRESET

**Values:** AMF\_VIDEO\_ENCODER\_AV1\_QUALITY\_PRESET\_ENUM : AMF\_VIDEO\_ENCODER\_AV1\_QUALITY\_PRESET\_HIGH\_QUALITY , AMF\_VIDEO\_ENCODER\_AV1\_QUALITY\_PRESET\_QUALITY , AMF\_VIDEO\_ENCODER\_AV1\_QUALITY\_PRESET\_BALANCED , AMF\_VIDEO\_ENCODER\_AV1\_QUALITY\_PRESET\_SPEED

**Default Value associated with usages:**

- Transcoding: AMF\_VIDEO\_ENCODER\_AV1\_QUALITY\_PRESET\_BALANCED
- Low latency: AMF\_VIDEO\_ENCODER\_AV1\_QUALITY\_PRESET\_SPEED

**Description:** Selects the quality preset to balance between encoding speed and video quality

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_QUERY\_TIMEOUT

**Values:** 0 ... 50

**Default Value:** 0 (no wait)

**Description:** Timeout for QueryOutput call in ms.

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_EXTRA\_DATA

**Values:** AMFBufferPtr

**Default Value:** NULL

**Description:** Buffer to retrieve coded sequence header.

---

Name (Prefix "AMF_VIDEO_ENCODER_AV1_")	Type
SCREEN_CONTENT_TOOLS	amf_bool
PALETTE_MODE	amf_bool
FORCE_INTEGER_MV	amf_bool

Name (Prefix "AMF_VIDEO_ENCODER_AV1_")	Type
ORDER_HINT	amf_bool
FRAME_ID	amf_bool
TILE_GROUP_OBU	amf_bool
ERROR_RESILIENT_MODE	amf_bool
COLOR_BIT_DEPTH	amf_int64
CDF_UPDATE	amf_bool
CDF_FRAME_END_UPDATE_MODE	amd_int64

Table 9. Encoder configuration

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_SCREEN\_CONTENT\_TOOLS

**Values:** true , false

**Default Value:** false

**Description:** If true, allow enabling screen content tools by AMF\_VIDEO\_ENCODER\_AV1\_PALETTE\_MODE\_ENABLE and AMF\_VIDEO\_ENCODER\_AV1\_FORCE\_INTEGER\_MV ; if false, all screen content tools are disabled.

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_PALETTE\_MODE

**Values:** true , false

**Default Value:** depends on USAGE

**Description:** If true, enable palette mode; if false, disable palette mode. Valid only when AMF\_VIDEO\_ENCODER\_AV1\_SCREEN\_CONTENT\_TOOLS is true.

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_FORCE\_INTEGER\_MV

**Values:** true , false

**Default Value:** depends on USAGE

**Description:** If true , enable force integer MV; if false , disable force integer MV. Valid only when AMF\_VIDEO\_ENCODER\_AV1\_SCREEN\_CONTENT\_TOOLS is true .

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_ORDER\_HINT

**Values:** true , false

**Default Value:** false

**Description:** Code order hint; if false , don't code order hint.

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_FRAME\_ID

Values: true , false

Default Value: depends on USAGE

Description: If true , code frame id; if false , don't code frame id.

---

Name: AMF\_VIDEO\_ENCODER\_AV1\_TILE\_GROUP\_OBU

Values: true , false

Default Value: depends on USAGE

Description: If true , code FrameHeaderObu + TileGroupObu and each TileGroupObu contains one tile; if false , code FrameObu .

---

Name: AMF\_VIDEO\_ENCODER\_AV1\_ERROR\_RESILIENT\_MODE

Values: true , false

Default Value: depends on USAGE

Description: If true , enable error resilient mode; if false , disable error resilient mode.

---

Name: AMF\_VIDEO\_ENCODER\_AV1\_COLOR\_BIT\_DEPTH

Values: AMF\_COLOR\_BIT\_DEPTH\_ENUM : AMF\_COLOR\_BIT\_DEPTH\_UNDEFINED , AMF\_COLOR\_BIT\_DEPTH\_8 , AMF\_COLOR\_BIT\_DEPTH\_10

Default Value: AMF\_COLOR\_BIT\_DEPTH\_8

Description: Sets the number of bits in each pixel’s color component in the encoder’s compressed output bitstream.

---

Name: AMF\_VIDEO\_ENCODER\_AV1\_CDF\_UPDATE

Values: true , false

Default Value: false

Description: If false , disable CDF update.

---

Name: AMF\_VIDEO\_ENCODER\_AV1\_CDF\_FRAME\_END\_UPDATE\_MODE

Values: AMF\_VIDEO\_ENCODER\_AV1\_CDF\_FRAME\_END\_UPDATE\_MODE\_ENUM : AMF\_VIDEO\_ENCODER\_AV1\_CDF\_FRAME\_END\_UPDATE\_MODE\_DISABLE , AMF\_VIDEO\_ENCODER\_AV1\_CDF\_FRAME\_END\_UPDATE\_MODE\_ENABLE\_DEFAULT

Default Value: AMF\_VIDEO\_ENCODER\_AV1\_CDF\_FRAME\_END\_UPDATE\_MODE\_DISABLE

Description: CDF frame end update mode.

---

Name (Prefix “AMF_VIDEO_ENCODER_AV1_”)	Type
INPUT_COLOR_PROFILE	amf_int64
INPUT_TRANSFER_CHARACTERISTIC	amf_int64

Name (Prefix "AMF_VIDEO_ENCODER_AV1_")	Type
INPUT_COLOR_PRIMARIES	amf_int64
OUTPUT_COLOR_PROFILE	amf_int64
OUTPUT_TRANSFER_CHARACTERISTIC	amf_int64
OUTPUT_COLOR_PRIMARIES	amf_int64
INPUT_HDR_METADATA	AMFBufferPtr

Table 10. Encoder color conversion parameters

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_INPUT\_COLOR\_PROFILE

**Values:** AMF\_VIDEO\_CONVERTER\_COLOR\_PROFILE\_ENUM : AMF\_VIDEO\_CONVERTER\_COLOR\_PROFILE\_UNKNOWN , AMF\_VIDEO\_CONVERTER\_COLOR\_PROFILE\_601 , AMF\_VIDEO\_CONVERTER\_COLOR\_PROFILE\_709 , AMF\_VIDEO\_CONVERTER\_COLOR\_PROFILE\_2020 , AMF\_VIDEO\_CONVERTER\_COLOR\_PROFILE\_JPEG , AMF\_VIDEO\_CONVERTER\_COLOR\_PROFILE\_FULL\_601 , AMF\_VIDEO\_CONVERTER\_COLOR\_PROFILE\_FULL\_709 , AMF\_VIDEO\_CONVERTER\_COLOR\_PROFILE\_FULL\_2020 , AMF\_VIDEO\_CONVERTER\_COLOR\_PROFILE\_COUNT

**Default Value:** AMF\_VIDEO\_CONVERTER\_COLOR\_PROFILE\_UNKNOWN

**Description:** Color profile of the input surface. SDR - Setting this parameter ( COLOR\_PROFILE ) can fully describe a surface for SDR use case. HDR – For HDR use case the TRANSFER\_CHARACTERISTIC , COLOR\_PRIMARIES , and NOMINAL\_RANGE parameters describe the surface.

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_INPUT\_TRANSFER\_CHARACTERISTIC

**Values:** AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_ENUM : AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_UNDEFINED , AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_BT709 , AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_UNSPECIFIED , AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_RESERVED , AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_GAMMA22 , AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_GAMMA28 , AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_SMPTE170M , AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_SMPTE240M , AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_LINEAR , AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_LOG , AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_LOG\_SQRT , AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_IEC61966\_2\_4 , AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_BT1361\_ECG , AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_IEC61966\_2\_1 , AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_BT2020\_10 , AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_BT2020\_12 , AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_SMPTE2084 , AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_SMPTE428 , AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_ARIB\_STD\_B67

**Default Value:** AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_UNDEFINED

**Description:** Characteristic transfer function of the input surface used to perform the mapping between linear light components (tristimulus values) and a nonlinear RGB signal. Used (alongside COLOR\_PRIMARIES and NOMINAL\_RANGE parameters) to describe surface in HDR use case.

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_INPUT\_COLOR\_PRIMARIES

**Values:** AMF\_COLOR\_PRIMARIES\_ENUM : AMF\_COLOR\_PRIMARIES\_UNDEFINED , AMF\_COLOR\_PRIMARIES\_BT709 , AMF\_COLOR\_PRIMARIES\_UNSPECIFIED , AMF\_COLOR\_PRIMARIES\_RESERVED , AMF\_COLOR\_PRIMARIES\_BT470M , AMF\_COLOR\_PRIMARIES\_BT470BG , AMF\_COLOR\_PRIMARIES\_SMPTE170M , AMF\_COLOR\_PRIMARIES\_SMPTE240M , AMF\_COLOR\_PRIMARIES\_FILM , AMF\_COLOR\_PRIMARIES\_BT2020 , AMF\_COLOR\_PRIMARIES\_SMPTE428 , AMF\_COLOR\_PRIMARIES\_SMPTE431 , AMF\_COLOR\_PRIMARIES\_SMPTE432 , AMF\_COLOR\_PRIMARIES\_JEDEC\_P22 , AMF\_COLOR\_PRIMARIES\_CCS

**Default Value:** AMF\_COLOR\_PRIMARIES\_UNDEFINED

**Description:** Color space primaries for the input surface which are the maximum red, green, and blue value permitted within the color space. Used (alongside TRANSFER\_CHARACTERISTIC and NOMINAL\_RANGE parameters) to describe surface in HDR use case.

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_OUTPUT\_COLOR\_PROFILE

**Values:** AMF\_VIDEO\_CONVERTER\_COLOR\_PROFILE\_ENUM : AMF\_VIDEO\_CONVERTER\_COLOR\_PROFILE\_UNKNOWN ,  
AMF\_VIDEO\_CONVERTER\_COLOR\_PROFILE\_601 , AMF\_VIDEO\_CONVERTER\_COLOR\_PROFILE\_709 , AMF\_VIDEO\_CONVERTER\_COLOR\_PROFILE\_2020 ,  
AMF\_VIDEO\_CONVERTER\_COLOR\_PROFILE\_JPEG , AMF\_VIDEO\_CONVERTER\_COLOR\_PROFILE\_FULL\_601 ,  
AMF\_VIDEO\_CONVERTER\_COLOR\_PROFILE\_FULL\_709 , AMF\_VIDEO\_CONVERTER\_COLOR\_PROFILE\_FULL\_2020 ,  
AMF\_VIDEO\_CONVERTER\_COLOR\_PROFILE\_COUNT

**Default Value:** AMF\_VIDEO\_CONVERTER\_COLOR\_PROFILE\_UNKNOWN

**Description:** Color profile of the compressed output stream. SDR - Setting this parameter ( COLOR\_PROFILE ) can fully describe a surface for SDR use case. HDR – For HDR use case the TRANSFER\_CHARACTERISTIC , COLOR\_PRIMARIES , and NOMINAL\_RANGE parameters describe the surface. Determines the optional VUI parameter “matrix\_coefficients”.

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_OUTPUT\_TRANSFER\_CHARACTERISTIC

**Values:** AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_ENUM : AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_UNDEFINED ,  
AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_BT709 , AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_UNSPECIFIED ,  
AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_RESERVED , AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_GAMMA22 ,  
AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_GAMMA28 , AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_SMPTE170M ,  
AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_SMPTE240M , AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_LINEAR ,  
AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_LOG , AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_LOG\_SQRT ,  
AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_IEC61966\_2\_4 , AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_BT1361\_ECG ,  
AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_IEC61966\_2\_1 , AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_BT2020\_10 ,  
AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_BT2020\_12 , AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_SMPTE2084 ,  
AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_SMPTE428 , AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_ARIB\_STD\_B67

**Default Value:** AMF\_COLOR\_TRANSFER\_CHARACTERISTIC\_UNDEFINED

**Description:** Characteristic transfer function of the compressed output stream used to perform the mapping between linear light components (tristimulus values) and a nonlinear RGB signal.

Used (alongside COLOR\_PRIMARIES and NOMINAL\_RANGE parameters) to describe surface in HDR use case.

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_OUTPUT\_COLOR\_PRIMARIES

**Values:** AMF\_COLOR\_PRIMARIES\_ENUM : AMF\_COLOR\_PRIMARIES\_UNDEFINED , AMF\_COLOR\_PRIMARIES\_BT709 ,  
AMF\_COLOR\_PRIMARIES\_UNSPECIFIED , AMF\_COLOR\_PRIMARIES\_RESERVED , AMF\_COLOR\_PRIMARIES\_BT470M ,  
AMF\_COLOR\_PRIMARIES\_BT470BG , AMF\_COLOR\_PRIMARIES\_SMPTE170M , AMF\_COLOR\_PRIMARIES\_SMPTE240M , AMF\_COLOR\_PRIMARIES\_FILM ,  
AMF\_COLOR\_PRIMARIES\_BT2020 , AMF\_COLOR\_PRIMARIES\_SMPTE428 , AMF\_COLOR\_PRIMARIES\_SMPTE431 , AMF\_COLOR\_PRIMARIES\_SMPTE432 ,  
AMF\_COLOR\_PRIMARIES\_JEDEC\_P22 , AMF\_COLOR\_PRIMARIES\_CCCS

**Default Value:** AMF\_COLOR\_PRIMARIES\_UNDEFINED

**Description:** Color space primaries for the compressed output surface which are the maximum red, green, and blue value permitted within the color space.

Used (alongside TRANSFER\_CHARACTERISTIC and NOMINAL\_RANGE parameters) to describe surface in HDR use case.

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_INPUT\_HDR\_METADATA

**Values:** AMFBuffer

**Default Value:** NULL

**Description:** Buffer to retrieve coded sequence header.

---

Name (Prefix "AMF_VIDEO_ENCODER_AV1_")	Type
NUM_TEMPORAL_LAYERS	amf_int64

Table 11. Encoder SVC parameters

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_NUM\_TEMPORAL\_LAYERS

**Values:** Maximum number of temporal layers supported

**Default Value:** 1

**Description:** Sets the number of temporal layers. SVC with temporal scalability is enabled when the number of layers is greater than 1. The maximum number of temporal layers supported is determined by the corresponding encoder capability.

Remarks:

- Actual modification of the number of temporal layers will be delayed until the start of the next temporal GOP.
- Intra-refresh feature is not supported with SVC.

---

Name (Prefix "AMF_VIDEO_ENCODER_AV1_")	Type
TL<TL_Num>.QL0.<Parameter_name>	

Table 12. Encoder SVC per-layer parameters

**Name:** TL<TL\_Num>.QL0.<Parameter\_name>

**Values:** Parameter-specific values

**Default Value:** N/A

**Description:** Configures rate-control parameter per SVC layer.

- TL\_Num — temporal layer number
- QL0 - quality layer 0
- Parameter\_name — rate-control parameter name (see below with AMF\_VIDEO\_ENCODER\_AV1\_prefix )

Rate-control parameters supported:

- TARGET\_BITRATE
- PEAK\_BITRATE
- VBV\_BUFFER\_SIZE
- FRAMERATE
- Max\_Q\_INDEX\_INTRA
- Max\_Q\_INDEX\_INTER



- Min\_Q\_INDEX\_INTRA
- Min\_Q\_INDEX\_INTER
- Q\_INDEX\_INTRA
- Q\_INDEX\_INTER
- FILLER\_DATA
- RATE\_CONTROL\_SKIP\_FRAME
- ENFORCE\_HRD
- MAX\_COMPRESSED\_FRAME\_SIZE

Table A-2. Input frame and encoded data parameters

Name (Prefix "AMF_VIDEO_ENCODER_AV1_")	Type
FORCE_INSERT_SEQUENCE_HEADER	amf_bool
FORCE_FRAME_TYPE	amf_int64
MARK_CURRENT_WITH_LTR_INDEX	amf_int64
FORCE_LTR_REFERENCE_BITFIELD	amf_int64
ROI_DATA	AMF_SURFACE_GRAY32

Table 13. Frame per-submission parameters

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_FORCE\_INSERT\_SEQUENCE\_HEADER

**Values:** true , false

**Default Value:** false

**Description:** If true , force insert sequence header with current frame.

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_FORCE\_FRAME\_TYPE

**Values:** AMF\_VIDEO\_ENCODER\_AV1\_FORCE\_FRAME\_TYPE\_ENUM : AMF\_VIDEO\_ENCODER\_AV1\_FORCE\_FRAME\_TYPE\_NONE , AMF\_VIDEO\_ENCODER\_AV1\_FORCE\_FRAME\_TYPE\_KEY , AMF\_VIDEO\_ENCODER\_AV1\_FORCE\_FRAME\_TYPE\_INTRA\_ONLY , AMF\_VIDEO\_ENCODER\_AV1\_FORCE\_FRAME\_TYPE\_SWITCH , AMF\_VIDEO\_ENCODER\_AV1\_FORCE\_FRAME\_TYPE\_SHOW\_EXISTING

**Default Value:** AMF\_VIDEO\_ENCODER\_AV1\_FORCE\_FRAME\_TYPE\_NONE

**Description:** Forces the frame type.

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_MARK\_CURRENT\_WITH\_LTR\_INDEX

**Values:** -1 ... MaxOfLTRFrames -1

**Default Value:** N/A

**Description:** If != -1 , the current picture is coded as a long-term reference with the given index. Remarks:

- When the user controls N LTRs (using the corresponding Create parameter), then the LTR Index the user can assign to a reference picture varies from 0 to N-1 . By default, the encoder will “use up” available LTR Indices (i.e. assign them to references) even if the user does not request them to be used.

- When LTR is used with SVC encoding, only base temporal layer pictures can be coded as LTR. In this case, the request to mark the current picture as LTR would be delayed to the next base temporal layer picture if the current picture is in an enhancement layer. If the user submits multiple requests to mark current as LTR between base temporal layer pictures, then only the last request is applied.

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_FORCE\_LTR\_REFERENCE\_BITFIELD

**Values:** Bitfield MaxOfLTRFrames (max possible 8 bits)

**Default Value:** 0

**Description:** Force LTR Reference allowed bitfield. If == 0, the current picture should predict from the default reference. If != 0, the current picture should predict from one of the LTRs allowed by the bitfield (bit# = LTR Index#).

Remarks:

- E.g. if Bit#0 = 1, then the existing LTR with LTR Index = 0 may be used for reference. The bitfield may allow more than one LTR for reference, in which case the encoder is free to choose which one to use. This bitfield also disallows existing LTRs not enabled by it from current/future reference.
- E.g. if Bit#1 = 0, and there is an existing reference with LTR Index = 1, then this LTR Index will not be used for reference until it is replaced with a newer reference with the same LTR Index.

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_ROI\_DATA

**Values:** Video surface in AMF\_SURFACE\_GRAY32 format

**Default Value:** N/A

**Description:** Important value for each 64x64 block ranges from 0 to 10, stored in 32bit unsigned format.

Name (Prefix "AMF_VIDEO_ENCODER_AV1_")	Type
OUTPUT_FRAME_TYPE	amf_int64
OUTPUT_MARKED_LTR_INDEX	amf_int64
OUTPUT_REFERENCED_LTR_INDEX_BITFIELD	amf_int64

Table 14. Encoded data parameters

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_OUTPUT\_FRAME\_TYPE

**Values:** AMF\_VIDEO\_ENCODER\_AV1\_OUTPUT\_FRAME\_TYPE\_ENUM : AMF\_VIDEO\_ENCODER\_AV1\_OUTPUT\_FRAME\_TYPE\_KEY , AMF\_VIDEO\_ENCODER\_AV1\_OUTPUT\_FRAME\_TYPE\_INTRA\_ONLY , AMF\_VIDEO\_ENCODER\_AV1\_OUTPUT\_FRAME\_TYPE\_INTER , AMF\_VIDEO\_ENCODER\_AV1\_OUTPUT\_FRAME\_TYPE\_SWITCH , AMF\_VIDEO\_ENCODER\_AV1\_OUTPUT\_FRAME\_TYPE\_SHOW\_EXISTING

**Default Value:** N/A

**Description:** Type of encoded frame.

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_OUTPUT\_MARKED\_LTR\_INDEX

**Values:** -1 ... MaxOfLTRFrames -1

Default Value: N/A

Description: Marked as LTR Index. If != -1 , then this picture was coded as a long-term reference with this LTR Index.

Name: AMF\_VIDEO\_ENCODER\_AV1\_OUTPUT\_REFERENCED\_LTR\_INDEX\_BITFIELD

Values: Bitfield MaxOfLTRFrames (max possible 8 bits)

Default Value: N/A

Description: Referenced LTR Index bitfield. If != 0 , this picture was coded to reference long-term references. The enabled bits identify the LTR Indices of the referenced pictures (e.g. if Bit #0 = 1 , then LTR Index 0 was used as a reference when coding this picture).

Table A-3. Encoder capabilities exposed in AMFCaps interface

Name (prefix with AMF_VIDEO_ENCODER_AV1_CAP_)	Type
NUM_OF_HW_INSTANCES	amf_int64
MAX_THROUGHPUT	amf_int64
REQUESTED_THROUGHPUT	amf_int64
COLOR_CONVERSION	amf_int64
PRE_ANALYSIS	amf_bool
MAX_BITRATE	amf_int64
MAX_PROFILE	amf_int64
MAX_LEVEL	amf_int64
MAX_NUM_TEMPORAL_LAYERS	amf_int64
MAX_NUM_LTR_FRAMES	amf_int64

Table 15. Encoder capabilities exposed in AMFCaps interface

Name: AMF\_VIDEO\_ENCODER\_AV1\_CAP\_NUM\_OF\_HW\_INSTANCES

Values: 0 ... number of instances - 1

Default Value: N/A

Description: Number of HW encoder instances.

Name: AMF\_VIDEO\_ENCODER\_AV1\_CAP\_MAX\_THROUGHPUT

Values: 0 ... number of instances - 1

Default Value: N/A

Description: MAX throughput for AV1 encoder in MB (16 x 16 pixel).

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_CAP\_REQUESTED\_THROUGHPUT

**Values:** 0 ... number of instances - 1

**Default Value:** N/A

**Description:** Currently total requested throughput for AV1 encode in MB (16 x 16 pixel).

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_CAP\_COLOR\_CONVERSION

**Values:** AMF\_ACCELERATION\_TYPE

**Default Value:** N/A

**Description:** Type of supported color conversion.

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_CAP\_PRE\_ANALYSIS

**Values:** true , false

**Default Value:** false

**Description:** Pre analysis module is available.

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_CAP\_MAX\_BITRATE

**Default Value:** N/A

**Description:** Maximum bit rate in bits.

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_CAP\_MAX\_PROFILE

**Values:** AMF\_VIDEO\_ENCODER\_AV1\_PROFILE\_ENUM : AMF\_VIDEO\_ENCODER\_AV1\_PROFILE\_MAIN

**Default Value:** N/A

**Description:** Maximum value of code profile.

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_CAP\_MAX\_LEVEL

**Values:** AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_ENUM : AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_2\_0 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_2\_1 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_2\_2 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_2\_3 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_3\_0 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_3\_1 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_3\_2 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_3\_3 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_4\_0 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_4\_1 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_4\_2 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_4\_3 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_5\_0 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_5\_1 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_5\_2 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_5\_3 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_6\_0 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_6\_1 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_6\_2 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_6\_3 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_7\_0 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_7\_1 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_7\_2 , AMF\_VIDEO\_ENCODER\_AV1\_LEVEL\_7\_3

**Default Value:** N/A

**Description:** Maximum value of codec level.

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_CAP\_MAX\_NUM\_TEMPORAL\_LAYERS

**Values:** 1 ... Maximum number of temporal layers supported

**Default Value:** N/A

**Description:** The cap of maximum number of temporal layers.

---

**Name:** AMF\_VIDEO\_ENCODER\_AV1\_CAP\_MAX\_NUM\_LTR\_FRAMES

**Values:** 0 ... 2

**Default Value:** N/A

**Description:** The cap of maximum number of LTR frames. This value is calculated based on current value of  
AMF\_VIDEO\_ENCODER\_AV1\_MAX\_NUM\_TEMPORAL\_LAYERS.

---