Advanced Media Framework – h.264 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

MIT license

Contents

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 H.264 video encoding functionality.

Figure 1 provides a system overview of the AMF Video Encoder Component.
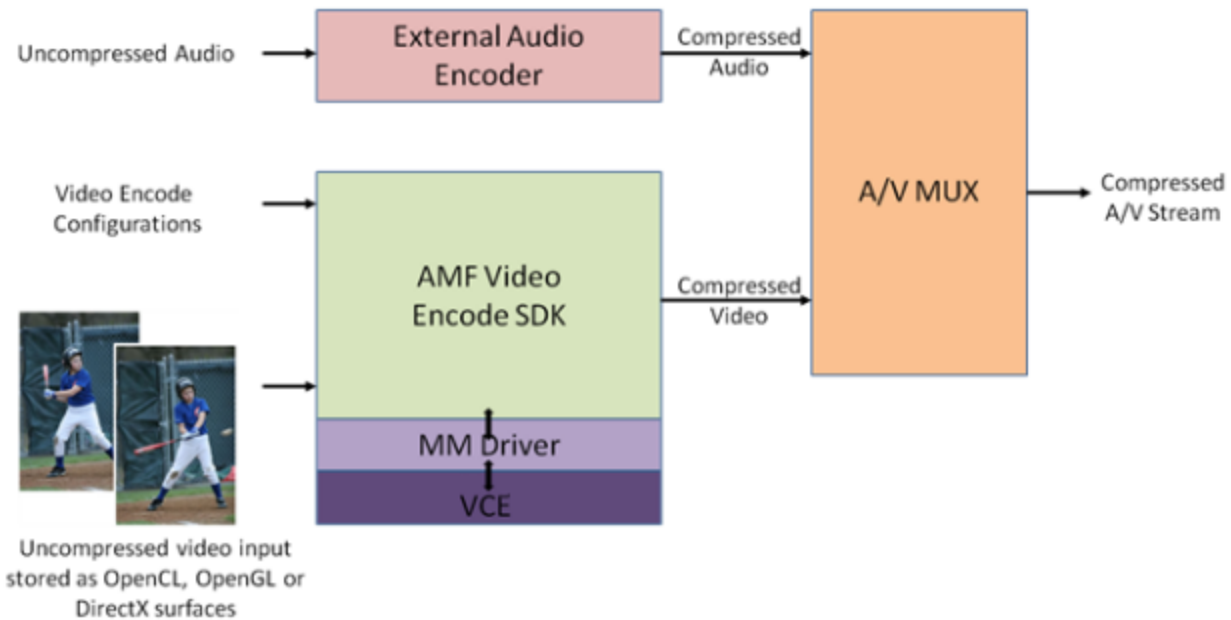
Figure 1 — System overview of the AMF Video Encode SDK

The AMF Video Encoder Component compresses RAW uncompressed video to an H.264 elementary bitstream.

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

The component provides four 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.
Ultra-low latency	Video game streaming	Optimize for extremely low latency use cases (e.g. cap the number of bits per frame), to enable high-interactivity

Usage Mode	Intended use-cases	Comments
		applications.
Low Latency	Video collaboration, remote desktop	Optimize for low latency scenarios but allow occasional bitrate overshoots to preserve quality.
Webcam	Video conferencing	Optimize for a low-latency video conferencing scenario.
HQ	High quality mode	Optimize for best subjective video quality with possible loss of performance.
HQLL	High quality low latency mode	Optimize for good quality with low latency.

Table 1. Encoding usage modes

Note: User can override the default settings for these pre-defined usages in Table A-3. Default values of parameters.

2 AMF Video Encoder VCE-AVC Component

The AMF Video Encoder VCE-AVC component provides hardware accelerated AVC/SVC encoding using AMD's VCE.

To instantiate the AMF Video Encoder component, call the `AMFFactory::CreateComponent` method passing `AMFVideoEncoderVCE_AVC` or `AMFVideoEncoderVCE_SVC` component IDs defined in the `public/include/components/VideoEncoderVCE.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.

In the Transcoding mode the encoder needs to accept at least 3 input frames before any output is produced. In low latency modes output becomes available as soon as the first submitted frame is encoded.

2.2 Encode Parameters

Annex A provides the detailed description of encoding parameters (i.e., encoder properties) exposed by the Video Encoder VCE-AVC component for the following four usages:

- Transcoding mode,
- Ultra-low latency mode,
- Low Latency mode,
- Webcam mode,
- HQ mode, and
- HQLL mode.

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

2.2.1 Static Properties

Static properties (e.g., profile, 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 SVC Properties

Scalable Video Coding (SVC) is enabled by setting `AMF_VIDEO_ENCODER_NUM_TEMPORAL_ENHANCMENT_LAYERS` to a value that is greater than 1. `AMF_VIDEO_ENCODER_NUM_TEMPORAL_ENHANCMENT_LAYER` is a dynamic property and can be changed at any time during an encoding session. To ensure proper support on Radeon RX 5000 Series or newer GPUs and Ryzen 2000 U/H series or newer APUs, `AMF_VIDEO_ENCODER_MAX_NUM_TEMPORAL_LAYERS` needs to be set before initializing the encoder to a value that is not smaller than the number of temporal enhancement layers. As an example, the maximum number of temporal layers shall be set to 4 if the number of temporal enhancement 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.TargetBitrate
```

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

```
TL1.QL0.TargetBitrate
```

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 on VCE 1.0. "QL0" must be used for quality layers.

2.2.5 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 16x16 macro block, ranging from 0 (least important) to 10 (most important), stored in 32bit unsigned format. Refer to SimpleROI sample application for further implementation details.

2.2.6 Encoder Statistics Feedback

If an application sets the `AMF_VIDEO_ENCODER_STATISTICS_FEEDBACK` flag on for an input picture, the encoder will feedback to the application statistics for this specific picture. After the encoding ends, the application can retrieve by name the specific statistic(s) it is interested in. The supported encoder statistics are listed in Table A-4. This feature is supported by Radeon RX 5000 Series or newer GPUs as well as Ryzen 2000 U/H series or newer APUs.

2.2.7 Picture Transfer Mode

If an application enables `AMF_VIDEO_ENCODER_PICTURE_TRANSFER_MODE` for a specific input picture, it can dump out the reconstructed picture after encoding and/or it can inject a picture to be used as the reference picture during the encoding. It is worth noting that reference picture injection is a feature that is intended for advanced algorithm testing and exploration. It needs to be used with care since the internal DPB in the current encoding session will be overridden by the injected reference picture(s). The reader can refer to `SimpleFrameInjection` sample application for further implementation details. This feature is supported by Radeon RX 5000 Series or newer GPUs as well as Ryzen 2000 U/H series or newer APUs.

2.2.8 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 AVC, maximum of 16 reference frames are supported according to the spec. These 16 reference frames are shared by SVC and LTR.

To use LTR, you need to set these properties as Static Properties:

`AMF_VIDEO_ENCODER_MAX_LTR_FRAMES`, Max number of LTR frames.

`AMF_VIDEO_ENCODER_LTR_MODE` default = `AMF_VIDEO_ENCODER_LTR_MODE_RESET_UNUSED`; remove/keep unused LTRs (not specified in property `AMF_VIDEO_ENCODER_FORCE_LTR_REFERENCE_BITFIELD`)

The `LTR_MODE` has two options:

```
enum AMF_VIDEO_ENCODER_LTR_MODE_ENUM
{
    AMF_VIDEO_ENCODER_LTR_MODE_RESET_UNUSED      = 0,
    AMF_VIDEO_ENCODER_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_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_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 `<= AMF_VIDEO_ENCODER_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_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 exiting in LTR slot will generate an Intra only frame.

Please note that LTR and B-frames (including fixed B-frames set through `AMF_VIDEO_ENCODER_B_PIC_PATTERN` and adaptive B-frames set through `AMF_VIDEO_ENCODER_ADAPTIVE_MINIGOP`) cannot be used simultaneously in the encoder. The encoder will check parameter settings during operation. If both LTR and B-frames are detected as enabled, the encoder will disable LTR.

2.2.9 SmartAccess Video

On supported APU + GPU systems, there is an opportunity to use SmartAccess Video. SmartAccess Video - an optimization logic which enables the parallelization of encode and decode streams across multiple Video Codec Engine (VCN) hardware instances – empowers apps to process streams faster through seamless job distribution across available hardware. With a simple enablement of the encoder and decoder control flags, the SmartAccess Video logic will optimally use hardware resources to benefit media apps. Follow the `SMART_ACCESS_VIDEO` tag in the documentation to search for the property flags to set. On systems without SmartAccess Video support, the `SMART_ACCESS_VIDEO` properties have no effect.

3 Sample Applications

The AMF Encoder Sample application show how to setup and use the AMF Video Encoder VCE-AVC 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: AVC or H264, HEVC or H265, AV1

Default Values: AVC

Description: Specify codec type.

Name: OUTPUT

Values: File name, relative or absolute path

Default Value: NULL

Description: Output file for encoded data.

Name: INPUT

Values: File name, relative or absolute path

Default Value: NULL

Description: Input file with frames.

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, Vulkan

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 VCEEncoderD3D 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: AVC or H264, HEVC or H265, AV1

Default Value: AVC

Description: Codec name.

Name: OUTPUT

Values: File name, relative or absolute path

Default Value: NULL

Description: Output H.264 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: Full screen.

Name: QueryInstanceCount

Values: `true, false`

Default Value: `false`

Description: If the flag is set, the number of independent VCE instances will be quried 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 (TranscodeHW.exe)

`TranscodeHW.exe -input input.h264 -output out.h264 –codec AVC -width 1280 -height 720`

This command transcodes H264 elementary stream to H.264 video. Encoder is created with "Transcoding" usage.

3.2.2 D3D application (VCEEncoderD3D.exe)

`VCEEncoderD3D.exe -output VideoSample_1024x768.h264 -width 1024 -height 768 -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 500kbits/sec.

4 Annex A: Encoding & frame parameters description

Table A-1. Encoder parameters

Name (prefix "AMF_VIDEO_ENCODER_")	Type
USAGE	amf_int64
INSTANCE_INDEX	amf_int64
PROFILE	amf_int64
PROFILE_LEVEL	amf_int64

Name (prefix "AMF_VIDEO_ENCODER_")	Type
MAX_LTR_FRAMES	amf_int64
LTR_MODE	amf_int64
LOWLATENCY_MODE	amf_bool
FRAMESIZE	AMFSize
ASPECT_RATIO	AMFRatio
MAX_REFRAMES	amf_int64
MAX_CONSECUTIVE_BPICTURES	amf_int64
ADAPTIVE_MINIGOP	amf_bool
PRE_ANALYSIS_ENABLE	amf_bool
COLOR_BIT_DEPTH	amf_int64
MAX_NUM_TEMPORAL_LAYERS	amf_int64
ENABLE_SMART_ACCESS_VIDEO	amf_bool

Table 4. Encoder static parameters

Name: `AMF_VIDEO_ENCODER_USAGE`

Values: `AMF_VIDEO_ENCODER_USAGE_ENUM: AMF_VIDEO_ENCODER_USAGE_TRANSCONDING`, `AMF_VIDEO_ENCODER_USAGE_TRANSCODING`, `AMF_VIDEO_ENCODER_USAGE_ULTRA_LOW_LATENCY`, `AMF_VIDEO_ENCODER_USAGE_LOW_LATENCY`, `AMF_VIDEO_ENCODER_USAGE_WEBCAM`, `AMF_VIDEO_ENCODER_USAGE_HIGH_QUALITY`, `AMF_VIDEO_ENCODER_USAGE_LOW_LATENCY_HIGH_QUALITY`

Default Value: N/A

Description: Selects the AMF usage (see Section 1.2).

Name: `AMF_VIDEO_ENCODER_INSTANCE_INDEX`

Values: `0`, `1`

Default Value: N\A

Description: Selects the encoder engine used for encoding.

Name: `AMF_VIDEO_ENCODER_PROFILE`

Values: `AMF_VIDEO_ENCODER_PROFILE_ENUM: AMF_VIDEO_ENCODER_PROFILE_BASELINE`, `AMF_VIDEO_ENCODER_PROFILE_MAIN`, `AMF_VIDEO_ENCODER_PROFILE_HIGH`

Default Value associated with usages:

- Transcoding: `AMF_VIDEO_ENCODER_PROFILE_MAIN`

- Ultra low latency: `AMF_VIDEO_ENCODER_PROFILE_MAIN`
- Low latency: `AMF_VIDEO_ENCODER_PROFILE_MAIN`
- Webcam: `AMF_VIDEO_ENCODER_PROFILE_MAIN`
- HQ: `AMF_VIDEO_ENCODER_PROFILE_HIGH`
- HQLL: `AMF_VIDEO_ENCODER_PROFILE_HIGH`

Description: Selects the H.264 profile.

Name: `AMF_VIDEO_ENCODER_PROFILE_LEVEL`

Values: `AMF_VIDEO_ENCODER_H264_LEVEL_ENUM`: `AMF_H264_LEVEL__1`,`AMF_H264_LEVEL__1_1`,
`AMF_H264_LEVEL__1_2`, `AMF_H264_LEVEL__1_3`, `AMF_H264_LEVEL__2`, `AMF_H264_LEVEL__2_1`,
`AMF_H264_LEVEL__2_2`, `AMF_H264_LEVEL__3`, `AMF_H264_LEVEL__3_1`, `AMF_H264_LEVEL__3_2`,
`AMF_H264_LEVEL__4`, `AMF_H264_LEVEL__4_1`, `AMF_H264_LEVEL__4_2`, `AMF_H264_LEVEL__5`,
`AMF_H264_LEVEL__5_1`, `AMF_H264_LEVEL__5_2`, `AMF_H264_LEVEL__6`, `AMF_H264_LEVEL__6_1`,
`AMF_H264_LEVEL__6_2`

Default Value: `AMF_H264_LEVEL__4_2`

Description: Selects the H.264 profile level. Automatically adjusted upwards based on frame size.

Name: `AMF_VIDEO_ENCODER_MAX_LTR_FRAMES`

Values: `0` ... `2`

Default Value: `0`

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

- When `== 0`, the encoder may or may not use LTRs during encoding.
- When `> 0`, the user has control over all LTR.
- With user control of LTR, B-pictures and Intra-refresh features are not supported.
- The actual maximum number of LTRs allowed depends on H.264 Annex A Table A-1 Level limits, which defines dependencies between the H.264 Level number, encoding resolution, and DPB size. The DPB size limit impacts the maximum number of LTR allowed.

Name: `AMF_VIDEO_ENCODER_LTR_MODE`

Values: `0`(reset unused) , `1` (keep unused)

Default Value: `0`

Description: Removes/keeps unused LTRs not specified inside the LTR reference bitfield.

Name: `AMF_VIDEO_ENCODER_LOWLATENCY_MODE`

Values: `true` (on), `false` (off)

Default Value associated with usages:

- Transcoding: `false`

- Ultra low latency: `true`
- Low latency: `false`
- Webcam: `false`
- HQ: `false`
- HQLL: `true`

Description: Enables low latency mode in the encoder and switches POC mode to 2.

Name: `AMF_VIDEO_ENCODER_FRAMESIZE`

Values: Width: 64 – 4096 Height: 64 – 4096

Default Value: `(0,0)`

Description: Frame width and height in pixels, maximum values are hardware-specific, should be queried through AMFCaps.

Name: `AMF_VIDEO_ENCODER_ASPECT_RATIO`

Values: `(1, 1)` ... `(INT_MAX, INT_MAX)`

Default Value: `(1,1)`

Description: Pixel aspect ratio.

Name: `AMF_VIDEO_ENCODER_MAX_NUM_REFRAMES`

Values: 0...16

Default Value: 4

Description: Maximum number of reference frames.

Name: `AMF_VIDEO_ENCODER_MAX_CONSECUTIVE_BPICTURES`

Values: 0...3

Default Value: 0

Description: Maximum number of consecutive B Pictures. It is suggested to set this value to 3 if `AMF_VIDEO_ENCODER_B_PIC_PATTERN` is not 0 as well as to enable Adaptive MiniGOP and PA.

Name: `AMF_VIDEO_ENCODER_ADAPTIVE_MINIGOP`

Values: `true`, `false`

Default Value: `false`

Description: Disable/Enable Adaptive MiniGOP, can enable with PA enabled.

Name: `AMF_VIDEO_ENCODER_PRE_ANALYSIS_ENABLE`

Values: `true, false`

Default Value associated with usages:

- Transcoding: `false`
- Ultra low latency: `false`
- Low latency: `false`
- Webcam: `false`
- HQ: `true`
- HQLL: `false`

Description: Some encoder properties require this property to be set. Enables the pre-analysis module. Refer to *AMF Video PreAnalysis API* reference for more details on the pre-analysis module and its settings under different usages.

Name: `AMF_VIDEO_ENCODER_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: 8

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

Name: `AMF_VIDEO_ENCODER_MAX_NUM_TEMPORAL_LAYERS`

Values: 1 … `Maximum number of temporal layers supported`

Default Value: 1

Description: Sets the maximum number of temporal layers. It shall not be exceeded by the number of temporal enhancement layers. The maximum number of temporal layers supported is determined by the corresponding encoder capability. This property is not supported on GPUs prior to Radeon RX 5000 Series or APUs prior to Ryzen 2000 U/H series.

Name: `AMF_VIDEO_ENCODER_ENABLE_SMART_ACCESS_VIDEO`

Values: `true, false`

Default Value: `false`

Description: When set to `true`, enables the SmartAccess Video feature, which optimally allocates the encoding task on supported APU/GPU pairings.

Name (prefix "AMF_VIDEO_ENCODER_")	Type
INPUT_COLOR_PROFILE	amf_int64
INPUT_TRANSFER_CHARACTERISTIC	amf_int64

Name (prefix "AMF_VIDEO_ENCODER_")	Type
INPUT_COLOR_PRIMARIES	amf_int64
OUTPUT_COLOR_PROFILE	amf_int64
OUTPUT_TRANSFER_CHARACTERISTIC	amf_int64
OUTPUT_COLOR_PRIMARIES	amf_int64

Table 5. Encoder color conversion parameters

Name: `AMF_VIDEO_ENCODER_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`

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_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_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_CCCS`

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_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`

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_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_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 (prefix "AMF_VIDEO_ENCODER_")	Type
TARGET_BITRATE	amf_int64
PEAK_BITRATE	amf_int64
RATE_CONTROL_METHOD	amf_int64
RATE_CONTROL_SKIP_FRAME_ENABLE	amf_bool
MIN_QP	amf_int64
MAX_QP	amf_int64
QP_I	amf_int64
QP_P	amf_int64
QP_B	amf_int64
QVBR_QUALITY_LEVEL	amf_int64
FRAMERATE	AMFRate
VBV_BUFFER_SIZE	amf_int64
INITIAL_VBV_BUFFER_FULLNESS	amf_int64
ENFORCE_HRD	amf_bool
MAX_AU_SIZE	amf_int64
B_PIC_DELTA_QP	amf_int64
REF_B_PIC_DELTA_QP	amf_int64
PREENCODE_ENABLE	amf_int64
FILLER_DATA_ENABLE	amf_bool
ENABLE_VBAQ	amf_bool

Table 6. Encoder rate-control parameters

Name: AMF_VIDEO_ENCODER_TARGET_BITRATE

Values: 10 000 - 100 000 000 bit/s

Default Value: 20 mbit/s

Description: Sets the target bitrate.

Name: AMF_VIDEO_ENCODER_PEAK_BITRATE

Values: 10 000 - 100 000 000 bit/s

Default Value: 30 mbit/s

Description: Sets the peak bitrate.

Name: AMF_VIDEO_ENCODER_RATE_CONTROL_METHOD

Values: AMF_VIDEO_ENCODER_RATE_CONTROL_METHOD_ENUM:
AMF_VIDEO_ENCODER_RATE_CONTROL_METHOD_UNKNOWN,
AMF_VIDEO_ENCODER_RATE_CONTROL_METHOD_CONSTANT_QP,
AMF_VIDEO_ENCODER_RATE_CONTROL_METHOD_CBR,
AMF_VIDEO_ENCODER_RATE_CONTROL_METHOD_PEAK_CONSTRAINED_VBR,
AMF_VIDEO_ENCODER_RATE_CONTROL_METHOD_LATENCY_CONSTRAINED_VBR,
AMF_VIDEO_ENCODER_RATE_CONTROL_METHOD_QUALITY_VBR,
AMF_VIDEO_ENCODER_RATE_CONTROL_METHOD_HIGH_QUALITY_VBR,
AMF_VIDEO_ENCODER_RATE_CONTROL_METHOD_HIGH_QUALITY_CBR

Default Value associated with usages:

- Transcoding: AMF_VIDEO_ENCODER_RATE_CONTROL_METHOD_PCVBR
- Ultra low latency: AMF_VIDEO_ENCODER_RATE_CONTROL_METHOD_LCVBR
- Low latency: AMF_VIDEO_ENCODER_RATE_CONTROL_METHOD_PCVBR
- Webcam: AMF_VIDEO_ENCODER_RATE_CONTROL_METHOD_PCVBR
- HQ: AMF_VIDEO_ENCODER_RATE_CONTROL_METHOD_QVBR/
  AMF_VIDEO_ENCODER_RATE_CONTROL_METHOD_PCVBR
- HQLL: AMF_VIDEO_ENCODER_RATE_CONTROL_METHOD_CBR

Description: Selects the rate control method:

- CQP – Constrained QP,
- CBR - Constant Bitrate,
- VBR - Peak Constrained VBR,
- VBR_LAT - Latency Constrained VBR,
- 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. Some RC parameters are not enabled with SVC encoding (e.g. all parameters related to B-pictures).
- 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 or VMAF).

Name: AMF_VIDEO_ENCODER_RATE_CONTROL_SKIP_FRAME_ENABLE

Values: true (on), false (off)

Default Value: Depends on USAGE

Description: Enables skip frame for rate control.

Name: AMF_VIDEO_ENCODER_MIN_QP

Values: 0 – 51

Default Value: 0

Description: Sets the minimum QP.

Name: AMF_VIDEO_ENCODER_MAX_QP

Values: 0 – 51

Default Value: 51

Description: Sets the maximum QP.

Name: AMF_VIDEO_ENCODER_QP_I

Values: 0 – 51

Default Value: 22

Description: Sets the constant QP for I-pictures. Remarks: Only available for CQP rate control method.

Name: AMF_VIDEO_ENCODER_QP_P

Values: 0 – 51

Default Value: 22

Description: Sets the constant QP for P-pictures. Remarks: Only available for CQP rate control method.

Name: AMF_VIDEO_ENCODER_QP_B

Values: 0 − 51

Default Value: 22

Description: Sets the constant QP for B-pictures. Remarks: Only available for CQP rate control method.

Name: AMF_VIDEO_ENCODER_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_FRAMERATE

Values: 1*FrameRateDen … 120* FrameRateDen

Default Value: 30 fps

Description: Frame rate numerator.

Name: AMF_VIDEO_ENCODER_VBV_BUFFER_SIZE

Values: 1000 − 100 000 000

Default Value associated with usages:

- Transcoding: 20 mbits
- Ultra low latency: 735 kbits
- Low latency: 4 mbits
- Webcam: 2 mbits
- HQ: 40 mbits
- HQLL: 10 mbits

Description: Sets the VBV buffer size in bits.

Name: AMF_VIDEO_ENCODER_INITIAL_VBV_BUFFER_FULLNESS

Values: 0 - 64

Default Value: 64

Description: Sets the initial VBV buffer fullness.

Name: AMF_VIDEO_ENCODER_ENFORCE_HRD

Values: true, false (On, Off)

Default Value associated with usages:

- Transcoding: `false`
- Ultra low latency: `true`
- Low latency: `false`
- Webcam: `false`
- HQ: `false`
- HQLL: `false`

Description: Disables/enables constraints on rate control to meet HRD model requirement(s) with peak_bitrate, VBV buffer size and VBV buffer fullness settings.

Name: `AMF_VIDEO_ENCODER_ENABLE_VBAQ`

Values: `true`, `false` (On, Off)

Default Value associated with usages:

- Transcoding: `false`
- Ultra low latency: `false`
- Low latency: `false`
- Webcam: `false`
- HQ: `true`
- HQLL: `true`

Description: Note: Cannot use when `RATE_CONTROL_METHOD` is CQP.

- VBAQ stands for *Variance Based Adaptive Quantization*.
- The basic idea of VBAQ: Human visual system is typically less sensitive to artifacts in highly textured area. In VBAQ mode, we use pixel variance to indicate the complexity of spatial texture. This allows us to allocate more bits to smoother areas. Enabling such feature leads to improvements in subjective visual quality with some content.Note: Cannot use when `RATE_CONTROL_METHOD` is CQP.

Name: `AMF_VIDEO_ENCODER_MAX_AU_SIZE`

Values: `0 - 100 000 000` bits

Default Value: `0`

Description: Maximum AU size in bits.

Name: `AMF_VIDEO_ENCODER_B_PIC_DELTA_QP`

Values: `-10 … 10`

Default Value associated with usages:

- Transcoding: 4
- Ultra low latency: 0
- Low latency: 4
- Webcam: 4
- HQ: 4
- HQLL: 4

Description: Selects the delta QP of non-reference B pictures with respect to I pictures. This feature is not supported by VCE 1.0. `BPicturesDeltaQP`, `ReferenceBPicturesDeltaQP`, `IntraRefreshNumMBsPerSlot`, `BPicturesPattern` and `BReferenceEnable` parameters are available only when:

- `MaxOfReferenceFrames` is greater than 1
- `NumOfLTR` is 0 (LTR is not used)

Name: `AMF_VIDEO_ENCODER_REF_B_PIC_DELTA_QP`

Values: -10 … 10

Default Value associated with usages:

- Transcoding: 2
- Ultra low latency: 0
- Low latency: 2
- Webcam: 2
- HQ: 2
- HQLL: 2

Description: Selects delta QP of reference B pictures with respect to I pictures. This feature is not supported by VCE 1.0. `BPicturesDeltaQP`, `ReferenceBPicturesDeltaQP`, `IntraRefreshNumMBsPerSlot`, `BPicturesPattern` and `BReferenceEnable` parameters are available only when:

- `MaxOfReferenceFrames` is greater than 1
- `NumOfLTR` is 0 (LTR is not used)

Name: `AMF_VIDEO_ENCODER_PREENCODE_ENABLE`

Values: `AMF_VIDEO_ENCODER_PREENCODE_DISABLED`, `AMF_VIDEO_ENCODER_PREENCODE_ENABLED`

Default Value associated with usages:

- Transcoding: `AMF_VIDEO_ENCODER_PREENCODE_DISABLED`
- Ultra low latency: `AMF_VIDEO_ENCODER_PREENCODE_DISABLED`
- Low latency: `AMF_VIDEO_ENCODER_PREENCODE_DISABLED`
- Webcam: `AMF_VIDEO_ENCODER_PREENCODE_DISABLED`
- HQ: `AMF_VIDEO_ENCODER_PREENCODE_DISABLED`
- HQLL: `AMF_VIDEO_ENCODER_PREENCODE_DISABLED`

Description: Enables or disables rate control pre-analysis.

Name: `AMF_VIDEO_ENCODER_FILLER_DATA_ENABLE`

Values: `true`, `false`

Default Value: `false`

Description: Enables/disables filler data to maintain constant bit rate.

Name (prefix "AMF_VIDEO_ENCODER_")	Type
HEADER_INSERTION_SPACING	amf_int64
IDR_PERIOD	amf_int64
INTRA_PERIOD	amf_int64
DE_BLOCKING_FILTER	amf_bool
INTRA_REFRESH_NUM_MBS_PER_SLOT	amf_int64
SLICES_PER_FRAME	amf_int64
B_PIC_PATTERN	amf_int64
B_REFERENCE_ENABLE	amf_int64
CABAC_ENABLE	amf_int64
HIGH_MOTION_QUALITY_BOOST_ENABLE	amf_bool

Table 7. Encoder picture-control parameters

Name: `AMF_VIDEO_ENCODER_HEADER_INSERTION_SPACING`

Values: `0 … 1000`

Default Value: `0`

Description: Sets the headers insertion spacing.

Name: `AMF_VIDEO_ENCODER_IDR_PERIOD`

Values: `0 … 1000`

Default Value associated with usages:

- Transcoding: `240`
- Ultra low latency: `240`
- Low latency: `240`
- Webcam: `30`
- HQ: `240`
- HQLL: `120`

Description: Sets IDR period. IDRPeriod = 0 turns IDR off. To get SPS/PPS for every IDR, header insertion spacing has to be the same as IDR period.

Name: `AMF_VIDEO_ENCODER_INTRA_PERIOD`

Values: `0 – 1000`

Default Value: `0`

Description: Number of frames between two intra frames.

Name: AMF_VIDEO_ENCODER_DE_BLOCKING_FILTER

Values: `true` (on), `false` (off)

Default Value: `true`

Description: Enable/disable the de-blocking filter.

Name: AMF_VIDEO_ENCODER_INTRA_REFRESH_NUM_MBS_PER_SLOT

Values: `0` - #MBs per frame

Default Value associated with usages:

- Transcoding: `0`
- Ultra low latency: 255
- Low latency: 255
- Webcam: `0`
- HQ: `0`
- HQLL: `0`

Description: Sets the number of slices per frame. `BPicturesDeltaQP`, `ReferenceBPicturesDeltaQP`, `IntraRefreshNumMBsPerSlot`, `BPicturesPattern` and `BReferenceEnable` parameters are available only when:

- `MaxOfReferenceFrames` is greater than `1`
- `NumOfLTR` is `0` (LTR is not used)

Name: AMF_VIDEO_ENCODER_SLICES_PER_FRAME

Values: `1` - #MBs per frame

Default Value: `1`

Description: Sets the number of slices per frame.

Name: AMF_VIDEO_ENCODER_B_PIC_PATTERN

Values: `0`, `1`, `2`, `3`

Default Value: `0`

Description: Sets the number of consecutive B-pictures in a GOP. `BPicturesPattern = 0` indicates that B-pictures are not used. This feature is not supported by VCE 1.0. `BPicturesDeltaQP`, `ReferenceBPicturesDeltaQP`, `IntraRefreshNumMBsPerSlot`, `BPicturesPattern` and `BReferenceEnable` parameters are available only when:

- `MaxOfReferenceFrames` is greater than `1`
- `NumOfLTR` is `0` (LTR is not used)

Name: AMF_VIDEO_ENCODER_B_REFERENCE_ENABLE

Values: `true (on)`, `false (off)`

Default Value associated with usages:

- Transcoding: `true`
- Ultra low latency: `true`
- Low latency: `true`
- Webcam: `true`
- HQ: `true`
- HQLL: `true`

Description: Enables or disables using B-pictures as references. This feature is not supported by VCE 1.0. `BPicturesDeltaQP`, `ReferenceBPicturesDeltaQP`, `IntraRefreshNumMBsPerSlot`, `BPicturesPattern` and `BReferenceEnable` parameters are available only when:

- `MaxOfReferenceFrames` is greater than `1`
- `NumOfLTR` is `0` (LTR is not used)

Name: `AMF_VIDEO_ENCODER_CABAC_ENABLE`

Values: `AMF_VIDEO_ENCODER_CODING_ENUM`: `AMF_VIDEO_ENCODER_UNDEFINED`, `AMF_VIDEO_ENCODER_CABAC`, `AMF_VIDEO_ENCODER_CALV`

Default Value: `AMF_VIDEO_ENCODER_UNDEFINED`

Description: Encoder coding method, when Undefined is selected, the behavior is profile-specific: CALV for Baseline, CABAC for Main and High.

Name: `AMF_VIDEO_ENCODER_HIGH_MOTION_QUALITY_BOOST_ENABLE`

Values: `true`, `false`

Default Value associated with usages:

- Transcoding: `false`
- Ultra low latency: `false`
- Low latency: `false`
- Webcam: `false`
- HQ: `true`
- HQLL: `true`

Description: Enable High motion quality boost mode. It pre-analysis the motion of the video and use the information for better encoding.

Name (prefix "AMF_VIDEO_ENCODER_")	Type
EXTRADATA	AMFBufferPtr
SCANTYPE	amf_int64
QUALITY_PRESET	amf_int64

Name (prefix "AMF_VIDEO_ENCODER_")	Type
FULL_RANGE_COLOR	amf_bool
PICTURE_TRANSFER_MODE	amf_int64
QUERY_TIMEOUT	amf_int64
INPUT_QUEUE_SIZE	amf_int64
OUTPUT_MODE	amf_int64
PSNR_FEEDBACK	amf_bool
SSIM_FEEDBACK	amf_bool
BLOCK_QP_FEEDBACK	amf_bool

Table 8. Encoder miscellaneous parameters

Name: AMF_VIDEO_ENCODER_EXTRADATA

Values: AMFBufferPtr

Default Value: NULL

Description: SPS/PPS buffer in Annex B format - read-only.

Name: AMF_VIDEO_ENCODER_SCANTYPE

Values: AMF_VIDEO_ENCODER_SCANTYPE_ENUM: AMF_VIDEO_ENCODER_SCANTYPE_PROGRESSIVE, AMF_VIDEO_ENCODER_SCANTYPE_INTERLACED

Default Value: AMF_VIDEO_ENCODER_SCANTYPE_PROGRESSIVE

Description: Selects progressive or interlaced scan.

Name: AMF_VIDEO_ENCODER_QUALITY_PRESET

Values: AMF_VIDEO_ENCODER_QUALITY_PRESET_ENUM: AMF_VIDEO_ENCODER_QUALITY_PRESET_BALANCED, AMF_VIDEO_ENCODER_QUALITY_PRESET_SPEED, AMF_VIDEO_ENCODER_QUALITY_PRESET_QUALITY

Default Value associated with usages:

- Transcoding: AMF_VIDEO_ENCODER_QUALITY_PRESET_BALANCED
- Ultra low latency: AMF_VIDEO_ENCODER_QUALITY_PRESET_SPEED
- Low latency: AMF_VIDEO_ENCODER_QUALITY_PRESET_SPEED
- Webcam: AMF_VIDEO_ENCODER_QUALITY_PRESET_SPEED
- HQ: AMF_VIDEO_ENCODER_QUALITY_PRESET_QUALITY
- HQLL: AMF_VIDEO_ENCODER_QUALITY_PRESET_QUALITY

Description: Selects the quality preset in HW to balance between encoding speed and video quality.

Name: AMF_VIDEO_ENCODER_FULL_RANGE_COLOR

Values: true, false

Default Value: false

Description: True indicates that the YUV range is 0...255.

Name: AMF_VIDEO_ENCODER_PICTURE_TRANSFER_MODE

Values: AMF_VIDEO_ENCODER_PICTURE_TRANSFER_MODE_ENUM:
AMF_VIDEO_ENCODER_PICTURE_TRANSFER_MODE_ON, AMF_VIDEO_ENCODER_PICTURE_TRANSFER_MODE_OFF

Default Value: AMF_VIDEO_ENCODER_PICTURE_TRANSFER_MODE_OFF

Description: The application can turn on this flag for a specific input picture to allow dumping the reconstructed picture and/or injecting a reference picture.

Name: AMF_VIDEO_ENCODER_QUERY_TIMEOUT

Values: ENCODER_TIMEOUT

Default Value associated with usages:

- Transcoding: 0 (no wait)
- Ultra low latency: 0 (no wait)
- Low latency: 0 (no wait)
- Webcam: 0 (no wait)
- HQ: 50
- HQLL: 50

Description: Timeout for a QueryOutput call in ms. Setting this to a nonzero value will reduce polling load when QueryOutput is called; it will be blocked until the frame is ready or until the timeout is reached.

Name: AMF_VIDEO_ENCODER_INPUT_QUEUE_SIZE

Values: 1 ... 32

Default Value: 16

Description: Set encoder input queue size. For high-resolution sequence, recommend to set a smaller value to save storage. For low-resolution sequence, recommend to set a larger value to improve encoding speed.

Name: AMF_VIDEO_ENCODER_OUTPUT_MODE

Values: AMF_VIDEO_ENCODER_OUTPUT_MODE_ENUM: AMF_VIDEO_ENCODER_OUTPUT_MODE_FRAME,
AMF_VIDEO_ENCODER_OUTPUT_MODE_TILE

Default Value: AMF_VIDEO_ENCODER_OUTPUT_MODE_FRAME

Description: Defines encoder output mode.

Name: AMF_VIDEO_ENCODER_PSNR_FEEDBACK

Values: true, false

Default Value: false

Description: Signal encoder to calculate PSNR score.

Name: AMF_VIDEO_ENCODER_SSIM_FEEDBACK

Values: true, false

Default Value: false

Description: Signal encoder to calculate SSIM score.

Name: AMF_VIDEO_ENCODER_BLOCK_QP_FEEDBACK

Values: true, false

Default Value: false

Description: Signal encoder to collect and feedback block level QP values.

Name (prefix "AMF_VIDEO_ENCODER_")	Type
MOTION_HALF_PIXEL	amf_bool
MOTION_QUARTERPIXEL	amf_bool

Table 9. Encoder miscellaneous parameters

Name: AMF_VIDEO_ENCODER_MOTION_HALF_PIXEL

Values: true (on), false (off)

Default Value: true

Description: Turns on/off half-pixel motion estimation.

Name: AMF_VIDEO_ENCODER_MOTION_QUARTERPIXEL

Values: true (on), false (off)

Default Value: false

Description: Turns on/off quarter-pixel motion estimation.

Name (prefix "AMF_VIDEO_ENCODER_")	Type
NUM_TEMPORAL_ENHANCMENT_LAYERS	amf_int64

Table 10. Encoder SVC parameters

Name: `AMF_VIDEO_ENCODER_NUM_TEMPORAL_ENHANCMENT_LAYERS`

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

Default Value: `1`

Description: Sets the number of temporal enhancement 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 enhancement layers will be delayed until the start of the next temporal GOP.
- B-pictures and Intra-refresh features are not supported with SVC.

`NumOfTemporalEnhancmentLayers` shall not exceed `MaxNumOfTemporalLayers`. SVC is supported in all usages on Radeon RX 5000 Series or newer GPUs and Ryzen 2000 U/H series or newer APUs. It is only supported in Webcam usage on products prior to the aforementioned.

Name (prefix "AMF_VIDEO_ENCODER_")	Type
TL\<TL_Num\>.QL\<QL_Num\>.\<Parameter_name\>	

Table 11. Encoder SVC per-layer parameters

Name: `AMF_VIDEO_ENCODER_TL<TL_Num>.QL<QL_Num>.<Parameter_name>`

Values: Parameter-specific values

Default Value: `N\A`

Description: Configures rate-control parameter per SVC layer.

- TL_Num — temporal layer number
- QL_Num — quality layer number
- Parameter_name — rate-control parameter name (see below)

Rate-control parameters supported

- TargetBitrate
- PeakBitrate
- VBVBufferSize
- FrameRate
- MinQP
- MaxQP
- QPI
- QPP
- FillerDataEnable
- RateControlSkipFrameEnable
- EnforceHRD

- MaxAUSize

(Refer to rate-control parameters section of this table for more details)

Remarks: Quality layers are not supported on VCE 1.0. "QL0" must be used for quality layers.

Table A-2. Input frame and encoded data parameters

Name (prefix "AMF_VIDEO_ENCODER_")	Type
INSERT_SPS	amf_bool
INSERT_PPS	amf_bool
INSERT_AUD	amf_bool
PICTURE_STRUCTURE	amf_int64
FORCE_PICTURE_TYPE	amf_int64
END_OF_SEQUENCE	amf_bool
END_OF_STREAM	amf_bool
MARK_CURRENT_WITH_LTR_INDEX	amf_int64
FORCE_LTR_REFERENCE_BITFIELD	amf_int64
ROI_DATA	AMF_SURFACE_GRAY32
STATISTICS_FEEDBACK	amf_bool
REFERENCE_PICTURE	AMFInterfacePtr

Table 12. Frame per-submission parameters

Name: `AMF_VIDEO_ENCODER_INSERT_SPS`

Values: `true` (on), `false` (off)

Default Value: `false`

Description: Inserts SPS.

Name: `AMF_VIDEO_ENCODER_INSERT_PPS`

Values: `true` (on), `false` (off)

Default Value: `false`

Description: Inserts PPS.

Name: `AMF_VIDEO_ENCODER_INSERT_AUD`

Values: `true` (on), `false` (off)

Default Value: `false`

Description: Inserts AUD.

Name: `AMF_VIDEO_ENCODER_PICTURE_STRUCTURE`

Values: `AMF_VIDEO_ENCODER_PICTURE_STRUCTURE_ENUM`: `AMF_VIDEO_ENCODER_PICTURE_STRUCTURE_NONE`, `AMF_VIDEO_ENCODER_PICTURE_STRUCTURE_FRAME`, `AMF_VIDEO_ENCODER_PICTURE_STRUCTURE_TOP_FIELD`, `AMF_VIDEO_ENCODER_PICTURE_STRUCTURE_BOTTOM_FIELD`

Default Value: `AMF_VIDEO_ENCODER_PICTURE_STRUCTURE_FRAME`

Description: Picture structure.

Name: `AMF_VIDEO_ENCODER_FORCE_PICTURE_TYPE`

Values: `AMF_VIDEO_ENCODER_PICTURE_TYPE_ENUM`: `AMF_VIDEO_ENCODER_PICTURE_TYPE_NONE`, `AMF_VIDEO_ENCODER_PICTURE_TYPE_SKIP`, `AMF_VIDEO_ENCODER_PICTURE_TYPE_IDR`, `AMF_VIDEO_ENCODER_PICTURE_TYPE_I`, `AMF_VIDEO_ENCODER_PICTURE_TYPE_P`, `AMF_VIDEO_ENCODER_PICTURE_TYPE_B`

Default Value: `AMF_VIDEO_ENCODER_PICTURE_TYPE_NONE`

Description: Forces the picture type (to use this feature, set `AMF_VIDEO_ENCODER_IDR_PERIOD` to 0) determined by the picture management with the following precedence IDR, I, P and B. Force picture type will only take effect to override type with lower precedence. B feature is not supported by VCE 1.0.

Name: `AMF_VIDEO_ENCODER_END_OF_SEQUENCE`

Values: `true` (on), `false` (off)

Default Value: `false`

Description: End of sequence.

Name: `AMF_VIDEO_ENCODER_END_OF_STREAM`

Values: `true` (on), `false` (off)

Default Value: `false`

Description: End of stream.

Name: `AMF_VIDEO_ENCODER_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_FORCE_LTR_REFERENCE_BITFIELD

Values: Bitfield MaxOfLTRFrames (max possible 16 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_ROI_DATA

Values: Video surface in AMF_SURFACE_GRAY32 format

Default Value: N\A

Description: Importance value for each 16x16 macro block ranges from 0 (least important) to 10 (most important), stored in 32bit unsigned format.

Name: AMF_VIDEO_ENCODER_STATISTICS_FEEDBACK

Values: true (on), false (off)

Default Value: false

Description: Instruct encoder to collect and feedback statistics.

Name: AMF_VIDEO_ENCODER_REFERENCE_PICTURE

Values: AMFSurface

Default Value: N\A

Description: Injected reference picture. Valid with `AMF_VIDEO_ENCODER_PICTURE_TRANSFER_MODE` turned on.

Name (prefix "AMF_VIDEO_ENCODER_")	Type
OUTPUT_DATA_TYPE	amf_int64
OUTPUT_MARKED_LTR_INDEX	amf_int64
OUTPUT_REFERENCED_LTR_INDEX_BITFIELD	amf_int64
OUTPUT_TEMPORAL_LAYER	amf_int64
OUTPUT_BUFFER_TYPE	amf_int64
RECONSTRUCTED_PICTURE	AMFSurface

Table 13. Encoded data parameters

Name: `AMF_VIDEO_ENCODER_OUTPUT_DATA_TYPE`

Values: `AMF_VIDEO_ENCODER_OUTPUT_DATA_TYPE_ENUM: AMF_VIDEO_ENCODER_OUTPUT_DATA_TYPE_IDR,`
`AMF_VIDEO_ENCODER_OUTPUT_DATA_TYPE_I, AMF_VIDEO_ENCODER_OUTPUT_DATA_TYPE_P,`
`AMF_VIDEO_ENCODER_OUTPUT_DATA_TYPE_B`

Default Value: N/A

Description: Type of encoded data. B feature is not supported by VCE 1.0.

Name: `AMF_VIDEO_ENCODER_OUTPUT_MARKED_LTR_INDEX`

Values: `-1 … MaxOfLTRFrames -1`

Default Value: `-1`

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_OUTPUT_REFERENCED_LTR_INDEX_BITFIELD`

Values: Bitfield `MaxOfLTRFrames` (max possible 16 bits)

Default Value: `0`

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).

Name: `AMF_VIDEO_ENCODER_OUTPUT_TEMPORAL_LAYER`

Values: `0 … Maximum number of temporal layers supported - 1`

Default Value: N\A

Description: Temporal layer of the encoded picture.

Name: `AMF_VIDEO_ENCODER_OUTPUT_BUFFER_TYPE`

Values: `AMF_VIDEO_ENCODER_OUTPUT_BUFFER_TYPE_ENUM`:
`AMF_VIDEO_ENCODER_OUTPUT_BUFFER_TYPE_FRAME`,`AMF_VIDEO_ENCODER_OUTPUT_BUFFER_TYPE_TILE`,
`AMF_VIDEO_ENCODER_OUTPUT_BUFFER_TYPE_TILE_LAST`

Default Value: N\A

Description: Encoder output buffer type.

Name: `AMF_VIDEO_ENCODER_RECONSTRUCTED_PICTURE`

Values: `AMFSurface`

Default Value: N\A

Description: Reconstructed picture. Valid with `AMF_VIDEO_ENCODER_PICTURE_TRANSFER_MODE` turned on.

Table A-3. Encoder capabilities exposed in AMFCaps interface

Name (prefix with AMF_VIDEO_ENCODER_CAP_)	Type
MAX_BITRATE	amf_int64
NUM_OF_STREAMS	amf_int64
MAX_PROFILE	amf_int64
MAX_LEVEL	amf_int64
BFRAMES	amf_bool
MIN_REFERENCE_FRAMES	amf_int64
MAX_REFERENCE_FRAMES	amf_int64
MAX_TEMPORAL_LAYERS	amf_int64
FIXED_SLICE_MODE	amf_bool
NUM_OF_HW_INSTANCES	amf_int64
COLOR_CONVERSION	amf_int64
PRE_ANALYSIS	amf_bool
ROI	amf_bool
MAX_THROUGHPUT	amf_int64
REQUESTED_THROUGHPUT	amf_int64
QUERY_TIMEOUT_SUPPORT	amf_bool

Name (prefix with AMF_VIDEO_ENCODER_CAP_)	Type
SUPPORT_SLICE_OUTPUT	amf_bool

Table 14. Encoder capabilities exposed in AMFCaps interface

Name: `AMF_VIDEO_ENCODER_CAP_MAX_BITRATE`

Values: Integers, >=0

Description: Maximum bit rate in bits.

Name: `AMF_VIDEO_ENCODER_CAP_NUM_OF_STREAMS`

Values: Integers, >=0

Description: Maximum number of encode streams supported.

Name: `AMF_VIDEO_ENCODER_CAP_MAX_PROFILE`

Values: `AMF_VIDEO_ENCODER_PROFILE_ENUM: AMF_VIDEO_ENCODER_PROFILE_BASELINE,`
`AMF_VIDEO_ENCODER_PROFILE_MAIN,AMF_VIDEO_ENCODER_PROFILE_HIGH,`
`AMF_VIDEO_ENCODER_PROFILE_CONSTRAINED_BASELINE,AMF_VIDEO_ENCODER_PROFILE_CONSTRAINED_HIGH`

Description: Maximum value of encoder profile.

Name: `AMF_VIDEO_ENCODER_CAP_MAX_LEVEL`

Values: `AMF_VIDEO_ENCODER_H264_LEVEL_ENUM: AMF_LEVEL_1, AMF_LEVEL_1_1, AMF_LEVEL_1_2,`
`AMF_LEVEL_1_3, AMF_LEVEL_2, AMF_LEVEL_2_1,AMF_LEVEL_2_2, AMF_LEVEL_3,`
`AMF_LEVEL_3_1,AMF_LEVEL_3_2, AMF_LEVEL_4, AMF_LEVEL_4_1, AMF_LEVEL_4_2,AMF_LEVEL_5,`
`AMF_LEVEL_5_1, AMF_LEVEL_5_2, AMF_LEVEL_6, AMF_LEVEL_6_1, AMF_LEVEL_6_2`

Description: Maximum value of codec level.

Name: `AMF_VIDEO_ENCODER_CAP_BFRAMES`

Values: `true, false`

Description: Are B-Frames supported?

Name: `AMF_VIDEO_ENCODER_CAP_MIN_REFERENCE_FRAMES`

Description: Minimum number of reference frames.

Name: `AMF_VIDEO_ENCODER_CAP_MAX_REFERENCE_FRAMES`

Description: Maximum number of reference frames.

Name: `AMF_VIDEO_ENCODER_CAP_MAX_NUM_TEMPORAL_LAYERS`

Values: 1 … `Maximum number of temporal layers supported`

Description: The cap of maximum number of temporal layers.

Name: `AMF_VIDEO_ENCODER_CAP_FIXED_SLICE_MODE`

Values: `true, false`

Description: Is fixed slice mode supported?

Name: `AMF_VIDEO_ENCODER_CAP_NUM_OF_HW_INSTANCES`

Values: 0… `number of instances - 1`

Description: Number of HW encoder instances.

Name: `AMF_VIDEO_ENCODER_CAP_COLOR_CONVERSION`

Values: `AMF_ACCELERATION_TYPE`

Description: Type of supported color conversion.

Name: `AMF_VIDEO_ENCODER_CAP_PRE_ANALYSIS`

Values: `true, false`

Description: Pre analysis module is available.

Name: `AMF_VIDEO_ENCODER_CAP_ROI`

Values: `true, false`

Description: ROI map support is available.

Name: `AMF_VIDEO_ENCODER_CAP_MAX_THROUGHPUT`

Values: Integers, >=0

Description: Maximum throughput for encoder in MB (16 x 16 pixels).

Name: `AMF_VIDEO_ENCODER_CAP_REQUESTED_THROUGHPUT`

Values: Integers, >=0

Description: Current total requested throughput for encoder in MB (16 x 16 pixels).

Name: `AMF_VIDEO_ENCODER_CAP_QUERY_TIMEOUT_SUPPORT`

Values: `true, false`

Description: Timeout supported for QueryOutout call.

Name: `AMF_VIDEO_ENCODER_CAP_SUPPORT_SLICE_OUTPUT`

Values: `true`, `false`

Description: If tile output is supported.

Table A-4. Encoder statistics feedback

Name (prefix "AMF_VIDEO_ENCODER_")	Type
STATISTIC_FRAME_QP	amf_int64
STATISTIC_AVERAGE_QP	amf_int64
STATISTIC_MAX_QP	amf_int64
STATISTIC_MIN_QP	amf_int64
STATISTIC_PIX_NUM_INTRA	amf_int64
STATISTIC_PIX_NUM_INTER	amf_int64
STATISTIC_PIX_NUM_SKIP	amf_int64
STATISTIC_BITCOUNT_RESIDUAL	amf_int64
STATISTIC_BITCOUNT_MOTION	amf_int64
STATISTIC_BITCOUNT_INTER	amf_int64
STATISTIC_BITCOUNT_INTRA	amf_int64
STATISTIC_BITCOUNT_ALL_MINUS_HEADER	amf_int64
STATISTIC_MV_X	amf_int64
STATISTIC_MV_Y	amf_int64
STATISTIC_RD_COST_FINAL	amf_int64
STATISTIC_RD_COST_INTRA	amf_int64
STATISTIC_RD_COST_INTER	amf_int64
STATISTIC_SATD_FINAL	amf_int64
STATISTIC_SATD_INTRA	amf_int64
STATISTIC_SATD_INTER	amf_int64
STATISTIC_VARIANCE	amf_int64

Table 15. Encoder statistics feedback

Name: `AMF_VIDEO_ENCODER_STATISTIC_FRAME_QP`

Description: QP of the first encoded macroblocks in a picture.

Name: `AMF_VIDEO_ENCODER_STATISTIC_AVERAGE_QP`

Description: Average QP of all encoded macroblocks in a picture.

Name: `AMF_VIDEO_ENCODER_STATISTIC_MAX_QP`

Description: Max QP among all encoded macroblocks in a picture.

Name: `AMF_VIDEO_ENCODER_STATISTIC_MIN_QP`

Description: Min QP among all encoded macroblocks in a picture.

Name: `AMF_VIDEO_ENCODER_STATISTIC_PIX_NUM_INTRA`

Description: Number of intra-coded pixels.

Name: `AMF_VIDEO_ENCODER_STATISTIC_PIX_NUM_INTER`

Description: Number of inter-coded pixels.

Name: `AMF_VIDEO_ENCODER_STATISTIC_PIX_NUM_SKIP`

Description: Number of skip-coded pixels.

Name: `AMF_VIDEO_ENCODER_STATISTIC_BITCOUNT_RESIDUAL`

Description: Frame level bit count of residual data.

Name: `AMF_VIDEO_ENCODER_STATISTIC_BITCOUNT_MOTION`

Description: Frame level bit count of motion vectors.

Name: `AMF_VIDEO_ENCODER_STATISTIC_BITCOUNT_INTER`

Description: Frame level bit count of inter macroblocks.

Name: `AMF_VIDEO_ENCODER_STATISTIC_BITCOUNT_INTRA`

Description: Frame level bit count of intra macroblocks.

Name: `AMF_VIDEO_ENCODER_STATISTIC_BITCOUNT_ALL_MINUS_HEADER`

Description: Frame level bit count of the bitstream excluding header.

Name: `AMF_VIDEO_ENCODER_STATISTIC_MV_X`

Description: Accumulated absolute values of MVX.

Name: AMF_VIDEO_ENCODER_STATISTIC_MV_Y

Description: Accumulated absolute values of MVY.

Name: AMF_VIDEO_ENCODER_STATISTIC_RD_COST_FINAL

Description: Frame level final RD cost.

Name: AMF_VIDEO_ENCODER_STATISTIC_RD_COST_INTRA

Description: Frame level RD cost for intra mode.

Name: AMF_VIDEO_ENCODER_STATISTIC_RD_COST_INTER

Description: Frame level RD cost for inter mode.

Name: AMF_VIDEO_ENCODER_STATISTIC_SATD_FINAL

Description: Frame level final SATD.

Name: AMF_VIDEO_ENCODER_STATISTIC_SATD_INTRA

Description: Frame level SATD for intra mode.

Name: AMF_VIDEO_ENCODER_STATISTIC_SATD_INTER

Description: Frame level SATD for inter mode.

Name: AMF_VIDEO_ENCODER_STATISTIC_VARIANCE

Description: Frame level variance for full encoding.

Table A-5. Encoder PSNR/SSIM feedback

Name (prefix "AMF_VIDEO_ENCODER_")	Type
STATISTIC_PSNR_Y	amf_double
STATISTIC_PSNR_U	amf_double
STATISTIC_PSNR_V	amf_double
STATISTIC_PSNR_ALL	amf_double
STATISTIC_SSIM_Y	amf_double
STATISTIC_SSIM_U	amf_double
STATISTIC_SSIM_V	amf_double
STATISTIC_SSIM_ALL	amf_double

Name: AMF_VIDEO_ENCODER_STATISTIC_PSNR_Y

Description: PSNR Y.

Name: AMF_VIDEO_ENCODER_STATISTIC_PSNR_U

Description: PSNY U.

Name: AMF_VIDEO_ENCODER_STATISTIC_PSNR_V

Description: PSNR V.

Name: AMF_VIDEO_ENCODER_STATISTIC_PSNR_ALL

Description: PSNR YUV.

Name: AMF_VIDEO_ENCODER_STATISTIC_SSIM_Y

Description: SSIM Y.

Name: AMF_VIDEO_ENCODER_STATISTIC_SSIM_U

Description: SSIM U.

Name: AMF_VIDEO_ENCODER_STATISTIC_SSIM_V

Description: SSIM V.

Name: AMF_VIDEO_ENCODER_STATISTIC_SSIM_ALL

Description: SSIM YUV.

Table A-6. Deprecated

Name (prefix "AMF_VIDEO_ENCODER_")	Type	Deprecated Starting
MAX_INSTANCES	amf_int64	v1.4.21
MULTI_INSTANCE_MODE	amf_bool	v1.4.21
CURRENT_QUEUE	amf_int64	v1.4.23
RATE_CONTROL_PREANALYSIS_ENABLE	amf_int64	v1.4.21
CAPS_QUERY_TIMEOUT_SUPPORT	amf_bool	v1.4.18

Table 17. Deprecated

Name: AMF_VIDEO_ENCODER_MAX_INSTANCES

Values: 1, 2

Default Value: 1

Description: Hardware-dependent, only some hardware supports 2 instances.

Name: AMF_VIDEO_ENCODER_MULTI_INSTANCE_MODE

Values: `true`, `false`

Default Value: `false`

Description: Enables or disables multi-instance mode.

Name: AMF_VIDEO_ENCODER_CURRENT_QUEUE

Values: `0`, `1`

Default Value: `0`

Description: Selects the encoder instance frames are being submitted to.

Name: AMF_VIDEO_ENCODER_RATE_CONTROL_PREANALYSIS_ENABLE

Values: AMF_VIDEO_ENCODER_PREENCODE_DISABLED, AMF_VIDEO_ENCODER_PREENCODE_ENABLED

Default Value: `0`

Description: Enables pre-encode assisted rate control. Deprecated, please use AMF_VIDEO_ENCODER_PREENCODE_ENABLE instead.

Name: AMF_VIDEO_ENCODER_CAPS_QUERY_TIMEOUT_SUPPORT

Values: `true`, `false`

Default Value: N/A

Description: Timeout supported for `QueryOutput` call. Deprecated, please use AMF_VIDEO_ENCODER_CAP_QUERY_TIMEOUT_SUPPORT instead.