Advanced Micro Devices

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

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; 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

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

This document provides a complete description of the AMD Advanced Media Framework (AMF) Video Encoder Component. This component exposes the AMD Video Compression Engine (VCE), which provides hardware accelerated H.264 video encoding functionality.

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

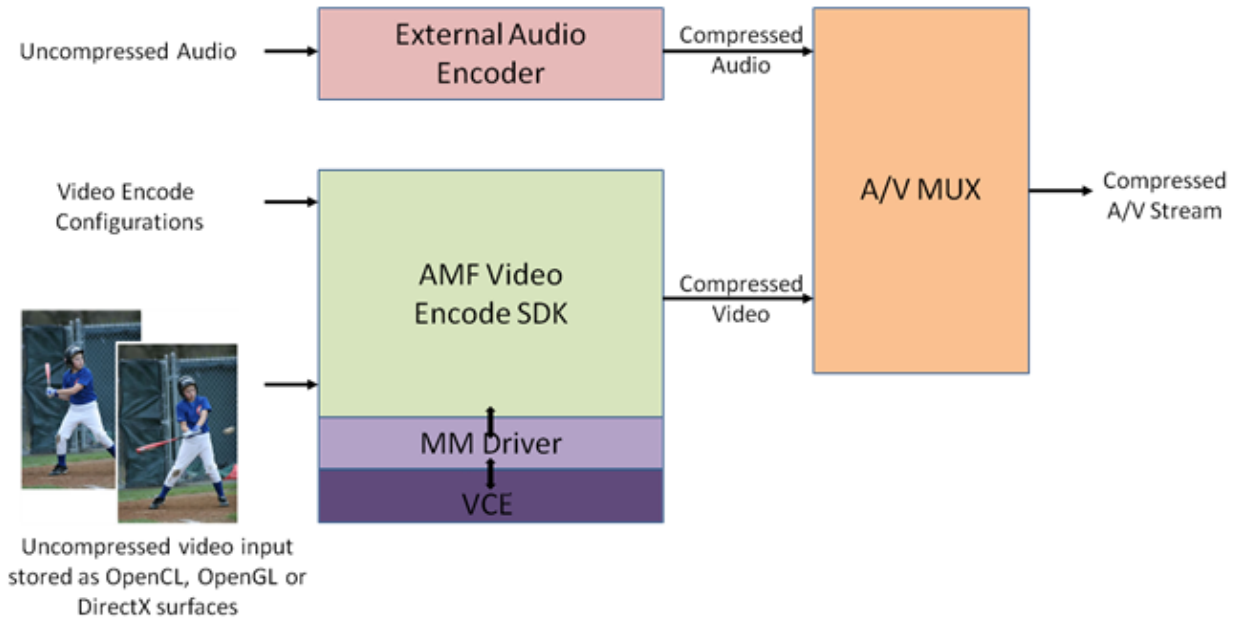

Figure 1 — System overview of the AMF Video Encode SDK

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 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, with scalable video coding (SVC) support.

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, and
- Webcam 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

For define frame-rate SVC parameters per layer the next format must be used:

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

For example, to configure "Target bitrate" for temporal layer 2 and quality layer 1 the next parameter name uses:

```
"TL2.QL0.TargetBitrate"
```

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 macro block, ranging from 0 to 10, stored in 32bit unsinged 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.

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 VCEEncoderRaw application:

Category	Name	Values	Description
Miscellaneous parameters	ApplyTo	Frame number	Forces all subsequent configuration parameters to be applied to a specific frame
	Input	File name, relative or absolute path	Input file with frames (YUV420, NV12 or BGRA)
	Output	File name, relative or absolute path	Output H.264 file for encoded data
	DX9	Flag (without any values)	Forces Direct3D 9 (default Direct3D 11)
	OpenCL	Flag (without any values)	Forces OpenCL
	MTMode	Flag (without any values)	Enables creating or reading from file of frames in separate thread.
	PerfStat	Flag (without any values)	Enables showing a performance statistic

Additional parameters of VCEEncoderD3D application:

Category	Name	Values	Description
Miscellaneous parameters	Frames	Number of frames to be encoded	Number of frames to be encoded
	ApplyTo	Frame number	Forces all subsequent configuration parameters to be applied to a specific frame
	Output	File name, relative or absolute path	Output H.264 file for encoded data
	DX9	Flag (without any values)	Use Direct3D 9 (default Direct3D 11) for rendering
	DX9EX	Flag (without any values)	The same as DX9 but using Device9Ex instead Device9
	OpenGL	Flag (without any values)	Use OpenGL for rendering
	Windowmode	Flag (without any values)	Shows rendering window for D3D sample application

	MTMode	Flag (without any values)	Enables creating or reading from file of frames in separate thread. Doesn't work for OpenGL.
	PerfStat	Flag (without any values)	Enables showing a performance statistic

3.2 Command line example

3.2.1 Transcoding application (TranscodingHW.exe)

```
VCEEncoderRaw.exe -input input.h264 -output out.h264 -width 1280 -height 720 -usage
transcoding -rateControlMethod cbr -targetBitrate 500000 -targetBitrate 100000
```

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 -usage
transcoding - rateControlMethod cbr -targetBitrate 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 500kbits/sec.

Annex A: Encoding & frame parameters description

Table A-1. Encoder configuration parameters

Category	Name	Values	Description
Encoder initialization parameters	AMF_VIDEO_ENCODER_USAGE	0, 1, 2, 3 (Transcoding, UltraLowLatency, LowLatency, Webcam)	Selects the AMF usage (see Section 1.2)
	AMF_VIDEO_ENCODER_PROFILE	66, 77, 100 (Baseline, Main, High)	Selects the H.264 profile
	AMF_VIDEO_ENCODER_PROFILE_LEVEL	1, 1.1, 1.2, 1.3, 2, 2.1, 2.2, 3, 3.1, 3.2, 4, 4.1, 4.2	Selects the H.264 profile level
	AMF_VIDEO_ENCODER_MAX_LTR_FRAMES	0 … 2	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.
	AMF_VIDEO_ENCODER_LOWLATENCY_MODE	True/False (On/Off); default is false	Enables low latency mode in the encoder and switches POC mode to 2
	AMF_VIDEO_ENCODER_FRAMESIZE	Width: 64 – 4096 Height: 64 – 4096	Frame width and height in pixels, maximum values are hardware-specific, should be queried through *AMFCaps*
	AMF_VIDEO_ENCODER_ASPECT_RATIO	Default 1:1	Pixel aspect ratio
	AMF_VIDEO_ENCODER_COLOR_BIT_DEPTH	8, 10, 16	Sets the number of bits in each pixel's color component in the encoder's compressed output bitstream. Default is 8.

Category	Name	Values	Description
Encoder color conversion parameters	AMF_VIDEO_ENCODER_INPUT_COLOR_PROFILE	UNKNOWN, 601, 709, 2020, JPEG, FULL_601, FULL_709, FULL_2020	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. See ColorSpace.h for enumeration.
	AMF_VIDEO_ENCODER_INPUT_TRANSFER_CHARACTERISTIC	UNDEFINED, BT709, UNSPECIFIED, RESERVED, GAMMA22, GAMMA28, SMPTE170M, SMPTE240M, LINEAR, LOG, LOG_SQRT, IEC61966_2_4, BT1361_ECG, IEC61966_2_1, BT2020_10, BT2020_12, SMPTE2084, SMPTE428, ARIB_STD_B67	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. See ColorSpace.h for enumeration.
	AMF_VIDEO_ENCODER_INPUT_COLOR_PRIMARIES	UNDEFINED, BT709, UNSPECIFIED, RESERVED, BT470M, BT470BG, SMPTE170M, SMPTE240M, FILM, BT2020, SMPTE428, SMPTE431, SMPTE432, JEDEC_P22	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. See ColorSpace.h for enumeration.
	AMF_VIDEO_ENCODER_OUTPUT_COLOR_PROFILE	UNKNOWN, 601, 709, 2020, JPEG, FULL_601, FULL_709, FULL_2020	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. See ColorSpace.h for enumeration. Determines the optional VUI parameter “matrix_coefficients”.

Category	Name	Values	Description
	AMF_VIDEO_ENCODER_OUTPUT_TRANSFER_CHARACTERISTIC	UNDEFINED, BT709, UNSPECIFIED, RESERVED, GAMMA22, GAMMA28, SMPTE170M, SMPTE240M, LINEAR, LOG, LOG_SQRT, IEC61966_2_4, BT1361_ECG, IEC61966_2_1, BT2020_10, BT2020_12, SMPTE2084, SMPTE428, ARIB_STD_B67	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. See ColorSpace.h for enumeration.
	AMF_VIDEO_ENCODER_OUTPUT_COLOR_PRIMARIES	UNDEFINED, BT709, UNSPECIFIED, RESERVED, BT470M, BT470BG, SMPTE170M, SMPTE240M, FILM, BT2020, SMPTE428, SMPTE431, SMPTE432, JEDEC_P22	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. See ColorSpace.h for enumeration.
	AMF_VIDEO_ENCODER_TARGET_BITRATE	10 000 - 100 000 000 bit/s	Sets the target bitrate
	AMF_VIDEO_ENCODER_PEAK_BITRATE	10 000 - 100 000 000 bit/s	Sets the peak bitrate

Category	Name	Values	Description
Encoder rate-control parameters	AMF_VIDEO_ENCODER_RATE_CONTROL_METHOD	0, 1, 2, 3, 4 (CQP, CBR, VBR, VBR_LAT, QVBR)	Selects the rate control method: • CQP – Constrained QP, • CBR - Constant Bitrate, • VBR - Peak Constrained VBR, • VBR_LAT - Latency Constrained VBR, • QVBR – Quality-Defined VBR Remarks: • When SVC encoding is enabled, all Rate-control parameters (with some restrictions) 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 mode is only supported if PreAnalysis is enabled.
	AMF_VIDEO_ENCODER_RATE_CONTROL_SKIP_FRAME_ENABLE	True/False (On/Off)	Enables skip frame for rate control
	AMF_VIDEO_ENCODER_MIN_QP	0 – 51	Sets the minimum QP
	AMF_VIDEO_ENCODER_MAX_QP	0 – 51	Sets the maximum QP
	AMF_VIDEO_ENCODER_QP_I	0 – 51	Sets the constant QP for I-pictures. Remarks: • Only available for CQP rate control method.
	AMF_VIDEO_ENCODER_QP_P	0 – 51	Sets the constant QP for P-pictures. Remarks: • Only available for CQP rate control method.
	AMF_VIDEO_ENCODER_QP_B	0 – 51	Sets the constant QP for B-pictures. Remarks: • Only available for CQP rate control method.

Category	Name	Values	Description
	AMF_VIDEO_ENCODER_QVBR_QUALITY_LEVEL	1 – 51	Sets the quality level for QVBR rate control method. Remarks: • Only available for QVBR rate control method.
	AMF_VIDEO_ENCODER_FRAMERATE	1*FrameRateDen … 120* FrameRateDen	Frame rate numerator
	AMF_VIDEO_ENCODER_VBV_BUFFER_SIZE	1000 – 100 000 000	Sets the VBV buffer size in bits
	AMF_VIDEO_ENCODER_INITIAL_VBV_BUFFER_FULLNESS	0 - 64	Sets the initial VBV buffer fullness
	AMF_VIDEO_ENCODER_ENFORCE_HRD	True/False (On/Off)	Disables/enables constraints on QP variation within a picture to meet HRD requirement(s)
	AMF_VIDEO_ENCODER_ENABLE_VBAQ	True/False (On/Off)	Enables/disables VBAQ 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.
	AMF_VIDEO_ENCODER_MAX_AU_SIZE	0 - 100 000 000 bits	Maximum AU size in bits
	AMF_VIDEO_ENCODER_B_PIC_DELTA_QP *	-10 ... 10	Selects the delta QP of non-reference B pictures with respect to I pictures
	AMF_VIDEO_ENCODER_REF_B_PIC_DELTA_QP *	-10 ... 10	Selects delta QP of reference B pictures with respect to I pictures
	AMF_VIDEO_ENCODER_PREENCODE_ENABLE	AMF_VIDEO_ENCODER_PREENCODE_DISABLED, AMF_VIDEO_ENCODER_PREENCODE_ENABLED	Enables or disables rate control pre-analysis, default is Disabled
	AMF_VIDEO_ENCODER_FILLER_DATA_ENABLE	True/False	Enables/disables filler data to maintain constant bit rate
Encoder picture-control parameters	AMF_VIDEO_ENCODER_HEADER_INSERTION_SPACING	0 … 1000	Sets the headers insertion spacing
	AMF_VIDEO_ENCODER_IDR_PERIOD	0 … 1000	Sets IDR period. IDRPeriod= 0 turns IDR off
	AMF_VIDEO_ENCODER_DE_BLOCKING_FILTER	True/False (On/Off)	Enable/disable the de-blocking filter
	AMF_VIDEO_ENCODER_INTRA_REFRESH_NUM_MBS_PER_SLOT	0 - #MBs per frame	Sets the number of intra-refresh macro-blocks per slot
	AMF_VIDEO_ENCODER_SLICES_PER_FRAME	1 - #MBs per frame	Sets the number of slices per frame
	AMF_VIDEO_ENCODER_B_PIC_PATTERN *	0, 1, 2, 3	Sets the number of consecutive B-pictures in a GOP. BPicturesPattern = 0 indicates that B-pictures are not used

Category	Name	Values	Description
	AMF_VIDEO_ENCODER_B_REFERENCE_ENABLE *	True/False (On/Off)	Enables or disables using B-pictures as references
	AMF_VIDEO_ENCODER_CABAC_ENABLE	AMF_VIDEO_ENCODER_UNDEFINED, AMF_VIDEO_ENCODER_CABAC, AMF_VIDEO_ENCODER_CALV	Encoder coding method, when Undefined is selected, the behavior is profile-specific: CALV for Baseline, CABAC for Main and High
	AMF_VIDEO_ENCODER_MAX_NUM_REFRAMES	0..16	Maximum number of reference frames
Encoder miscellaneous parameters	AMF_VIDEO_ENCODER_SCANTYPE	0, 1 (Progressive, Interlaced)	Selects progressive or interlaced scan
	AMF_VIDEO_ENCODER_QUALITY_PRESET	0, 1, 2 (Balanced, Speed, Quality)	Selects the quality preset
	AMF_VIDEO_ENCODER_FULL_RANGE_COLOR	True/False	True indicates that the YUV range is 0..255
	AMF_VIDEO_ENCODER_MAX_INSTANCES	1 or 2	Hardware-dependent, only some hardware supports 2 instances
	AMF_VIDEO_ENCODER_MULTI_INSTANCE_MODE	True or False	Enables or disables multi-instance mode, default - disabled
	AMF_VIDEO_ENCODER_CURRENT_QUEUE	0 or 1	Selects the encoder instance frames are being submitted to
	AMF_VIDEO_ENCODER_PICTURE_TRANSFER_MODE	AMF_VIDEO_ENCODER_PICTURE_TRANSFER_MODE_ON, AMF_VIDEO_ENCODER_PICTURE_TRANSFER_MODE_OFF	The application can turn on this flag for a specific input picture to allow dumping the reconstructed picture and/or injecting a reference picture
Encoder motion estimation parameters	AMF_VIDEO_ENCODER_MOTION_HALF_PIXEL	True/False (On/Off)	Turns on/off half-pixel motion estimation
	AMF_VIDEO_ENCODER_MOTION_QUARTERPIXEL	True/False (On/Off)	Turns on/off quarter-pixel motion estimation
Encoder SVC parameters (only webcam usage)	AMF_VIDEO_ENCODER_NUM_TEMPORAL_ENHANCMENT_LAYERS	0 … MaxOfTemporalEnhansmentLayers	Change the number of temporal enhancement layers. The maximum number allowed is set by the corresponding create parameter. 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.

Category	Name	Values	Description
Encoder SVC per-layer parameters (only webcam usage)	TL<TL_Num>. QL<QL_Num>. <Parameter_name>	Parameter-specific values	Configures SVC frame-rate parameter per SVC layer. • TL_Num — temporal layer number • QL_Num — quality layer number • Parameter_name — frame rate parameter name (look at frame-rate parameters on this table). Remarks: • Quality layers are not supported on VCE 1.0. "QL0" must be used for quality layers.

* this feature is not supported by VCE 1.0

Table A-2. Input frame and encoded data parameters

Category	Name	Values	Description
Frame per-submission parameters	AMF_VIDEO_ENCODER_INSERT_SPS	True/False (On/Off)	Inserts SPS
	AMF_VIDEO_ENCODER_INSERT_PPS	True/False (On/Off)	Inserts PPS
	AMF_VIDEO_ENCODER_INSERT_AUD	True/False (On/Off)	Inserts AUD
	AMF_VIDEO_ENCODER_PICTURE_STRUCTURE	0, 1, 2, 3 (None, Frame, TopField, BottomField)	Picture structure
	AMF_VIDEO_ENCODER_FORCE_PICTURE_TYPE	0, 1, 2, 3, 4, 5* (NONE, SKIP, IDR, I, P, B*)	Forces the picture type (to use this feature, set AMF_VIDEO_ENCODER_IDR_PERIOD to 0)
	AMF_VIDEO_ENCODER_END_OF_SEQUENCE	True/False (On/Off)	End of sequence
	AMF_VIDEO_ENCODER_END_OF_STREAM	True/False (On/Off)	End of stream
	AMF_VIDEO_ENCODER_MARK_CURRENT_WITH_LTR_INDEX	-1 … (MaxOfLTRFrames - 1)	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.
	AMF_VIDEO_ENCODER_FORCE_LTR_REFERENCE_BITFIELD	Bitfield (MaxOfLTRFrames (max possible 16 bits))	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.
	AMF_VIDEO_ENCODER_ROI_DATA	Video surface in AMF_SURFACE_GRAY32 format	Important value for each macro block ranges from 0 to 10, stored in 32bit unsigned format.
	AMF_VIDEO_ENCODER_STATISTICS_FEEDBACK	True/False (On/Off)	Instruct encoder to collect and feedback statistics
	AMF_VIDEO_ENCODER_REFERENCE_PICTURE	AMFSurface	Injected reference picture. Valid with AMF_VIDEO_ENCODER_PI

			CTURE_TRANSFER_MODE turned on
Encoded data parameters	AMF_VIDEO_ENCODER_OUTPUT_DATA_TYPE	0, 1, 2, 3* (IDR, I, P, B*)	Type of encoded data
	AMF_VIDEO_ENCODER_OUTPUT_MARKED_LTR_INDEX	-1 … (MaxOfLTRFrames - 1)	Marked as LTR Index. If != -1, then this picture was coded as a long-term reference with this LTR Index.
	AMF_VIDEO_ENCODER_OUTPUT_REFERENCED_LTR_INDEX_BITFIELD	Bitfield (MaxOfLTRFrames (max possible 16 bits))	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).
	AMF_VIDEO_ENCODER_RECONSTRUCTED_PICTURE	AMFSurface	Reconstructed picture. Valid with AMF_VIDEO_ENCODER_PICTURE_TRANSFER_MODE turned on

* this feature is not supported by VCE 1.0

Table A-3. Default values of parameters

Category	Name	Transcoding	Ultra low latency	Low latency	Webcam
Initialization Parameters	AMF_VIDEO_ENCODER_PROFILE	Main	Main	Main	Main
	AMF_VIDEO_ENCODER_PROFILE_LEVEL	4.2	4.2	4.2	4.2
Rate control	AMF_VIDEO_ENCODER_TARGET_BITRATE	20 mbps	20 mbps	20 mbps	20 mbps
	AMF_VIDEO_ENCODER_PEAK_BITRATE	30 mbps	20 mbps	30 mbps	30 mbps
	AMF_VIDEO_ENCODER_MIN_QP	18	22	22	22
	AMF_VIDEO_ENCODER_MAX_QP	46	48	48	48
	AMF_VIDEO_ENCODER_QP_I	22	22	22	22
	AMF_VIDEO_ENCODER_QP_P	22	22	22	22
	AMF_VIDEO_ENCODER_QP_B	22	22	22	22
	AMF_VIDEO_ENCODER_FRAMERATE	30 fps	30 fps	30 fps	30 fps
	AMF_VIDEO_ENCODER_VBV_BUFFER_SIZE	20 mbits	735 kbits	4 mbits	2 mbits
	AMF_VIDEO_ENCODER_INITIAL_VBV_BUFFER_FULLNESS	64	64	64	64
	AMF_VIDEO_ENCODER_ENFORCE_HRD	false	true	false	false
	AMF_VIDEO_ENCODER_MAX_AU_SIZE	0	0	0	0
	AMF_VIDEO_ENCODER_FILLER_DATA_ENABLE	false	false	false	false
	AMF_VIDEO_ENCODER_B_PIC_DELTA_QP*	+4	0	+4	+4
	AMF_VIDEO_ENCODER_REF_B_PIC_DELTA_QP*	+2	0	+2	+2
	AMF_VIDEO_ENCODER_HEADER_INSERTION_SPACING**	0	0	0	0
	AMF_VIDEO_ENCODER_IDR_PERIOD	30	300	300	30
Picture Control	AMF_VIDEO_ENCODER_DE_BLOCKING_FILTER	true	true	true	true
	AMF_VIDEO_ENCODER_INTRA_REFRESH_NUM_MBS_PER_SLOT*	0	255	255	0
	AMF_VIDEO_ENCODER_SLICES_PER_FRAME	1	1	1	1
	AMF_VIDEO_ENCODER_B_PIC_PATTERN*	3	0	0	0
	AMF_VIDEO_ENCODER_B_REFERENCE_ENABLE*	true	false	true	true
	AMF_VIDEO_ENCODER_SCANTYPE	0	0	0	0
	AMF_VIDEO_ENCODER_QUALITY_PRESET	Balanced	Speed	Speed	Speed
	AMF_VIDEO_ENCODER_MOTION_HALF_PIXEL	1	1	1	1

Motion estimation	AMF_VIDEO_ENCODER_MOTION_QUARTERPIXEL	1	1	1	1
SVC	AMF_VIDEO_ENCODER_NUM_TEMPORAL_ENHANCMENT_LAYERS	disable	disable	disable	0
Per-submission parameters	AMF_VIDEO_ENCODER_INSERT_SPS	0	0	0	0
	AMF_VIDEO_ENCODER_INSERT_PPS	0	0	0	0
	AMF_VIDEO_ENCODER_PICTURE_STRUCTURE	0	0	0	0
	AMF_VIDEO_ENCODER_FORCE_PICTURE_TYPE	0	0	0	0
	AMF_VIDEO_ENCODER_INSERT_AUD	false	false	false	false
	AMF_VIDEO_ENCODER_END_OF_SEQUENCE	false	false	false	false
	AMF_VIDEO_ENCODER_END_OF_STREAM	false	false	false	false
	AMF_VIDEO_ENCODER_MARK_CURRENT_WITH_LTR_INDEX	-1	-1	-1	-1
	AMF_VIDEO_ENCODER_FORCE_LTR_REFERENCE_BITFIELD	0x0	0x0	0x0	0x0
	AMF_VIDEO_ENCODER_ROI_DATA	N/A	N/A	N/A	N/A

* `BPicturesDeltaQP`, `ReferenceBPicturesDeltaQP`, `IntraRefreshNumMBsPerSlot`, `BPicturesPattern` and `BReferenceEnable` parameters are available only when:

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

** HeaderInsertionSpacing: Every IDR frame has SPS and PPS regardless of default value of HeaderInsertionSpacing per VCE logic.

Table A-4. Encoder statistics feedback

Statistic Name	Description
AMF_VIDEO_ENCODER_STATISTIC_FRAME_QP	QP of the first encoded macroblocks in a picture
AMF_VIDEO_ENCODER_STATISTIC_AVERAGE_QP	Average QP of all encoded macroblocks in a picture
AMF_VIDEO_ENCODER_STATISTIC_MAX_QP	Max QP among all encoded macroblocks in a picture
AMF_VIDEO_ENCODER_STATISTIC_MIN_QP	Min QP among all encoded macroblocks in a picture
AMF_VIDEO_ENCODER_STATISTIC_PIX_NUM_INTRA	Number of intra-coded pixels
AMF_VIDEO_ENCODER_STATISTIC_PIX_NUM_INTER	Number of inter-coded pixels
AMF_VIDEO_ENCODER_STATISTIC_PIX_NUM_SKIP	Number of skip-coded pixels
AMF_VIDEO_ENCODER_STATISTIC_BITCOUNT_RESIDUAL	Frame level bit count of residual data
AMF_VIDEO_ENCODER_STATISTIC_BITCOUNT_MOTION	Frame level bit count of motion vectors
AMF_VIDEO_ENCODER_STATISTIC_BITCOUNT_INTER	Frame level bit count of inter macroblocks
AMF_VIDEO_ENCODER_STATISTIC_BITCOUNT_INTRA	Frame level bit count of intra macroblocks
AMF_VIDEO_ENCODER_STATISTIC_BITCOUNT_ALL_MINUS_HEADER	Frame level bit count of the bitstream excluding header
AMF_VIDEO_ENCODER_STATISTIC_MV_X	Accumulated absolute values of MVX
AMF_VIDEO_ENCODER_STATISTIC_MV_Y	Accumulated absolute values of MVY