**3GPP TSG- Meeting #**

**, , -**

|  |
| --- |
| *CR-Form-v12.3* |
| **CHANGE REQUEST** |
|  |
|  |  | **CR** |  | **rev** |  | **Current version:** |  |  |
|  |
| *For* [***HE******LP***](http://www.3gpp.org/3G_Specs/CRs.htm#_blank)*on using this form: comprehensive instructions can be found at* [*http://www.3gpp.org/Change-Requests*](http://www.3gpp.org/Change-Requests)*.* |
|  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***Proposed change affects:*** | UICC apps |  | ME | **x** | Radio Access Network |  | Core Network | **x** |

|  |
| --- |
|  |
| ***Title:***  |  |
|  |  |
| ***Source to WG:*** |  |
| ***Source to TSG:*** | S4 |
|  |  |
| ***Work item code:*** |  |  | ***Date:*** |  |
|  |  |  |  |  |
| ***Category:*** |  |  | ***Release:*** |  |
|  | *Use one of the following categories:****F*** *(correction)****A*** *(mirror corresponding to a change in an earlier release)****B*** *(addition of feature),* ***C*** *(functional modification of feature)****D*** *(editorial modification)*Detailed explanations of the above categories canbe found in 3GPP [TR 21.900](http://www.3gpp.org/ftp/Specs/html-info/21900.htm). | *Use one of the following releases:Rel-8 (Release 8)Rel-9 (Release 9)Rel-10 (Release 10)Rel-11 (Release 11)…Rel-17 (Release 17)Rel-18 (Release 18)Rel-19 (Release 19) Rel-20 (Release 20)* |
|  |  |
| ***Reason for change:*** | Errors and omissions identified in the detailed algorithmic description. |
|  |  |
| ***Summary of change:*** | Corrections and insertion of additional description to correct the algorithmic description. |
|  |  |
| ***Consequences if not approved:*** | TS 26.253 describing the IVAS codec algorithm will be incorrect and missing description of certain operations, which can be misleading. |
|  |  |
| ***Clauses affected:*** | 2, 4.2.10, 5.2.2.3.1.1, 5.2.2.3.1.2.1, 5.2.2.3.1.2.2, 5.2.2.3.1.2.4, 5.2.2.3.2.3., 5.2.2.3.2.5.3, 5.2.4.3.2.1, 5.2.4.5.3.2, 5.3.2.3.4, 5.3.2.4.4.5, 5.3.2.4.7, 5.3.2.4.12, 5.3.2.4.13, 5.3.5.1.1, 5.5.1, 5.5.3.2.7, 5.7.3.6.3, 5.8.1, 5.8.4, 5.9.4.2, 5.9.10, 6.3.1.2.2, 6.3.2.3.10.1, 6.3.2.3.10.3, 6.3.5.1.2, 6.3.5.1.3, 6.3.5.1.4, 6.4.11, 6.6.7.2, 6.8.3, 6.8.4, 6.8.5, 6.9.2, 6.9.3, 6.9.4.1, 6.9.4.2, 6.9.4.3, 6.9.4.4., 6.9.5, 6.9.7.1.1, 6.9.7.3.1, 6.9.7.3.2, 6.9.7.4, 6.9.10, 6.9.11, 7.4.1, 7.4.2, 7.4.3, 7.4.5.1, 7.4.7.3.6, 7.5.1, 8.4.1, 8.4.2, 8.8 |
|  |  |
|  | **Y** | **N** |  |  |
| ***Other specs*** |  | **x** |  Other core specifications  | TS/TR ... CR ...  |
| ***affected:*** |  | **x** |  Test specifications | TS/TR ... CR ...  |
| ***(show related CRs)*** |  | **x** |  O&M Specifications | TS/TR ... CR ...  |
|  |  |
| ***Other comments:*** |  |
|  |  |
| ***This CR's revision history:*** | Initial version: S4-240663 (SA4#127-bis-e endorsed)Rev 1: Additional corrections that were identified after SA4#127-bis-e |

CHANGE 1

# 2 References

[1] 3GPP TR 21.905: "Vocabulary for 3GPP Specifications".

[2] 3GPP TS 26.441: "Codec for Enhanced Voice Services (EVS); General Overview".

[3] 3GPP TS 26.445: "Codec for Enhanced Voice Services (EVS); Detailed Algorithmic Description".

[4] 3GPP TS 26.447: "Codec for Enhanced Voice Services (EVS); Error concealment of lost packets".

[5] 3GPP TS 26.448: "Codec for Enhanced Voice Services (EVS); Jitter Buffer Management"

[6] 3GPP TS 26.250: "Codec for Immersive Voice and Audio Services (IVAS); General overview".

[7] 3GPP TS 26.251: "Codec for Immersive Voice and Audio Services (IVAS); C code (fixed-point)".

[8] 3GPP TS 26.252: "Codec for Immersive Voice and Audio Services (IVAS); Test Sequences".

[9] 3GPP TS 26.254: "Codec for Immersive Voice and Audio Services (IVAS); Rendering".

[10] 3GPP TS 26.255: "Codec for Immersive Voice and Audio Services (IVAS); Error concealment of lost packets".

[11] 3GPP TS 26.256: "Codec for Immersive Voice and Audio Services (IVAS); Jitter Buffer Management".

[12] 3GPP TS 26.258: "Codec for Immersive Voice and Audio Services (IVAS); C code (floating point)".

[13] C. de Boor and K. Höllig (1987), B-splines without divided differences, in Geometric Modeling, G. Farin ed., SIAM, 21–27.

[14] Borß, C. A Polygon-Based Panning Method for 3D Loudspeaker Setups. In *Audio Engineering Society Convention 137*, Los Angeles, USA, Oct. 2014.

[15] Zotter, F. and Frank, M., All-Round Ambisonic Panning and Decoding, *J. Audio Eng. Soc.*, vol. 60, no. 10, pp. 807-820 (Oct. 2012)

[16] Allen, J. B., & Berkley, D. A., Image method for efficiently simulating small room acoustics. *Journal of the Acoustical Society of America*, 65(4), 943-950, (1979).

[17] Box, G. E. P. and Cox, D. R. (1964). An analysis of transformations, Journal of the Royal Statistical Society, Series B, 26, 211-252.

[18] 3GPP TS 26.118: “Virtual Reality (VR) profiles for streaming applications”.

[19] Ivanic, J., & Ruedenberg, K., Rotation matrices for real spherical harmonics. Direct determination by recursion. *The Journal of Physical Chemistry*, *100*(15), 6342-6347, 1996.

[20] C. R. Helmrich and B. Edler, “Signal-Adaptive Transform Kernel Switching for Stereo Audio Coding,” in Proc. IEEE WASPAA, New Paltz, NY, USA, Oct. 2015.

[21] AES, “AES69-2022: AES standard for file exchange - Spatial acoustic data”, Audio Engineering Society, 2022.

[22] F. Thomas, "Approaching Dual Quaternions From Matrix Algebra," in *IEEE Transactions on Robotics*, vol. 30, no. 5, pp. 1037-1048, Oct. 2014.

[23] J, Daniel “R,eprésentation de champs acoustiques, application à la transmission et à la reproduction de scènes sonores complexes dans un contexte multimedia”, Thèse de doctorat de l’Université Paris 6, 2001

[24] M. Chapman, “A Standard for Interchange of Ambisonic Signal Sets. Including a file standard with metadata”, Ambisonics Symposium 2009, Graz, June 25-27

[25] ISO/IEC 23091-3:2018 - Information technology Coding-independent code points Part 3: Audio

[26] ISO/IEC 23008-3:2015 - Information technology High efficiency coding and media delivery in heterogeneous environments Part 3: 3D audio

[r1] IETF RFC 4566 (2006): "SDP: Session Description Protocol", M. Handley, V. Jacobson and C. Perkins.

[r2] 3GPP TS 26.114: "IP Multimedia Subsystem (IMS); Multimedia Telephony; Media handling and interaction".

CHANGE 2

4.2.10 Discontinuous Transmission (DTX) Operation

DTX is a functionality of operation where the encoder encodes frames containing only background noise with a lower bit rate and lower packet frequency than normally used for encoding speech. A terminal and the network may adapt their transmission scheme to take advantage of the smaller frames and longer transmission interval to reduce power consumption, average bit rate and network activity. The discontinuous transmission (DTX) functionality of the IVAS codec includes voice activity detection (VAD) and comfort noise generation (CNG). DTX functionality is supported for IVAS operation points, i.e., audio formats and bitrates, that are especially optimized for efficient stereo and immersive conversational voice transmissions.

The size of the SID frames for EVS interoperable modes is unchanged relative to EVS, 48 bits for EVS primary modes. For IVAS modes, the SID frame size is 104 bits. The default SID frame interval is once per 8 frames, but other update intervals are also supported.

DTX is supported in IVAS formats and bitrates as summarized in Table 4.2-3. The algorithmic description of DTX/CNG is then discussed in related IVAS format specific clauses.

Table 4.2-3: DTX support overview in the IVAS codec

|  |  |
| --- | --- |
| **Input audio format**  | **DTX support** |
| Stereo  | Yes, up to 256 kbps |
| Scene-based audio (SBA)  | Yes, up to 80 kbps |
| Metadata assisted spatial audio (MASA)  | Yes, up to 512 kbps |
| Object-based audio (ISM)(1)  | Yes, up to 512 kbps |
| Multi-channel audio (MC)  | No |
| Combined ISM and SBA (OSBA)  | No |
| Combined ISM and MASA (OMASA)  | No |

CHANGE 3

##### 5.2.2.3.1 Core-coder pre-processing

5.2.2.3.1.1 Selection of internal sampling rate

The LP-based core-coder within the IVAS codec operates at two internal sampling rates, 12.8 kHz and 16 kHz and the the MDCT-based core-coder operates at four internal sampling rates, 12.8, 16, 25.6 and 32 kHz.

The selection of the internal sampling rate of the LP-based core-coder is based on bitrate, the operating mode, the bandwidth and the ACELP16k binary flag, . The ACELP16k binary flag is used as an indicator that 16kHz sampling rate shall be selected for the LP-based core coder. The ACELP16k binary flag is set as follows.

In the DFT strereo mode, the ACELP16k binary flag is set to 1 for SID and NO\_DATA frames when the codec operates in the DTX mode with LP-CNG encoding (see clause 5.2.2.3.5.5) but only if the bandwidth of the input signal is WB. For all other cases within the DFT stereo mode, the ACELP16k binary flag is set to 1 when the element bitrate is higher or equal to 24.4 kbps and to 0 when the bitrate is lower than 24.4 kbps. In the TD stereo mode the ACELP16k binary flag is set to 1 for the core-coder in the primary channel when the bitrate is higher or equal to 24.4 kbps. However, when LRTD sub-mode has been selected in the TD stereo encoder the ACELP16k binary flag is set to 1 when the bitrate is higher than 24.4 kbps and not when the bitrate is equal to 24.4 kbps. For all bitrates lower than 24.4 kbps in the TD stereo mode the ACELP16k binary flag is set to 0.

For the encoding of SCE (single-channel element) the ACELP16k binary flag is set to 1 when the element bitrate is higher than or equal to 17 kbps. For all other bitrates it is set to 0.

In case of EVS mono operation the ACELP16k binary flag is set based on the logic described in clause 5.4.4 of [3].

The internal sampling rate of the LP-based core-coder and the MDCT-based core-coder (TCX) is initially set to 16kHz when the ACELP16k binary flag is set to 1. Otherwise, the internal sampling rate is initially set to 12.8 kHz. When the IVAS codec operates in the MDCT stereo mode and the element bitrate is higher than or equal to 64 kbps TCX core-coder is selected and its internal sampling rate is set to 32kHz. Similarly, for the encoding of SCEs, TCX core-coder is selected and its internal sampling rate is set to 32kHz when the core-coder bitrate is higher than 48 kbps. The TCX core is also selected when the IVAS codec operates in the ISM mode (object-based audio) and the core-coder bitrate is higher than 40 kbps and lower than or euqal to 48 kbps. In this case, however, the internal sampling rate of the TCX core-coder is set to 25.6 kHz.

During the DTX operation, the internal sample rate of the core-coder is not allowed to change during NO\_DATA frames. The internal sampling rate is also not allowed to change in SID frames following short segments of active frames.

CHANGE 4

##### 5.2.2.3.1.2 Core-coder technology selection

5.2.2.3.1.2.1 Overview

The selection of the core-coder technology in the IVAS codec is based on the mechanism described in clause 5.1.14 in [3]. The core-coder technology is selected from the following list

- ACELP core-coder technology

- GSC core-coder technology

- TCX core-coder technology

- HQ core-coder technology

The selection of the core-coder technology follows the pre-selection mechanism, described in detail in clause 5.2.2.2.12, which is part of the front pre-processing module.

The selection of the core-coder technology is based on the pre-selected core-coder technology which is stored in the form of the S/M binary flags and . The selection of the core-coder technology takes into account bitrate limitations, bandwidth limitations, VAD flag and other auxiliary parameters. Table 5.2‑7 shows the selection of the core-coder technology based on core-coder bitrate, bandwidth and content type.

Table 5.2‑7: Selection of core-coder technology based on bitrate, bandwidth and content type

|  |  |  |
| --- | --- | --- |
| bandwidth | content | core-coder total bitrate [kbps] |
| 13.2 | 16.4 | 24.4 | 32 | 48 | >48 |
| WB | speech | ACELP | ACELP | ACELP | ACELP | ACELP | TCX |
| audio | GSC/TCX | GSC/TCX | TCX/HQ | TCX/HQ | TCX/HQ | TCX |
| inactive | GSC/TCX | GSC /TCX | GSC /TCX | ACELP/TCX | ACELP/TCX | TCX |
| SWB | speech | ACELP | ACELP | ACELP | ACELP | ACELP | TCX |
| audio | GSC/TCX | GSC/TCX | TCX/HQ | TCX/HQ | TCX/HQ | TCX |
| inactive | GSC /TCX | GSC /TCX | GSC I/TCX | ACELP/TCX | ACELP/TCX | TCX |
| FB | speech |  |  |  | ACELP | ACELP | TCX |
| audio |  |  |  | TCX/HQ | TCX/HQ | TCX |
| inactive |  |  |  | ACELP/TCX | ACELP/TCX | TCX |

Please, note that the core-coder bitrate may be different from the bitrates listed in Table 5.2-6 due to variable bitrate allocation mechanism in the ACELP/TCX coder, described in clauses 5.2.2.3.2 and 5.2.2.3.3.

As can be seen in Table 5.2‑6 the ACELP core-coder technology is selected when the content is classified as “speech” which is signalled to the selector by the S/M binary flags. The content is considered “speech” when and . ACELP core is also selected for the encoding of SID frames and NO\_DATA frames in DTX mode. In the ISM low-rate mode (see clause 5.6.2.3.2) ACELP core is selected for the encoding of inactive frames. In this case, is set to INACTIVE.

For “music” content the IVAS codec selects among GSC, TCX or HQ core-coder technologies. The selection between the GSC core-coder technology and the TCX/HQ core-coder technology is based on the values of the and S/M binary flags. The GSC core is selected when and . In case when and the IVAS codec selects either the TCX core-coder technology or the HQ core-coder technology.The selection between the TCX core-coder technology and the HQ core-coder technology is based on the output of the HQ classifier, described in detail in clause 5.2.2.3.1.2.4.

For “inactive” content the selection of the core-coder technology is based on the coding mode classification, described in clause 5.2.2.2.10. The coding mode classification is based, among other parameters, on the SAD module, described in clause 5.2.2.2.5 and on the output of the IVAS S/M classifier, described in clause 5.2.2.2.11. For core-coder bitrates below 9 kbps and in cases where LP-CNG type has been chosen by the pre-processor (see clause 5.6 in [3]], GSC core-coder technology is selected for the encoding of inactive content. In all other cases TCX core-coder technology is selected.

The TCX core-coder technology is used for any content at bitrates higher than 48 kbps except of the ISM format coding where it is used for any content higher than 40 kbps. The TCX core-coder technology is also selected when the IVAS codec operates in the MDCT stereo mode. In case TCX core-coder technology has been selected by the logic described so far but the bitrate is lower than 9 kbps GSC core is used instead. In this case is reset to 0 and is set to AUDIO. However, if the IVAS codec operates in the ISM low-rate mode is set to INACTIVE instead of AUDIO.

In the special case when the IVAS codec operates in LRTD stereo submode with ACELP core-coder technology running at bitrate lower than or equal to 16.4 kbps, the selection mechanism prevents the IVAS codec from switching directly into DFT stereo mode with TCX core-coder technology. This is to avoid excessive computational complexity. The IVAS codec allows for switching into DFT stereo mode but enforces ACELP core-coder technology in the first frame where the switch occurs. In the successive frames core-coder technology is selected without any restrictions using the logic described so far.

Please, note that the selected core-coder technology may be changed in some other special situations which is described in other clauses in this document.

CHANGE 5

5.2.2.3.1.2.2 TD/FD BWE technology selection

The ACELP core-coder in the IVAS codec uses the TD and FD bandwidth extension (BWE) technology from the EVS codec. The selection between the TD BWE and the FD BWE is described in detail in clause 5.1.14.4 of [3]. In the IVAS codec, the selection mechanism has been further modified to accomodate the variable-rate of the ACELP core-coder and the new SWB TBE modes at 1.10 kbps and 1.75 kbps, described in clause 5.2.2.3.2.7.

The selection between the TD BWE and the FD BWE technology is based on the characteristics of the input signal, the bandwidth and the core-coder of the low-band signal. Table 5.2-7b lists the bitrates of all TD BWE and FD BWE technologies.

Table 5.2‑7b: Bitrates of TD BWE and FD BWE technologies

|  |  |  |
| --- | --- | --- |
| **bandwidth** | **time-domain** | frequency-dmain |
| WB | WB TBE at 0.35 kbps | WB BWE at 0.35 kbps |
| WB TBE at 1.05 kbps |  |
| SWB | SWB TBE at 0.95 kbps |  |
| SWB TBE at 1.10 kbps |  |
| SWB TBE at 1.75 kbps |  |
| SWB TBE at 1.6 kbps | SWB BWE at 1.6 kbps |
| SWB TBE at 2.8 kbps |  |
| FB | FB TBE at 1.8 kbps | FB BWE at 1.8 kbps |
| FB TBE at 3.0 kbps |  |

For WB signals, the bandwidth extension technology is selected as follows. When the core-coder bitrate is lower than 7.15 kbps the bandwidth extension is done in frequency domain without any side information transmitted in the bitstream. This can be referred to as the WB BWE at 0 kbps. The same technology is applied also in the TD stereo mode for the encoding of the secondary channel. If the core-coder bitrate is higher than or equal to 7.15 kbps and the binary flag ACELP16k is set to 0, the bandwidth extension technology is set based on the S/M binary flags and . described in clause 5.2.2.2.12.6. When and , the selected bandwidth extension technology is WB BWE at 0.35 kbps. The same WB BWE technology is selected for inactive frames, i.e. when the coder type is set to INACTIVE. For all other combinations of S/M binary flags and , WB TBE technology is selected but only under the condition that the coder type is different than INACTIVE. The bitrate of the selected WB TBE technology is set with the following logic. In case the core-coder bitrate is lower than 9.65 kbps, WB TBE at 0.35 kbps is used. Similarly, when the IVAS codec operates in the TD stereo mode, the primary channel is encoded with WB TBE at 0.35 kbps. For all other situations, WB TBE at 1.05 kbps is used.

For SWB and FB signals, the bandwidth extension technology is selected as follows. The WB BWE technology at 0 kbps, described in the previous parapraph, may also be applied for SWB and FB signals when neither of the following three conditions is fullfilled. The first condition is fullfilled when the core-coder bitrate is higher than or equal to 7.8 kbps. The second condition is specific only to the LRTD stereo sub-mode and it is fullfilled when the core-coder bitrate is higher than or equal to 5 kbps. The third condition is specific to the TD stereo mode (both LRTD sub-mode and regular sub-mode) and it is fullfilled when the core-coder bitrate is higher than or equal to 5 kbps and the CPE bitrate is lower than 16.4 kbps. When all three conditions are fullfilled, the IVAS codec selects one of the following SWB or FB bandwidth extension technologies. Note, that the selected BWE technology is always encoded with non-zero side information that is sent to the decoder in the bitstream.

The selection of the SWB/FB bandwidth extension technology is based on the S/M binary flags and . described in clause 5.2.2.2.12.6 and the SWB noisy speech flag, , described in clause 5.1.13.6.9 of [3]. When and, at the same time, and , the selected bandwidth extension technology is the SWB BWE at 1.6 kbps. In the case of , the SWB BWE at 1.6 kbps is also selected for all INACTIVE signals. Furthermore, if the input bandwidth is FB, then FB BWE at 1.8 kbps is selected on top of SWB BWE at 1.6 kbps for the encoding of the upper band from 14 to 20 kHz. If none of the above conditions is fullfilled SWB TBE technology is selected. The bitrate of the SWB TBE technology is based on the core-coder bitrate and the state of some auxiliary parameters. The selection of the SWB TBE bitrate is set as follows. The initial bitrate of the SWB TBE is set to 1.6 kbps. The initial bitrate may be modified under some special conditions. In case of SCE encoding, when the binary flag ACELP16k set to 1, the SWB TBE bitrate is increased to 2.8 kbps but only when the core-coder bitrate is higher than or equal to 24.4 kbps. In the LRTD stereo sub-mode, the bitrate of the SWB TBE is either decreased to 1.1 kbps when the CPE bitrate is lower than 24.4 kbps or increased to 1.75 kbps when the CPE bitrate is higher than or equal to 24.4 kbps. Finally, the SWB TBE bitrate is decreased to 0.95 kbps if the core-coder bitrate is lower than 13.2 kbps. This is the minimum SWB TBE bitrate. If the input bandwidth is FB, then the SWB TBE technology is complemented with the FB TBE technology for the encoding of the upper band from 14 kHz to 20 kHz. The bitrate of the selected FB TBE is set as follows. The initial bitrate of the FB TBE technology is set to 1.8 kbps. In case of SCE encoding, when the binary flag ACELP16k set to 1, the FB TBE bitrate is increased to 3.0 kbps but only when the core-coder bitrate is higher than or equal to 24.4 kbps.

The IVAS codec also contains the IC-BWE technology, optimized for the encoding of the bandwidth extension of stereo signals. The IC-BWE encoder is described in detail in clause 5.3.2.2.1. The IC-BWE bitrate is set as follows. In the TD stereo mode, the IC-BWE technology is applied only in the regular submode (not LRTD submode). The IC-BWE bitrate is set either to 0.35 kbps when the binary flag ACELP16k is equal to 1 or to 0.25 kbps when the binary flag ACELP16k is equal to 0. In the DFT stereo mode, the IC-BWE bitrate is set either to 0.5 kbps when the binary flag ACELP16k is equal to 1 or to 0.4 kbps when the binary flag ACELP16k is equal to 0. The bitrate of the IC-BWE may be further increased by 0.5 kbps in case when ACELP core coder has been selected for the encoding of the lower band. However, this increase of bitrate is not applied in the LRTD sub-mode within the TD stereo mode.

CHANGE 6

5.2.2.3.1.2.4 HQ classifier

For SWB bitrates at 24.4 and 32, there are 4 modes supported, Transient, Harmonic, HVQ and Generic as described in 5.3.4.2 in [3]. A frame is considered harmonic following the classification described in 5.3.4.2.3 [3]. An HQ classifier is used to switch to Generic mode for harmonic frames if a sparse harmonic structure is not detected in the spectrum, otherwise the classification follows that described in 5.3.4.2.3 [3]. A spectrum sparseness analysis is performed based on obtaining a peakyness measure and a noise band detection measure derived from the MDCT coefficients.

First, the magnitude of a critical frequency region is obtained by

 (5.2-75)

where is the MDCTs spectrum computed, is the first bin the critical frequency region and is the last bin in the critical frequency region. and are set to and respectively where the input sampling rate is 32 kHz and the frame length is 640. The critical frequency region is the upper half of the MDCT spectrum.

The peakyness measure is obtained by obtaining the crest in accordance with equation (5.2-76) below

 (5.2-76)

where is the number of bins in the critical band. A complimentary peakyness measure is obtained in accordance with equation (5.2-77)

 (5.2-77)

where

 (5.2-78)

A noise band detection measure is obtained according to equation

 (5.2-79)

where is a moving mean of the absolute spectrum using a window size fixed to 21. The moving mean is computed according to equation (5.2-80) below.

 (5.2-80)

 gives a measure of local concentration of energy, indicating a noise band in the spectrum. To stabilize the decision, and are low pass filtered according to equation.

 (5.2-81)

 (5.2-82)

where is set to 0.97.

The spectrum sparseness analysis obtains a harmonic decision for the current frame in accordance with equation (5.2-83):

 (5.2-83)

**Table 5.2‑8: Thresholds for HQ classifier**

|  |  |
| --- | --- |
| **threshold** | **value** |
|  | 7.0 |
|  | 2.128 |
|  | 220 |

The Generic mode is selected if the condition below is fulfilled, otherwise the mode decision is left unchanged and follows that described in clause 5.3.4.2.3 of [3] to select between HQ\_HARMONIC and HQ\_HVQ mode.

 . (5.2-84)

The audio frame is then encoded using the HQ Core encoder with the selected mode.

CHANGE 7

5.2.2.3.2.3 LP coefficients encoding

The encoding of the LP coefficients is performed on the LSF representation. The structure of the quantizer is similar to the EVS lattice based quantizer described in clause 5.2.2.1.4 of [3]. A safety net, predictive, or switched safety-net predictive multi-stage vector quantizer (MSVQ) is used to quantize the full-length frame-end LSF vector for all modes where it is used. In addition to the bitrates supported by the EVS LSF quantizer, in the IVAS codec, due to the fine granularity of the variable bit allocation between the coding blocks, more bitrate values are added for the multiple scale lattice vector quantizer (MSLVQ) LSF quantizer. The corresponding lattice structure structures are defined for the number of bits presented in Table 5.2‑9 according to the signal coding type.

Table 5.2‑9: Number of bits for which the MSLVQ are defined

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Coding type | LSF quantizer - Number of bits | VQ bits | Predictive LSF quantizer  | VQ bits |
| Unvoiced WB | 14,15,18,19,25,28 |  | 14-28 | 12 |
| Unvoiced NB | 18,19,24,25,29,32 |  | 18,19,22,23,24,29,32 | 8 |
| Voiced WB | 17-37 | 8 | 9-39 | 6 |
| Voiced NB | 17,18,22,23,24,25,27,29,34,37 | 8 | 8,19,23,24,25,27-36,39 | 6 |
| Generic WB | 12-32 | 9 | 15-35 | 6 |
| Generic NB | 12,16,19,20,21,22,23,25-32 | 9 | 15,19,22-26,28-35 | 6 |
| Transition WB | 17-33 | 9 |  |  |
| Transition NB | 17,18,22,23,24,25,28,31,32 | 9 |  |  |
| Generic 16k | 31,32 |  | 26-37 | 5 |
| Transition 16k | 32,33 | 8 |  |  |
| Audio WB | 17-36 | 4 | 21-40 | 0 |
| Audio NB | 17,21,22,25,26,28 | 4 | 21,25,26,30,31,32 | 0 |
| Audio WB 16k | 26,36 |  | 26-37 | 5 |
| Voiced 16k | 22-37 | 8 | 24-39 | 6 |
| Inactive NB |  |  | 17,21,22,25,26,27,36 | 5 |
| Inactive WB |  |  | 17-36 | 5 |
| Inactive 16k |  |  | 17,21,22,25,26,27,36 | 5 |

The VQ structures are defined by the corresponding unstructured codebooks and the lattice quantizers by a set of 6 scales and 6 numbers of leader vectors defining the structures as presented in clause 5.2.2.1.4 of [3].

To accommodate many more bitrates than are used for EVS, a different representation of the structures is used. For each bitrate of each coding mode there are the lattice structure definition and the lattice scales. One lattice structure definition consists of two groups of 3 integer numbers specifying the number of leader vectors in each of the three lattice truncations of each 8-dimensional half of the LSF vector. The lattice scales consist of two groups of 3 floating point scale values, one for each lattice truncation. In order to represent one lattice structure definition, a vector of two numeric pointers is used, a first pointer and a second pointer. The values of the pointers point to entries in a table comprising all (275) sub vectors defining the possible lattice structures for the lattice quantizers part for the first, and for the second half of the LSF vector. Where the first and second pointers each point to a sub vector half of the LSF vector, each sub vector half combining to give the full vector. The first and second pointers are contained as an entry in a table of pointers. A lattice structure for an 8-dimensional quantizer corresponding to a half of the LSF vector is represented as a 3-dimensional vector of integers, each integer indicating how many leader classes form each lattice truncation. Because there are more than 256 different lattice structures, and the pointers are represented using 8 bits, pointer values that are larger than 256 are stored as their value minus 256. Following on, if the first pointer value is smaller than the second pointer value then this indicates that the first pointer value has been stored using modulo 256 arithmetic. In this case the first pointer is given by adding 256 to the value of the stored first pointer. However, if the first pointer value is larger (or the same value) as the second pointer value then the value of the first pointer is directly used to access the LSF sub vector.

CHANGE 8

5.2.2.3.2.5.3 Unused Bit Encoding

Figure 5.2‑12 shows a flow diagram of the number of unused bits encoding. In the figure, represents a targeted SV, which is either of or . For encoding of unused bits, Remainder bits, RB, is computed according to

 (5.2-109)

Before computation of unused bits for the , one overflow bit is added to when overflow is detected (i.e., RB=4), for the remaining possible values of RB, RB bits are subtracted from the usable bits () before computing unused bits for the

 (5.2-110)

Finally, the number of unused bits is encoded, and the encoding process is completed.

For the case of is , due to re-ordering of sub-vectors some of the sub-vectors in Group2 with 0 value is also encoded which may lead to inadequate bits for representing , to avoid those cases following additional operations are performed before encoding unused bits.

Where NCNV is the number of consecutive null vectors.

When the above condition detect possibility of wasting bit(s) because of encoding null vector(s), the number of available bits () for encoding is updated by additionally adding () bits, or bits, or () bits. Number of bits added in this process depends on the result of as shown in Figure 5.2‑13



**Figure 5.2‑12: Flow of Encoding the number of unused bits**

Finally, the number of unused bits is encoded according to Table 5.2.2-1, and the encoding process is completed. This table can be customized based on the number of usable bits, meaning codes for unused bits exceeding the usable bits are not necessary and may be removed, and the stop bit “0” of the longest codeword for the unused bit can be omitted. To ensure bit efficiency, typically this stop bit omitted code is used when the codebook indicator is 0 (i.e codebook is Q0), because the unused bits become the largest in that case.

Table 5.2.2-1: Correspondence between number of unused bits and an unused bit encoding code

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Unused bits | 0-4 | 5-9 | 10-14 | 15-19 | .. |
| Codeword | 10 | 0 | 110 | 1110 | .. |
| Number of bits | 2 | 1 | 3 | 4 |  |



**Figure 5.2‑13: Process of updating the number of available bits for encoding SV3**

CHANGE 9

##### 5.2.4.3.2 Direction metadata quantization

5.2.4.3.2.1 Joint azimuth elevation quantization

The spatial audio parameters, azimuth and elevation of each time-frequency (TF) tile are indexed to a point of a spherical grid. The indexing is performed by first joint quantizing the two spatial audio direction parameters, or direction metadata. The direction metadata quantization refers to the quantization of azimuth and elevation values for each time frequency tile of each frame. The direction parameters can be quantized on the spherical grid whose structure is defined for each of the following number of bits: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, and 11. The spherical grid or codebook structure, for each number of bits is formed such that it approximates the covering of the sphere with smaller spheres, wherein the centres of the smaller spheres define the points of the spherical grid.

CHANGE 10

5.2.4.5.3.2 Variable rate encoding of direction parameters within EC3

For sake of simplicity, we will use in the following the index to denote the subband that is currently being encoded, instead of . If the number of subframes is larger than 1 the quantization is performed as a switched method deciding whether to use joint entropy encoding of the azimuth index and the elevation index or whether to consider sending an average common direction per subband as reference, followed by the corresponding azimuth differences, the elevation values being set to the value in the average common direction. Within the switched method, the decision to use the average common direction is enabled by the calculation, for each sub band , of 2 distortion measures and . The first distortion measure represents the estimation of the L2 norm distance on a surface of a sphere between the point on the sphere given by the elevation and azimuth and the point on the sphere given by the quantized elevation and quantized azimuth according to the joint quantization scheme.

where are the number of TF tiles in a sub band, and is the quantised elevation.

CHANGE 11

##### 5.3.2.3.4 Near out-of-phase operation (NOOP)

###### 5.3.2.3.4.1 NOOP signal detection

The TD downmix may have a problem if the signal in the left channel and the signal in the right channel are close to an opposite phase. In that case the passive mono downmix obtained by summing the left channel and the right channel would lead to canceling of certain harmonic components, making it unsuitable for encoding. To ensure that this problem does not occur the TD stereo coder contains a special sub-mode for input signals that are near-out-of-phase (NOOP). The NOOP decision process is based on the analysis of the passive mono downmix and the side signal defined in the previous clause and some other auxiliary parameters. The NOOP decision process is outlined below.

First, a preliminary out-of-phase signal detection is based on the energy differencebetween the side signal defined in equation (5.3-59) and the passive mono downmix defined in equation (5.3-58). The instantaneous side-to-mono energy difference is calculated as

 (5.3-73)

Then, the long-term side-to-mono energy difference is calculated as

 (5.3-74)

where the superscript indicates the previous frame. The content is considered INACTIVE when the VAD hangover counter, , is different than 0.

In addition to the long-term side-to-mono energy difference the last OL pitch correlation as defined in clause 5.1.10 of Reference [1] is calculated for both the primary channel and the secondary channel. The OL pitch correlation is then evaluated to decide whether the TD stereo coder should run in the NOOP sub-mode. Let denote the calculated maximum OL pitch correlation of the primary channel Y in the previous frame. Let denote the calculated maximum OL pitch correlation of the secondary channel in the previous frame. A sub-optimality flag is then set to 1 based on the following criterion.

 (5.3-75)

where is the VAD hangover counter of the primary channel and is the VAD hangover counter of the secondary channel. The VAD hangover counters are calculated using the procedure defined in clause 5.1.10 of [3].

The sub-optimality flag indicates that the TD stereo coder is running in the NOOP sub-mode. When the sub-optimality flag is set to 0 the LRTD sub-mode is selected within the TD stereo coder.

To improve the stability of the NOOP decision a simple hangover logic is applied to the sub-optimality flag . The hangover logic uses the OL pitch stability measure calculated for the primary and the secondary channel as

 (5.3-76)

The sub-optimality flag is set to 1 in the current frame if it was set to 1 in three consecutive previous frames and if the pitch stability is higher than a certain pre-defined threshold. That is

 (5.3-77)

where for is the OL pitch parameter of the *k*th subframe calculated using the OL pitch analysis specified in clause 5.1.10 of [3]. The complete procedure of NOOP detection is outlined in the schematic diagram in Figure 5.3‑20.



Figure 5.3‑20: NOOP detection in the TD stereo mode

###### 5.3.2.3.4.2 NOOP sub-mode selection

Following the NOOP signal detection described in clause 5.3.2.3.4.1 the selection of the NOOP sub-mode within the TD stereo coder is made using a series of conditions based on the signal classification of the primary and the secondary channel and the output of the NOOP signal detection block described in Fig. 5.3-20. Let the detected NOOP signal from Fig. 5.3-20 be denoted with a binary flag . Let the initial selection of the NOOP sub-mode be defined with the binary parameter and the final selection of the NOOP sub-mode with the binary parameter . Note, that all binary parameters have either the value of 1 or the value of 0.

The initial selection of the NOOP sub-mode is made as follows

where [-1] indicates the value of the binary parameter in the previous frame.

The final selection of the NOOP sub-mode is then performed based on the initial selection of the NOOP sub-mode and a modification flag related to the NOOP-specific mixing ratio, . The modification flag indicates that the NOOP-specific mixing ratio needs or does not need to be modified. If the modification flag in the previous frame indicates that the mixing ratio needs to be modified, then the final selection of the NOOP sub-mode is set to 0, i.e. . If the modification flag of the previous frame indicates that the mixing ratio does not need to be modified, then the final selection of the NOOP sub-mode is set based on a series of conditions involving the signal classification of the primary and the secondary channel. The following conditions are specified for the final selection of the NOOP sub-mode:

Condition1: the frame type of a primary channel signal in a previous frame (tdm\_SM\_last\_clas) is UNVOICED\_CLAS and its previous frame (tdm\_SM\_last2\_clas) is VOICED\_TRANSITION, or frame type of a secondary channel signal in a previous frame (tdm\_SM\_last\_clas) is UNVOICED\_CLAS and its previous frame (tdm\_SM\_last2\_clas) is VOICED\_TRANSITION;

Condition2: neither of the primary channel signal and the secondary channel signal in the previous frame (last\_coder\_type\_raw) is a coding type corresponding to VOICED;

Condition3: a quantity of consecutive frames before the previous frame (tdm\_NOOP\_cnt) that use the channel combination scheme used by the previous frame is greater than 5;

Condition4: the frame type of the primary channel signal in the previous frame (tdm\_SM\_last\_clas) is UNVOICED\_CLAS, or the frame type of the secondary channel signal in the previous frame (tdm\_SM\_last\_clas) is UNVOICED\_CLAS;

Condition5: long-term root mean square energy values of the left channel (rms\_L) and right channel (rms\_R) are less than 400;

If Condition 1, Condition 2 and Condition 3 are all satisified at the same time, then the final selection of the NOOP sub-mode is set to 1. i.e. . Alternatively, if Condition 2, Condition 3, Condition 4 and Condition 5 are all satisfied at the same time, then the final selection of the NOOP sub-mode is set to 1. i.e. .

###### 5.3.2.3.4.3 NOOP signal coding

When operating in the NOOP sub-mode, the TD stereo coder modifies the TD stereo downmix described in clause 5.3.2.3.3. In this mode, the left and right channels of the input stereo signal are combined using a NOOP-specific mixing factor, incorporating fade-in and fade-out smoothing techniques.

The primary and secondary channel signals in the current frame are calculated as follows:

where is the sample index, indicates a fade-in factor and indicates a fade-out factor. Furthermore, indicates a transition processing length which can be set to 0. The signal indicates the left channel signal in the current frame, indicates the right channel signal in the current frame, *Y*(*n*) indicates the primary channel signal in the current frame that is obtained through the time-domain processing, and *X*(*n*) indicates the secondary channel signal that is in the current frame and that is obtained through the time-domain processing. The parameter indicates encoding delay compensation which can be set to 0. The matrix is the downmixing matrix corresponding to the mixing ratio for the NOOP signal in the current frame and is the downmixing matrix corresponding to the mixing ratio for the NOOP signal in the previous frame. The mixing ratio for the NOOP signal in the current frame is calculated based on the difference of two correlation measures between the left and right channel and the passive mono downmix. The downmixing matrices are defined as follows

,

where and . The factor is the mixing ratio corresponding to the NOOP signal in the previous frame. Furthermore, and and is the mixing ratio corresponding to the NOOP signal in the current frame. The calculation of the mixing ratio corresponding top the NOOP signal is described in the next clause.

To obtain the primary channel signal and the secondary channel signal in the TD stereo mode for the NOOP signal, a segmented time-domain downmix process is performed based on the channel combination scheme for the current frame and the channel combination scheme in the previous frame when the channel combination scheme for the current frame is different from the channel combination scheme for the previous frame. The channel combination scheme for the previous frame is the correlated signal channel combination scheme, and the channel combination scheme for the current frame is the anticorrelated signal channel combination scheme. To obtain the start/first middle segments of the primary and secondary channel signals, time-domain downmix processing on the start/first middle segments is performed by using the mixing ratio corresponding to the correlated signal channel combination scheme for the previous frame and a time-domain downmix processing manner corresponding to the correlated signal channel combination scheme for the previous frame. To obtain the end/second middle segments of the primary and secondary channel signals, time-domain downmix processing on the end/second middle segments is performed by using the mixing ratio corresponding to the anticorrelated signal channel combination scheme for the previous frame and a time-domain downmix processing manner corresponding to the anticorrelated signal channel combination scheme for the previous frame. To obtain the middle segments of the primary and secondary channel signals, a weighted summation processing on the first middle segments and the second middle segments is performed. The segments can be calculated according to

wherein indicates the start segment of the primary channel signal, indicates the start segment of the secondary channel signal, indicates the end segment of the primary channel signal, indicates the end segment of the secondary channel signal, indicates the middle segment of the primary channel signal, and indicates the middle segment of the secondary channel signal; . indicates the primary channel signal; indicates the secondary channel signal.

 indicates the fade-in factor, indicates the fade-out factor, and a sum of and is 1; n indicates a sampling point number, and , <<; indicates the first middle segment of the primary channel signal, indicates the first middle segment of the secondary channel signal, indicates the second middle segment of the primary channel signal, and indicates the second middle segment of the secondary channel signal:

###### 5.3.2.3.4.4 Adaptive mixing ratio for the NOOP signal

The NOOP is a stereo signal whose phase difference between the left channel signal and the right channel signal falls within [180°-, 180°+], where is an angle between 0° and 90°. The mixing ratio for the NOOP signal is calculated using the same procedure as outlined in clause 5.3.2.3.3 with the notable difference that the passive mono downmix defined in eq. (5.3-58) is calculated in the following way

and the side channel resulting from the passive mono downmix defined in eq. (5.3-59) is calculated as follows

The mixing ratio for the NOOP signal is calculated based on the difference of two correlation measures, and defined by eq. (5.3-63). Note, that the calculation of the auxiliary parameters is done by following eqs. (5.3-60) to (5.3-52) using the mono downmix and the side channel as defined above.

The correlation measures and are then smoothed similarly as in eq. (5.3-64). That is

where is the convergence speed. Note, that has the same meaning as the convergence speed used in eq. (5.3-64) but it’s set specifically to the NOOP sub-mode. Finally, the long-term correlation difference is calculated as

i.e. similarly to eq. (5.3-65). The long-term correlation difference is then limited to the interval between -1.5 and +1.5. This is done as follows

The long-term correlation difference is linearly scaled using piece-wise linear mapping. This is done as follows:

After the linearization, the long-term correlation difference is converted to the NOOP-specific mixing factor using the cosine function. That is

CHANGE 12

###### 5.3.2.4.4.5 Refined ITD control mechanism

The ITD is critical to keep the stereo image stable. Sometimes ITD values calculated by the frequency domain correlation are not continuous. A difference parameter is used to represent a difference between the ITD of the current frame and the ITD of the previous frame . ITD is determined based on the difference parameter and a characteristic parameter of the current frame. The characteristic parameter is calculated as follows.

Divide a low frequency part of the left-channel frequency-domain signal of the current frame into M sub-bands, where each sub-band includes N frequency domain amplitude values. Calculate a correlation parameter of the current frame and a previous frame according to

, where represents a frequency domain amplitude value of an sub-band in the low frequency part of the left-channel frequency-domain signal of the current frame, represents a frequency domain amplitude value of an sub-band in a low frequency part of a left-channel frequency-domain signal of the previous frame, and represents a normalized cross-correlation value corresponding to an sub-band in the M sub-bands.

Calculate a peak-to-average ratio of each sub-band of the current frame .

If the ITD value of the current frame and an ITD value of the previous frame meet at least of one of the preset conditions, determine whether to reuse the ITD value of the previous frame for the current frame. The preset conditions set to

* condition\_1: the absolute ITD value of the previous frame is greater than the absolute ITD value of the current frame,

condition\_1 = || > 0.2 \* ||

* condition\_2: the average value of the normalized cross-correlation values of the sub-bands is greater than 0.85,

condition\_2 = avrg() > 0.85

* condition\_3: the average value of the normalized cross-correlation values of the sub-bands is greater than 0.7, and a normalized cross-correlation value of a sub-band is greater than 0.9,

condition\_3 = avrg() > 0.7 and ( > 0.9 or > 0.9 or > 0.9)

* condition\_4: the average value of the peak-to-average ratios of the sub-bands is greater than 0.6,

condition\_4 = avrg() > 0.6

* condition\_5: the ITD value of the previous frame is not equal to 0,

condition\_5 = ≠ 0

* condition\_6\_a: a product of the ITD value of the previous frame and the ITD value of the current frame is negative,

condition\_6\_a = itd \* prev\_itd < 0

* condition\_6\_b: a product of the ITD value of the previous frame and the ITD value of the current frame is 0,

condition\_6\_b = itd \* prev\_itd = 0

* condition\_6\_c: an absolute value of a difference between the ITD value of the previous frame and the ITD value of the current frame is greater than half of a target value, where the target value is an ITD value whose absolute value is larger in the ITD value of the previous frame and the ITD value of the current frame,

The ITD fine control result sets to

where

, condition\_1234 = condition\_1 and (condition\_2 or condition\_3 or condition\_4)

when the signal-to-noise ratio meets the signal-to-noise ratio condition, stopping reusing the ITD value of the previous frame as the ITD value of the current frame. The signal-to-noise ratio condition is set as the SNR value is less than 0.006 or the SNR value is greater than 2 000 000.

In addition, the ITD of the current frame could reuse the ITD of the previous frame ITD(m-1) based on the hangover counter. The hangover counter is the quantity of target frames that are allowed to appear consecutively. A characteristic information which is described by the signal-to-noise ratio and the peak feature of cross correlation coefficients of the stereo signal is used to control the hangover counter. When the signal-to-noise ratio and the peak feature of the cross-correlation coefficients meet at least one of preset conditions, the hangover counter will be reduced by adjusting at least one of a target frame count and a threshold of the target frame count.

CHANGE 13

##### 5.3.2.4.7 IPD calculation, stabilization and encoding scheme

5.3.2.4.7.1 IPD calculation and stabilization

For the downmix, a single global inter-channel-phase-difference (IPD) *gIPD* is calculated over the first 8 subbands of the ERB 4 partitioning (up to DFT bin 84) as

 *gIPD* = arg()

 (5.3-119)

where denotes the complex conjugate of .

To provide a more stable *gIPD* estimate, a stability mechanism is employed which is described in detail below and shown more comprehensively as a flow diagram in Figure 5.3‑27.



Figure 5.3‑27: Flow diagram of global IPD stabilization

The stabilization first requires the calculation of additional bandwise phase differences *IPDt,b* as

 *IPDt,b* = arg()

 (5.3-120)

for each of the 8 subbands over which the global IPD is calculated. Note that also for bitrates ≤ 16.4 kbps the ERB 4 bands are used for IPD calculation.

Additionally, in each subband bandwise mean IPDs over the 5 previous frames are calculated. Since distances between phases are ambiguous (2 possible directions on a circle) a meaningful bandwise mean IPD cannot always be calculated by standard averaging (only if all phases are within the same semi-circle). Instead, the bandwise mean IPD of a subband, denoted as IPDmean,b, may be initialized with 0 and then updated iteratively with

 (5.3-121)

where is the index over the previous IPD values of the band. After each iteration, the distance of the current result to the next value in the buffer is calculated:

 (5.3-122)

If is greater than , i.e. more than a half-circle rotation in the given direction, needs to be temporarily shifted outside of the range by adding or subtracting 2 depending on which side of the circle it lies on:

 (5.3-123)

or

 (5.3-124)

Then the mean will be updated using this shifted version which now has a distance of less than to the next value in . If after the update is still outside the shift is reversed before the next iteration.

Now the bandwise IPD change, denoted as IPDchange,b, between the current bandwise and bandwise mean is computed for each subband with

 (5.3-125)

with

 (5.3-126)

From the individual bandwise IPD changes in each subband a mean bandwise change, denoted as IPDchange, over all subbands is computed:

 (5.3-127)

Themeanbandwise IPD changeis taken as an overall indication of the stability of the bandwise IPD in the current frame and is now used to force a similar level of stability on theglobal IPD estimate, i.e., to calculate a stabilized IPD estimate using the current global IPD estimate, the transmitted stabilized estimate of the previous frame and the mean bandwise IPD change*.* For small values of the mean bandwise IPD change (smaller than 0.3) the current global IPD is overwritten with the stabilized IPD estimate of the previous frame:

 (5.3-128)

For larger values, a modulus of a difference between the transmitted IPD of the previous frame and the global IPD of the current frame*, denoted as* gIPDdiff*,* is computed*:*

 (5.3-129)

with

 (5.3-130)

If

 (5.3-131)

which means that themodulus of the difference between the transmitted IPD of the last previous frame and the global IPD estimate of the current frame is larger than the mean bandwise IPD change, the maximum alloweddifference to the previously transmitted IPD is limited to the mean bandwise IPD change, so that the global IPD is calculated as:

 (5.3-132)

or

 (5.3-133)

If, however,

 ,

 (5.3-134)

which means that the modulus of the difference between the previously transmitted IPD and the global IPD estimate of the current frame is equal to or smaller than the mean bandwise IPD change, the original global IPD estimate *gIPD* is used for the current frame.

5.3.2.4.7.2 IPD encoding scheme

In addition, a reference parameter is used to determine the IPD parameter encoding scheme. The reference parameter includes at least one of the signal characteristic parameter and the signal characteristic parameters of previous frame. The signal characteristic parameter is calculated as the correlation between left channel and right channel of the current frame. The signal characteristic parameters of previous frame include at least one of the correlations between left channel and right channel of previous frame, an ITD parameter of previous frame, a signal type of previous frame.

The correlation between left channel and right channel is obtained by using the following calculation formula:

wherein

and

wherein indicates an energy sum of left channel, indicates an energy sum of right channel, indicates a real part of a kth frequency value of left channel frequency domain signal, indicates a real part of a kth frequency value of right channel frequency domain signal, indicates an imaginary part of the kth frequency value of left channel frequency domain signal, indicates an imaginary part of the kth frequency value of right channel frequency domain signal, L indicates a quantity of sub-band spectral coefficients, and N indicates a quantity of sub-bands, n indicates an index value of a time domain signal, k indicates an index value of a frequency domain signal, indicates a frame length, indicates left channel time domain signal, indicates right channel time domain signal, indicates a kth frequency value that is of left channel frequency domain signal and that is used to calculate the IPD parameter, and indicates a kth frequency value that is of right channel frequency domain signal and that is used to calculate the IPD parameter, wherein and indicate sequences of real numbers.

If the correlation between the left channel and right channel of the current frame is greater than or equal to 0.75, the IPD parameter encoding scheme of the current frame is skipping encoding an IPD parameter; If the IPD parameter encoding scheme of the previous frame is skipping encoding an IPD parameter, and the signal type of previous frame is music, the IPD parameter encoding scheme of the current frame is skipping encoding an IPD parameter. If the IPD parameter encoding scheme of the current frame is not skipping encoding an IPD parameter, the IPD parameter encoding scheme of the current frame is encoding sub-band IPD parameters of some or all of sub-bands of the current frame.

CHANGE 14

##### 5.3.2.4.12 Residual coding

###### 5.3.2.4.12.1 Overview

The residual coding is achieved after synthesis the signal back in time-domain at a sampling-rate of 8 kHz through the inverse DFT. No overlap-adding is realized since the windowed synthesized signal is directly transformed by a forward MDCT after applying the same windows as the analysis STFT window to achieve an analysis MDCT window equivalent to a sine window.

The residual coding operates in the MDCT domain at a target quantization SNR, specified together with the maximum number of bits that are allowable for each frame. If the target SNR requires a larger number of bits than the maximum specified, the SNR is gradually decreased so that the actual number of bits will satisfy the bit constraint.

The target SNR is derived from the psychoacoustic consideration that quantization errors are more perceptible if the restored stereo channels are out-of-phase. Therefore, the target SNR is made dependent on an out-of-phase estimator. Considering the stereo upmix, the left are right channels are simplistically generated by a mid signal, a side gain as well a residual signal,

 (5.3-176)

Since one can consider that the components and are always in-phase, while and are obviously out-of-phase. The out-of-phase ratio retained is the maximum out-of-phase ratio among the two channels:

 (5.3-177)

given The in-phase ratio is the 1-complementary of , and can be expressed as:

 (5.3-178)

The target SNR is the maximum of the interpolations calculated for each frequency band between a SNR of 10 dB for in-phase components and 40 dB for out-of-phase components:

 (5.3-179)

The real-valued MDCT coefficients of the residual signal are truncated above a maximum frequency given as input configuration parameter and form the vector. The output of the encoder is the global gain index and the entropy coded bits generated by the arithmetic coder. The target SNR is achieved by choosing a suitable global gain index , which is then used to quantize the vector into the integer vector that will be entropy coded.

The global gain index is dequantized to the global gain by the relation

 (5.3-180)

and the global gain is quantized to the global gain index by the relation

 (5.3-181)

where is used for rounding instead of to achieve optimal mean-squared-error reconstruction.

The special global gain index value is used to indicate that all values in the vector are zero. The global gain index is coded raw using 7 bits, and it is always placed before the entropy coded bits, therefore using the special value signals there are no entropy coded bits to follow.

The vector is converted to the quantized vector using the dequantized global gain derived from the chosen global gain index by the relation

 (5.3-182)

which represents scaling by and uniform scalar quantization with rounding to the nearest integer.

Let the block on position be extracted as , for , where the block sizes are . For each block, a parameter which identifies the model used for coding the block is selected and coded as side information. The value , indicating the very low entropy case, uses a model which allows for coding of nonzero values of , while the rest of the values are . This includes the case where all the values in the block are . The values use a model assuming the values are generated by a Laplace distribution with scale parameter .

The parameter value can be used to code only the blocks that satisfy the corresponding model constraints, the other parameter values can encode any arbitrary block, however with different number of bits. For a block, the encoder selects the optimal parameter from those that can be used to code it, such that the total number of bits for coding both the parameter and the block is minimized.

Entropy coding for of a block starts by coding , the number of nonzero values of , with raw coding using 2 bits. Then, the nonzero mask is coded, which contains ones and zeros, with raw coding of the sign bits of the nonzero positions.

encode\_low\_entropy\_block(block, blk\_length)

{

 nz\_count = 0

 for (i = 0; i < blk\_length; i++)

 {

 if (block[i] != 0)

 {

 nz\_count++

 }

 }

 rc\_uni\_enc\_encode\_bits(nz\_count, 2)

 left\_1 = nz\_count

 left\_0 = blk\_length - nz\_count

 for (i = 0; i < blk\_length; i++)

 {

 val = block[i]

 if ((left\_0 == 0) || (left\_1 == 0))

 {

 /\* only ones left or only zeros left \*/

 }

 else

 {

 count\_0 = left\_0 \* ECSQ\_tab\_inverse[left\_0 + left\_1]

 rc\_uni\_enc\_encode\_bit\_prob\_fast(abs(val), count0, 14)

 }

 if (val != 0)

 {

 rc\_uni\_enc\_encode\_bits(get\_sign(val), 1)

 left\_1--

 }

 else

 {

 left\_0--

 }

 }

}

The precomputed table is computed as . Also, a helper function is used to obtain the sign bit, .

During coding of a block, if there are only ones or zeros left, the nonzero mask is already determined. Otherwise, the mask is coded with an adaptive probability model giving the probability of a zero as . This probability is approximately mapped to a 14-bit frequency count, without using a division operation as , where both and , and the second term in the approximation is available in a precomputed table. The value of is derived implicitly from the relation .

The obtained code length in bits of the nonzero mask is exactly the same as would be obtained by optimal combinatorial coding, which would use bits.

Entropy coding with of a block starts by computing , which represents the number of least significant bits of the absolute values that are coded approximately uniformly or with raw coding. The most significant bits of each absolute value are coded using a probability model selected by together with escape coding. For , coding of LSBs takes into account that for absolute values the probability of zero ( maps to one value) is half of the probability of nonzeros ( maps to two values, ). For larger shifts, the length difference is negligible and raw coding is used using bits. Finally, if the value is nonzero, the sign is coded raw.

encode\_normal\_block(block, blk\_length, param)

{

 shift = max(0, param - 3)

 for (i = 0; i < blk\_length; i++)

 {

 val = block[i]

 sym = abs(val)

 if (shift != 0)

 {

 lsbs = sym & ((1 << shift) – 1)

 sym = sym >> shift

 arith\_encode\_prob\_escape(ECSQ\_tab\_vals[param – 1], 16, sym)

 if ((sym > 0) || (shift > 4))

 {

 rc\_uni\_enc\_encode\_bits(lsbs, shift)

 }

 else /\* (sym == 0) && (shift <= 4) \*/

 {

 rc\_uni\_enc\_encode\_symbol\_fast(lsbs, ECSQ\_tab\_abs\_lsbs[shift], 14)

 }

 }

 else

 {

 arith\_encode\_prob\_escape(ECSQ\_tab\_vals[param – 1], 16, sym)

 }

 if (val != 0)

 {

 rc\_uni\_enc\_encode\_bits(get\_sign(val), 1)

 }

 }

}

The encoding of the entire quantized vector can be expressed in terms of the previous two functions, which encode low entropy blocks and normal blocks, together with a helper function *find\_optimal\_parameter*, which computes for a block the optimal parameter to use for encoding.

encode\_raw\_vector(q\_input, N)

{

 block\_cnt = (N + 7) / 8

 for (k = 0; k < block\_cnt; k++)

 {

 blk\_length[k] = min(8, N – 8 \* k)

 for (i = 0; i < blk\_length[k]; i++)

 {

 block[k][i] = q\_input[8 \* k + i]

 }

 param[k] = find\_optimal\_parameter(block[k], blk\_length[k])

 rc\_uni\_enc\_encode\_symbol\_fast(param[k], ECSQ\_tab\_param, 14)

 if (param[k] == 0)

 {

 encode\_low\_entropy\_block(block[k], blk\_length[k])

 }

 else

 {

 encode\_normal\_block(block[k], blk\_length[k], param[k])

 }

 }

}

###### 5.3.2.4.12.2 Adaptive residual signal encoding

5.3.2.4.12.2.1 Adaptive residual signal encoding parameter

For the 32kbps WB coding mode, an adaptive residual signal encoding parameter is used to determine whether to encode the residual signals of the M sub-bands in the current frame. The residual signal encoding parameter is calculated based on downmixed signal energy and residual signal energy of each of M sub-bands in the current frame, wherein spectral coefficients of the current frame are divided to obtain N sub-bands, the M sub-bands are at least some of the N sub-bands, N is a positive integer greater than 1, M ≤ N, and M is a positive integer. The residual signal encoding parameter of the current frame is determined based on the dmx\_res\_all, frame\_nrg\_ratio and res\_dmx\_ratio\_lt. The residual signal encoding parameter is calculated according to

when res\_cod\_mode\_flag is equal to 1 means encode the residual signals, otherwise do not encode the residual signals. The res\_dmx\_ratio is a parameter described relationship between the downmixed signal energy and the residual signal energy of each of the M sub-bands. The res\_dmx\_ratio\_lt is a parameter described a long-term smoothing parameter of the previous frame of the current frame. The long-term smoothing parameter res\_dmx\_ratio\_lt is calculated according to

wherein res\_dmx\_ratio\_lt\_prev represents the long-term smoothing parameter of the previous frame of the current frame, wherein

The res\_dmx\_ratio is calculated according to

res\_dmx\_ratio[b] = res\_cod\_NRG\_S[b]/(res\_cod\_NRG\_S[b] + (1 – g(b))∙(1 – g(b)) res\_cod\_NRG\_M[b] + 1)

wherein res\_dmx\_ratio[b] represents the energy parameter of the sub-band whose sub-band index number is b, b is greater than or equal to 0 and is less than or equal to a preset maximum sub-band index number, res\_cod\_NRG\_S[b] represents residual signal energy of the sub-band whose sub-band index number is b, res\_cod\_NRG\_M[b] represents downmixed signal energy of the sub-band whose sub-band index number is b, and g(b) represents a function of a side gain side\_gain[b] of the sub-band whose sub-band index number is b. The frame\_nrg\_ratio is a parameter described relationship between a sum of residual signal energy and downmixed signal energy of the M sub-bands, and a sum of residual signal energy and downmixed signal energy of M sub-bands in a frequency-domain signal of a previous frame of the current frame.

5.3.2.4.12.2.2 Adaptive downmix for stereo coding

In DFT stereo with WB 32kbps, a corrected downmixed signal is calculated in a preset frequency band of the current frame when a previous frame of a current frame of a stereo signal is not a switching frame and a residual signal in the previous frame does not need to be encoded, or when a current frame is not a switching frame and a residual signal in the current frame does not need to be encoded. The corrected downmix signal is determined by a sum of the downmixed signal and the compensated downmixed signal in the current frame. The compensated downmixed signal in the sub-band b in the subframe i of the current frame is calculated according to:

wherein represents the compensated downmixed signal in the sub-band b in the subframe i of the current frame, k represents a frequency bin index value, and . The downmix compensation factor αi(b) in a sub-band b in the subframe i of the current frame is calculated according to

wherein represents an energy sum of a left channel frequency-domain signal in the sub-band b in the subframe i of the current frame; represents an energy sum of a right channel frequency-domain signal in the sub-band b in the subframe i of the current frame; represents an energy sum of the energy of the left channel frequency-domain signal and the energy of the right channel frequency-domain signal in the sub-band b in the subframe i of the current frame; represents a minimum frequency bin index value of the sub-band b in the subframe i of the current frame; represents a minimum frequency bin index value of a sub-band b + 1 in the subframe i of the current frame; represents a left channel frequency-domain signal that is in the sub-band b in the subframe i of the current frame and that is obtained after adjustment based on a stereo parameter; represents a right channel frequency-domain signal that is in the subband b in the subframe i of the current frame and that is obtained after adjustment based on the stereo parameter; and k represents a frequency bin index value, wherein each subframe of the current frame comprises sub-bands, the downmix compensation factor of the subframe i of the current frame comprises the downmix compensation factor of the subband b in the subframe i of the current frame, b is an integer, , and .

5.3.2.4.12.2.3 Calculation of downmixed signal and residual signal during transitional frame

In stereo coding, obtaining an initial downmixed signal and an initial residual signal of a sub-band corresponding to a preset frequency band in a current frame. If the previous frame is a switching frame, the downmixed signal and the residual signal of the sub-band corresponding to the preset frequency band in the current frame are calculated based on a switch fade-in/fade-out factor of a current frame, the initial downmixed signal, and the initial residual signal. The switch fade-in/fade-out factor of the current frame is determined based on a residual signal coding parameter and an inter-frame energy fluctuation parameter. The residual signal coding parameter is used to represent an energy relationship between a downmixed signal and a residual signal of the current frame, and the inter-frame energy fluctuation parameter is used to represent an energy or amplitude relationship between the current frame and a frame previous to the current frame. The switch fade-in/fade-out factor of the current frame is determined according to

when and , ; when and , ; in another case, ; wherein

 represents the inter-frame energy fluctuation parameter of the current frame which is defined as a ratio of total energy of the downmixed signal and the residual signal to total energy of a downmixed signal of a previous frame and a residual signal of the previous frame; represents the residual signal coding parameter of the current frame; represents the switch fade-in/fade-out factor of the current frame.

The initial downmixed signal and the initial residual signal are calculated according to

wherein represents downmixed signal of a sub-band b in a subframe i in the current frame; represents an initial downmixed signal of the sub-band b in the subframe i in the current frame; represents a compensated downmixed signal of the sub-band b in the subframe i in the current frame; represents an initial residual signal of the sub-band b in the subframe i in the current frame; represents residual signal of the sub-band b in the subframe i in the current frame; the sub-band b in the subframe i in the current frame is a sub-band in the at least one sub-band corresponding to the preset frequency band; k represents a frequency bin index of the sub-band b in the subframe i in the current frame; and , wherein represents a quantity of subframes comprised in the current frame.

when residual coding flag is unequal to residual coding flag value of previous frame, and a modification flag of the residual coding flag of the previous frame is 0 which indicates that the residual coding flag value of the previous frame has not been modified, the residual coding switching flag set to 1 which indicates the frame is a switching frame and the residual signal should be encoded.

5.3.2.4.12.2.4 Adaptive downmix

The encoding mode indication information of the residual signal is obtained by at least one of the following information: residual signal encoding status of previous frame, updating manner flag for a long-term smooth parameter, status change parameter relative to previous frame. The encoding status of the previous frame is used to indicate at least one of the following cases: the quantity of consecutive frames whose residual signals are encoded before the current frame, a quantity of consecutive frames whose residual signals are not encoded before the current frame, and encoding modes of residual signals of previous frame. The status change parameter is a ratio of energy of the current frame to energy of previous frame. The encoding mode that used to indicate whether to encode the residual signal of the current frame is determined based on the encoding mode indication information and the initial encoding mode of the residual signal. The initial encoding mode of the residual signal of the current frame is determined based on the downmixed signal energy and the residual signal energy.

If the following conditions are met, the encoding mode of the current frame is the encoding mode of the previous frame. The conditions include that the initial encoding mode is different from the encoding mode of the previous frame which is closely adjacent to the current frame, and the encoding mode of the previous frame indicates to encode the residual signal of the previous frame, and an additional condition. The additional condition is the quantity of consecutive frames whose residual signals are encoded before the current frame is less than a threshold or the updating manner flag for the long-term smooth parameter is 0, and the encoding mode of the residual signal of the previous frame is not modified. If the additional condition is not met, the encoding mode of the current frame is the initial encoding mode. If the initial encoding mode is the same as the encoding mode of previous frame, the encoding mode of the residual signal of the current frame is the initial encoding mode.

If the following conditions are met, the encoding mode of the current frame is the encoding mode of the residual signal of the previous frame. The conditions include that the initial encoding mode is different from the encoding mode of previous frame, and the encoding mode of previous frame indicates not to encode the residual signal of the previous frame, and a second additional condition. The second additional condition is the quantity of consecutive frames whose residual signals are not encoded before the current frame is less than a threshold or the value of the status change parameter is not less than a second threshold, and not greater than a third threshold. If the second condition is not met, the encoding mode of the current frame is the initial encoding mode.

If the encoding mode of the residual signal of the current frame is different from the encoding mode of previous frame, and the encoding mode of previous frame is not modified, the encoding mode of the current frame is used to indicate the encoding mode of the current frame.

CHANGE 15

5.3.2.4.13 Reverberation gain parameter determination

The left channel signal and the right channel signal are treated as the first channel signal and the second channel signal. Encoder quantizes the first channel signal and the second channel signal based on the downmixed signal, the initial reverberation gain parameter, and the identification information, and writes the quantized first channel signal and a quantized second channel signal into the bitstream.

The reverberation gain parameters correspond to different sub-bands of the first channel signal and the second channel signal. The target reverberation gain parameter indicates those reverberation gain parameters that needs to be encoded. The target reverberation gain parameter is determined based on at least one of coherence between energy of the first channel signal and energy of the downmixed signal and coherence between energy of the second channel signal and the energy of the downmixed signal, wherein each of the first channel signal and the second channel signal comprises a plurality of frequency bins. The identification information is used to indicate a sub-band corresponding to the target reverberation gain parameter and whether the initial reverberation gain parameter needs to be adjusted. The identification information is determined based on the target difference value which is the larger difference value in the first difference value and the second difference value. The identification information uses 1bit to indicate the first frequency band. A target attenuation factor used to adjust initial reverberation gain parameter of a target channel signal is calculated based on the first difference value and the second difference value. Each of the plurality of attenuation factors corresponds to at least one sub-band of the target channel signal, and any sub-band corresponds to only one attenuation factor. The first difference value is a sum of absolute values of difference values between energy of the first channel signal and energy of the downmixed signal at a plurality of frequency bins, and the second difference value is a sum of absolute values of difference values between energy of the second channel signal and energy of the downmixed signal at the plurality of frequency bins. When the first difference value or the second difference value is greater than 120, the reverberation gain parameter corresponding to a sub-band of a first frequency band is the target reverberation gain parameter, wherein the first frequency band is a part of all frequency bands of each of the first channel signal and the second channel signal, wherein a frequency of the first frequency band is less than a frequency of another frequency band different from the first frequency band in the first channel signal and the second channel signal. The plurality of frequency bins are in a second frequency band of each of the first channel signal and the second channel signal, and a frequency of the second frequency band is greater than a frequency of another frequency band, different from the second frequency band, in the first channel signal and the second channel signal.

CHANGE 16

##### 5.3.5.1.1 Signal activity detection in Unified stereo

The signal activity detection is run on the down-mix signal as described in 5.2.2.2.5. To aid in the stereo classification and selection between the TD-based stereo and DFT-based stereo, as well as activating the Stereo CNG mode, an additional signal activity detection is run on the input stereo signals coordinating the selection of encoding mode, see clause 5.3.2.2.2. Based on the encoding mode selected for each channel (as determined by and ), CNG encoding is applied in accordance with the joint VAD decision as obtained from equation (5.3-41). If active encoding is selected for at least one of the channels (i.e., and/or ), active encoding is selected for both channels. However, if a speech pause is detected (i.e., and ), the DFT-based stereo mode is selected to be prepared to encode and transmit CNG frames. Within the DFT-based stereo mode, the IVAS core signal activity detector (see clause 5.2.2.2.5) is further run on the downmix signal, however this time without DTX hangover addition. Although the signal activity detection may not have been triggered for each channel separately, the combination of the two input channels may still trigger the signal activity detection.

In the case signal activity is detected in the downmix signal (, a VAD decision determining whether to apply active encoding mode or CNG encoding mode within the stereo encoding is determined by:

 (5.3-304a)

If , active encoding mode is applied. In the case signal activity is neither detected in the downmix signal, nor in the individual input signals, CNG encoding is applied. Figure 5.3‑45 illustrates the signal activity detection logic.

Figure 5.3‑45: Signal activity detection in Unified stereo

The encoder is determined to be in DTX hangover mode if and the local VAD flag for each of the front VAD channels and the downmix VAD. The local VAD represents an instantaneous VAD decision and is defined in clause 5.1.12.3 of [3].

CHANGE 17

5.5.1 MASA format overview

The metadata-assisted spatial audio (MASA) operation encodes the IVAS encoder inputs that use the MASA format. This is a parametric spatial audio format that can be used with any multi-microphone array with suitable capture analysis. The MASA format is optimised for immersive audio capture by smartphones and other form factors that may utilize irregular microphone arrays.

The MASA format is based on audio channels and an associated set of metadata parameters. The audio signals can be one or two, i.e., mono or stereo. These can be denoted as mono-MASA (MASA1) and stereo-MASA (MASA2), respectively. The metadata parameters include spatial metadata parameters providing information about the captured spatial audio scene for transmission and reproduction of the spatial audio, and descriptive metadata parameters providing further description about the capture configuration and source format of the spatial audio content represented by the MASA format.

Each MASA metadata frame, corresponding to 20 ms of audio, includes the descriptive metadata (consisting of a format descriptor and a channel audio format field that further defines the number of directions described by the spatial metadata, number of audio channels, the source format configuration, and a variable description depending on the previous information) and the spatial metadata parameters that are: direction index, direct-to-total energy ratio, diffuseto-total energy ratio, remainder-to-total energy ratio, spread coherence, and surround coherence.

The direction index (decodable with an elevation and an azimuth component) provides an efficient representation of the multitude of possible spatial directions with about 1-degree accuracy in any arbitrary direction. The direction indices define a spherical grid that covers a sphere with several smaller spheres with centres of the spheres giving the points corresponding with the directions.

Each spatial metadata parameter is provided (through capture, analysis, or creation) for each of 96 time-frequency (TF) tiles corresponding to 4 temporal (or time) subframes and 24 frequency bands. The MASA frequency band borders as CLDFB frequency bins are shown in table 5.5-1.

Table 5.5-1: MASA frequency band borders as CLDFB frequency bins

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Band**  | 𝟎  | 𝟏  | 𝟐  | **3**  | **4**  | **5**  | **6**  | **7**  | **8**  | **9**  | **10**  | **11** |
| **First bin** | 0  | 1  | 2  | 3  | 4  | 5  | 6  | 7  | 8  | 9  | 10  | 11 |
| **Last bin** | 0  | 1  | 2  | 3  | 4  | 5  | 6  | 7  | 8  | 9  | 10  | 11 |
| **Band**  | 𝟏𝟐  | 𝟏𝟑  | **14**  | **15**  | **16**  | **17**  | **18**  | **19**  | **20**  | **21**  | **22**  | **23** |
| **First bin** | 12  | 13  | 14  | 15  | 16  | 17  | 18  | 19  | 20  | 25  | 30  | 40 |
| **Last bin** | 12  | 13  | 14  | 15  | 16  | 17  | 18  | 19  | 24  | 29  | 39  | 59 |

The direct-to-total energy ratio and spread coherence parameters are associated with the direction (parameter). The direction index, direct-to-total energy ratio, and spread coherence parameters are therefore given for each direction described per TF tile (as given by the number of directions descriptive metadata parameter). For each TF tile, the sum of the different energy ratio parameters is 1.0.

The metadata for MASA format inputs shall be provided as defined in detail in Annex A of [12]. The uncompressed MASA metadata size according to definitions in [12] is between 272.8 and 426.4 kbps (268.8 and 422.4 kbps for spatial metadata only), depending on the number of directions in spatial metadata. As IVAS supports encoding of both mono-MASA and stereo-MASA at IVAS bitrates between 13.2 and 512 kbps, as described in Table 4.2‑1, significant compression of the MASA metadata is performed during the encoding process. The compression is based on several overall strategies, including a bitrate dependent configuration for MASA metadata, simplification of the spatial metadata based on combining of spatial metadata parameters in time or frequency, decreasing the number of directions from 2 to 1, and selecting the method of compression based on the configuration parameter indicating the source format.

CHANGE 18

##### 5.5.3.2.7 Combining of MASA spatial audio metadata across multiple directions

This clause describes the combining of multiple MASA spatial audio parameter metadata of the same type for a frequency band, where each of the MASA spatial audio parameters which are combined correspond to a different direction.

The input to the combining process is the MASA spatial audio parameters corresponding to two directions for each frequency band . In effect, the combining process receives two-direction MASA spatial metadata, for each frequency band , consisting of two azimuth , two elevation , and two spread coherence spatial audio parameters, where is the direction index and holds the values of 0 and 1 corresponding to the two different directions. For a frequency band , the two azimuth values, the two elevation values and the two spread coherence values are then each combined into a single value, giving a combined azimuth value, a combined elevation value, and a combined spread coherence value.

Initially, the azimuth and elevation angles are converted to Cartesian coordinate vectors. The length of the vectors is determined based on the direct-to-total energy ratio . This is performed for each direction (where and correspond to the specific subframe and frequency band in which the combining takes place)

Then, the vectors are summed over the two directions

and the length of the sum vector is computed by

Then, an importance metric is determined, specific to the frequency band , which represents the importance of having two directions instead of one for the frequency band . It is determined by

where is the number of subframes (i.e., 4).

Then, based on the importance metric , it is determined whether the encoded values for the frequency band comprises the original azimuth, elevation, and spread coherence for each separate direction, or whether the encoded values comprise the combined azimuth, elevation, and spread coherence. The importance metric is determined for all frequency bands.

Determining whether to encode two separate spatial audio parameter values or combined spatial audio parameter value for a particular frequency band is performed with the use of a “two-direction frequency band” variable, which has the value of 1 for the frequency bands that use the two directions and the value of 0 for the frequency bands that use a combined (single direction) value. The value of the variable, for each frequency band , is determined by sorting the values of for all frequency bands in order of magnitude to determine the frequency bands that have the largest values of .

 corresponds to the factor twoDirBands determined in clause 5.5.3.2.4. For the frequency bands which have the largest values of , is set to 1. For the other frequency bands , is set to 0.

Then, for the frequency bands where , the directions are combined as follows.

The combined azimuth and elevation angles are determined by

Then, a ratio sum variable is determined as

Then, an ambient energy value is determined using the direct-to-total energy ratios by

Then, the combined direct-to-total energy ratio is determined using the ratio sum variable, the ambient energy value, and the length of the sum vector by

Then, the combined spread coherence is determined by

Then, a further ambient energy value (corresponding to the combined direct-to-total energy ratio) is determined by

Then, original surround coherence energy and new surround coherence energy variables are determined as follows

The surround coherence is then adjusted to a new value representing the combined surround coherence by using the original surround coherence energy and new surround coherence energy

CHANGE 19

##### 5.7.3.6.3 LFE-to-total energy ratio encoding

For the operating bitrates of 13.2 and 16.4 kbps, the LFE-to-total energy ratios for each subframe of the current frame are quantized as a single ratio for the whole frame. For higher bitrates, an additional ratio is quantized for each subframe using residual VQ when the LFE frames have a higher energy. Inactive LFE frames for all bitrates are indicated with one bit (0) in the bitstream, and no further LFE-to-total ratio energy coding is performed. The bit allocation, dependent on the McMASA bitrate, is provided in table 5.7‑6.

Table 5.7‑6: Bit allocation for LFE-to-total energy ratio

|  |  |  |  |
| --- | --- | --- | --- |
| Bitrate(kbps) | Bits used(inactive frames) | Bit allocation (active frames) | Bits used(active frames) |
| 13.2 | 1 | 1 (activity / energy ratio modulation) | 1 |
| 16.4 | 1 | 1 (activity) + 3 (scalar quantization) | 4 |
| ≥ 24.4 | 1 | 1 + 3 (scalar quant) + 0…4 (subframe VQ for high energy ratio frames) | 4…8 |

The activity of LFE is detected from the subframe LFE-to-total energy ratios. LFE-to-total energy ratio is only sent when any of the subframes in the frame have a which is above a threshold of 0.005. Otherwise, if the maximum of all the subframes in the frame is less than the threshold, one bit is sent with a zero (0) index for all bitrates.

In the case when a is above the threshold for any of the subframes in the current frame, the encoding process comprises determining an averaged LFE-to-total energy ratio for a frame by

where the LFE-to-total energy ratio for each subframe is clamped between [-9,1] by

At the lowest bitrate of 13.2 kbps only one bit is allocated for the LFE-to-total energy ratios for the frame. In this case the LFE-to-total energy ratio bit is set to (1) if is higher than both a threshold value (MCMASA\_LFE\_1BIT\_THRES=0.03) and a value depending on the previous frame’s quantized value . The comparisons are made in the linear domain by converting the averaged LFE-to-total energy ratio according to . Therefore, the condition for setting the LFE-to-total energy ratio bit to 1, is expressed as when and are true the LFE-to-total energy ratio bit is set to (1). If either of conditions are not met LFE-to-total energy ratio bit is set to (0). In practice on a frame-by-frame basis, when the LFE-to-total energy ratio bit is one this signals an increase in the LFE-to-total energy ratio by a predetermined value and when the LFE-to-total energy ratio bit is zero this signals the dampening of the LFE-to-total energy ratio by a factor.

At the second lowest bitrate of 16.4 kbps 1 (activity) + 3 = 4 bits are used to encode the LFE-to-total energy ratios for the frame when the frame is classified as an active frame, with the first bit being used to signal that the frame is an active frame. When the first bit is 1 (active) next three bits are used to scalar quantize for the frame using a linear scalar quantizer using a codebook of with step of 1.0. This codebook is given in Table 5.7‑7 as the second column.

CHANGE 20

5.8.1 OSBA format overview

The encoder supports combined input with 1 – 4 ISMs and an SBA signal of order 1 – 3. Depending on the IVAS total bitrate, different OSBA coding modes summarized in Table 5.8-1 are employed to combine these input signals.

Table 5.8-1: Overview of coding modes in OSBA format

|  |  |
| --- | --- |
| IVAS bitrate[kbps] | **number of ISMs** |
| **1** | **2** | **3** | **4** |
| 13.2 – 80 | Pre-rendering | Pre-rendering | Pre-rendering | Pre-rendering |
| 96 | Discrete | Pre-rendering | Pre-rendering | Pre-rendering |
| 128 – 512 | Discrete | Discrete | Discrete | Discrete |

Input to IVAS, consisting of audio signals in SBA and ISM formats and the associated metadata, is processed through a simplification stage and an encoding stage. At the simplification stage, the SBA and ISM signals are converted into a mezzanine format, as described in clauses 5.8.2 and 5.8.3, where the mezzanine format depends on the IVAS bitrate and SBA coding as described in clause 5.4. At bitrates less than 256 kbps, the mezzanine format is the First order Ambisonics (FOA) format whereas at bitrates greater than or equal to 256 kbps it includes FOA channels, selected HOA channels and all discrete ISM objects. At the encoding stage, the simplified audio output of the simplification stage is encoded into IVAS bitstream which is then transmitted to the decoder.

CHANGE 21

5.8.4 OSBA bitrate switching

When the bitrate is switched in OSBA format, both encoders (SBA and ISM) are re-configured. The configuration is the same as is there were running as separate instances of IVAS. One special case for OSBA is the switching between bitrates corresponding to different OSBA coding modes. Then the encoder switches between the pre-rendering and the discrete coding mode.

CHANGE 22

#### 5.9.4.2 Low-bitrate pre-rendering coding method

Once the MASA and audio object content have been merged to the MASA format, consisting of the 2 transport audio signal channels and MASA metadata, encoding of the aforementioned “merged” MASA format is performed on the basis that the “merged” MASA format is treated as a stereo-MASA format content at the corresponding overall bitrate. The coding format is signalled as a MASA format at the beginning of the bitstream. In addition, the number of input audio objects is encoded into the bitstream using two bits reserved from the encoding of the MASA metadata, which is as follows:

- ‘01’ if there are 4 objects,

- ‘10’ if there are 3 objects,

- ‘11’ if there are 1 or 2 objects.

In the instance that the number of input audio objects is 1 or 2, i.e., encoded as ‘11’ according to the list above, the bit used to signal the number of transport channels for the MASA format (the MASA number of transport channel signal bit) is repurposed for use in distinguishing between 1 and 2 audio objects. Thus, when the MASA number of transport channel signal bit is ‘0’ this indicates the case of 1 input audio object, and when the MASA number of transport channel signal bit is ‘1’ this indicates the case of 2 input audio objects. In the instance that the number of objects is 3 or 4, the MASA number of transport channel signal bit is used for the encoding of the combined MASA format audio signal (formed by combining the input audio objects converted into the MASA format with the input MASA audio signal comprising the transport audio signals and MASA metadata).

CHANGE 23

### 5.9.10 OMASA bitstream structure

Four bitstream structures are defined for the OMASA case, corresponding to each of the four coding modes. For the *Rend OMASA* coding mode the bitstream has the MASA bitstream structure presented in clause 5.5.6 with the modification indicated in clause 5.9.4.2 on the signalling of the number of input objects. For the *One MASA* coding mode the bitstream has the following order of components: IVAS format bits, the separated audio object data and spatial metadata, MASA transport channels, MASA metadata in which it has been inserted the number of objects and their importance flags. The *Param OMASA* coding mode has the following order of components in the bitstream: IVAS format bits, the separated object audio content, the MASA transport channels, MASA metadata including the ISM energy ratios and MASA-to-total energy ratios, the importance of audio objects, the index of separated audio object, and number of input audio objects. For the *Disc OMASA* coding mode, the bitstream is formed by: IVAS format bits, ISM bitstream, MASA transport channels, MASA metadata, number of input audio objects. In all coding modes except *Rend OMASA* the number of input audio objects is at the end of the bitstream.

CHANGE 24

##### 6.3.1.2.2 Inter-channel Alignment (ICA) decoder

The ICA decoder operates as an inter-channel re-aligner. After receiving the ICA bitstream, dequantization of the inter-channel temporal shift and gain factor parameters takes place, i.e., prevNCShift, currentNCShift, and the targetGain are generated. Similar to what occurred at the ICA encoder, there is a channel identification and target adjustment stage. The reference and target channels are identified. If the temporal shifts between the current and previous frames differ, the target signal is adjusted by the same structure as the Target Sample Adjuster as described in the encoder. However, the shift of the target signal is in the opposite direction as was performed in the ICA Encoder, i.e., the Inter-Channel Re-Aligner places back the left and right channels to their original temporal difference. The adjusted target channel is scaled by a targetGain to balance the audio levels between the two channels.

Decoder obtains the primary-channel signal and the secondary-channel signal, and the ITD from bitstream. Time-domain upmixing processing on the primary-channel signal and the secondary-channel signal is performed to obtain the left-channel reconstructed signal and the right-channel reconstructed signal. Delay of the left-channel reconstructed signal and the right-channel reconstructed signal is adjusted based on the ITD after an interpolation processing which is performed based on the ITD from current frame and previous frame. The ITD after the interpolation processing is calculated according to a formula

wherein A is the inter-channel time difference after the interpolation processing in the current frame, B is the inter-channel time difference in the current frame, C is the inter-channel time difference in the previous frame of the current frame, is a pre-stored interpolation coefficient which can be set to .

CHANGE 25

6.3.2.3.10.1 General

In case of a packet loss, the PLC operation is activated for the DFT stereo. The decoded down-mix signal is generated by running the down-mix decoder PLC to obtain a down-mix PLC frame . A DFT representation of the down-mix PLC frame is obtained in the same way as in error-free decoding as described by in (6.3-11) in 6.3.2.3.2. The stereo parameters that were decoded in the previous frame are generally reused as substitution parameters together with in the same was as in 6.3.2.3.10.2, where the side gain prediction residual concealment frame (if present) is generated as described in 6.3.2.3.10.3. The generated down-mix PLC frame is used together with the substituted parameters and the generated side prediction residual to perform a DFT stereo synthesis as described in 6.3.2.3.7 and 6.3.2.3.9.

CHANGE 26

6.3.2.3.10.3 Side prediction residual PLC

In case of a packet loss for a DFT stereo frame where a decoded side prediction residual is present, the PLC operation is activated to produce a concealment frame of the side prediction residual. This is achieved by combining the Phase ECU with the predicted stereo residual obtained by the stereo filling algorithm as described in 6.3.2.3.6.2. First, the DFT domain down-mix PLC frame is run through the frequency domain decorrelator (see decorrelation of in 6.3.2.3.6.2) to obtain . The magnitude of the previously decoded side prediction residual is combined with the phase from to retain the correlation property with respect to the down-mix PLC frame . This could be done by matching the magnitude of with the magnitude of . A low-complex adjustment is made by matching the order of the absolute values of the real and imaginary part and the signs for each bin of with each bin of , as expressed in (6.3-108). Following this principle, the phase matched is calculated according to

 (6.3-106)

where is

 (6.3-107)

in the case where the order of the absolute values for the real and imaginary components are the same, i.e.

 (6.3-108)

and otherwise

 (6.3-109)

An example of the low complex phase matching is illustrated inFigure 6.3‑7, where the operation moves the phase within the correct section of the unit circle.



Figure 6.3‑7: Low complex phase matching within of target

The Phase ECU algorithm as described in [4] 5.4.3.5.2 and 5.4.3.5.3 is applied on , where the peaks are identified and then refined to on a fractional frequency scale. As described in 6.3.2.3.2, the second analysis window of the subframe of the DFT stereo is a time-reversed version of the first analysis window. Since the last residual subframe is generated from the second subframe, the first ECU subframe is time-reversed to create the matching window shape. The phase adjustment for the time-reversed ECU frame is calculated according to

 (6.3-110)

where is the number of samples between the start of the second DFT stereo subframe of the previous frame to the start of the first subframe of the current frame, is the number of consecutively lost frames and is the length of the DFT analysis frame. For the first lost frame . The second subframe is not time reversed and the phase adjustment is computed similar to [4] 5.4.3.5.2, except the frame length and frame alignment gives different constants.

 (6.3-111)

As in [4] 5.4.3.5.3, the peaks of are adjusted by applying the phase adjustment to the peak bins and their neighbourhood bins according to

 (6.3-112)

for , where denotes complex conjugate and results in a time reversal for the 1st subframe and denotes the set of bins that are part of the peaks and their neighbouring bins in the current frame that are within the limits of the spectrum.

 (6.3-113)

The Phase ECU algorithm in [4] 5.4.3.5.2 and 5.4.3.5.3 is complemented with a separate source for the noise component of the spectrum. The side prediction residual concealment spectrum is formed by combining the phase adjusted peaks and their neighbouring bins with the energy adjusted decorrelated down-mix signal , i.e.

 (6.3-114)

The non-peak bins may be seen as the noise component of the spectrum. If no peaks are found, will comprise only the energy adjusted noise component. The concealment spectrum for the decoded residual signal will be transformed to time domain and included in the reconstructed stereo signal as described in 6.3.2.3.7.

CHANGE 27

##### 6.3.5.1.2 Stereo CNG spectral shape extraction

The EVS CNG decoder operates in two modes, LP-CNG and FD-CNG. In Unified stereo CNG, the CNG is generated in DFT domain based on the spectral shape of the decoded EVS CNG frame. For the LP-CNG, the filter LP synthesis filter described in clause 6.7.2.1.4 of [3] is convolved with the denominator of the de-emphasis filter as described in clause 6.4 of [3], to produce a deemphasized synthesis filter

 (6.3-143)

where ‘’ denotes convolution. is then transformed to DFT domain where the synthesis shape is formed according to

 (6.3-144)

where is a low-pass filtered energy of the low-band excitation signal defined in clause 6.8.4 of [3] and is the FFT length. For a core coding of kHz, and for kHz, . For the high band (up to 14 kHz for kHz and up to 16 kHz for kHz) a corresponding shaping spectrum is obtained as the inverse magnitude DFT spectrum of the LP-filter for SHB-CNG obtained as defined in clause 6.7.2.1.7 of [3].

For TD-based stereo, the CNG is still utilizing the DFT-based stereo mode. For a smooth transition from active TD-based stereo coding to CNG, background noise parameters estimated for the active coding are adapted and combined with background noise parameters from the SID frame. The correlation between the decoded left and right stereo signals of the TD-based stereo mode is estimated according to

 (6.3-145)

where , and and are energies of the output signals from the previous active frame. A low pass filtering of the inter-channel correlation is performed over frames according to

 (6.3-146)

with .

For the first SID after active coding, coherence values for the frequency bands are obtained differently based on whether active frame coding is done in TD-based stereo mode or DFT-based stereo mode, according to

 (6.3-147)

where is defined by equation (6.3-175) and is a flag indicating that the last active frame was running the TD-based stereo mode.

For subsequent inactive the frames the coherence is updated as follows

 (6.3-148)

where number of TD frames for which has been estimated, number of DFT-based stereo frames, is the number of SID frames received by the decoder, andis set to 0.8. is the latest value of estimated during TD-based stereo coding.

For the transition from active TD stereo coding to CNG a crossfade is performed between two noise spectra, one being the background noise parameters representing frames of the active TD stereo coding mode, estimated at the decoder, and the other one based on parameters provided in the SID.

For LP CNG the crossfade is based on background noise parameters estimated in active TD frames, here denoted an for the latest active frame. A crossfade length is determined in the first SID after TD coding according to

 (6.3-149)

where is the maximum crossfade or transition length allowed and is the energy ratio of the two background estimates determined by

 (6.3-150)

Over a transition period comfort noise parameters are generated as the weighted average of the two noise spectra over a transition period according to

 (6.3-151)

where is the number of inactive frames and is a compensation factor used to scale the background noise parameters of TD stereo active coding mode primary (downmix) signal such that its energy corresponds to the energy of a corresponding downmix signal of the CNG coding mode (which is based on the DFT-based stereo coding mode) The scaling parameter is computed by summing scaling factors over all bands and dividing by the number of bands, where

(6.3-152)

with being the TD-based stereo mixing ratio controlling the downmixing, see clause 6.3.2.2, and a target gain applied to the right channel during upmixing, see ICA encoder target gain in clause 5.3.4.2. is given by

(6.3-153)

where is the side gain parameter as described in clause 6.3.5.1.3.

For low band LP CNG, a random noise generator is used to generate noise for the real and imaginary parts. The random noise is scaled with , where is the length of the synthesis frame.

For high band LP CNG, denoted as , a scale factor for the high band CNG is computed as

 (6.3-154)

where the synthesis gain is defined by equation (1971) in [3]. A random noise generator is used to generate noise for the real and imaginary parts. The random noise is scaled with a flipped spectrum of and.

Two uncorrelated noise spectra and in the case of LP CNG are accordingly generated as

 (6.3-155)

 (6.3-156)

where and are two random gaussian noises generated with different seeds.

For FD CNG the background noise estimate at the decoder is updated using in a similar way as in clause 6.7.3.2.3.2 in [3] by combining SID and shaping parameters estimated at the decoder, with the modification provided in equation (6.3-157). The modifications provide a smooth transition when switching from TD active mode to FD CNG by applying a crossfade over a fixed transition length between the adapted background noise parameters based on the background noise parameters estimated during active TD-based stereo and the background noise parameters received in the SID frame, as described below. A full resolution CNG spectrum is obtained as

 (6.3-157)

where is a frequency sub-band index and is determined by

 .

 (6.3-158)

 is the ratio of two noise spectra determined by

 =

 (6.3-159)

with , and as defined in clause 6.7.3.2.3.2 in [3].

 is equivalent to

 (6.3-160)

For the first part of the CNG spectrum, the frequency resolution of the noise shaping function is twice the DFT-based stereo resolution since DFT-based stereo uses two subframes for each frame. The CNG spectrum is formed by averaging the bins two by two according to

 (6.3-161)

For higher frequency coefficients, Two uncorrelated noise spectra and are generated as in clause 6.7.3.3.2 of [3] based on pseudo-random Gaussian noise with different random seeds, scaled with , where is the length of the synthesis frame. Accordingly

 (6.3-162)

 (6.3-163)

where and are two random gaussian noises generated with different seeds.

CHANGE 28

##### 6.3.5.1.3 Stereo CNG side gain, ITD and IPD decoding

The side gain is decoded the same way as in active frames, but with the band resolution as described in Table 5.3‑22. The decoded side gain parameters are low-pass filtered during CNG frames according to

 (6.3-164)

except for the first CNG frame after active coding where is directly set to .

The ITD parameter is decoded similar to the active frames but without the option of Huffman coding and included an extra step due to the reduced resolution.

 (6.3-165)

where is the received ITD index and is the decoded sign bit.

Whether residual encoding is enabled or not during the active encoding mode indicates whether the foreground and background signals are efficiently separated for active frames, and if it is enabled, is used directly for synthesis in the CNG encoding mode, i.e. . However, for CNG stereo synthesis where the active frames are encoded at a bitrate , where residual coding is not utilized (indicating the foreground and background signals are not efficiently separated for active frames), the ITD is adjusted by a gradual fading from the ITD of the previous frame towards the received ITD target, as

(6.3-166)

where is a counter of number of frames for which the fade has been performed and is the total length of the fade, unless interrupted by active frames. One exception is for SID frames following active segments of at most N\_(xfade\\_reset)=2 active frames, which instead are handled according to equation (6.3-168). denotes the ITD of the previous frame, being the latest ITD value of the fading and starting from the ITD of the last active frame prior the CNG period. The size of the steps taken towards the target ITD is set in the beginning of the CNG period, and updated whenever a new is received, according to

 (6.3-167)

The fading counter is increased by one for each frame the fade is being performed and reset to zero when there has been at more than active frames. Following segments of at most active frames, the counter is not reset and the ITD fade is resumed from the ITD of the previous CNG frame instead of being restarted from the last active frame ITD, i.e.

 (6.3-168)

where is the latest ITD value of the gradual fade from the previous CNG period. If a new ITD target is received, the step size is updated as

(6.3-169)

The IPD is decoded according to

 (6.3-170)

where is the decoded IPD index. As for the ITD, if residual coding is enabled for the active frames, is used directly for stereo CNG synthesis, . However, for active frame bitrates where residual coding is not utilized, the IPD is adjusted by a gradual fading towards as

(6.3-171)

where is a counter of number of frames for which the fade has been performed and is an upper threshold for the number of fading frames. denotes the IPD of the previous frame, being the latest IPD value of the fading and starting from the IPD of the last active frame prior the CNG period. The size of the steps taken towards the target IPD is set in the beginning of the CNG period, and updated whenever a new is received, according to

(6.3-172)

The fading counter is increased by one for each frame the fade is being performed and reset to zero when there has been at more than active frames. Following segments of at most active frames, the counter is not reset and the IPD fade is resumed from the IPD of the previous CNG frame instead of being restarted from the last active frame IPD, i.e.

 (6.3-173)

where is the latest IPD value of the gradual fade from the previous CNG period. If a new IPD target is received, the step size is updated as

(6.3-174)

CHANGE 29

##### 6.3.5.1.4 Stereo CNG coherence decoding

The intra-frame predictor index is obtained from the bitstream and the intra-frame predictor is selected. Based on the available bit budget for the encoded stereo coherence for the current frame , the weighting factor is obtained according to Table 5.3‑23, where the decoded bit now indicates whether to select or . The coherence for each band is obtained according to

 (6.3-175)

where the coherence prediction residual is now obtained from the bitstream and decoded according to Table 5.3‑24. The intra-frame prediction and the inter-frame prediction are obtained in the same way as in (5.3-319), using the previously reconstructed coherence values. In case the reconstructed values fall outside of the valid range , the value is clamped to this range according to

 (6.3-176)

The coherence is used together with the remaining decoded stereo parameters and the decoded CNG down-mix signal to produce a stereo CNG synthesis as described in clause 6.3.5.1.5.

CHANGE 30

### 6.4.11 SBA decoding with TSM

For mono and stereo output the all processing steps to produce the final output are done before the TSM in the first processing step. The transport channel buffer is a simple output buffer.

For all other output formats the following processing steps are done before the TSM:

 SPAR and DirAC parameter (meta data) and transport channel decoding

 Application of AGC/PCA on the transport channels

 In case of rendering to binaural with the parametric renderer or the parametric room renderer

a. Calculation of the SPAR upmix matrix

b. Application of the gain for binaural rendering on the transport channels

After the TSM and the transport channel buffer management according to clause 6.2.7.2, the local subframes are calculated according to 6.2.7.4.3.1, the meta data mapping for the SPAR upmix parameters is determined according to clause 6.2.7.4.2.1 Eqs. (6.2-91) and (6.2-93) with

CHANGE 31

#### 6.6.7.2 Mono downmix

To generate mono output for one time frame, a proto signal is first computed from the transport channels, i.e., the transmitted SCEs:

 (6.6-52)

Here, is the time sample index, is the number of time samples per frame, denotes the number of transport channels, is the transport channel index, and describes the -th transport channel.

Furthermore, an input energy and a proto energy are calculated with the help of local energies according to:

 (6.6-53)

 (6.6-54)

 (6.6-55)

 (6.6-56)

The smoothing coefficient is defined as and both and are initialized to 0 for the first fame to be processed. Afterward, the input and proto energies and of the current frame are retained and stored as and for use in the next frame to be processed.

With the maximum allowed downmix gain and , an equalization factor is determined from the input and proto energies:

 (6.6-57)

The mono downmix is finally obtained by equalizing the proto signal according to:

 (6.6-58)

CHANGE 32

6.8.3 OSBA PLC

For low-bitrate pre-rendering OSBA coding mode, the OSBA decoding is mostly identical to the SBA decoding and, consequently, PLC is handled exactly the same way as in clause 6.4.8.

For high-bitrate discrete OSBA coding mode, PLC processing is performed in the SBA and ISM metadata decoders according to clauses 6.4.8 and 6.6.5, respectively, and the MCT according to clause 6.2.3.4.10.

6.8.4 OSBA bitrate switching

In OSBA format, bitrate switching entails re-configuration of both the SBA and ISM decoders. The configuration is the same as if these two decoders were running in separate instances of the IVAS decoder. This is described in clauses 6.4.9 for SBA and 6.6.6 for ISM.

A special case for OSBA is the switching between bitrates employing different OSBA coding modes. Then the decoder switches between the pre-rendering and the discrete coding modes.

When the high-bitrate mode is switched on, additional re-configurations are required as compared to bitrate switching for SBA in clause 6.4.9. Specifically, the number of MCT channels is set according to the SBA configuration and the number of objects. The ISM mode flag is set to signal discrete object coding.

When the high-bitrate mode is switched off, the ISM mode flag is set to signal pre-rendering mode.

6.8.5 OSBA output format conversion

In the pre-rendering OSBA coding mode, the decoder-side processing is identical to that in SBA format. The objects are pre-rendered into the SBA scene on the encoder side. Consequently, the output format and the specific processing associated with it are the same as described in clauses 6.4.10 and 6.4.6.5.8.

In the discrete OSBA coding mode, the output is generated by the SBA and ISM decoders concurrently. Hence, both decoders must be configured to provide the requested output format. The signals from both decoders are then summed up. The SBA output processing is again performed according to clauses 6.4.10 and 6.4.6.5.8. The ISM output format conversions are applied according to clause 6.6.7.

CHANGE 33

### 6.9.2 Low-bitrate pre-rendering (Rend OMASA) decoding mode

The low-bitrate pre-rendering mode is signalled as MASA format and decoded as such, with the following differences:

* The 2 reserved bits from the MASA metadata frame are checked, and if they are not ‘00’ they are interpreted as it is specified in clause 5.9.4.2.
* When the two reserved bits are read as ‘11’, this signals 1 or 2 objects. In this instance the bit for the number of transport channels (the MASA number of transport channel signal bit) is also read in order to distinguish between 1 audio object and 2 audio objects. One audio object is signalled by ‘0’ and 2 audio objects are signalled by ‘1’. When the MASA number of transport channel signal bit is ‘0’ this indicates the case of 1 input audio object, and when the MASA number of transport channel signal bit is ‘1’ this indicates the case of 2 input audio objects.

When the two reserved bits from the MASA metadata frame are read as either- ‘01’ signifying 4 objects or - ‘10’ signifying 3 objects, then the MASA number of transport channels signal bit is used for the decoding of the combined MASA format audio signal.

### 6.9.3 One object with MASA (One OMASA) decoding mode

When the IVAS format is signalled as OMASA, the decoder reads the number of audio objects (i.e. the number of audio objects at the input to the encoder), the importance flag of the separated object and the ISM related flags signalling the presence of ISM metadata (see clause 5.6.5.2). Knowing the IVAS total bitrate and the number of audio objects, the decoding mode is obtained from table 5.9-1, along with the bitrate allocated to the separated object and to the MASA format data. The MASA configuration is realized based on the nominal bitrate initially allocated to the MASA part from the configurations table. The MASA metadata is next decoded, from the end of the bitstream, according to the procedures described in clause 6.5.3. After decoding the MASA metadata, the bit allocation between the one separated object and the MASA content is adjusted to conform to the procedure of clause 5.9.8. The separated object together with its metadata is then decoded. The last part of the decoding obtains the decoded MASA transport audio signal channels according to clause 5.5.4. In OMASA there are always 2 transport channels.

### 6.9.4 Parametric one object (Param OMASA) decoding mode

#### 6.9.4.1 Overview

The one audio object with parametric representation mode decodes, in addition to the decoded parameters from sub clause 6.9.3, the parametric mix representation parameters consisting of the MASA-to-total energy ratios and the ISM energy ratios. The direct-to-total energy ratios are obtained as presented in clause 6.9.6 and are subsequently used for rendering the OMASA content which is presented in clause 6.9.7.

The overall decoding procedure for OMASA, is described by the following flow:

- IVAS format is read

- Number of objects is read

- The index of the separated object is read

- The number of MASA directions is read on 1 bit

- The MASA-to-total energy ratios are decoded (see clause 6.9.4.2)

- The ISM energy ratios are decoded (see clause 6.9.4.3)

- The separated object metadata is decoded (see clause 6.9.4.4)

- The MASA metadata is decoded (see clause 6.5.3)

#### 6.9.4.2 MASA-to-total ratios decoding

The decoding of the encoded data corresponding to the MASA-to-total energy ratios depends on the number of sub frames and sub bands, which can result in one of 4, 5, 8, 12, 20, or 32 indexes being read. The case of 32 indexes is read in 4 streaks of 8 indexes each.

Reading of one streak is as follows:

1. If it is the single streak or the first one from the group of 4

a. The sign of the DCT coefficient of order 0 is read (1 for positive, 0 for negative)

2. End if

3. Read on the next 6 bits the value of the first DCT coefficient of the streak

4. Multiply the coefficient with its sign

5. If the first DCT coefficient is not null

a. If the length of the streak is larger than 8

i. Read on 4 bits the position of the last index, *i\_min*, that has been encoded with Golomb Rice of order GR2

ii. Read first Golomb Rice order GR1 on 1 bit; GR1 is 1 or 2

iii. If GR1 == 2

1. Read 1 bit for GR2 (0 or 1)

iv. Else

1. GR2 = 0

v. End if

vi. Decode *i\_min* indexes with Golomb Rice decoder with order GR2

vii. Decode the rest of indexes with Golomb Rice decoder of order GR1

b. Else

i. Read Golomb Rice order, GR1, on 1 bit (1 or 0)

ii. Decode all remaining indexes with Golomb Rice decoder of order GR1

c. End if

d. Reorder the indexes and dequantize the DCT coefficients using the decoded indexes and the quantization step of 0.1.

6. Else

a. All DCT coefficients are null

7. End if

8. Group the coefficients into a matrix

9. Inverse DCT transform the matrix to obtain the decoded MASA-to-total energy ratios

#### 6.9.4.3 ISM energy ratios decoding

The decoder is configured to receive encoded ISM energy ratio indexes relating to each audio object, which on a per audio object basis consists of the ISM energy ratio index corresponding to each sub frame and sub band of the audio frame. The encoded ISM energy ratio indexes for each audio object are then decoded to give the ISM energy ratio quantization index for each sub frame and sub band of the audio frame. Each quantization index is then used to retrieve the respective corresponding quantized ISM energy ratio value.

Before the decoding procedure, a number of verifications are first performed.

In a first verification, the MASA to total energy ratio of each TF tile is compared against a threshold (whose value is 0.98) in order to determine whether the ratio value if greater than the threshold. If it is determined that all MASA to total energy ratios of the frame are greater that the threshold then the ISM energy ratio indexes are determined to be evenly distributed across the TF tiles of the audio frame such that the ISM energy ratio indexes of the frame sum to K in a manner similar to that laid out in step 1.a of clause 5.9.6.3.3. However, if it is determined that the MASA to total energy ratios for the TF tiles of the frame are not all above the threshold then the encoded information relating to the quantized ISM energy ratio index for each TF tile of the frame is read from the bitstream and decoded.

In a second verification, a combination (for the frame) of whether the separated audio object is the last audio object and the number of audio objects is greater than two is checked. If the check is in the affirmative, then the index corresponding to the ISM energy ratio of the separated audio object is set to zero.

Since the encoding procedure is a combination of absolute coding for the first sub frame, followed by differential coding for the following sub frames of the frame, data from previous sub frames is then stored during the decoding process. The generic decoding procedure is as follows:

1. *T* = - 1

2. For *sf* = 1:*nblocks*

a. Read and decode information relating to the quantized ISM energy ratio index vectors for all sub bands for *T* audio objects to give the quantized ISM energy ratio index vectors for all the sub bands and the T objects of the subframe.

b. Save current subframe quantized ISM energy ratio index vectors for use as a previous sub frame data

c. If and the separated object is the last audio object

i. Interchange the quantized ISM energy ratio index for the first audio object with the last audio object.

d. End if

e. If *sf ==1* and all decoded quantized ISM energy ratio indexes of sub bands are zero

i. *T* =  *–* 2

f. Else

i. *T* =  *–* 1

g. End if

h. Reconstruct quantized ISM energy ratios from the quantized ISM energy ratio indexes

3. End for

The decoding uses one of the following: a deindexing procedure, a differential decoding based on previous sub band or a differential decoding based on the previous sub frame.

The deindexing procedure is applied to the value *index* which is read from the bitstream as a fixed number of bits depending on the total number objects. The number of bits read is one of 3, 6, or 7 corresponding to the number of objects of 2, 3, and 4 respectively. The deindexing procedure, for the first subframe is presented in the following pseudo-code and is executed as step 2.a in the above generic decoding procedure. The result is the group or vector of indexes which sum up to , and represent the indexes of the quantized ISM energy ratios for all the objects, for one TF tile. The number *nb* is the number of bits used for the quantization of an individual ISM energy ratio. is the total number of audio objects.

CHANGE 34

#### 6.9.4.4 ISM metadata decoding

Encoding of the directional parameter for each audio object over a frame occurs when the IVAS codec is operating in the one separated audio object with parametric representation coding mode.

In this mode, for the decoding of the audio object directional parameter, the MASA-to-total energy ratios for all the TF tiles in the frame are initially decoded together with the ISM energy ratios for all audio objects and for all TF tiles in the frame. These decoded values are then used to derive the priority values for all the audio objects in the frame according to equations (5.9-1) and (5.9-2).

A bit allocation for the directional parameter of each audio object is then calculated using the priority value calculated for the respective audio object according to equation (5.9-3).

The directional parameter (for each audio object) is then decoded using the calculated respective bit allocation. This is performed by the following procedure:

If the bit allocation is less than a threshold bit allocation value of 8 bits, a signal bit is read. If the value of the signal bit is 1, this indicates that the current audio object’s azimuth and elevation directional parameter values are the same as the directional parameter values of the audio object from of the previous frame. However, if the value of the bit is zero, then a spherical grid quantizer is used to decode/de-index the azimuth and elevation values of the directional parameter of the audio object. The quantizer is based on the remaining bits of the bit allocation after the signal bit has been taken into account.

If the bit allocation is larger or equal to the threshold bit allocation of 8 bits, then the spherical direction index for the azimuth and elevation values of the directional parameter of the audio object decoded/de-indexed using a spherical grid quantizer based on the bit allocation number of bits for the directional metadata.

For the case when the bit allocation is lower than 8 and the directional metadata differs from the previous frame values, an additional smoothing procedure is performed for the azimuth value as detailed below:

CHANGE 35

### 6.9.5 Discrete (Disc OMASA) decoding mode

In the Disc OMASA coding mode, first the adapted ISM total bitrate and adapted MASA total bitrate are set in the inter-format adaptation module (clause 5.9.8). Then, in a sequential order, the ISM format decoder (clause 6.6) decodes the ISM channels including their metadata followed by the MASA format decoder (clause 6.5) decoding the two MASA transport channels including their metadata. The transport audio channels and metadata related to both ISM and MASA parts are supplied to the decoder and rendered to the requested output configuration.

CHANGE 36

### 6.9.7 OMASA rendering

#### 6.9.7.1 Binaural rendering

##### 6.9.7.1.1 Discrete coding mode

First, the separately coded audio signals are separated to a separate signal , having channels. The MASA transport audio signals are separated to another signal , having 2 channels.

The MASA transport signals are rendered to binaural output signals , using the parametric binaural renderer presented in clause 7.2.2.3.

The separated signals are attenuated by a gain of 0.7943 and delayed by 5 milliseconds to match the delay caused by the CLDFB processing of the parametric binaural renderer. The resulting signals are are rendered to binaural output signals , using the time-domain binaural renderer presented in clause 7.2.2.2.

Then, the rendered output signals from the two renderers are combined

forming the output signals of the rendering process.

##### 6.9.7.1.2 Other coding modes

Binaural (with and without a room effect) rendering is performed using the parametric binauralizer and stereo renderer presented in clause 7.2.2.3.

#### 6.9.7.2 Stereo rendering

Stereo rendering is performed using the parametric binauralizer and stereo renderer presented in clause 7.2.2.3.

#### 6.9.7.3 Multi-channel loudspeaker and Ambisonics rendering

##### 6.9.7.3.1 Overview

The multi-channel loudspeaker and Ambisonics (SBA) rendererrenders multi-channel loudspeaker and Ambisonics output signals from decoded OMASA transport audio signals, separated object signals, spatial metadata, and object metadata.

The rendering is performed in subframes, where denotes the subframe index. A subframe containsCLDFB slots (in non-JBM operation, , in JBM operation 7). The data determined at previous subframes (i.e., subframes -1 and earlier) affects the rendering of the present subframe due to temporal averaging and interpolation.

The fetching of temporally correct spatial metadata parameter values for the current subframe so that they are in sync with the audio signals is described in clause 6.2.7. It operates differently for the JBM and non-JBM use. Fetching the correct spatial metadata values is not discussed in the following, it is assumed that it has already been correctly performed, as described in the aforementioned clause.

As an input, the renderer obtains (or receives) audio signals. First, the separately encoded audio signals are separated to a separate signal , having channels, depending on the number of objects and the OMASA coding mode. The MASA transport audio signals are separated to another signal , having 2 channels.

In addition, the renderer obtains (or receives) spatial metadata associated with the MASA transport audio signals. The spatial metadata obtained (or received) by the renderer contains the following parameters: azimuth , elevation , direct-to-total energy ratio , spread coherence , and surround coherence

In addition, the renderer obtains (or receives) object metadata containing azimuth and elevation angles associated with each separated object.

First, the MASA transport signals are rendered to multi-channel or Ambisonic output signals using the methods described in clause 6.9.7.3.2.

Then, the separated signals are delayed by 5 ms to match the delay caused by the CLDFB processing of the MASA part rendering. The resulting signals are .

Then, the panning gains are determined based on the azimuth and the elevation angles for each separate object and output channel . If the output is multi-channel loudspeakers, the panning gains are calculated using VBAP as presented in clause 7.2.1.2.2 (where the VBAP is initialized before the gain determination using the methods described in clause 7.2.1.2.1). If the output is Ambisonics, the panning gains are calculated by determining the spherical harmonic gain vectors , which correspond to the azimuth and elevation angles, using the ACN channel order (see clause 6.4.6.5.1) and the SN3D normalization, and setting the determined spherical harmonic gain vectors as the panning gains ().

Then, an interpolator is determined

Using these gains, the rendered signals are created by panning the object signals by

where refers to the last subframe of the previous frame when . The rendering is performed for all subframes .

Then, the rendered output signals from the two renderers are combined

forming the output signals of the rendering.

##### 6.9.7.3.2 Rendering of the MASA part

First, the direct and diffuse power factors and surround coherence ratios are computed based on the direct-to-total energy ratio and the surround coherence , using the methods described in clause 6.5.7.2.2.

Then, directional responses are computed based on the MASA spatial metadata using the methods described in clause 6.5.7.2.3.

Then, diffuse responses are computed using the methods described in clause 6.5.7.2.4.

Then, the transport audio signals are transformed to the time-frequency domain with a 60-bin (with the sampling rate of 48 kHz) complex low-delay filter-bank (CLDFB) (see clause 6.2.5 for details), resulting in , where is the frequency bin index, is the CLDFB temporal slot index, and is the transport audio signal channel index.

Then, prototype audio signals (direct and diffuse prototype audio signals) are determined based on the transport audio signals) using the methods presented in clauses 6.5.7.2.5 and 6.5.7.2.6.

Then, the determined diffuse prototype audio signals are decorrelated using the methods presented in clause 6.5.7.2.7.

Then, the spatial audio signals are synthesized using the methods presented in clause 6.5.7.2.8, yielding .

Finally, the time-frequency domain signals are converted to the time domain via the inverse CLDFB (see clause 6.2.5 for details), yielding , which are the synthesized time domain multi-channel loudspeaker or Ambisonic signals associated with the MASA part.

#### 6.9.7.4 Mono rendering

The transport audio signals and the separately coded audio signals are first summed together. Then, the energy of the summed signal and the energies of the transport audio signals and the separately coded audio signals are computed. A gain is then applied to the summed signal such that the energy of the summed signal equals the sum of the energies of the transport audio signals and the separately coded audio signals. The resulting signal is then outputted as the mono audio signal.

CHANGE 37

### 6.9.10 OMASA decoding with Time Scale Modification (TSM)

OMASA decoding with time scale modification follows the procedures from clause 6.2.7 and clause 6.9.

### 6.9.11 OMASA decoding to original combined input format

#### 6.9.11.1 Overview

In this mode, the IVAS codec outputs the audio data in the same format that was input to the encoder. For the OMASA case, this means that the input should be formed by the MASA audio transport channels, MASA metadata, audio objects audio content and the objects metadata. The following sub-clauses describe the obtention of the OMASA format data in the 4 coding modes of OMASA.

#### 6.9.11.2 Decoding to original combined input format in pre-rendering mode

In the pre-rendering coding mode of OMASA, the data is encoded and signalled in MASA format. Consequently, after reading the coding format as MASA format, additional information is read to determine if the original input format was OMASA. If that’s the case, the number of objects in the input is also read, as described in clause 6.9.2. The output data is formed by the decoded MASA format data together with the corresponding number of null objects. The null objects have zero valued audio content and zero valued metadata.

#### 6.9.11.3 Decoding to original combined input format in one object with MASA representation mode

According to what has been described in clause 6.9.3, for the one object with MASA representation coding mode, the information pertaining to the combined spatial audio signal in MASA format and the audio objects is obtained by decoding, for each frame, the encoded spatial audio signal to produce audio metadata in the MASA format and audio channel signals, the information used for decoding the separated object audio content and metadata, and the information related to the number of input audio objects to produce the same number of audio objects in the output data.

The number of objects is read on the two last bits of the bitstream. A corresponding number of null objects is generated following the steps as described further below.

The combined spatial audio signal represented in MASA format is formed of the transport audio signals and the corresponding MASA metadata (azimuth , elevation , direct-to-total energy ratio , spread coherence , surround coherence , and diffuse-to-total energy ratio .

The encoded separate audio object is decoded to give a separate audio object, containing the separately coded object audio signal and the corresponding ISM direction (azimuth and ) (see clause 6.9.3).

The following paragraphs describe how the original input format of MASA format spatial audio format with audio objects are obtained from the decoded separated audio object and from the decoded combined audio signal in MASA format.

First, the MASA transport audio signals are transformed to the time-frequency domain with a 60-bin (with the sampling rate of 48 kHz) complex low-delay filter-bank (CLDFB) (see clause 6.2.5 for details), resulting in , where is the time-domain signal sample index, is the frequency bin index, is the CLDFB temporal slot index, and is the transport audio signal channel index. The audio signal of the separately coded object is also transformed to the time-frequency domain with the same filter bank, resulting in .

Then, energies in MASA frequency bands and subframes are determined for the MASA transport audio signals and the separate signal by

where and are the first and the last bin of the frequency band , and and are the first and the last temporal slot of the subframe .

Then, the decoded separate audio object is converted to an object-based MASA stream, i.e., to an object-based MASA spatial metadata and to an object-based MASA transport audio signal. First, the object-based MASA spatial metadata parameters are determined as follows. The decoded ISM direction is set to the MASA direction (azimuth and elevation) of the created stream

The direct-to-total energy ratio is set to 1

The spread coherence, the surround coherence, and the diffuse-to-total energy ratio are set to 0

Then, object-based MASA transport audio signals are determined. Stereo panning gains are determined based on the decoded ISM direction (, ) using the methods described in clause 7.2.2.3.6 (for the “stereo” mode operation).

Then, an interpolator is determined

where is the length of the subframe in samples, and takes the values corresponding to a single subframe. The interpolator for a single subframe is concatenated four times to get the interpolator for the whole frame . Then, interpolated stereo panning gains are determined by

Using the determined interpolated stereo panning gains, the object-based MASA transport audio signals are determined

Then, the decoded MASA metadata (, , , , ) (see clause 6.9.3) and the determined object-based MASA metadata (, , , , , ) are combined using the methods described in clause 5.9.3.2 using the determined energies and . The result is the combined MASA metadata (, , , , ), which is set to the output.

Then, the decoded MASA transport audio signals (see clause 6.9.3) and the determined object-based MASA transport audio signals are combined

The resulting combined MASA transport audio signals are set to the output.

Then, a number of null audio objects are generated as a substitute for the plurality of audio object inputted in the encoder. The number of the null audio objects corresponds to the number of objects read from the end of the current frame bitstream. The null audio objects contain an audio object channel signal having zero sample values and an audio object direction which is a predetermined fixed value.

The audio channel signal having zero sample values is determined by

The determined null object audio signals for each object are set to the output.

Then, the corresponding object metadata is determined by setting the azimuth and the elevation angles to zero

The resulting object metadata is set to the output.

#### 6.9.11.4 Decoding to original combined input format in parametric one object mode

According to what has been described in clause 6.9.4, for the one object with parametric representation coding mode, the information pertaining to the combined spatial audio signal in MASA format and the audio objects is obtained by decoding, for each frame, the information corresponding to the audio content of one separated audio object, the information corresponding to the MASA format representation of the initial MASA format audio signal and the rest of the audio objects, the MASA spatial metadata, the information to produce the metadata for the set of audio objects, and the information related to the audio object energy ratios and audio signal energy ratios which are the parameters defining the mix between the MASA format data and audio object data.

The number of objects is read on the two last bits of the bitstream. If there were more than one object, two additional bits are read signalling the index of the separated object. The importance of the separated object is read on two bits and serves at the bitrate adaptation according to clause 5.9.8.

The combined spatial audio signal represented in MASA format is formed of the transport audio signals and the corresponding MASA metadata. The following paragraphs describe how the original input format of MASA format spatial audio format with audio objects are obtained from the decoded separated audio object and the decoded combined audio signal in MASA format and from the parametric signal mix information decoded according to the methods described in 6.9.4.2 and 6.9.4.3.

First, the object audio signals are generated for each audio object from the transport audio signals , ISM ratios, the MASA-to-total energy ratios, and the object directions. As a first step, the transport audio signals are transformed to the time-frequency domain with a 60-bin (with the sampling rate of 48 kHz) complex low-delay filter-bank (CLDFB) (see clause 6.2.5 for details), resulting in , where is the time-domain signal sample index, is the frequency bin index, is the CLDFB temporal slot index, and is the transport audio signal channel index.

Then, stereo panning gains are determined for each object based on the decoded ISM direction (, ) (see clause 6.9.4) using the methods described in clause 7.2.2.3.6 (for the “stereo” mode operation). An interpolator is determined

where is the number of CLDFB slots per frame. Interpolated stereo panning gains are determined by

where are the stereo panning gains determined for the previous frame.

Then, prototype object audio signals are determined for each object by

Then, the energies of the prototype object audio signals and the MASA transport audio signals are determined

Then, the target energies for the object audio signals are determined

where is the rendering direct-to-total energy ratio for the object (determined in clause 6.9.6 based on the ISM ratios and the MASA-to-total energy ratios), and is the subframe index.

The energies are smoothed over time by

Then, the object processing gains are determined by

and using them the rendered object audio signals are determined by

The time-frequency domain rendered object audio signals are converted to the time domain via the inverse CLDFB (see clause 6.2.5 for details), yielding the rendered object audio signals in the time domain.

The encoded separate audio object is decoded to give a separate audio object, containing a separately coded object audio signal . The decoded separately coded object audio signal is delayed by 5 milliseconds to match the delay caused by the CLDFB processing, yielding . Then, the delayed separately coded audio object signal and the rendered object audio signals are combined

where is the object index of the separately coded object audio signal (see clause 6.9.4.1).

The audio objects are created by assigning a corresponding directional value for each audio object signal . The directional values (azimuths and elevations for each object ) are obtained from the decoded object directions determined in clause 6.9.4.4. The determined audio objects (containing the object audio signals and the corresponding directions) are set to the output.

Then, MASA transport audio signals associated with the MASA transport audio signals inputted in the encoder are separated from the MASA transport audio signals obtained in the decoder (also containing the object audio signal content) by processing them based on the MASA-to-total energy ratios.

The target energies for the MASA transport audio signals are determined

where is the MASA-to-total energy ratio (determined in clause 6.9.4.2).

The energies are smoothed over time by

MASA processing gains are determined by

and using them the MASA transport audio signals are processed by

The determined processed MASA transport audio signals are set to the output. Alongside the determined MASA transport audio signals, the decoded MASA spatial metadata (determined in clause 6.9.4.1) is set to the output.

#### 6.9.11.5 Decoding to original combined input format in discrete mode

The decoding to original combined input format in the discrete mode is straightforward. After reading the number of objects from the end of the bitstream, and the format bitrate distribution based on object importance as described in clause 5.9.8, each of the two input formats are decoded separately and output independently.

CHANGE 38

### 7.4.1 Rendering control overview

Rendering control means allow for real-time control of rendered audio and for customizing the immersive experience. This clause discusses on two groups of rendering control means: scene and listener orientation control and rendering customization control. The basic aspects of scene and listener orientation are provided in clause 7.4.2. The scene and listener orientation control include means to track, compensate, modify, and process scene orientation, listener pose handling and capture device orientation handling, including several orientation tracking or compensation mechanisms, and combining the above. These aspects are discussed in clauses 7.4.3 through 7.4.6. The customization control mechanisms include HRTF and BRIR sets handling for binauralization (clause 7.4.7) and room acoustics parameters handling for room acoustics synthesis (clause 7.4.8). All the above are applicable to binaural rendering to support immersive experience. Next to that, custom loudspeaker layout control is discussed in clause 7.4.9.

### 7.4.2 Scene and listener orientation

#### 7.4.2.1 Scene orientation

The spherical coordinates and correspond to the azimuth and elevation of a point on the surface of the unit sphere, respectively. Values for azimuth are positive left and are expected to be in the range from -180° (exclusive) to 180°. Values for elevation are positive up and are expected to lie in the range from -90° to 90°. Any values outside these ranges will be wrapped to within these ranges. Loudspeakers are positioned on the surface of the unit sphere, audio objects may in addition have a radius and directivity (cf. clauses 5.6.4.3 and 7.2.2.2.7).

Scene orientation control can be used for modifying the scene, e.g., in response to information about the default scene orientation or in response to potential capture device orientation changes. If available at the renderer, such orientation parameters can be used to rotate the rendered immersive scene. This can for example be done to compensate for a change of the capture device orientation in case such a compensation has not already been carried out in a preceding processing step, e.g., during capture at the sending device. This control can also be used to undo the compensation of a capture device orientation change, e.g., in case the application requires that capture device orientation changes have a corresponding effect in the rendered immersive scene. Note that compensation of a capture device orientation change means that the immersive scene is displaced in the same rotation sense as the capture device. Undoing a compensation means that the scene is displaced by the inverse rotation of capture device. Details of scene orientation control with such kind of external orientation input are described in clause 7.4.5. When used in combination with head tracking, the corresponding details are described in clause 7.4.6.

#### 7.4.2.2 Listener orientation

The listener orientation is defined with respect to the coordinate system described in clause 7.4.2.1. By default the listener’s head is positioned at the origin (0,0,0) and faces towards the positive x-axis (1, 0, 0). This can be described by a vector which is therefore . The listener’s head only rotates around the origin and may not be repositioned for three degrees of freedom cases. IVAS codec also supports six degrees of freedom where the listener can move away from the origin over three axes in Cartesian coordinates as explained in clause 7.4.3.1. Listener orientation may be modified by the orientation tracking module which is further described in clause 7.4.4.

### 7.4.3 Head tracking

#### 7.4.3.1 Head tracking via scene displacement

Head tracking in the IVAS codec is implemented via scene displacement. The coordinate system is depicted in Figure 7.4‑1 the immersive scene. To track head movement of the user, the scene is displaced by the **inverse** rotation of the listener’s head. For example, if the listener orientation points towards the front center speaker of a 5.1 layout and is updated to point towards the front-right loudspeaker (cf. clause 4.3.2) at -30° azimuth, the scene must rotate +30° in yaw around the z-axis to reposition the speaker, as if it were the front center loudspeaker directly in front

where and are basis vectors. The IVAS\_QUATERNION structure used for this data is:

typedef struct

{

 float w, x, y, z;

} IVAS\_QUATERNION;

A special value of -3 (which cannot occur for unit quaternions) for can be used to indicate that the values and correspond to Euler angles of yaw, pitch and roll respectively.

IVAS codec supports six degrees of freedom with optional listener position parameters *pos.x, pos.y* and *pos.z* in the head tracking structure. The default values for the listener position parameters are set to ‘zero’ placing the listener at the origin. Listener position parameters are defined in Cartesian coordinates and no limit is set for the movement of the listener.

CHANGE 39

### 7.4.5 External orientation input handling

#### 7.4.5.1 Overview

The external orientation input provides to the IVAS renderer any orientation information separate from the head orientation (rotation) data of the listener. The external orientation data, when available, is therefore processed and applied in addition to the head-tracking data, which is described in clause 7.4.3.

Figure 7.4‑4 presents a block diagram describing the external orientation inputs at the IVAS renderer.



Figure 7.4‑4: Block diagram of external orientation input handling for binaural rendering

CHANGE 40

##### 7.4.7.3.6 Conversion from Time domain to SH CLDFB domain for parametric binauralizer

7.4.7.3.6.1 Overview

The conversion of the time domain head-related impulse response HRIR and BRIR data to CLDFB-domain spherical harmonic binaural rendering matrices is described here. The data is assumed to be at 48 kHz sampling rate.

7.4.7.3.6.2 HRIR conversion

The HRIR database consists of time domain binaural responses corresponding to a multitude of directions in anechoic conditions. The responses are assumed to be time-aligned. If not, time-alignment can be achieved by measuring the average group delay of each response pair, and temporally aligning all HRIR pairs in the dataset so that they have the same average group delay.

As the first step, a common group delay in samples is formulated as a median group delay value in samples over all directions and both left and right response channels, and this value is rounded to the nearest integer value.

Next, a reference impulse is generated that is zeros otherwise, but one at the sample corresponding to the common group delay.

The reference impulse and the HRIRs are then converted to a time-frequency representation using the CLDFB. Prior to this conversion, the responses are zero-padded at the end so that all time domain data is represented at the converted CLDFB domain responses.

Let us denote the CLDFB domain responses as whereis the bin index, is the temporal slot index, is the channel index, and is the source direction index in the HRIR database. is the reference impulse CLDFB domain response. Non-equalized HRTFs for each direction is formulated by

where is the imaginary value.

Then, diffuse field equalization is performed by first determining a spatially even set of 240 directions in 3D, and then selecting for each of these directions the closest data point at the HRIR dataset. The indices of these data points are , and the total number of these indices is denoted . A single data point of the HRIR set can be represented multiple times in the set .

The HRTF set is then equalized by

where is the median of , and finally,

The HRTFs are then transformed to the spherical harmonic representation. Let us denote matrix as a matrix of size containing at its columns the third order spherical harmonic encoding gains for each of the source directions. Let us denote matrix as a matrix having the HRTF gains as its elements, where index populates the first axis and the second axis. For bins where the bin center frequency exceeds 2kHz, then the elements of are *.*

The spherical harmonic domain rendering matrix for bin is then

where is the pseudoinverse of . These matrices can then be used in determining HRTFs as described in clause 7.2.2.3.6 where is used as .

7.4.7.3.6.3 BRIR conversion

Typical HRTF datasets are obtained in a high number of spatial positions. For example, by use of simulations or moving loudspeakers arcs at the anechoic chamber, one can obtain up to hundreds or thousands of measurement positions, resulting in a high spatial precision.

On the contrary, typical BRIR datasets are obtained in listening rooms with lesser number of loudspeakers. Typical surround loudspeaker arrangements have in the range of 6-12 loudspeakers, whereas some listening rooms are equipped with extensive arrangements over 20 loudspeakers.

However, even the setups with 20 to 30 loudspeakers are not close to the spatial fidelity of the typical HRTF sets. Therefore, the parametric renderer selects a hybrid approach: The directional sound is rendered with HRTF-based data, but the rendering is modified so that it attains characteristics of the given BRIR dataset. The spatial rendering in such a scheme is described in clause 7.2.2.3. Here, it is described how these relevant characteristics are estimated from a given BRIR dataset.

First, from the given BRIR set, five responses are selected, which are the responses closest to reference positions 30°, 0°, -30°, 110°, -110° at the horizontal plane. Other BRIR responses are discarded. Then, the BRIR set is high-pass filtered with a 5th order Butterworth high-pass filter with the cut-off frequency at 80 Hz.

Next, the maximum energy sample indices of the BRIRs are formulated. The energy response of each of the five BRIR pairs is formulated as a where is the BRIR of index , is the sample index, and is the left or right ear channel index.

A 120-sample length lowpass filter window is determined by

Then, is convolved with to find temporally smoothed energies, and the maximum energy position of the BRIR response is determined based on the maximum energy of the temporally smoothed response, for each independently.

Then, each of the BRIR responses are independently truncated from the beginning so that the maximum energy position corresponds to the 61st sample of the truncated response. This procedure aligns the obtained BRIR responses so that the direct sound portion is aligned to a first frame ranging from sample indices 1 to 120.

Next, the BRIR set is transformed to a time-frequency representation via STFT, using 120-sample length FFT operation, 60-sample length hop size and a complex 120-length window that is defined by

In the above STFT procedure, no zero-padding is applied at the beginning. In other words, the first frame temporal indices range from 1 to 120. Only the first 60 bins of the STFT operation are preserved. These bins have the same centre frequencies as the CLDFB frequency domain data, and the same temporal resolution.

The energies of the STFT operation results are formulated by

where is the STFT domain (as described in the foregoing) BRIR where is the bin index and is the temporal index. The energy data is also converted to decibels by

The BRIR analysis is performed based on this energy data for each frequency bin independently. First, for each bin , the start index of the late reverberation is determined as the one that has the maximum energy at range .

Next, a line is fitted in the least squares sense to all sequences of that have the starting point and ending point . The value is tested for each so that , up until is the temporal length of .

Then the best line is selected. First, an estimation error value is defined by

Then, is selected as that which has the largest that satisfies the condition , where is the minimum of over all . The purpose of this procedure is to find the longest sequence that does not contain the noise floor.

The selected values are then collected by . These values are then median-filtered by a 5-length median filter, truncated at the edges (e.g., 2nd bin has median filter from 1st bin to 4th bin), resulting in .

Then, the reverberation time , for each bin is determined based on fitting a linear line to for indices , and the is the time when that line reduces by 60 dB.

For highest bins the estimates may be noisy. Therefore, a linear fit is performed for in range , resulting in . Then the values for are modified so that the resulting estimates linearly interpolate from to at range and then after it remain at The result is the estimated set of reverberation times.

Then, early and late part energies are formulated by

Then, the reference HRTF energies are formulated from HRTFs as determined in clause 7.4.7.3.6.2, which are stored in spherical harmonic representation, denoted . From the spherical harmonic representation HRTF pairs are determined by where are the spherical harmonic encoding gains for the five direction (i.e., 30°, 0°, -30°, 110°, -110°). Energy is the mean energy of these five HRTF pairs, i.e.

These reference HRTF pairs are further modified by where is the median of over all .

Then, early part energy correction gains are formulated by

An exception is for where

which adds 1 decibel to compensate for the effect of the applied high-pass filter.

CHANGE 41

### 7.5.1 Overview

Pre-rendering in IVAS is generally the operation to transform one or more audio inputs into another encoder input format supported by the IVAS codec. Pre-rendering is possible using the rendering tools available in the codec framework. Pre-rendering can be desirable when several audio inputs are to be encoded using a single encoder instance.

IVAS pre-rendering methods allow pre-rendering into scene-based audio (SBA) format, multichannel (MC) format, binaural formats (cf. clause 7.2.2.1). Pre-rendering to metadata-assisted spatial audio (MASA) format using the IVAS pre-rendering methods may only be performed if one or more of the audio inputs to the pre-rendering is a MASA format input. The latter option allows to avoid a pre-rendering delay of 5 ms, if one or more of the audio inputs to the pre-rendering is a MASA format input. Such delay would be incurred when rendering MASA into another format, such as SBA. Therefore, IVAS pre-rendering does not increase the overall codec delay.

Table 7.5‑1 summarises the available prerendering combinations. Channel based formats in the table refers to mono, stereo, supported MC layouts or custom loudspeaker layouts.

Table 7.5‑1: Overview of Pre-rendering from/to supported formats

|  |  |
| --- | --- |
| Input format | Output Format |
|  | **Channel based** | **SBA** | **MASA** | **BINAURAL\*** |
| Channel based | Clause 7.5.5.4 | Clause 7.5.2.2 | Clause 7.5.3.3 | Clause 7.5.4.1 |
| SBA | Clause 7.5.5.2 | N/A | Clause 7.5.3.4 | Clause 7.5.4.1 |
| MASA | Clause 7.5.5.3 | Clause 7.5.2.3 | Clause 7.5.3.5.1 | Clause 7.5.4.3 |
| ISM | Clause 7.5.5.1 | Clause 7.5.2.1 | Clause 7.5.3.2 | Clause 7.5.4.2 |

CHANGE 42

## 8.4 Bit allocation for metadata-assisted spatial audio (MASA)

### 8.4.1 Bit allocation for MASA in active frames

Active frame signalling for IVAS MASA operation (MASA) is summarized in Table 8.4‑1.

Table 8.4‑1: MASA frame signalling, active frames

|  |  |  |  |
| --- | --- | --- | --- |
| IVAS format | configuration | number of bits | Value |
| MASA | - | 3 | 7 |

Detailed bit allocation principles at different bitrates of the MASA operation are provided in Table 8.4‑1A, Table 8.4‑1B, and Table 8.4‑1C.

Table 8.4‑1A: Bit allocation at 13.2, 16.4, 24.4, and 32 kbps

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Description | Ordering of bits | 13.2 kbps | 16.4 kbps | 24.4 kbps | 32 kbps |
| **total bits** | Forward ordering of bits | 264 | 324 | 488 | 640 |
| **IVAS common header (format)** | 3 | 3 | 3 | 3 |
| **Core-coder – SCE/CPE** | variable | variable | variable | variable |
| **No. of transport channels**  | Reverse ordering of bits | 1 | 1 | 1 | 1 |
| **Reserved** | 2 | 2 | 2 | 2 |
| **No. of spatial directions** | 1 | 1 | 1 | 1 |
| **Subframe mode** **(SF = 0 or 1)** | 1 | 1 | 1 | 1 |
| **Low bitrate mode** | **1 subframe (SF=1)** | 0 | 0 | 0 | 0 |
| **4 subframes****(SF=0)** | 1 | 1 | 1 | 1 |
| **MASA metadata** | **LR mode** | variable, max 45-(1-SF) | variable, max 45-(1-SF)  | variable, max 55-(1-SF)  | variable, max 65-(1-SF)  |
| **Normal mode** | variable, max 45-(1-SF) | variable, max 55-(1-SF) | variable, max 65-(1-SF) | variable, max 80-(1-SF) |

Table 8.4‑1B: Bit allocation at 48, 64, 80, and 96 kbps

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Description | Ordering of bits | 48 kbps | 64 kbps | 80 kbps | 96 kbps |
| **total bits** | Forward ordering of bits | 960 | 1280 | 1600 | 1920 |
| **IVAS common header (format)** | 3 | 3 | 3 | 3 |
| **Core-coder – SCE/CPE** | variable | variable | Variable | variable |
| **No. of transport channels**  | Reverse ordering of bits | 1 | 1 | 1 | 1 |
| **Reserved** | 2 | 2 | 2 | 2 |
| **No. of spatial directions** | 1 | 1 | 1 | 1 |
| **Subframe mode**  | 1 | 1 | 1 | 1 |
| **MASA metadata** | variable, max 135 | variable, max 175  | variable, max 215  | variable, max 251  |

Table 8.4‑1C: Bit allocation at 128, 160, 192, 256, 384, and 512 kbps

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Description | Ordering of bits | 128 kbps | 160 kbps | 192 kbps | 256 kbps | 384 kbps | 512 kbps |
| **total bits** | Forward ordering of bits | 2560 | 3200 | 3840 | 5210 | 7680 | 10240 |
| **IVAS common header (format)** | 3 | 3 | 3 | 3 | 3 | 3 |
| **Core-coder – SCE/CPE** | variable | variable | variable | variable | variable | variable |
| **No. of transport channels**  | Reverse ordering of bits | 1 | 1 | 1 | 1 | 1 | 1 |
| **Reserved** | 2 | 2 | 2 | 2 | 2 | 2 |
| **No. of spatial directions** | 1 | 1 | 1 | 1 | 1 | 1 |
| **Subframe mode**  | 1 | 1 | 1 | 1 | 1 | 1 |
| **MASA metadata** | variable, max 325 | variable, max 427 | variable, max 523  | variable, max 827 | variable | variable |

### 8.4.2 Bit allocation for MASA in SID frames

SID frame signalling for IVAS MASA operation (MASA) is summarized in Table 8.4‑2 and the corresponding bitstream description in Table 8.4-3.

Table 8.4‑2: MASA frame signalling, SID frames

|  |  |  |  |
| --- | --- | --- | --- |
| IVAS format | configuration | number of bits | Value |
| MASA | - | 2 | 0x3 |
| - | 2 | 0x7 |

Table 8.4‑3: Bit allocation for MASA in SID frames

|  |  |  |
| --- | --- | --- |
| Description | Ordering of bits | Number of bits |
| **total bits** | Forward ordering of bits | 104 |
| **SID format bits** | 3 |
| **Core-coder – SCE/CPE** | 48 |
| **MASA metadata** | Reverse ordering of bits | 53 |

CHANGE 43

## 8.8 Bit allocation for combined Object-based audio and MASA (OMASA)

Active frame signalling for IVAS OMASA operation (OMASA) is summarized in Table 8.8‑1. DTX operation is not supported with OMASA inputs.

Table 8.8‑1: OMASA frame signalling, active frames

|  |  |  |  |
| --- | --- | --- | --- |
| IVAS format | configuration | number of bits | Value |
| OMASA | Rend mode | 3 | 7 |
| other modes | 4 | 10 |

Detailed bit allocation principles for OMASA format in pre-rendering coding mode are presented in Tables 8.8-2 and 8.8-3 for different bitrates and different number of objects.

Table 8.8‑2: Bit allocation for pre-rendering coding mode – part 1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Description | Ordering of bits | 13.2 kbps1 and 2 obj | 16.4 kbps1 and 2 obj | 24.4 kbps 2 obj |
| **total bits** | Forward ordering of bits | 264 | 324 | 488 |
| **IVAS common header (format)** | 3 | 3 | 3 |
| **Core-coder – CPE** | variable | variable | variable |
| **Additional info on number of objects**  | Reverse ordering of bits | 1 | 1 | 1 |
| **Initial info on number of objects** | 2 | 2 | 2 |
| **No. of spatial directions** | 1 | 1 | 1 |
| **Subframe mode (SF = 0 or 1)** | 1 | 1 | 1 |
| **Low bitrate mode**  | **1 subframe (SF=1)** | 0 | 0 | 0 |
| **4 subframes****(SF=0)** | 1 | 1 | 1 |
| **MASA metadata** | variable, max 45 – (1-SF) | variable, max 45 – (1-SF)  | variable, max 55 – (1-SF)  |

Table 8.8‑3:  Bit allocation for pre-rendering coding mode – part 2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Description | Ordering of bits | 13.2 kbps3 and 4 obj | 16.4 kbps3 and 4 obj | 24.4 kbps3 and 4 obj |
| **total bits** | Forward ordering of bits | 264 | 324 | 488 |
| **IVAS common header (format)** | 3 | 3 | 3 |
| **Core-coder – CPE** | variable | variable | variable |
| **No. of spatial directions** | Reverse ordering of bits | 1 | 1 | 1 |
| **Info on number of objects** | 2 | 2 | 2 |
| **Subframe mode** **(SF = 0 or 1)** | 1 | 1 | 1 |
| **Low bitrate mode** | **1 subframe (SF=1)** | 0 | 0 | 0 |
| **4 subframes (SF=0)** | 1 | 1 | 1 |
| **MASA metadata** | variable, max 45-(1-SF) | variable, max 45-(1-SF)  | variable, max 55-(1-SF)  |

Detailed bit allocation principles for OMASA format in one object with MASA representation coding mode are presented in Table 8.8-4 for the bitrates and corresponding number of objects for which the coding mode is used.

Table 8.8‑4: Bit allocation for one object with MASA representation coding mode

|  |  |  |  |
| --- | --- | --- | --- |
| Description | Ordering of bits | 32 kbps3 and 4 obj | 48 kbps3 and 4 obj |
| **total bits** | Forward ordering of bits | 640 | 960 |
| **IVAS common header (format)** | 4 | 4 |
| **Separated object (SCE) with object metadata** | variable | Variable |
| **Core-coder – CPE** | variable | Variable |
| **Number of objects**  | Reverse ordering of bits | 2 | 2 |
| **Separated object importance flags** | 2 or 4 | 2 or 4 |
| **Reserved MASA bits** | 2 | 2 |
| **No. of spatial directions** | 1 | 1 |
| **Subframe mode** | 1 | 1 |
| **Low bitrate mode** | **1 subframe (LRSF)** | 0 | 0 |
| **4 subframes (LRSF)** | 1 | 0 |
| **MASA metadata** | variable, max 52-LRSF | variable, max 62-SF |

Detailed bit allocation principles for OMASA format in parametric one object coding mode are presented in Table 8.8-5 for the bitrates and corresponding number of objects for which the coding mode is used.

Table 8.8‑5: Bit allocation for parametric one object coding mode

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Description | Ordering of bits | 32 kbbps2 obj | 64 kbps 3 and 4 obj | 80 kbps3 and 4 obj | 96 kbps 4 obj |
| **total bits** | Forward ordering of bits | 640 | 1280 | 1600 | 1920 |
| **IVAS common header (format)** | 4 | 4 | 4 | 4 |
| **Separated object (SCE) with object metadata** | variable | variable | variable | variable |
| **Core-coder – CPE** | variable | variable | variable | variable |
| **Number of objects**  | Reverse ordering of bits | 2 | 2 | 2 | 2 |
| **Index of separated object** | 2 | 2 | 2 | 2 |
| **Separated object importance flags** | 2 | 2 | 2 | 2 |
| **Reserved MASA bits** | 2 | 2 | 2 | 2 |
| **No. of spatial directions** | 1 | 1 | 1 | 1 |
| **Subframe mode (SF = 0 or 1)** | 1 | 1 | 1 | 1 |
| **Low bitrate mode**  | **1 sub frame (SF=1)** | 0 | 0 | 0 | 0 |
| **4 sub frames****(SF=0)** | 1 | 0 | 0 | 0 |
| **MASA to total energy ratios****ISM energy ratios****ISM metadata****MASA metadata** | variable, max 65-(1-SF) | variable, max 130 | variable, max 170 | variable, max 210 |

Detailed bit allocation principles for OMASA format in discrete coding mode are presented in Table 8.8-6, 8.8-7, and 8.8-8 for the bitrates and corresponding number of objects for which the coding mode is used.

Table 8.8‑6: Bit allocation for discrete coding mode – part 1

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Description | Ordering of bits | 24 kbps\*1 | 32 kbps=1  | 48 kbps | 64 kbps \*=1,2 |
| **total bits** | Forward ordering of bits | 640 | 1280 | 1600 | 1920 |
| **IVAS common header (format)** | 4 | 4 | 4 | 4 |
| **ISM format data** | variable | variable | variable | variable |
| **Core-coder – CPE** | variable | variable | variable | variable |
| **Number of objects**  | Reverse ordering of bits | 2\* | 2\* | 2\* | 2\* |
| **Objects importance flags** | 2  | 2 | 2 | 2 |
| **Reserved MASA bits** | 2 | 2 | 2 | 2 |
| **No. of spatial directions** | 1 | 1 | 1 | 1 |
| **Subframe mode** **(SF = 0 or 1)** | 1 | 1 | 1 | 1 |
| **Low bitrate mode** | **1 subframe** **(SF = 1)** | 0 | 0 | 0 | 0 |
| **4 subframes (SF = 0)** | 1 | 1 | 1 | 0 |
| **MASA metadata** | variable, max 42-(1-SF) | variable, max 52-(1-SF) | variablemax 62-(1-SF) | variable, max 62-(1-SF) |

Table 8.8‑7: Bit allocation for discrete coding mode – part 2

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Description | Ordering of bits | 80 kbps = 1, 2  | 96 kbps = 1, 2, 3 | 128 kbps = 1, 2 | 128 kbps = 3, 4 |
| **total bits** | Forward ordering of bits | 640 | 1280 | 1600 | 1920 |
| **IVAS common header (format)** | 4 | 4 | 4 | 4 |
| **Separated object (SCE) with object metadata** | variable | variable | variable | variable |
| **Core-coder – CPE** | variable | variable | variable | variable |
| **Number of objects**  | Reverse ordering of bits | 2 | 2 | 2 | 2 |
| **Objects importance flags** | (2 or 4) \* | (2 or 4)\* | (2 or 4)\* | (2 or 4)\* |
| **OMASA bitrate flag** | 0 | 0 | 0 | 1 |
| **Reserved MASA bits** | 2 | 2 | 2 | 2 |
| **No. of spatial directions** | 1 | 1 | 1 | 1 |
| **Subframe mode (SF)** | 1 | 1 | 1 | 1 |
| **Low bitrate mode** | **1 subframe** | 0 | 0 | 0 | 0 |
| **4 subframes** | 0 or 1 | 0 or 1 | 0 | 0 or 1 |
| **MASA metadata** | variable  | variable  | variable  | variable  |

Table 8.8‑8: Bit allocation for discrete coding mode – part 3

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Description | Ordering of bits | 160 kbps = 1..4  | 192 kbps = 1..4 | 256 kbps = 1..4 | 384 kbps = 1..4 | 512 kbps = 1.. 4 |
| **total bits** | Forward ordering of bits | 3200 | 3840 | 5210 | 7680 | 10240 |
| **IVAS common header (format)** | 4 | 4 | 4 | 4 | 4 |
| **Separated object (SCE) with object metadata** | variable | variable | variable | variable | variable |
| **Core-coder – CPE** | variable | variable | variable | variable | variable |
| **Number of objects**  | Reverse ordering of bits | 2 | 2 | 2 | 2 | 2 |
| **Objects importance flags** | (2 or 4) \* | (2 or 4) \* | (2 or 4)\* | (2 or 4)\* | (2 or 4)\* |
| **Reserved MASA bits** | 2 | 2 | 2 | 2 | 2 |
| **No. of spatial directions** | 1 | 1 | 1 | 1 | 1 |
| **Subframe mode (SF)** | 1 | 1 | 1 | 1 | 1 |
| **MASA metadata** | variable  | variable  | variable  | variable  | variable  |

END OF CHANGES