INTERNATIONAL STANDARD

ISO/IEC 23003-2

First edition
2010-10-01

Information technology — MPEG audio technologies —

Part 2: Spatial Audio Object Coding (SAOC)

Technologies de l'information — Technologies audio MPEG —

Partie 2: Codage d'objet audio spatial (SAOC)

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

Reference number
ISO/IEC 23003-2:2010(E)

© ISO/IEC 2010

PDF disclaimer

This PDF file may contain embedded typefaces. In accordance with Adobe's licensing policy, this file may be printed or viewed but shall not be edited unless the typefaces which are embedded are licensed to and installed on the computer performing the editing. In downloading this file, parties accept therein the responsibility of not infringing Adobe's licensing policy. The ISO Central Secretariat accepts no liability in this area.

Adobe is a trademark of Adobe Systems Incorporated.

Details of the software products used to create this PDF file can be found in the General Info relative to the file; the PDF-creation parameters were optimized for printing. Every care has been taken to ensure that the file is suitable for use by ISO member bodies. In the unlikely event that a problem relating to it is found, please inform the Central Secretariat at the address given below.

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

COPYRIGHT PROTECTED DOCUMENT

© ISO/IEC 2010

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Contents

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

ISO/IEC 23003-2 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*.

ISO/IEC 23003 consists of the following parts, under the general title *Information technology — MPEG audio technologies*:

— *Part 1: MPEG Surround*

— *Part 2: Spatial Audio Object Coding (SAOC)*

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

Introduction

The International Organization for Standardization (ISO) and International Electrotechnical Commission (IEC) draw attention to the fact that it is claimed that compliance with this document may involve the use of patents.

ISO and IEC take no position concerning the evidence, validity and scope of these patent rights.

The holders of these patent rights have assured ISO and IEC that they are willing to negotiate licences under reasonable and non-discriminatory terms and conditions with applicants throughout the world. In this respect, the statements of the holders of these patent rights are registered with ISO and IEC. Information may be obtained from the companies listed in Annex G.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights other than those identified in Annex G. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

Information technology — MPEG audio technologies —

Part 2: Spatial Audio Object Coding (SAOC)

1 Scope

This part of ISO/IEC 23003 specifies the reference model of the Spatial Audio Object Coding (SAOC) technology that is capable of recreating, modifying and rendering a number of audio objects based on a smaller number of transmitted channels and additional parametric data. In the preferred modes of operating the SAOC system, the transmitted signal can be either mono or stereo. The audio objects can be represented by a mono and stereo signal or have the MPEG Surround (MPS) Multi-channel Background Object (MBO) format. The additional parametric data exhibits a significantly lower data rate than required for transmitting all objects individually, making the coding very efficient. At the same time this ensures compatibility of the transmitted signal with legacy devices.

When a multi-channel rendering setup (e.g. a 5.1 loudspeaker setup) is required, the SAOC system acts as a transcoder, converting the additional parametric data to MPS parameters, and interfaces to the MPS decoder that acts as rendering device. For certain rendering setups (e.g. a binaural or plain stereo setup), the SAOC system behaves as a decoder, using its own rendering engine. Another key feature is that the SAOC parametric data from different streams can be merged at parameter level to allow for the combination of SAOC streams, similar to the functionality of a Multi-point Control Unit (MCU).

2 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 13818-7:2006, *Information technology — Generic coding of moving pictures and associated audio information — Part 7: Advanced Audio Coding (AAC)*

ISO/IEC 14496-3:2009, *Information technology — Coding of audio-visual objects — Part 3: Audio*

ISO/IEC 23003-1:2007, *Information technology — MPEG audio technologies — Part 1: MPEG Surround*

3 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

3.1
audio object
input audio signal consisting of one, two or multiple channels, including Multi-channel Background Object (MBO)

3.2
frame
time segment to which SAOC processing is applied according to the data conveyed in the corresponding SAOCFrame() syntax element

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

3.3
hybrid filterbank
hybrid filter bank structure, consisting of a quadrature mirror filter (QMF) bank and oddly modulated Nyquist filter banks, used to transform time domain signals into hybrid subband samples

3.4
hybrid filtering
filtering step on a quadrature mirror filter (QMF) subband signal resulting in multiple hybrid subbands

NOTE The resulting hybrid subbands can be non-consecutive in frequency.

3.5
hybrid subband
subband obtained after hybrid filtering of a quadrature mirror filter (QMF) subband

NOTE The hybrid subband can have the same time/frequency resolution as a QMF subband.

3.6
input channel
input audio channel corresponding to the channels of an audio object

3.7
output channel
audio channel corresponding to a specific speaker

NOTE Channel abbreviations and loudspeaker positions are given in Table 1.

3.8
parameter band
one or more hybrid subbands applicable to one parameter

3.9
parameter time slot
specific time slot for which the parameter is defined

3.10
parameter set
parameters associated with a specific parameter time slot

3.11
parameter subset
parameters associated with a specific parameter time slot and a specific One-To-Two (OTT) box or Two-To-Three (TTT) box

3.12
processing band
one or more hybrid subbands defining the finest frequency resolution that could be controlled by the parameters

3.13
QMF bank
bank of complex exponentially modulated filters

3.14
QMF subband
subband obtained after QMF filtering of a time-domain signal, without any additional hybrid filtering stage

3.15
time segment
group of consecutive time slots

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

3.16
time slot
finest resolution in time for spatial audio coding (SAC) time borders

NOTE One time slot equals one subsample in the hybrid quadrature mirror filter (QMF) domain.

4 Symbols, notation and abbreviated terms

4.1 Notation

The description of the SAOC system uses the following notation:

- Vectors are indicated by bold lower-case names, e.g. **vector**.
- Matrices (and vectors of vectors) are indicated by bold upper-case single letter names, e.g. **M**.
- Variables are indicated by italic, e.g. *variable*.
- Functions are indicated as *func(x)*.

For equations (and flowcharts), normal mathematical (and pseudo-code) interpretation is assumed with no rounding or truncation unless explicitly stated.

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

4.2 Operations

4.2.1 Scalar operations

X^*	is the complex conjugate of X.
$y = \log_{10}(x)$	is the base-10 logarithm of x.
$y = \min(\ldots,\ldots)$	is the minimum value in the argument list.
$y = \max(\ldots,\ldots)$	is the maximum value in the argument list.

4.2.2 Vector and matrix operations

$\mathbf{m} = diag(\mathbf{M})$	is diagonal of matrix $\mathbf{M}$.
$\mathbf{y} = sort(\mathbf{x})$	is equal to the sorted vector $\mathbf{x}$, where the elements of $\mathbf{x}$ are sorted in ascending order.
$y = trace(\mathbf{M})$	is sum of all diagonal elements of matrix $\mathbf{M}$.

4.3 Constants

ε	is an additive constant to avoid division by zero, e.g. $\varepsilon = 10^{-9}$.

4.4 Variables

$a_{i,y}^{l,m}$	is the virtual speaker transfer function, defined for binaural output channel i, audio object y and all parameter time slots l and processing bands m.
$\mathbf{D}$	is the downmix matrix.
$\mathbf{D}_{\mathrm{CLD}}$	is the three dimensional matrix holding the dequantized, and mapped CLD data for every OTT box, every parameter set, and M_{proc} bands.

$\mathbf{D}_{\mathrm{ICC}}$	is the three dimensional matrix holding the dequantized, and mapped ICC data for every OTT or TTT box, every parameter set, and M_{proc} bands.
$\mathbf{D}_{\mathrm{CPC\_1}}$, $\mathbf{D}_{\mathrm{CPC\_2}}$	are the three dimensional matrices holding the dequantized, and mapped first and second CPC data for every TTT box, every parameter set, and M_{proc} bands.
$\mathbf{D}_{\mathrm{CLD\_1}}$, $\mathbf{D}_{\mathrm{CLD\_2}}$	are the three dimensional matrices holding the dequantized, and mapped first and second CLD data for every TTT box, every parameter set, and M_{proc} bands.
$\mathbf{D}_{\mathrm{DCLD}}$	is the two dimensional matrix holding the dequantized, and mapped DCLD data for every input channel, and every parameter set.
$\mathbf{D}_{\mathrm{DMG}}$	is the two dimensional matrix holding the dequantized, and mapped DMG data for every input channel, and every parameter set.
$\mathbf{D}_{\mathrm{IOC}}$	is the four dimensional matrix holding the dequantized, and mapped IOC data for every input channel pair, every parameter set, and M_{proc} bands.
$\mathbf{D}_{\mathrm{NRG}}$	is the two dimensional matrix holding the dequantized, and mapped NRG data for the highest energy within every parameter set, and M_{proc} bands.
$\mathbf{D}_{\mathrm{OLD}}$	is the three dimensional matrix holding the dequantized, and mapped OLD data for every input channel, every parameter set, and M_{proc} bands.
$\mathbf{D}_{\mathrm{PDG}}$	is the three dimensional matrix holding the dequantized, and mapped PDG data for every downmix channel, every parameter set, and M_{proc} bands.
$H_{i,\{L,R\}}^{m}$	is the HRTF parameter which represents the average level with respect to the left and right ear {*L*, *R*} for the HRTF database index i and all processing bands m.
$\mathbf{idx}XXX(\ldots,\ldots)$	is a three dimensional matrix holding the Huffman and delta decoded indices. *XXX* can be any of OLD, IOC, NRG, DCLD, DMG, PDG.
K	is the number of hybrid subbands.
L	is the number of parameter sets.
M	is the number of downmix channels.
M_{proc}	is the number of processing bands.
M_{QMF}	is the number of QMF subbands depending on sampling frequency.
$\mathbf{M}^{l,m}$	is the OTN/TTN upmix matrix for the prediction mode of operation
$\mathbf{M}_{Energy}^{l,m}$	is the OTN/TTN upmix matrix for the energy mode of operation
$\mathbf{M}_{1}^{n,k}$, $\mathbf{M}_{2}^{n,k}$	are the time and frequency variant pre-matrices, defined for all time slots n and all hybrid subbands k.
$\mathbf{M}_{\mathrm{ren}}^{l,m}$	is the time and frequency variant rendering matrix, defined for all parameter time slots l and all processing bands m.
N	is the number of SAOC input channels of audio objects.

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

N_{EAO}	is the number of EAO channels.
N_{MPS}	is the number of MPS output channels.
N_{HRTF}	is the number of different HRTFs in the HRTF database.
P	frame length.
$\mathbf{W}_{\mathrm{ADG}}^{l,m}$	is the time and frequency variant matrix including ADGs, defined for all parameter time slots l and all processing bands m.
$\mathbf{W}_{h}^{l,m}$	is the time and frequency variant sub-rendering matrix, defined for OTT box h (of the MPS "5-1-5" tree-structure), all parameter time slots l and all processing bands m.
$\mathbf{W}_{\mathrm{PDG}}^{l,m}$	is the time and frequency variant matrix including PDGs, defined for all parameter time slots l and all processing bands m.
$\mathbf{s}^{n,k}$	is a vector with the hybrid subband (encoder) input channels, defined for all time slots n and all hybrid subbands k.
$\mathbf{x}^{n,k}$	is a vector with the hybrid subband (transcoder/decoder) input signals (downmix and residuals), defined for all time slots n and all hybrid subbands k.
$\mathbf{y}^{n,k}$	is a vector with the (transcoder/decoder) output hybrid subband signals, which are fed into the hybrid synthesis filter banks, defined for all time slots n and all hybrid subbands k.
ϕ_{i}^{m}	is the HRTFs parametric representation of the average phase difference, defined for the HRTF database index i and all processing bands m.

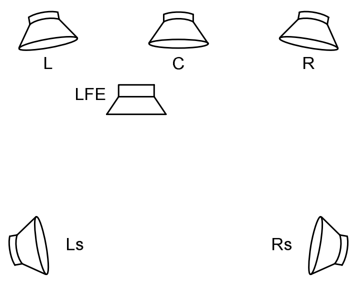
4.5 Abbreviated terms

ADG	Arbitrary Downmix Gain
CLD	Channel Level Difference, describes the energy difference between two channels
CPC	Channel Prediction Coefficient, used for recreating three or more channels from two channels
DCLD	Downmix Channel Level Difference describes the gain differences of objects contributing to the left and right downmix channel in case of a stereo downmix
DCU	Distortion Control Unit
DMG	DownMix Gain, gains applied to each object before downmixing
EAO	Enhanced Audio Object
HRTF	Head Related Transfer Function
ICC	Inter Channel Correlation, describes the correlation between two channels
IOC	Inter Object Correlation, describes the correlation between two channels of audio objects
LD	Low Delay

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

MBO	Multi-channel Background Object
MCU	Multi-point Control Unit
MPS	MPEG Surround
N/A	Not Applicable
NRG	absolute object eNeRGy, specifies the absolute energy of the object with the highest energy for the corresponding frequency band
OLD	Object Level Difference, describes intensity differences between one object and the object with the highest energy for the corresponding frequency band
OTN	conceptual “One-To-N” unit that takes one channel as input and produces N channels as output
OTT	conceptual “One-To-Two” unit that takes one channel as input and produces two channels as output
PDG	Post(processing) Downmix Gains, describes intensity differences between the encoder-generated downmix and the post(processed) downmix for the corresponding frequency band
QMF	Quadrature Mirror Filter
SAC	Spatial Audio Coding
SAOC	Spatial Audio Object Coding
TTN	conceptual "Two-To-N" unit that takes two channels as input and produces N channels as output
TTT	conceptual "Two-To-Three" unit that takes two channels as input and produces three channels as output

Table 1 – Channel abbreviations and loudspeaker positions

Channel abbreviation	Loudspeaker position	Figure
L	Left Front	
R	Right Front	
C	Center Front	
LFE	Low Frequency Enhancement	
Ls	Left Surround	
Rs	Right Surround	

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

5 SAOC overview

5.1 Introduction

Spatial Audio Object Coding (SAOC) is a parametric multiple object coding technique. It is designed to transmit a number of audio objects in an audio signal that comprises M channels. Together with this backwards compatible downmix signal, object parameters are transmitted that allow for recreation and manipulation of the original object signals. An SAOC encoder produces a downmix of the object signals at its input and extracts these object parameters. The number of objects that can be handled is in priniciple not limited.

The object parameters are quantized and coded efficiently into an SAOC bitstream.

The downmix signal can be compressed and transmitted without the need to update existing coders and infrastructures. The object parameters, or SAOC side information, are transmitted in a low bitrate side channel, e.g. the ancillary data portion of the downmix bitstream.

On the decoder side, the input objects are reconstructed and at the same time rendered to a certain number of playback channels. The rendering information containing reproduction level and panning position for each object is user supplied or can be extracted from the SAOC bitsream (e.g. preset information). The rendering information can be time variant. Output scenarios can range from mono to multi-channel (e.g. 5.1) and are independent from both, the number of input objects and the number of downmix channels. Binaural rendering of objects is possible including azimuth and elevation of virtual object positions. An optional effects interface allows for advanced manipulation of object signals, besides level and panning modification.

The objects themselves can be mono signals, stereophonic signals, as well as multi-channel signals (e.g. 5.1 channels). Typical downmix configurations are mono and stereo.

5.2 Basic structure of the SAOC transcoder/decoder

The SAOC transcoder/decoder module described below may act either as a stand-alone decoder or as a transcoder from an SAOC to an MPS bitstream, depending on the intended output channel configuration. The following table illustrates the differences between the two modes of operation:

Table 2 – Operation modes of the SAOC

Output signal configuration	# of output channels	SAOC module mode	SAOC module output	MPS decoder required
mono/stereo/binaural	1 or 2	Decoder	PCM output	No
multi-channel configuration	> 2	Transcoder	MPS bitstream, downmix signal	Yes

Figure 1 shows the basic structure of the SAOC transcoder/decoder architecture. The residual processor extracts the EAOs from the incoming downmix using the residual information contained in the SAOC bitstream. The downmix pre-processor processes the regular audio objects. The EAOs and processed regular audio objects are combined to the output signal for the SAOC decoder mode or to the MPS downmix signal for the SAOC transcoder mode. The detailed descriptions of these processing blocks are given in the corresponding subclauses, namely, 7.6 and 7.7 describe the SAOC transcoder/decoder functionality and 7.8 explains handling of extended audio objects and residual processing.

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

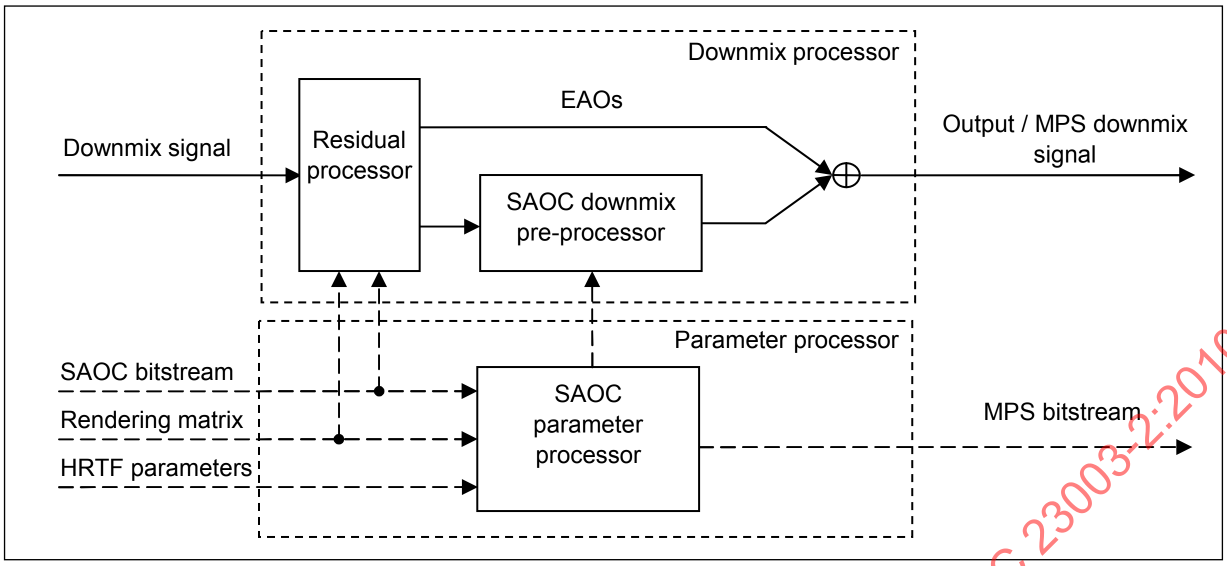


Figure 1 – Overall structure of the SAOC transcoder/decoder architecture

Figure 2 (left) shows a block diagram of an SAOC transcoder unit. It consists of an SAOC parameter processor and a downmix processor module. The SAOC parameter processor decodes the SAOC bitstream and has furthermore a user interface from which it receives additional input in form of generally time variant rendering information. It provides steering information for the downmix processor. The SAOC transcoder outputs an MPS bitstream and downmix signal, as an input to the MPS decoder. In case of a mono downmix, the downmix pre-processor leaves the downmix signal unchanged. However, in case of a stereo downmix, it is functional to pre-process the downmix signal to allow more flexible object panning than is supported by the MPS rendering engine alone. In case of a mono/stereo/binaural output configuration the SAOC system works in decoder mode and MPS decoding is omitted, see Figure 2 (right). Here the downmix processing module directly provides the output signal.

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

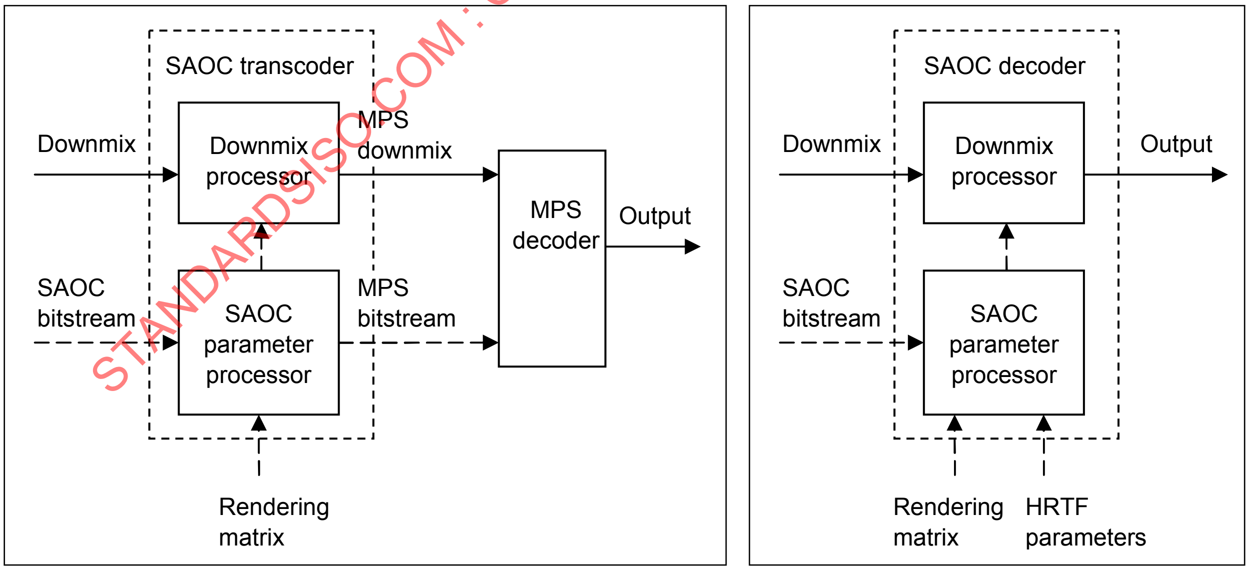


Figure 2 – Block diagrams of the SAOC transcoder (left) and decoder (right) processing modes

© ISO/IEC 2010 – All rights reserved

5.3 Tools and functionality

5.3.1 General SAOC tools

5.3.1.1 Introduction

The SAOC system incorporates a number of tools that allow for flexible complexity and/or quality trade-off, as well as a diverse set of functionality. In the following subclauses some key-features of SAOC are briefly outlined.

5.3.1.2 Binaural decoding

The SAOC system can be operated in a binaural mode. This enables a multi-channel impression over headphones by means of Head Related Transfer Function (HRTF) filtering.

5.3.1.3 Efficient multipoint control unit support

In order to use the SAOC concept for teleconferencing applications a Multipoint Control Unit (MCU) functionality of combining the signals of several communication partners without decoding/re-encoding the corresponding audio objects is provided. The MCU combines the input SAOC side information streams into one common SAOC bitstream in a way that the parameters representing all audio objects from the input bitstreams are included in the resulting output bitstream. These calculations are performed in the parameter domain without the need to analyze the downmix signals and, therefore, introduce no additional delay in the signal processing chain.

5.3.1.4 External downmix

The SAOC system is capable of handling not only encoder-generated downmixes but also post(processed) downmixes supplied to the encoder in addition to the input audio object signals. In this case, Post Downmix Gains (PDGs) are calculated in the encoder and conveyed as a part of the SAOC bitstream. The difference of the downmix signals is compensated for at the SAOC decoder side.

5.3.1.5 Multichannel background object

The audio input to a SAOC encoder can contain a so-called Multi-channel Background Object (MBO). Generally, the MBO can be considered as a complex sound scene involving a large and often unknown number of sound sources, for which no controllable rendering functionality is required. The MBO is represented by a downmix of the MPS encoded complex sound scene and corresponding MPS parameters.

5.3.1.6 Enhanced audio object processing

A special "Karaoke-type" application scenario requires a total suppression of specific objects, typically the lead vocals, while keeping the perceptual quality of the background sound scene unharmed. High sound quality is assured by the incorporation of residual coding enabling a better separation of the background object and foreground objects. The current EAO processing mode supports reproduction of both EAO and regular objects exclusively and arbitrary mixtures of these object groups.

5.3.1.7 Distortion control unit

The distortion control unit is incorporated into the SAOC system in order to provide a flexible control for users and audio content providers over the SAOC rendering functionality and audio output quality.

5.3.1.8 Predefined rendering information

The SAOC system is capable of starting playback with some initial predefined settings which can be stored and/or transmitted in SAOC bitstream. These settings can be dynamically updated. The SAOC system allows instantaneous switching between them if more than one set of predefined settings is available.

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

5.3.1.9 Effects interface

The SAOC effects interface operates on the downmix and therefore is part of the downmix processor of the SAOC transcoder or decoder. The effects interface allows objects or linear combinations of objects to be extracted from the downmix for effects processing. Two types of effects processing interfaces are supported. The first type, referred to as the insert effects interface, allows effects processing to individual objects in the downmix. The second type, referred to as the send effects interface, allows effect processing on individual objects or linear combinations thereof.

5.3.2 High Quality, Low Power and Low Delay

The SAOC decoder can be implemented in a High Quality (HQ) version, a Low Power (LP) version, and a Low Delay (LD) version. The main difference are outlined by the following Table 3 and given in detail in 7.13 and 7.14.

Table 3 – Outline of difference between the HQ, LP and LD SAOC system

Tool or functionality	HQ system	LP system	LD system
Filterbank	Complex valued QMF	Partially complex valued QMF	LD QMF, no Nyquist FB
Aliasing reduction	Not Applicable	Tool operates on the real-valued part of the frequency range	Not Applicable
Residual coding	Supported over the entire frequency range	Only supported over the complex valued part of the frequency range	Not supported
Decorrelators		No Decorrelator for stereo downmix	

5.4 Delay and synchronization

5.4.1 Overview

The SAOC decoder introduces a delay when processing the time domain signal coming from a downmix decoder. Depending on whether the SAOC module is working as a decoder or as a transcoder for a Multichannel (MC) renderer, i.e. MPS decoder, two different cases are to be taken into account.

For all the different cases described in this subclause, the transmission of the SAOC side information with respect to the transmission of the coded downmix signal is done in such a manner that there is no need to further delay the downmix signal before the SAOC processing. This means that synchronization of the spatial data and downmix is achieved at the SAOC decoder/transcoder, following the temporal relationships described in Clause 8.

5.4.2 High Quality and Low Power processing

5.4.2.1 SAOC Decoding mode

The SAOC decoder (mono, stereo or binaural up mix modes) introduces a total delay of 1281 time domain samples for the High Quality (HQ) mode and 1601 samples for the Low Power (LP) mode. As shown in Figure 3 the analysis filterbank as outlined in 7.3 introduces a delay of 704 samples, while the synthesis filterbank introduces 257 for HQ and 577 for LP. The analysis filterbank processing delay consists of the QMF and hybrid processing delays, 320 and 384 time domain samples respectively. If no real to complex conversion is performed, this delay shall be compensated by a delay. This leads to 320 time samples which are added on top the analysis processing delay for both HQ and LP decoders. The synthesis filterbank introduces 257 samples delay which is introduced by the QMF synthesis filtering. The hybrid synthesis does not introduce further delay. For the LP decoder the complex to real conversion adds 320 time domain samples to the synthesis processing delay.

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

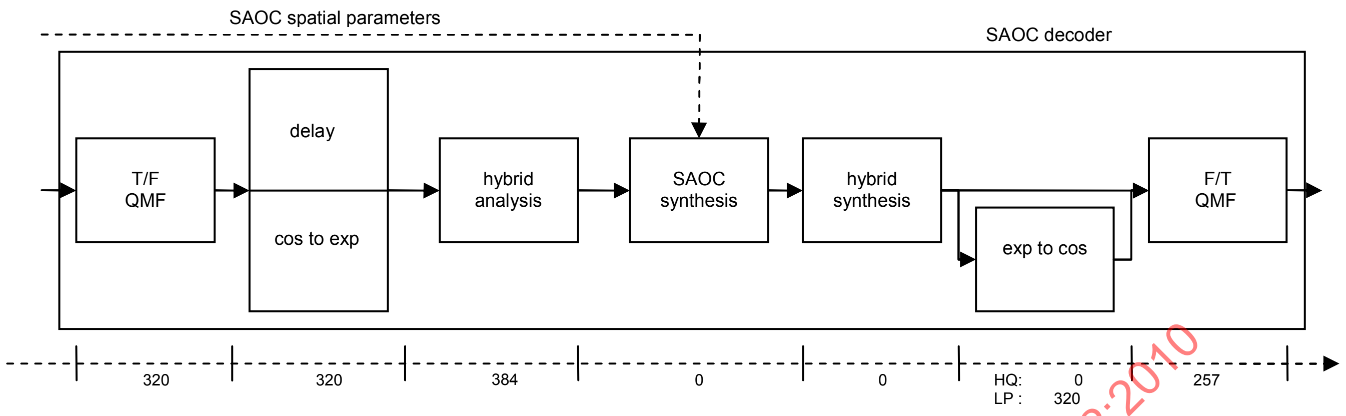


Figure 3 – Delay for the different parts of the SAOC decoder

5.4.2.2 SAOC transcoding mode

The MPS renderer introduces a delay when processing the downmix from the SAOC transcoder. Two cases may be distinguished, mono and stereo downmix.

In case of a mono downmix the signal from the downmix decoder is passed directly to the MPS decoder as no further processing is applied to the downmix, depicted in Figure **4**. The MC SAOC processing delay shall be equal to the one given by the SAOC decoder, as the SAOC to MPS parameter processing does not introduce any additional delay.

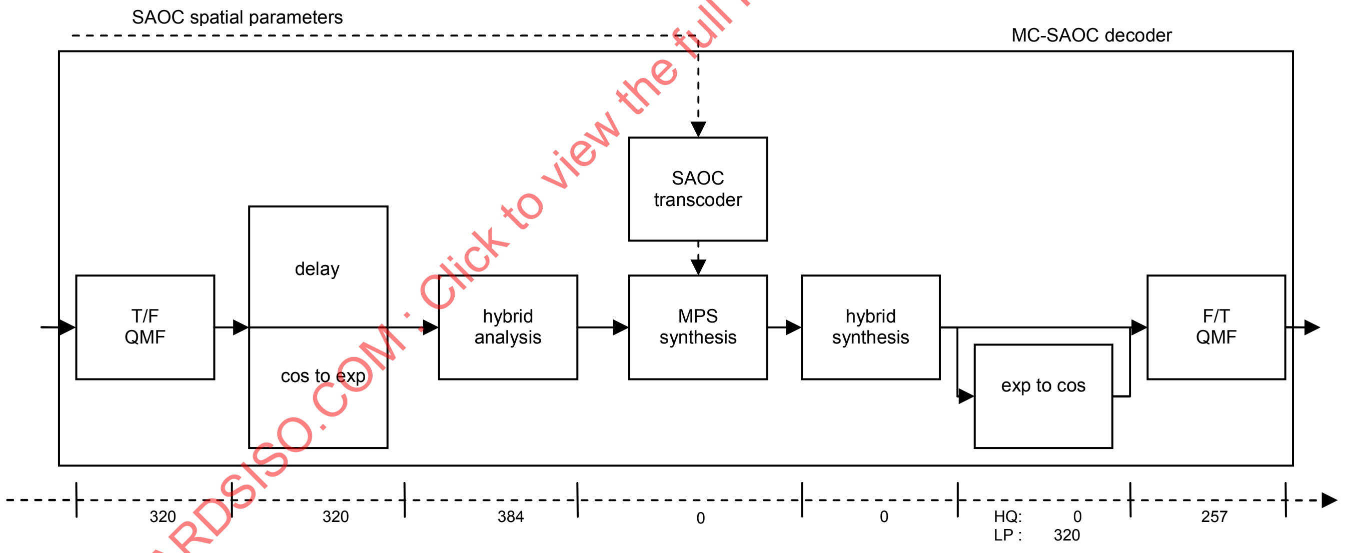


Figure 4 – Delay for the different parts of the MC-SAOC decoder with mono downmix signal

In case of a stereo downmix the SAOC transcoder processes the core coder signal to adapt it to the subsequent MPS decoding, as shown in Figure **5**. The delay of the MPS renderer shall be added on top of the SAOC processing delay. Both, the SAOC transcoder and the MPS decoder introduce a delay of 1281 samples for the HQ mode or of 1601 samples for the LP mode. Their analysis/synthesis delay is distributed as described above for the SAOC decoder.

If both modules are not integrated, i.e. they interface via the time domain, the total processing delay shall be the sum of the delays introduced by each module plus buffering needed for synchronization of the MPS parameters to the downmix.

The size of buffer B (spatial parameters buffer) is a multiple of the frame length. The downmix buffer A is needed to synchronize the delayed bitstream and processed downmix. To achieve synchronization the following equation must be met, both buffer sizes are given in time domain samples:

B = N*Frame length so that B ≥ 1281 (HQ) or B ≥ 1601 (LP); N=0,1,2,3,….

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

A(HQ) = B – 1281 or A(LP) = B – 1601

Given an interface via the hybrid QMF domain, the overall processing delay of the SAOC MC rendering is equal to the delay of the SAOC decoding mode.

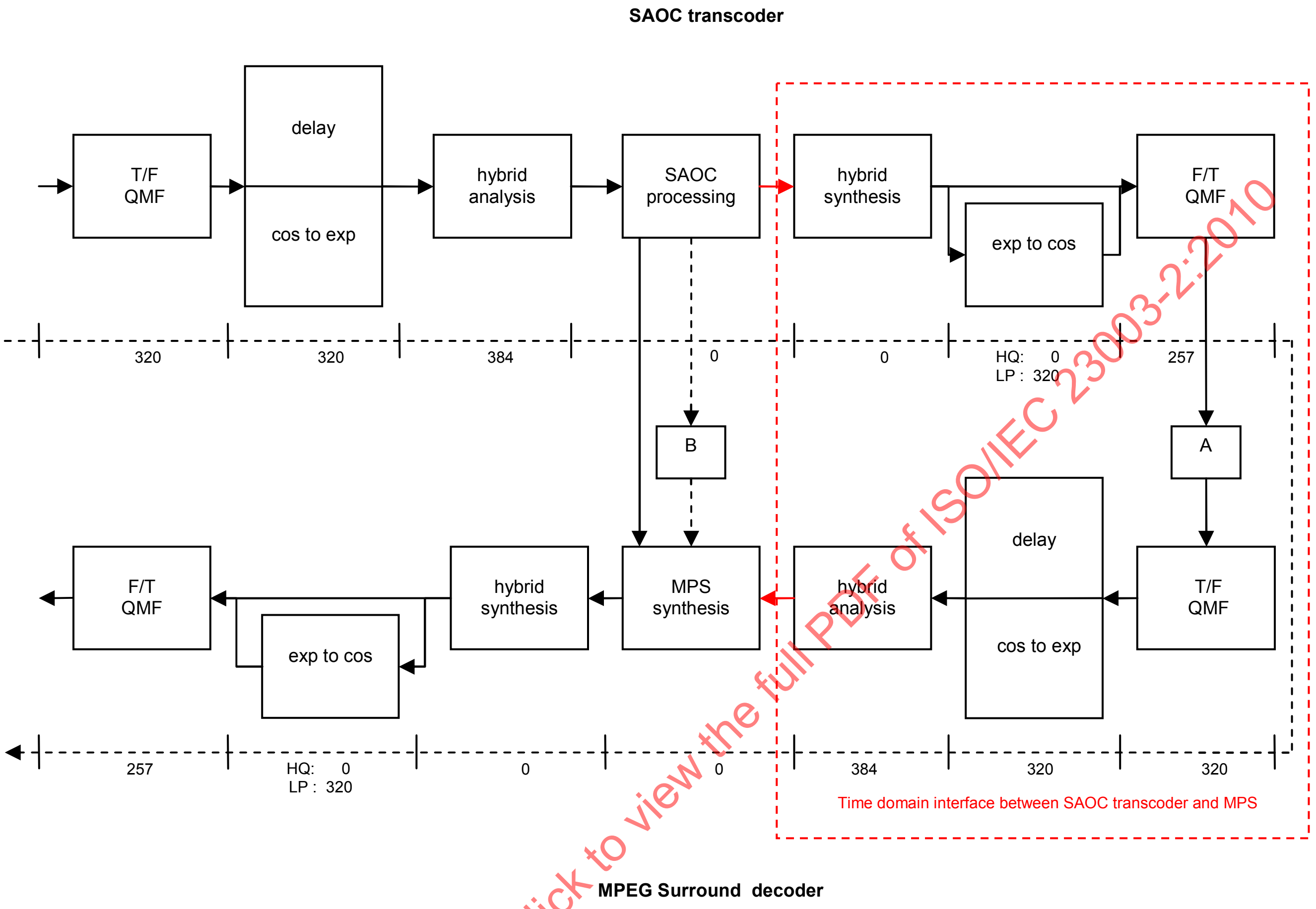


Figure 5 – Delay for the different parts of the MC-SAOC decoder with stereo downmix signal

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

5.4.2.3 Connection to an arbitrary core coder

If the (MC-)SAOC decoder is connected with an arbitrary downmix coder (including HE-AAC) via the time domain, as shown in Figure **6**, the additional delay introduced by the (MC-)SAOC decoder processing is described in the previous subclauses.

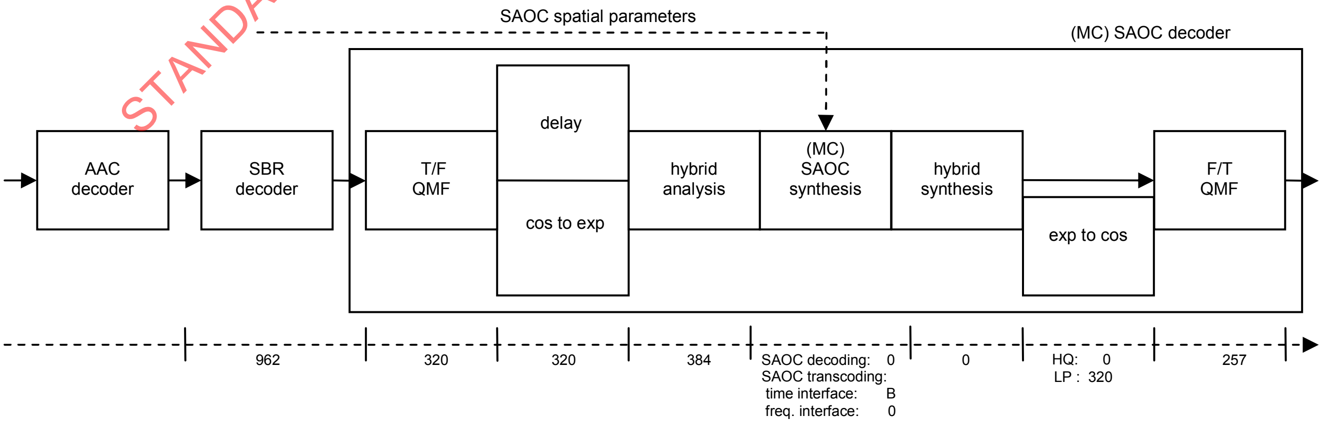


Figure 6 – Delay when connecting (MC-)SAOC in the time-domain for an arbitrary core codec (including HE-AAC)

© ISO/IEC 2010 – All rights reserved

If the (MC-)SAOC decoder is directly connected with a High Efficiency AAC decoder via the QMF domain, as shown in Figure 7, the delay shall be reduced as outlined in 4.5 of ISO/IEC 23003-1:2007 for a MPS decoder. The only additional delay is introduced by the real to complex converter for a LP decoder or the delay compensation for a HQ decoder. No hybrid analysis delay is introduced due to the fact that the look-ahead of 384 time domain samples is already available on the SBR tool of HE-AAC, as described in 8.A.3 of ISO/IEC 14496-3:2009.

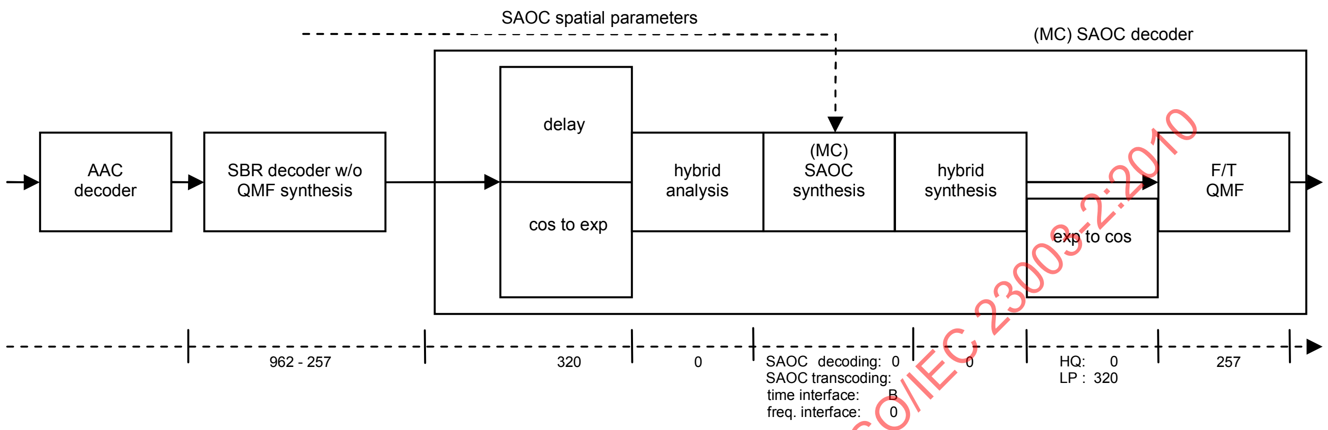


Figure 7 – Delay when connecting (MC-)SAOC with HE-AAC in the QMF domain

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

5.4.3 LD processing

5.4.3.1 SAOC Decoding

As outlined in 7.13 for the LD SAOC processing the Hybrid QMF filterbank is substituted by a LD QMF, in addition no LP mode is allowed and hereafter no complex conversion is to be performed. As a result, the analysis and synthesis processing delays shall only depend on the LD QMF filtering, i.e. 160 time domain samples for analysis and 96 for synthesis. This is shown in Figure 8.

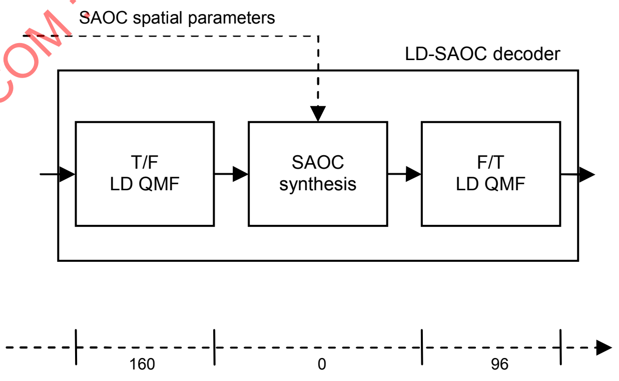


Figure 8 – Delay for the different parts of the LD SAOC decoder

5.4.3.2 SAOC Transcoding

Equally to the non LD MC decoders, depending on the number of downmix channels (mono/stereo) the processing for the LD-SAOC MC rendering scenarios is performed differently. Due to the delay constraints for a LD-SAOC system no time domain interface is allowed for the stereo processing case. Hereafter, both mono and stereo downmix cases may be equally described with respect to the systems delay, matching the SAOC decoder delay distribution, as depicted in Figure 9 and Figure 10.

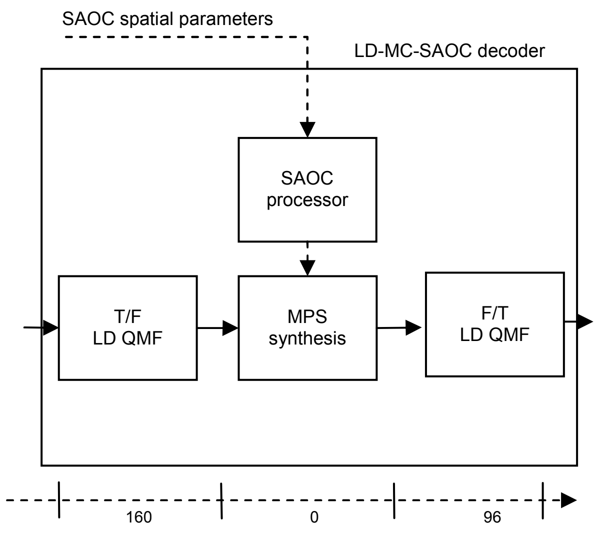


Figure 9 – Delay for the different parts of the LD-MC-SAOC decoder with mono downmix signal

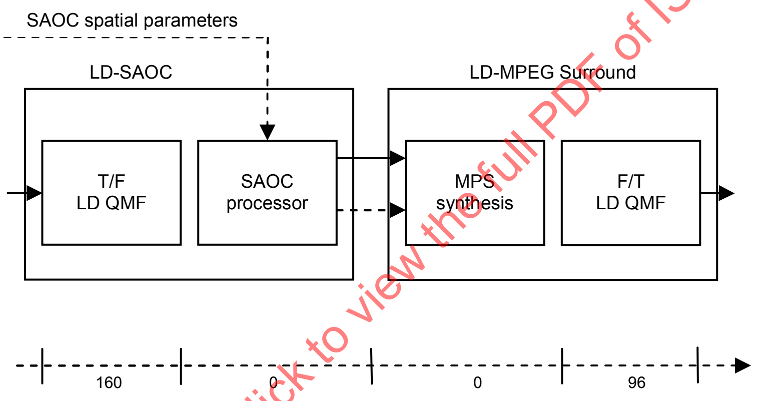


Figure 10 – Delay for the different parts of the LD-MC-SAOC decoder with stereo downmix signal

5.4.3.3 LD-(MC-) SAOC connection to LD core codec

If the LD-(MC-)SAOC decoder is connected with a LD downmix coder (e.g. AAC-LD or AAC-ELD, with/-out SBR) via the time domain, as shown in Figure 11, the additional delay introduced by the LD-(MC-)SAOC decoder processing is described above.

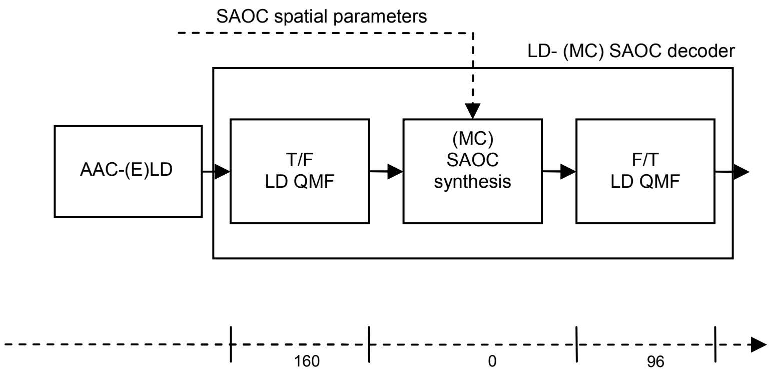


Figure 11 – Delay when connecting LD-(MC-)SAOC in the time-domain for a LD core codec (e.g. AAC-LD or AAC-ELD)

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

If the LD-(MC-) SAOC decoder is connected with a LD core coder via the frequency domain (e.g. AAC-ELD with SBR), as shown in Figure 12, the additional delay introduced by the LD-(MC-) SAOC decoder processing shall be reduced with respect to the description above, as no analysis is performed in the LD-(MC-) SAOC decoder.

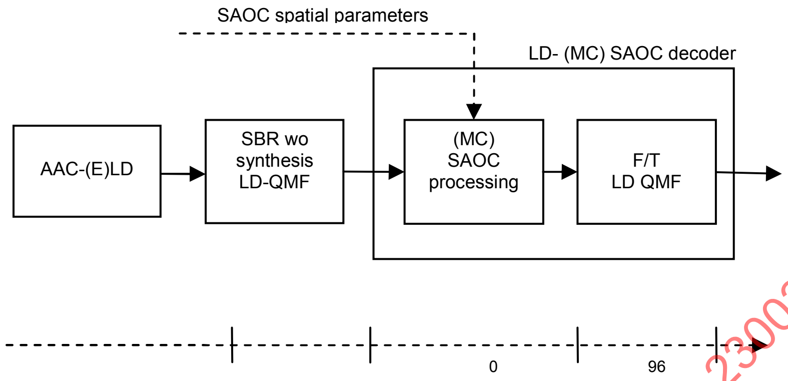


Figure 12 – Delay when connecting LD-(MC-) SAOC in the frequency-domain for a LD core codec (e.g. AAC-ELD with SBR)

5.5 SAOC Profiles and Levels

5.5.1 Introduction

This Subclause defines profiles and their levels for SAOC.

Complexity units are defined to give an approximation of the decoder complexity in terms of processing power and RAM usage required for the SAOC decoding/transcoding process. The approximated processing power is given in “Processor Complexity Units” (PCU), specified in MOPS. The approximated RAM usage is given in “RAM Complexity Units” (RCU), specified in kWords (1000 words).

Please note that the following two profiles are independent from each other and are not intended to be interoperable on a bitstream or decoder level. Due to the underlying concept of SAOC the number of output channels is independent of the bitstream. Depending on the decoder capabilities, the output channel configuration is selected via the user interface of the decoder.

5.5.2 Baseline Profile

This profile is the generic SAOC profile for all applications that are not delay critical. Within this profile all SAOC decoders use the Hybrid QMF filterbank and allow rendering to at least two output channels, based on the maximum number of supported downmix channels of two. Rendering to more than two output channels is optional.

An envisioned application scenario for the baseline profile includes e.g. the following interactive re-mix application: a stereo (or mono) compressed signal, or “track” can be provided to the consumer along with SAOC data describing the objects present in the track. The user can with this create his or her own "re-"mix of the music, or the sounds in the stereo (or mono) track, by playing back such object-based tracks with a control similar to that of a multi-track mixing desk and are therefore free to adjust relative level, spatial position, etc. of instruments, sounds, or dialogue according to their preferences.

Note that ISO/IEC 23000-12 (Information technology – Multimedia application format (MPEG-A) – Part 12: Interactive music spplication format) defines several brands that refer to the SAOC Baseline Profile.

5.5.3 LD Profile

This SAOC profile is targeted at applications that require low-delay operation. Within this profile all SAOC decoders use the LD-QMF filterbank and allow rendering to at least two output channels. Rendering to more than two output channels is optional, with one exception: Level 3 requires the support for at least 5 output channels and includes a low-delay version of MPS as the rendering engine.

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

An envisioned application scenario for the LD profile includes e.g. the following telecommunication application: Existing monophonic infrastructures for telecommunication may be extended in their functionality easily by introducing SAOC with a single downmix channel. Telecommunication terminals equipped with an SAOC extension are able to pick up several sound sources (objects), mix them into a single (monophonic) downmix signal which is transmitted in a compatible way by using the existing coders (e.g. speech coders). The side information is conveyed in a hidden, backward compatible way. While such advanced terminals produce an output object stream containing several objects, legacy terminals will simply be considered as producing an object stream with a single embedded object.

The definition of the profiles and levels of SAOC is given in Table 4.

Table 4 – SAOC Profiles and Levels

Profiles	**Baseline profile**				**LD profile**		
Levels	**1**	**2**	**3**	**4**	**1**	**2**	**3**
Hybrid QMF bank	X	X	X	X	-	-	-
LD-QMF bank	-	-	-	-	X	X	X
Max number of residual channels	0	2	4	4	-	-	-
Max sampling rate [kHz]	48	48	48	96	48	48	48
Max number of objects	8	16	32	32	8	32	32
Max number of downmix channels	2	2	2	2	1	2	2
Min number of required output channels *)	2	2	2	2	2	2	5
PCU HQ decoder	12.2	20.4	33.9	67.8	8.4	20.7	39.3 **)
PCU LP decoder	6.6	12.2	23.0	46.0	N/A	N/A	N/A
PCU addition for transcoding	1.1	1.1	1.1	2.3	0.7	1.1	N/A
PCU reduction for integrated transcoding	-6.8	-6.8	-6.8	-6.8	-3.6	-6.5	N/A
RCU HQ decoder	5.7	9.8	13.5	17.5	3.6	4.2	17.9 ***)
RCU LP decoder	4.8	5.4	5.7	10.3	N/A	N/A	N/A
RCU reduction for integrated transcoding	-1.3	-1.3	-1.3	-1.3	-0.6	-1.3	N/A

*) Every SAOC decoder has to support rendering of SAOC content for every number of output channels up to the maximum number, as selected by the user interface. If the number of output channels > 2, MPS is used to render the output signals.

**) PCU number includes SAOC (20.7), transcoding overhead (1.1), reduction for integrated transcoding (-6.5), and an LD-MPS “5-2-5” decoder (24).

***) RCU number includes SAOC (4.2), reduction for integrated transcoding (-1.3), and an LD-MPS “5-2-5” decoder (15).

The SAOC decoder type is defined by the four conditions:

- Profile: baseline profile or LD profile
- Level: 1 - 4 (for baseline) or 1 - 3 (for LD)
- HQ/LP: only applicable for baseline profile
- MPS transcoding support if the number of output channels > 2

For all profiles and levels the following features are supported:

- Decoding to mono/stereo/binaural output

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

6 Syntax

6.1 Payloads for SAOC

Table 5 – Syntax of SAOCSpecificConfig()

Syntax	No. of bits	Mnemonic
SAOCSpecificConfig()		
{		
bsSamplingFrequencyIndex;	**4**	**uimsbf**
if (bsSamplingFrequencyIndex == 15) {		
bsSamplingFrequency;	**24**	**uimsbf**
}		
bsLowDelayMode;	**1**	**uimsbf**
bsFreqRes;	**3**	**uimsbf**
if (bsLowDelayMode == 0) {		
bsFrameLength;	**7**	**uimsbf**
} else {		
bsFrameLength;	**5**	**uimsbf**
}		
bsNumObjects;	**5**	**uimsbf**
for (i=0; i<bsNumObjects+1; i++) {		
bsRelatedTo[i][i] = 1;		
for(j=i+1; j<bsNumObjects+1; j++) {		
bsRelatedTo[i][j];	**1**	**uimsbf**
bsRelatedTo[j][i] = bsRelatedTo[i][j];		
}		
}		
bsTransmitAbsNrg;	**1**	**uimsbf**
bsNumDmxChannels;	**1**	**uimsbf**
if (bsNumDmxChannels == 1) {		
bsTttDualMode;	**1**	**uimsbf**
if (bsTttDualMode) {		
bsTttBandsLow;	**5**	**uimsbf**
bsTttBandsHigh = numBands;		Note 1
} else {		
bsTttBandsLow = numBands;		
}		
}		
bsPdgFlag;	**1**	**uimsbf**
bsOneIOC;	**1**	**uimsbf**
bsDcuFlag;	**1**	**uimsbf**
if (bsDcuFlag == 1) {		
bsDcuMandatory;	**1**	**uimsbf**
bsDcuDynamic;	**1**	**uimsbf**
if (bsDcuDynamic == 0) {		
bsDcuMode;	**1**	**uimsbf**
bsDcuParam;	**4**	**uimsbf**
}		
} else {		
bsDcuMandatory = 0**;**		
bsDcuDynamic = 0**;**		
bsDcuMode = 0;		
bsDcuParam = 0;		
}		
ByteAlign();		
SAOCExtensionConfig();		
}		
Note 1: numBands is defined in Table 33 and depends on bsFreqRes.		

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

Table 6 – Syntax of SAOCExtensionConfig()

Syntax	No. of bits	Mnemonic
SAOCExtensionConfig()		
{		
SaocExtNum = 0;		
while (BitsAvailable() >= 8) {		Note 1
bsSaocExtType;	**4**	**uimsbf**
SaocExtType[SaocExtNum] = bsSaocExtType;		
SaocExtNum++;		
cnt = **bsSaocExtLen**;	**4**	**uimsbf**
if (cnt==15) {		
cnt += **bsSaocExtLenAdd**;	**8**	**uimsbf**
}		
if (cnt==15+255) {		
cnt += **bsSaocExtLenAddAdd**;	**16**	**uimsbf**
}		
bitsRead = SAOCExtensionConfigData(bsSaocExtType)		Note 2
nFillBits = 8*cnt-bitsRead;		
bsFillBits;	**nFillBits**	**bslbf**
}		
}		
}		
Note 1: The function BitsAvailable() returns the number of bits available to be read.		
Note 2: SAOCExtensionConfigData() returns the number of bits read.		

Table 7 – Syntax of SAOCExtensionConfigData(0)

Syntax	No. of bits	Mnemonic
SAOCExtensionConfigData(0)		
{		
if (bsDcuFlag == 1) {		
bsDcuFlag2;	**1**	**uimsbf**
if ((bsDcuFlag2 == 1) && (bsDcuDynamic == 0)) {		
bsDcuMode2;	**1**	**uimsbf**
bsDcuParam2;	**4**	**uimsbf**
}		
}		
ResidualConfig();		
}		

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

Table 8 – Syntax of ResidualConfig()

Syntax	No. of bits	Mnemonic
ResidualConfig()		
{		
bsResidualSamplingFrequencyIndex;	**4**	**uimsbf**
bsResidualFramesPerSAOCFrame;	**2**	**uimsbf**
bsNumGroupsFGO;	**2**	**uimsbf**
for (i=0; i<bsNumGroupsFGO + 1; i++) {		
bsResidualPresent[i];	**1**	**uimsbf**
if (bsResidualPresent[i]) {		
bsResidualBands[i];	**5**	**uimsbf**
}		
bsTtnDualMode[i];	**1**	**uimsbf**
if (bsTtnDualMode[i]) {		
bsTtnBandsLow[i];	**5**	**uimsbf**
} else {		
bsTtnBandsLow[i] = numBands;		Note 1
}		
}		
}		
Note 1: numBands is defined in Table 33 and depends on bsFreqRes.		

Table 9 – Syntax of SAOCExtensionConfigData(1)

Syntax	No. of bits	Mnemonic
SAOCExtensionConfigData(1)		
{		
}		

Table 10 – Syntax of SAOCExtensionConfigData(2)

Syntax	No. of bits	Mnemonic
SAOCExtensionConfigData(2)		
{		
SpatialSpecificConfig();		Note 1
}		
Note 1: SpatialSpecificConfig() is defined in ISO/IEC 23003-1:2007, Table 5.		

Table 11 – Syntax of SAOCExtensionConfigData(3)

Syntax	No. of bits	Mnemonic
SAOCExtensionConfigData(3)		
{		
}		

Table 12 – Syntax of SAOCExtensionConfigData(8)

Syntax	No. of bits	Mnemonic
SAOCExtensionConfigData(8)		
{		
ObjectMetaData();		
}		

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

Table 13 – Syntax of ObjectMetaData()

Syntax	No. of bits	Mnemonic
ObjectMetaData()		
{		
for (i=0; i<bsNumObjects+1; i++) {		
bsNumByteMetaData[i];	**8**	**uimsbf**
for (j=0; j<bsNumByteMetaData[i]; j++) {		
bsMetaData[i][j];	**8**	**bslbf**
}		
}		
}		

Table 14 – Syntax of SAOCExtensionConfigData(9)

Syntax	No. of bits	Mnemonic
SAOCExtensionConfigData(9)		
{		
PresetConfig();		
}		

Table 15 – Syntax of PresetConfig()

Syntax	No. of bits	Mnemonic
PresetConfig()		
{		
bsNumPresets;	**4**	**uimsbf**
for (i=0; i<bsNumPresets+1; i++) {		
bsNumBytePresetLabel[i];	**8**	**uimsbf**
for (j=0; j<bsNumBytePresetLabel[i]; j++) {		
bsPresetLabel[i][j];	**8**	**bslbf**
}		
bsPresetMatrix;	**1**	**uimsbf**
if (bsPresetMatrix) {		
PresetMatrixData();		
} else {		
PresetUserData();		
}		
}		
}		

Table 16 – Syntax of PresetMatrixData()

Syntax	No. of bits	Mnemonic
PresetMatrixData()		
{		
bsPresetMatrixType;	**2**	**uimsbf**
for(i=0; i<bsNumObjects+1; i++) {		
numChannels = PresetMatrixType[bsPresetMatrixType];		
for(j=0; j<numChannels; i++) {		
bsPresetMatrixElements[i][j];	**5**	**uimsbf**
}		
}		
}		

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

Table 17 – Syntax of PresetUserData()

Syntax	No. of bits	Mnemonic
PresetUserData()		
{		
for(i=0; i<6; i++) {		
bsPresetUserDataIdentifier[i];	**8**	**bslbf**
}		
bsPresetUserDataLen;	**12**	**uimsbf**
PresetUserDataContainer(bsPresetUserDataIdentifier);		
}		

Table 18 – Syntax of SAOCExtensionConfigData(10)

Syntax	No. of bits	Mnemonic
SAOCExtensionConfigData(10)		
{		
SeparationMetaData();		
}		

Table 19 – Syntax of SeparationMetaData()

Syntax	No. of bits	Mnemonic
SeparationMetaData()		
{		
bsNumSeparationPairs;	**4**	**uimsbf**
for (i=0; i<bsNumSeparationPairs+1; i++) {		
bsSeparationMainObjectID[i];	**5**	**uimsbf**
bsSeparationSubObjectID[i];	**5**	**uimsbf**
}		
}		

Table 20 – Syntax of SAOCFrame()

Syntax	No. of bits	Mnemonic
SAOCFrame()		
{		
SAOCFramingInfo();		
bsIndependencyFlag;	**1**	**uimsbf**
for(i=0; i<bsNumObjects+1; i++) {		
idxOLD[i] = EcDataSaoc(OLD, i, numBands);		
}		
if (bsTransmitAbsNrg) {		
idxNRG = EcDataSaoc(NRG, 0, numBands);		
}		
k=0;		
iocIdx1=0;		
iocIdx2=0;		
for(i=0; i<bsNumObjects+1; i++) {		
idxIOC[i][i] = 0;		
for(j=i+1; j<bsNumObjects+1; j++) {		
if (bsRelatedTo[i][j] != 0) {		
if (bsOneIOC == 0) {		
idxIOC[i][j] = EcDataSaoc(IOC, k, numBands);		
k++;		
} else {		
if (k == 0)		
idxIOC[i][j] = EcDataSaoc(IOC, k, numBands);		
k++;		

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

Table 20 (*continued*)

Syntax	No. of bits	Mnemonic
iocIdx1=i;		
iocIdx2=j;		
} else {		
idxIOC[i][j] = idxIOC[iocIdx1][iocIdx2];		
}		
}		
} else {		
idxIOC[i][j] = 5;		
}		
idxIOC[j][i] = idxIOC[i][j];		
}		
}		
idxDMG = EcDataSaoc(DMG, 0, bsNumObjects+1);		
if (bsNumDmxChannels == 1) {		
idxDCLD = EcDataSaoc(DCLD, 0, bsNumObjects+1);		
}		
if (bsPdgFlag == 1) {		
for (i=0; i<bsNumDmxChannels + 1; i++) {		
idxPDG[i] = EcDataSaoc(PDG, i, numBands);		Note 1
}		
}		
if (bsDcuFlag == 1) && (bsDcuDynamic == 1) {		
if (bsIndependencyFlag == 1) {		
bsDcuDynamicUpdate = 1**;**		
} else {		
bsDcuDynamicUpdate;	**1**	**uimsbf**
}		
if (bsDcuDynamicUpdate == 1) {		
bsDcuMode;	**1**	**uimsbf**
bsDcuParam;	**4**	**uimsbf**
}		
}		
ByteAlign();		
SAOCExtensionFrame();		
}		
Note 1: numBands is defined in Table 33 and depends on bsFreqRes.		

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

Table 21 – Syntax of SAOCFramingInfo()

Syntax	No. of bits	Mnemonic
SAOCFramingInfo()		
{		
bsFramingType;	**1**	**uimsbf**
If (bsLowDelayMode == 0) {		
bsNumParamSets;	**3**	**uimsbf**
} else {		
bsNumParamSets;	**1**	**uimsbf**
}		
if (bsFramingType) {		
for (ps=0; ps<numParamSets; ps++) {		Note 1
bsParamSlot[ps];	**nBitsParamSlot**	**uimsbf** Note 2
}		
}		
}		
Note 1: numParamSets is defined by numParamSets = bsNumParamSets + 1. Note 2: nBitsParamSlot is defined according to nBitsParamSlot = ceil(log2(numSlots)).		

© ISO/IEC 2010 – All rights reserved

Table 22 – EcDataSaoc()

Syntax	No. of bits	Mnemonic
EcDataSaoc(dataType, paramIdx, stopIdx)		Note 1
{		
dataSets = 0;		
for (ps=0; ps<bsNumParamSets + 1; ps++) {		
bsXXXdataMode[paramIdx][ps];	**2**	**uimsbf**
if (bsXXXdataMode[paramIdx][ps] == 3) {		
dataSets++;		
}		
}		
setIdx = 0;		
while (setIdx < dataSets) {		
If (dataSets-setIdx > 1) {		
bsDataPairXXX[paramIdx][setIdx];	**1**	**uimsbf**
} else {		
bsDataPairXXX[paramIdx][setIdx] = 0;		
}		
bsQuantCoarseXXX[paramIdx][setIdx];	**1**	**uimsbf**
bsFreqResStrideXXX[paramIdx][setIdx];	**2**	**uimsbf**
dataDim = (stopIdx-1)/pbStride+1;		Note 2
SAOCEcDataPair(dataType, paramIdx, setIdx, dataDim,		
bsDataPairXXX[paramIdx][setIdx],		
bsQuantCoarseXXX[paramIdx][setIdx]);		
if (bsDataPairXXX[paramIdx][setIdx]) {		
bsXXXQuantCoarse[paramIdx][setIdx+1] =		
bsXXXQuantCoarse[paramIdx][setIdx];		
bsFreqResStrideXXX[paramIdx][setIdx+1] =		
bsFreqResStrideXXX[paramIdx][setIdx];		
}		
setIdx += bsDataPairXXX[paramIdx][setIdx]+1;		
}		
stopIdxXXX[paramIdx] = stopIdx;		
}		
Note 1: XXX is to be replaced by the value of dataType (OLD, IOC, NRG, DCLD, DMG, PDG). Note 2: pbStride is defined in ISO/IEC 23003-1:2007, Table 70 and depends on bsFreqResStride[][]. Furthermore the division shall be interpreted as ANSI C integer division.		

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

Table 23 – Syntax of SAOCEcDataPair()

Syntax	No. of bits	Mnemonic
SAOCEcDataPair(dataType, paramIdx, setIdx, dataBands, pairFlag, coarseFlag)		Note 1
{		
mixedTimePair_flag = 0;		
bsPcmCodingXXX[paramIdx][setIdx];	1	**uimsbf**
if (bsPcmCoding[paramIdx][setIdx]) {		
if (coarseFlag) {		
numQuantSteps = numQuantStepsXXXCoarse;		Note 2
} else {		
numQuantSteps = numQuantStepsXXXFine;		Note 2
}		
aaDataPair = GroupedPcmData(dataType, pairFlag, numQuantSteps, dataBands);		Note 3
} else {		
allowDiffTimeBack = (!bsIndependencyFlag) || (setIdx>0);		
(aaDataPairMsbDiff, aPgOffset, mixedTimePair_flag) = SAOCDiffHuffData(dataType, pairFlag, allowDiffTimeBack, dataBands);		
aaDataPairLsb[0] = LsbData(dataType, coarseFlag, dataBands);		Note 4
if (pairFlag) {		
aaDataPairLsb[1] = LsbData(dataType, coarseFlag, dataBands);		Note 4
}		
}		
bsDiffTypeXXX[paramIdx][setIdx] = bsDiffType[0];		
bsDiffTimeDirectionXXX[paramIdx][setIdx] = bsDiffTimeDirection[0];		
mixedTimePairXXX[paramIdx][setIdx] = mixedTimePair_flag;		
if (pairFlag) {		
bsDiffTypeXXX[paramIdx][setIdx+1] = bsDiffType[1];		
bsDiffTimeDirectionXXX[paramIdx][setIdx+1] = bsDiffTimeDirection[1];		
bsPcmCodingXXX[paramIdx][setIdx+1] = bsPcmCodingXXX[paramIdx][setIdx];		
mixedTimePairXXX[paramIdx][setIdx+1] = mixedTimePair_flag;		
}		
for (pg=0; pg<dataBands; pg++) {		
if (bsPcmCodingXXX[paramIdx][setIdx]) {		
bsXXXpcm[paramIdx][setIdx][pg] = aaDataPair[0][pg];		
} else {		
bsXXXmsbDiff[paramIdx][setIdx][pg] = aaDataPairMsbDiff[0][pg];		
bsXXXlsb[paramIdx][setIdx][pg] = aaDataPairLsb[0][pg];		
}		
if (pairFlag) {		
if (bsPcmCodingXXX[paramIdx][setIdx+1]) {		
bsXXXpcm[paramIdx][setIdx+1][pg] = aaDataPair[1][pg];		
} else {		
bsXXXmsbDiff[paramIdx][setIdx+1][pg] =aaDataPairMsbDiff[1][pg];		
bsXXXlsb[paramIdx][setIdx+1][pg] = aaDataPairLsb[1][pg];		
}		
}		
}		
}		
Note 1: XXX is to be replaced by the value of dataType (OLD, IOC, NRG, DCLD, DMG, PDG). Note 2: numQuantStepsXXXCoarse and numQuantStepsXXXFine is defined in Table 42 and depends on dataType. Note 3: GroupedPcmData() is defined in ISO/IEC 23003-1:2007, Table 25. Note 4: LsbData() is defined in ISO/IEC 23003-1:2007, Table 31.		

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

Table 24 – Syntax of SAOCDiffHuffData()

Syntax	No. of bits	Mnemonic
SAOCDiffHuffData(dataType, pairFlag, allowDiffTimeBackFlag, dataBands)		
{		
mixedTimePair_flag = 0;		
bsDiffType[0] = DIFF_FREQ;		
bsDiffType[1] = DIFF_FREQ;		
if (pairFlag || allowDiffTimeBackFlag) {		
bsDiffType[0];	**1**	**uimsbf**
}		
if (pairFlag && ((bsDiffType[0] == DIFF_FREQ) || allowDiffTimeBackFlag)) {		
bsDiffType[1];	**1**	**uimsbf**
}		
bsCodingScheme;	**1**	**uimsbf**
if (bsCodingScheme == HUFF_1D) {		
(aaHuffData[0]) = SAOCHuffData1D(dataType, aDiffType[0],dataBands);		
if (pairFlag) {		
(aaHuffData[1]) =		
SAOCHuffData1D(dataType, aDiffType[1],dataBands);		
}		
} else {		
bsPairing = FREQ_PAIR;		
(aaHuffData[0]) =		
SAOCHuffData2DFreqPair(dataType, aDiffType[0], dataBands);		
If (pairFlag) {		
(aaHuffData[1]) =		
SAOCHuffData2DFreqPair(dataType, aDiffType[1], dataBands);		
}		
}		
if ((bsDiffType[0] == DIFF_TIME) || (bsDiffType[1] == DIFF_TIME)) {		
bsDiffTimeDirection[0] = BACKWARDS;		
if (pairFlag) {		
bsDiffTimeDirection[1] = BACKWARDS;		
}		
}		
return (aaHuffData, aPgOffset, mixedTimePair_flag);		
}		

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

Table 25 – Syntax of SAOCHuffData1D()

Syntax	No. of bits	Mnemonic
SAOCHuffData1D(dataType, diffType, dataBands)		
{		
pgOffset = 0;		
if (diffType == DIFF_FREQ) {		
aHuffData1D[0] = 1Dhuff_dec(hcodFirstBand_XXX, **bsCodeW**);	**1..x**	**vlclbf** Note 1,3
pgOffset = 1;		
}		
for (i=pgOffset; i<dataBands; i++) {		
aHuffData1D[i] = 1Dhuff_dec(hcod1D_XXX_YY, **bsCodeW**);	**1..x**	**vlclbf** Note 1,2,3
if (aHuffData1D[i] != 0) {		
bsSign;	**1**	**uimsbf**
if (bsSign) {		
aHuffData1D[i] = -aHuffData1D[i];		
}		
}		
}		
return (aHuffData1D);		
}		
Note 1: XXX is to be replaced by the value of dataType (OLD, IOC, NRG, DCLD, DMG, PDG).		
Note 2: YY is to be replaced by "DF" or "DT", depending on the value of diffType.		
Note 3: 1Dhuff_dec() is defined in ISO/IEC 23003-1:2007, Annex A.1		

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

Table 26 – Syntax of SAOCHuffData2DFreqPair()

Syntax	No. of bits	Mnemonic
SAOCHuffData2DFreqPair(dataType, diffType, dataBands)		
{		
LavIdx = 1Dhuff_dec(hcodLavIdx, **bsCodeW**);	**1..3**	**vlclbf**
lav = lavTabXXX[LavIdx];		Note 1
pgOffset = 0;		
if (diffType == DIFF_FREQ) {		
aHuffData2D[0] = 1Dhuff_dec(hcodFirstBand_XXX, **bsCodeW**);	**1..x**	**vlclbf** Note 6
pgOffset = 1;		
}		
escapeCode = hcod2D_XXX_YY_FP_LL_escape;		Note 2,3,4,5
escCntr = 0;		
for (i=pgOffset; i<dataBands; i+=2) {		
(aTmp[0], aTmp[1]) = 2Dhuff_dec(hcod2D_XXX_YY_FP_LL, **bsCodeW**);	**1..x**	**vlclbf** Note 3,4,5,6
if (bsCodeWord != escapeCode) {		
aTmpSym = SymmetryData(aTmp);		
aHuffData2D[i] = aTmpSym[0];		
aHuffData2D[i+1] = aTmpSym[1];		
} else {		
aEscList[escCntr++] = i;		
}		
}		
if (escCntr > 0) {		
aaEscData = GroupedPcmData(dataType, 1, 2*lav+1, escCntr);		
for (i=0; i<escCntr; i++) {		
aHuffData2D[aEscList[i]] = aaEscData[0][i] - lav;		
aHuffData2D[aEscList[i]+1] = aaEscData[1][i] - lav;		
}		
}		
if ((dataBands-pgOffset) % 2) {		Note 7
aHuffData2D[dataBands-1] = 1Dhuff_dec(hcod1D_XXX_YY, **bsCodeW**);	**1..x**	**vlclbf** Note 3,4,6
if (aHuffData2D[dataBands-1] != 0) {		
bsSign;	**1**	**uimsbf**
if (bsSign) {		
aHuffData2D[dataBands-1] = -aHuffData2D[dataBands-1];		
}		
}		
}		
return (aHuffData2D);		
}		
Note 1: lavTabXXX is defined in Table 40 and Table 41. Note 2: The escape code tables are defined in Table A.6, Table A.7, Table A.8. Note 3: XXX is to be replaced by the value of dataType (OLD, IOC, NRG, DCLD, DMG, PDG). Note 4: YY is to be replaced by “DF” or “DT”, depending on the value of diffType. Note 5: LL is to be replaced by the value of lav. Note 6: 1Dhuff_dec() and 2Dhuff_dec() are defined in ISO/IEC 23003-1:2007, Annex A.1. Note 7: % denotes the modulo operator (ANSI C integer math) and returns the remainder of the division.		

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

Table 27 – Syntax of SAOCExtensionFrame()

Syntax	No. of bits	Mnemonic
SAOCExtensionFrame()		
{		
for (ec=0; ec<SaocExtNum; ec++) {		
if (SaocExtType[ec]<8) {		
cnt = **bsSaocExtLen**;	**8**	**uimsbf**
if (cnt==255) {		
cnt += **bsSaocExtLenAdd**;	**16**	**uimsbf**
}		
if (cnt>0) {		
bitsRead = SAOCExtensionFrameData(SaocExtType[ec])		Note 1
}		
nFillBits = 8*cnt-bitsRead;		
bsFillBits;	**nFillBits**	**bslbf**
}		
}		
}		
Note 1: SAOCExtensionFrameData() returns the number of bits read.		

Table 28 – Syntax of SAOCExtensionFrameData(0)

Syntax	No. of bits	Mnemonic
SAOCExtensionFrameData(0)		
{		
if ((bsDcuFlag == 1) && (bsDcuFlag2 == 1) && (bsDcuDynamic == 1)) {		
if (bsIndependencyFlag == 1) {		
bsDcuDynamicUpdate2 = 1**;**		
} else {		
bsDcuDynamicUpdate2;	**1**	**uimsbf**
}		
if (bsDcuDynamicUpdate2 == 1) {		
bsDcuMode2;	**1**	**uimsbf**
bsDcuParam2;	**4**	**uimsbf**
}		
}		
SAOCResidualData();		
}		

Table 29 – Syntax of SAOCResidualData()

Syntax	No. of bits	Mnemonic
SAOCResidualData()		
{		
for (i=0; i<bsNumEAO + 1; i++) {		
if (bsResidualPresent[i]) {		
individual_channel_stream(0);		Note 1
}		
}		
}		
Note1: individual_channel_stream(0) according to MPEG-2 AAC Low Complexity profile bitstream syntax described in 6.3 of ISO/IEC 13818-7:2006.		

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

Table 30 – Syntax of SAOCExtensionFrameData(1)

Syntax	No. of bits	Mnemonic

SAOCExtensionFrameData(1)
{
    PresetConfig();
}		

Table 31 – Syntax of SAOCExtensionFrameData(2)

Syntax	No. of bits	Mnemonic

SAOCExtensionFrameData(2)
{
    SpatialFrame();
}
		
		Note 1

Note 1: SpatialFrame() is defined in ISO/IEC 23003-1:2007, Table 15.		

Table 32 – Syntax of SAOCExtensionFrameData(3)

Syntax	No. of bits	Mnemonic

SAOCExtensionFrameData(3)
{
    ObjectMetaData();
}		

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

6.2 Definition

ByteAlign()	Up to 7 fill bits to achieve byte alignment with respect to the beginning of the syntactic element in which ByteAlign() occurs.
SAOCSpecificConfig()	Syntactic element that contains the SAOC configuration data (header).
bsSamplingFrequencyIndex	see 1.6.3.4 in ISO/IEC 14496-3:2009.
bsSamplingFrequency	see 1.6.3.3 in ISO/IEC 14496-3:2009.
bsFreqRes	Defines the number of parameter bands according to:

Table 33 – bsFreqRes

bsFreqRes	**numBands**	
	bsLowDelayMode==0	bsLowDelayMode==1
0	Reserved	Reserved
1	28	23
2	20	15
3	14	12
4	10	9
5	7	7
6	5	5
7	4	4

bsLowDelayMode Indicates whether the SAOC operates in HQ or LD mode according to:

Table 34 – bsLowDelayMode

bsLowDelayMode	**Meaning**
0	HQ operation mode
1	LD operation mode (see 7.14)

bsTttBandsHigh	Same as bsTttBandsLow but for high band range. The high band range is bsTttBandsLow <= pb < **bsTttBandsHigh**.
bsFrameLength	Defines the number of time slots in an SAOC frame.
bsNumObjects	Defines the number of object channels for which SAOC parameters are transmitted. In case of stereo or multi-channel objects, each channel counts as separate object.
bsRelatedTo[i][j]	Defines whether an object has relation with other objects. More precisely, **bsRelatedTo**[i][j] = 1 means that object i and object j are related, i.e., have correlation, whereas **bsRelatedTo**[i][j] = 0 means that this is not the case.
bsTransmitAbsNrg	Defines whether absolute energy parameters are transmitted. These are necessary for delay-free merging of SAOC bitstreams in an MCU bitstream combiner.
bsNumDmxChannels	Defines the number of downmix channels.
bsTttDualMode	Indicates if transcoding to the TTT box operates in different modes for a low and high band range according to:

Table 35 – bsTttDualMode

bsTttDualMode	Meaning
0	same TTT transcoding mode for full band range (i.e. no separate high band range)
1	different TTT transcoding modes for low and high band ranges

bsTttBandsLow	Defines the number of parameter bands for which TTT transcoding should be processed according to prediction or energy based scheme. Prediction based scheme should be used for the parameter band range: 0 <= pb < **bsTttBandsLow**. Energy based scheme should be used for the parameter band range: **bsTttBandsLow** <= pb < **numBands**.
bsTtnDualMode	Indicates if transcoding to the OTN/TTN box operates in different modes for a low and high band range according to:

Table 36 – bsTtnDualMode[i]

bsTtnDualMode[i]	Meaning
0	same OTN/TTN transcoding mode for full band range (i.e. no separate high band range)
1	different OTN/TTN transcoding modes for low and high band ranges

bsTtnBandsLow	Defines the number of parameter bands for which OTN/TTN transcoding should be processed according to prediction or energy based scheme. Prediction based scheme should be used for the parameter band range: 0 <= pb < bsTtnBandsLow. Energy based scheme should be used for the parameter band range: bsTtnBandsLow <= pb < numBands.
bsPdgFlag	Defines whether PDG parameters are transmitted according to:

Table 37 – bsPdgFlag

bsPdgFlag	Meaning
0	PDG parameters are not transmitted
1	PDG parameters are transmitted

bsDcuFlag	Defines whether the values **bsDcuMode** and **bsDcuParam** are transmitted in the bitstream.

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

bsDcuFlag2 Defines whether the values **bsDcuMode2** and **bsDcuParam2** are transmitted in the bitstream.

bsDcuMandatory If **bsDcuMandatory** == 1, then the DCU must be applied using the parameters **bsDcuMode** and **bsDcuParam** as transmitted in the bitstream. If **bsDcuMandatory** == 0, then the DCU parameters **bsDcuMode** and **bsDcuParam** transmitted in the bitstream are only recommended values, and also other DCU settings could be used.

bsDcuDynamic Enables dynamic signaling of the values **bsDcuMode** and **bsDcuParam**.

bsDcuDynamicUpdate Defines whether the values **bsDcuMode** and **bsDcuParam** are updated. More precisely, **bsDcuDynamicUpdate** == 1 means that the values **bsDcuMode** and **bsDcuParam** are updated in the current frame, whereas **bsDcuDynamicUpdate** == 0 means that the previously transmitted values are kept.

bsDcuDynamicUpdate2 Same as **bsDcuDynamicUpdate** but for application only in strict EAO mode.

bsDcuMode Defines the distortion-free target matrix type for the DCU according to Table 40.

Table 38 – bsDcuMode

bsDcuMode	Meaning
0	Downmix-similar target matrix
1	Best-effort target matrix

bsDcuMode2 Same as **bsDcuMode** but for application only in strict EAO mode.

bsDcuParam Defines the parameter value for the DCU algorithm according to Table 41.

Table 39 – bsDcuParam parameters quantization table

idx	0	1	2	3	4	5	6	7
DcuParam[idx]	0.00	0.05	0.10	0.15	0.20	0.25	0.30	0.35
idx	8	9	10	11	12	13	14	15
DcuParam[idx]	0.40	0.45	0.50	0.60	0.70	0.80	0.90	1.00

bsDcuParam2 Same as **bsDcuParam** but for application only in strict EAO mode.

bsOneIOC Indicates if only a single IOC parameter is conveyed common to all objects which have relation with other, signalled by bsRelatedTo[i][j] = 1.

bsObjectMetaDataAvailable

Defines whether or not metadata information for the encoded objects is transmitted.

ObjectMetaData() Syntactic element that contains a metadata description of the the encoded objects.

bsNumEAO Defines the number of EAO channels.

bsNumByteMetaData[i] Defines the number of bytes used for the metadata description of an object i.

bsMetaData[i][j] Defines the metadata description (interpreted as string in UTF-8 encoding format).

SAOCFrame() Syntactic element that contains the data of an SAOC frame (payload).

SAOCFramingInfo() Syntactic element that contains information about the number of parameter sets and their associated time slots.

bsIndependencyFlag Indicates if lossless coding of the current SAOC frame is done independently of the previous SAOC frame, i.e. whether the current SAOC frame can be decoded without knowledge about the previous SAOC frame.

EcDataSaoc() Syntactic element that contains all parameter subsets of a given parameter in the SAOC frame. However the following changes apply:

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

Table 40 – lavTabXXXdf

LavIdx	**lavTabDCLD/DMG/PDGdf [LavIdx]**	**lavTabIOCdf [LavIdx]**	**lavTabOLDdf [LavIdx]**	**lavTabNRGdf [LavIdx]**
0	3	1	3	3
1	5	3	6	5
2	7	5	9	7
3	9	7	12	9
Note: Table 40 replaces ISO/IEC 23003-1:2007, Table 76, and shall be applied only in case diffType == DIFF_FREQ.				

Table 41 – lavTabXXXdt

LavIdx	**lavTabDCLD/DMG/PDGdt [LavIdx]**	**lavTabIOCdt [LavIdx]**	**lavTabOLDdt [LavIdx]**	**lavTabNRGdt [LavIdx]**
0	3	1	3	3
1	5	3	6	6
2	7	5	9	9
3	9	7	12	12
Note: Table 41 replaces ISO/IEC 23003-1:2007, Table 76, and shall be applied only in case diffType == DIFF_TIME.				

SAOCEcDataPair()	Syntactic element that contains one or two temporally subsequent parameter subsets of a given parameter in the SAOC frame.
bsPcmCodingXXX	Indicates whether PCM coding is applied.
GroupedPcmData()	Syntactic element that contains one or two temporally subsequent parameter subsets of a given parameter in the SAOC frame, where groups of quantized values are represented by a single PCM code. numQuantSteps depends on the data type XXX and whether coarse or fine quantization is used and is defined in Table 42.

Table 42 — numQuantSteps

XXX (dataType)	**numQuantStepsXXXCoarse**	**numQuantStepsXXXFine**
DCLD, DMG, PDG	15	31
IOC	4	8
OLD	8	16
NRG	32	64

SAOCExtensionConfig()	Syntactic element that acts as container to carry extensions to the SAOC audio configuration. The presence of the SAOC extension of type bsSaocExtType (see Table 43) is signaled by the presence of a SAOCExtensionConfigData(bsSaocExtType) element in SAOCExtensionConfig().
SaocExtNum	Helper variable storing the number of the SAOC extension containers present.
bsSaocExtType	Indicates type of the SAOC extension data according to:

Table 43 – bsSaocExtType

bsSaocExtTyp	**Meaning**	**SAOCExtensionFrameData()**
0	Residual coding data	present
1	Dynamic preset information	
2	MBO MPS data	
3	Dynamic object metadata	
4...7	Reserved	
8	Static object metadata	not present
9	Static preset information	
10	Separation metadata	
11…15	Reserved	

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

SaocExtType[i]	Helper variable storing the type of spatial extension data carried in the extension container i.
bsSaocExtLen	Number of bytes in SAOCExtensionConfigData() or SAOCExtensionFrameData().
bsSaocExtLenAdd	Additional number of bytes in SAOCExtensionConfigData() or SAOCExtensionFrameData().
bsSaocExtLenAddAdd	Further additional number of bytes in SAOCExtensionConfigData().
bsFillBits	Fill bits, to be ignored.
SAOCExtensionConfigData(bsSacExtType)	Instance of the SAOCExtensionConfigData that carries configuration data for the SAOC extension of type bsSaocExtType (see Table 43).
SAOCExtensionConfigData(0)	Syntactic element that, if present, indicates that residual coding information and corresponding DCU parameters are available.
SAOCExtensionConfigData(1)	Syntactic element that, if present, indicates that dynamic preset information is available.
SAOCExtensionConfigData(2)	Syntactic element that, if present, indicates that the MBO MPS data is available.
SAOCExtensionConfigData(3)	Syntactic element that, if present, indicates that dynamic object metadata is available.
SAOCExtensionConfigData(8)	Syntactic element that, if present, indicates that metadata information is available.
SAOCExtensionConfigData(9)	Syntactic element that, if present, indicates that static preset information is available.
SAOCExtensionConfigData(10)	Syntactic element that, if present, indicates that separation metadata is available.
SeparationMetaData()	Syntactic element that contains a metadata description of the separated objects.
bsNumSeparationPairs	Defines the number of pairs of the separated objects.
bsSeparationMainObjectID[i]	Defines the object ID of the main object which is separated from mixed signals. Its value is in the range of 0 to bsNumObjects.
bsSeparationSubObjectID[i]	Defines the object ID of the sub object which is separated from mixed signals. Its value is in the range of 0 to bsNumObjects.
PresetConfig()	Syntactic element that contains all preset rendering data.
bsNumPresets	Defines the number of available rendering presets. The case **bsNumPresets** = 0 means that no such data are available.
bsNumBytePresetLabel[i]	Defines the number of bytes used for the preset i.
bsPresetLabel[i][j]	Defines the text label for the preset i (interpreted as string in UTF-8 encoding format).

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

bsPresetMatrix Defines the preset data representation type according to:

Table 44 – bsPresetMatrix

bsPresetMatrix	**Meaning**
0	User-defined preset representation format
1	Matrix-based preset representation format

PresetMatrixData() Syntactic element that contains the preset rendering parameters in the matrix-based representation format.

bsPresetMatrixType Defines the rendering configuration according to:

Table 45 – bsPresetMatrixType

bsPresetMatrixType	**Meaning**
0	Mono playback system
1	Stereo playback system
2	5.0 playback system
3	Reserved

PresetMatrixType[bsPresetMatrixType]

Defines the number of upmix channels according to:

Table 46 – PresetMatrixType

bsPresetMatrixType	0	1	2	3
PresetMatrixType[bsPresetMatrixType]	1	2	5	N/A

bsPresetMatrixElements[i][j]

Defines the preset rendering matrix. Namely, the output level of the audio object channel i for the upmix channel j.

PresetUserData() Syntactic element that contains the preset rendering parameters in the user-defined preset representation format.

bsPresetUserDataIdentifier[i]

Defines the identifier for the user-defined preset i representation format (interpreted as string in UTF-8 encoding format).

bsPresetUserDataLen Defines the length in bytes for the preset data included into PresetUserDataContainer() container element.

PresetUserDataContainer()

Syntactic element that contains preset rendering data in the user-defined preset representation format and has a length of exactly **bsPresetUserDataLen** bytes.

7 SAOC processing

7.1 Compressed data stream decoding and dequantization of SAOC data

7.1.1 Introduction

This Subclause describes the decoding and dequantization of the bitstream payload into variables that are used in the SAOC transcoder/decoder.

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

7.1.2 Dequantization of the SAOC parameters

The dequantization of the IOC parameters follows the same rules as defined in 6.1.8 of ISO/IEC 23003-1:2007 for the MPS ICC parameters, the decoding of the DMG, DCLD and PDG parameters those of the MPS CLD parameters.

The dequantization of the OLD (i.e. relative energy) parameters is done by applying the dequantization function $deq\left(index, parameterType\right)$ as defined in 6.1.8 of ISO/IEC 23003-1:2007 in combination with the parameter quantization table given in Table 47.

Table 47 – OLD parameter quantization table

idx	0	1	2	3	4	5	6	7
OLD[idx]	$10^{-15.00}$	$10^{-4.50}$	$10^{-4.00}$	$10^{-3.50}$	$10^{-3.00}$	$10^{-2.50}$	$10^{-2.20}$	$10^{-1.90}$
idx	8	9	10	11	12	13	14	15
OLD[idx]	$10^{-1.60}$	$10^{-1.30}$	$10^{-1.00}$	$10^{-0.80}$	$10^{-0.60}$	$10^{-0.40}$	$10^{-0.20}$	1

The dequantization of the NRG (i.e. absolute energy) parameters is done by applying the dequantization function $deq\left(index, parameterType\right)$ as defined in 6.1.8 of ISO/IEC 23003-1:2007 in combination with the parameter quantization table given in Table 48.

Table 48 – NRG parameter quantization table

idx	0	1	2	3	4	5	6	7
NRG[idx]	$50 \cdot 10^{-9.45}$	$50 \cdot 10^{-9.30}$	$50 \cdot 10^{-9.15}$	$50 \cdot 10^{-9.00}$	$50 \cdot 10^{-8.85}$	$50 \cdot 10^{-8.70}$	$50 \cdot 10^{-8.55}$	$50 \cdot 10^{-8.40}$
idx	8	9	10	11	12	13	14	15
NRG[idx]	$50 \cdot 10^{-8.25}$	$50 \cdot 10^{-8.10}$	$50 \cdot 10^{-7.95}$	$50 \cdot 10^{-7.80}$	$50 \cdot 10^{-7.65}$	$50 \cdot 10^{-7.50}$	$50 \cdot 10^{-7.35}$	$50 \cdot 10^{-7.20}$
idx	16	17	18	19	20	21	22	23
NRG[idx]	$50 \cdot 10^{-7.05}$	$50 \cdot 10^{-6.90}$	$50 \cdot 10^{-6.75}$	$50 \cdot 10^{-6.60}$	$50 \cdot 10^{-6.45}$	$50 \cdot 10^{-6.30}$	$50 \cdot 10^{-6.15}$	$50 \cdot 10^{-6.00}$
idx	24	25	26	27	28	29	30	31
NRG[idx]	$50 \cdot 10^{-5.85}$	$50 \cdot 10^{-5.70}$	$50 \cdot 10^{-5.55}$	$50 \cdot 10^{-5.40}$	$50 \cdot 10^{-5.25}$	$50 \cdot 10^{-5.10}$	$50 \cdot 10^{-4.95}$	$50 \cdot 10^{-4.80}$
idx	32	33	34	35	36	37	38	39
NRG[idx]	$50 \cdot 10^{-4.65}$	$50 \cdot 10^{-4.50}$	$50 \cdot 10^{-4.35}$	$50 \cdot 10^{-5.20}$	$50 \cdot 10^{-4.05}$	$50 \cdot 10^{-3.90}$	$50 \cdot 10^{-3.75}$	$50 \cdot 10^{-3.60}$
idx	40	41	42	43	44	45	46	47
NRG[idx]	$50 \cdot 10^{-3.45}$	$50 \cdot 10^{-3.30}$	$50 \cdot 10^{-3.15}$	$50 \cdot 10^{-3.00}$	$50 \cdot 10^{-2.85}$	$50 \cdot 10^{-2.70}$	$50 \cdot 10^{-2.55}$	$50 \cdot 10^{-2.40}$
idx	48	49	50	51	52	53	54	55
NRG[idx]	$50 \cdot 10^{-2.25}$	$50 \cdot 10^{-2.10}$	$50 \cdot 10^{-1.95}$	$50 \cdot 10^{-1.80}$	$50 \cdot 10^{-1.65}$	$50 \cdot 10^{-1.50}$	$50 \cdot 10^{-1.35}$	$50 \cdot 10^{-1.20}$
idx	56	57	58	59	60	61	62	63
NRG[idx]	$50 \cdot 10^{-1.05}$	$50 \cdot 10^{-0.90}$	$50 \cdot 10^{-0.75}$	$50 \cdot 10^{-0.60}$	$50 \cdot 10^{-0.45}$	$50 \cdot 10^{-0.30}$	$50 \cdot 10^{-0.15}$	50

The decoding of the SAOCFrame() data results in the parameter indices idxXXX of the quantized OLD, IOC, NRG, DCLD, DMG and PDG parameters that are listed in Table 49, where

$$pi = 0 \ldots N-1,\quad ps = 0 \ldots L-1,\quad pb = 0 \ldots M_{proc}-1.$$

Table 49 – Dimensions and value ranges of the parameter indices idxXXX

Parameter	idxOLD	idxNRG	idxIOC	idxDMG	idxDCLD	idxPDG
Dimension	[pi][ps][pb]	[ps][pb]	[pi][pi][ps][pb]	[ps][pi]	[ps][pi]	[ps][pi]
Value range	0 ... 15	0 … 63	0 ... 7	-15 ... 15	-15 ... 15	-15 ... 15

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

The decoding follows largely the procedure described in ISO/IEC 23003-1:2007, taking into account the following differences. The pseudo code defining the preprocessing step in 6.1.2.3.2 is altered in the following way:

```
setIdxStart = dataSetIdx[ps];
startBand = startBandXXX[pi];
stopBand = stopBandXXX[pi];
pbStride = pbStrideTable[bsFreqResStrideXXX[pi][setIdx]];
dataBands = (stopBand - startBand - 1)/pbStride + 1; /*ANSI C integer math*/
aGroupToBand = createMapping(startBand, stopBand, pbStride);
for (pg=0; pg<dataBands; pg++) {
    pb = aGroupToBand[pg];
    tmp = idxXXX[pi][ps-1][pb];
    switch (XXX) {
    case DCLD, DMG, PDG:
        if (bsQuantCoarseXXX[pi][setIdx]) {
            tmp = (tmp/2)+7;    /* ANSI C integer math */
        }
        else {
            tmp = tmp+15;
        }
        break;
    case IOC:
        if (bsQuantCoarseXXX[pi][setIdx]) {
            tmp = tmp/2;    /* ANSI C integer math */
        }
        break;
    case OLD:
        if (bsQuantCoarseXXX[pi][setIdx]) {
            tmp = tmp/2;    /* ANSI C integer math */
        }
        break;
    case NRG:
        if (bsQuantCoarseXXX[pi][setIdx]) {
            tmp = tmp/2;    /* ANSI C integer math */
        }
        break;
    }
    idxXXXmsb[pi][setIdxStart-1][pg] = tmp;
}
```

The pseudo code defining the postprocessing step in 6.1.2.3.2 is altered in the following way:

```
for (i=0; i<=bsDataPairXXX[pi][setIdxStart]; i++) {
    setIdx = setIdxStart+i;
    ps = paramSet[setIdx];
    paramHandled[ps] = 1;
    for (pg=0; pg<dataBands; pg++) {
        tmp = idxXXXnotMapped[pi][setIdx][pg];
        switch (XXX) {
        case DCLD, DMG, PDG:
            if (bsQuantCoarseXXX[pi][setIdx]) {
                tmp = (tmp-7)*2;
                if (tmp==-14) tmp=-15;
                if (tmp==14) tmp=15;
            }
            else {
                tmp = tmp-15;
            }
            break;
            case IOC:
```

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

```
                if (bsQuantCoarseXXX[pi][setIdx]) {
                    tmp = tmp*2;
                }
                break;
                case OLD:
                if (bsQuantCoarseXXX[pi][setIdx]) {
                    if ( tmp > 0 ) {
                        tmp = tmp*2 + 1;
                    }
                }
                break;
                case NRG:
                if (bsQuantCoarseXXX[pi][setIdx]) {
                    tmp = tmp*2;
                }
                break;

            }
            pbStart = aGroupToBand[pg];
            pbStop = aGroupToBand[pg+1];
            for (pb=pbStart; pb<pbStop; pb++) {
                idxXXX[pi][ps][pb] = tmp;
            }
        }
    }
```

The pseudo code in 6.1.2.2 is altered in the following way:

```
    while (ps=0; ps<numParamSet; ps++) {
        switch (bsXXXdataMode[pi][ps]) {
        case 0: /* default */
            for (pb=0; pb<numBands, pb++) {
                switch (XXX) {
                case OLD, NRG, IOC, DCLD, DMG, PDG:
                    idxXXX[pi][ps][pb] = 0;
                    break;
                }
            }
            break;
        case 1: /* keep */
        case 2: /* interpolate */
            for (pb=0; pb<numBands, pb++) {
                idxXXX[pi][ps][pb] = idxXXX[pi][ps-1][pb];
            }
            break;
        case 3: /* coded */
            if (!paramHandled[ps]) {
                DecodeDataPair(); /*see subclause 6.1.2.3 in ISO/IEC 23003-1:2007*/
            }
            break;
        }
    }
```

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

7.2 Compressed data stream encoding and quantization of MPS data

7.2.1 Introduction

This Subclause describes the encoding and quantization of the variables into bitstream payload that are used in the MPS decoder.

7.2.2 Quantization of the MPS parameters

The obtained CLD (ADG), ICC and CPC parameters are quantized according to Tables 82, 83 and 84 defined in ISO/IEC 23003-1:2007. The delta and Huffman coding corresponds the description given in ISO/IEC 23003-1:2007, Clause 6.

7.2.3 Unquantized interface for the MPS parameters

For an efficient practical implementation and to prevent a loss in precision, the parameter interface to the MPS decoder may alternatively be established in a direct, unquantized way. Rather than writing an actual MPS bitstream, the relevant parameters may be passed directly to the MPS decoder.

7.2.4 List of MPS bitstream variables

The following table contains the list of all MPS bitstream variables (with the corresponding references to tables defined in ISO/IEC 23003-1:2007) which are needed to be specified for the MPS bitstream generation and their values:

Table 50 – List of MPS bitstream variables

MPS bitstream variable	Value / SAOC bitstream variable / description	Reference*)
bsSamplingFrequencyIndex	bsSamplingFrequencyIndex	Table 5
bsSamplingFrequency	bsSamplingFrequency	Table 5
bsFrameLength	bsFrameLength	Table 5
bsFreqRes	bsFreqRes	Table 5
bsTreeConfig	0 (for mono downmix) / 2 (for stereo downmix)	Table 5
bsQuantMode	0	Table 5
bsOneIcc	0	Table 5
bsArbitraryDownmix	1 (for mono downmix) / 0 (for stereo downmix)	Table 5
bsFixedGainsSur	0	Table 5
bsFixedGainsLFE	0	Table 5
bsFixedGainsDMX	0	Table 5
bsMatrixMode	0	Table 5
bsTempShapeConfig	0	Table 5
bsDecorrConfig	0	Table 5
bs3DaudioMode	0	Table 5
bsOttBands[0]	The number of parameter bands for LFE channel for which OTT information is present	Table 6
bsTttDualMode[0]	bsTttDualMode	Table 7
bsTttModeLow[0]	1	Table 7
bsTttModeHigh[0]	5	Table 7
FramingInfo()	SAOCFramingInfo()	Table 15
bsIndependencyFlag	Indicates if lossless coding of current frame is done independently of previous frame	Table 15
bsSmoothMode	0	Table 19
*) defined in ISO/IEC 23003-1:2007		

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

7.3 Time/frequency transforms

The same hybrid filterbank as described in ISO/IEC 23003-1:2007 is applied.

7.4 Post(processing) downmix compensation

If the post(processed) downmix $\mathbf{X}^{n,k}_{\text{post(processed)}}$ is used, the following modification should be taken prior to SAOC decoding/transcoding:

$$\mathbf{X}^{n,k} = \mathbf{W}^{n,k}_{\text{PDG}} \mathbf{X}^{n,k}_{\text{post(processed)}} \,,$$

where $\mathbf{X}^{n,k}$ represents the input signal to the SAOC decoder/transcoder.

The matrix $\mathbf{W}^{n,k}_{\text{PDG}}$ is defined for every time-slot n and every hybrid subband k. Its elements are obtained from the transmitted PDG parameters which are defined for a given parameter time-slot l and a given processing band m. The mapping to the hybrid domain is done according to Table A.31, ISO/IEC 23003-1:2007. If post(processed) downmix compensation is applied (bsPdgFlag = 1), the matrix $\mathbf{W}^{l,m}_{\text{PDG}}$ is defined as:

$$\mathbf{W}^{l,m}_{\text{PDG}} = \left(PDG_0^{l,m} \right), \qquad \text{for mono downmix,}$$

$$\mathbf{W}^{l,m}_{\text{PDG}} = \begin{pmatrix} PDG_0^{l,m} & 0 \\ 0 & PDG_1^{l,m} \end{pmatrix}, \qquad \text{for stereo downmix,}$$

where $PDG_j^{l,m} = \mathbf{D}_{\text{PDG}}\left(j,l,m\right)$.

7.5 Signals and parameters

7.5.1 Dimensionality of signals and parameters

The audio signals are defined for every time slot n and every hybrid subband k. The corresponding SAOC parameters are defined for each parameter time slot l and processing band m. The subsequent mapping between the hybrid and parameter domain is specified by Table A.31, ISO/IEC 23003-1:2007. Hence, all calculations are performed with respect to the certain time/band indices and the corresponding dimensionalities are implied for each introduced variable.

The data available at the SAOC decoder/transcoder consists of the downmix signal $\mathbf{X}$, covariance matrix $\mathbf{E}$, rendering matrix $\mathbf{M}_{\text{ren}}$ and downmix matrix $\mathbf{D}$.

7.5.2 Input signal

The input signal $\mathbf{X}$ to the SAOC decoder/transcoder is represented as

$$\mathbf{X} = \mathbf{x}^{n,k} = \begin{pmatrix} l_0 \\ r_0 \end{pmatrix}, \qquad \text{for stereo downmix,}$$

$$\mathbf{X} = \mathbf{x}^{n,k} = \begin{pmatrix} d_0 \\ 0 \end{pmatrix}, \qquad \text{for monoo downmix.}$$

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

7.5.3 Object parameters

The covariance matrix $\mathbf{E}$ of size $N \times N$ with elements $e_{i,j}$ represents an approximation of the original signal covariance matrix $\mathbf{E} \approx \mathbf{SS}^*$ and is obtained from the OLD and IOC parameters as

$$e_{i,j} = \sqrt{OLD_i OLD_j} IOC_{i,j} .$$

Here, the dequantized object parameters are obtained according to 7.1.2 as

$$OLD_i = \mathbf{D}_{\text{OLD}}(i,l,m), \quad IOC_{i,j} = \mathbf{D}_{\text{IOC}}(i,j,l,m).$$

7.5.4 Rendering matrix

The rendering matrix $\mathbf{M}_{\text{ren}}$ applied to the input audio objects $\mathbf{S}$ determines the target rendered output as $\mathbf{Y} = \mathbf{M}_{\text{ren}}\mathbf{S}$. The rendering matrix $\mathbf{M}_{\text{ren}}$ with elements $m_{i,j}$ maps all input objects i to the desired output channels j. The rendering matrix $\mathbf{M}_{\text{ren}}$ is given by

$$\mathbf{M}_{\text{ren}} = \begin{pmatrix} m_{0,Lf} & \cdots & m_{N-1,Lf} \\ m_{0,Rf} & \cdots & m_{N-1,Rf} \\ m_{0,C} & \cdots & m_{N-1,C} \\ m_{0,Lfe} & \cdots & m_{N-1,Lfe} \\ m_{0,Ls} & \cdots & m_{N-1,Ls} \\ m_{0,Rs} & \cdots & m_{N-1,Rs} \end{pmatrix}, \quad \text{for 5.1 output configuration,}$$

$$\mathbf{M}_{\text{ren}} = \begin{pmatrix} m_{0,L} & \cdots & m_{N-1,L} \\ m_{0,R} & \cdots & m_{N-1,R} \end{pmatrix}, \quad \text{for stereo output configuration,}$$

$$\mathbf{M}_{\text{ren}} = \begin{pmatrix} m_{0,C} & \cdots & m_{N-1,C} \end{pmatrix}, \quad \text{for mono output configuration.}$$

7.5.5 Downmix matrix

The downmix matrix $\mathbf{D}$ applied to the input audio objects $\mathbf{S}$ determines the downmix signal as $\mathbf{X} = \mathbf{DS}$. For the stereo downmix case the downmix matrix $\mathbf{D}$ of size $2 \times N$ with elements $d_{i,j}$ $(i = 0,1;\ j = 0,\ldots,N-1)$ is obtained from the DMG and DCLD parameters as

$$d_{0,j} = 10^{0.05 DMG_j} \sqrt{\frac{10^{0.1 DCLD_j}}{1+10^{0.1 DCLD_j}}}, \quad d_{1,j} = 10^{0.05 DMG_j} \sqrt{\frac{1}{1+10^{0.1 DCLD_j}}}.$$

For the mono downmix case the downmix matrix $\mathbf{D}$ of size $1 \times N$ with elements $d_{i,j}$ $(i = 0;\ j = 0,\ldots,N-1)$ is obtained from the DMG parameters as

$$d_{0,j} = 10^{0.05 DMG_j} .$$

Here, the dequantized downmix parameters are obtained according to 7.1.2 as

$$DMG_j = \mathbf{D}_{\text{DMG}}(j,l), \quad DCLD_j = \mathbf{D}_{\text{DCLD}}(j,l).$$

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

7.6 Transcoding modes

7.6.1 Introduction

In this Subclause the method for combining SAOC parameters and panning information associated with each audio object in a standards compliant MPS bitstream is explained. The SAOC transcoder is depicted in Figure 2 (left) and consists of the SAOC parameter processor and downmix processor applied for a stereo downmix.

7.6.2 Mono downmix ("x-1-5") processing mode

7.6.2.1 Introduction

The following subclauses give a description of the SAOC transcoding mode for the mono downmix case. The object parameters (OLD, IOC, DMG, DCLD) from the SAOC bitstream are transcoded into spatial parameters (CLD, ICC, CPC, ADG) for the MPS bitstream according to the rendering information. The downmix is not modified.

7.6.2.2 Sub-rendering matrices for each OTT element

In case of a mono downmix, the MPS decoder employs a tree-structured parameterization. The tree is populated using OTT elements that split each (mono) input into two output signals, see Figure 13.

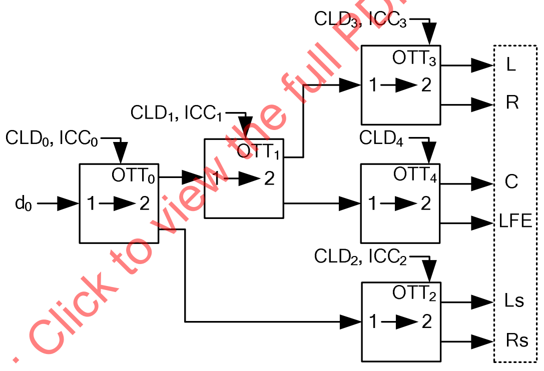


Figure 13 – "5-1-5$_1$" tree structure for the MPS decoder (mono downmix)

Associated with each OTT element is the channel level difference (CLD) parameter that describes the relative level differences between the two output channels, and the inter-channel correlation (ICC) parameter for the desired cross-correlation between the output signals.

In contrast to the MPS the SAOC bitstream comprises the relative level of each audio object in the downmix signal, termed the object level difference (OLD) and the inter-object correlation (IOC). Additional information is provided by the rendering matrix $\mathbf{M}_{\text{ren}}^{l,m}$ with elements $m_{i,j}^{l,m}$, yielding the mapping of all audio input channels i to the desired output channels j. The rendering matrix $\mathbf{M}_{\text{ren}}^{l,m}$ for the 5.1 output configuration is given by:

$$\mathbf{M}_{\text{ren}}^{l,m} = \begin{pmatrix} m_{0,Lf}^{l,m} & \cdots & m_{N-1,Lf}^{l,m} \\ m_{0,Rf}^{l,m} & \cdots & m_{N-1,Rf}^{l,m} \\ m_{0,C}^{l,m} & \cdots & m_{N-1,C}^{l,m} \\ m_{0,Lfe}^{l,m} & \cdots & m_{N-1,Lfe}^{l,m} \\ m_{0,Ls}^{l,m} & \cdots & m_{N-1,Ls}^{l,m} \\ m_{0,Rs}^{l,m} & \cdots & m_{N-1,Rs}^{l,m} \end{pmatrix}.$$

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

The task of the SAOC parameter processing unit is to estimate all CLD and ICC parameters from the SAOC data and the rendering matrix. This process is performed for each OTT element independently.

The respective contribution of each object to the two outputs of OTT element 0 is obtained by summation of the corresponding elements in $\mathbf{M}_{\text{ren}}^{l,m}$. This summation gives a sub-rendering matrix $\mathbf{W}_0^{l,m}$ of OTT element 0:

$$\mathbf{W}_0^{l,m} = \begin{pmatrix} w_{0,0}^0 & \cdots & w_{0,N-1}^0 \\ w_{1,0}^0 & \cdots & w_{1,N-1}^0 \end{pmatrix} =$$

$$= \begin{pmatrix} m_{0,Lf}^{l,m} + m_{0,Rf}^{l,m} + m_{0,C}^{l,m} + m_{0,Lfe}^{l,m} & \cdots & m_{N-1,Lf}^{l,m} + m_{N-1,Rf}^{l,m} + m_{N-1,C}^{l,m} + m_{N-1,Lfe}^{l,m} \\ m_{0,Ls}^{l,m} + m_{0,Rs}^{l,m} & \cdots & m_{N-1,Ls}^{l,m} + m_{N-1,Rs}^{l,m} \end{pmatrix}.$$

The CLDs and ICCs of the subsequent OTT boxes ($CLD_h^{l,m}$, $ICC_h^{l,m}$, $h = 0,\ldots,4$) are calculated using the sub-rendering matrices defined as:

$$\mathbf{W}_1^{l,m} = \begin{pmatrix} w_{0,0}^1 & \cdots & w_{0,N-1}^1 \\ w_{1,0}^1 & \cdots & w_{1,N-1}^1 \end{pmatrix} = \begin{pmatrix} m_{0,Lf}^{l,m} + m_{0,Rf}^{l,m} & \cdots & m_{N-1,Lf}^{l,m} + m_{N-1,Rf}^{l,m} \\ m_{0,C}^{l,m} + m_{0,Lfe}^{l,m} & \cdots & m_{N-1,C}^{l,m} + m_{N-1,Lfe}^{l,m} \end{pmatrix},$$

$$\mathbf{W}_2^{l,m} = \begin{pmatrix} w_{0,0}^2 & \cdots & w_{0,N-1}^2 \\ w_{1,0}^2 & \cdots & w_{1,N-1}^2 \end{pmatrix} = \begin{pmatrix} m_{0,Ls}^{l,m} & \cdots & m_{N-1,Ls}^{l,m} \\ m_{0,Rs}^{l,m} & \cdots & m_{N-1,Rs}^{l,m} \end{pmatrix},$$

$$\mathbf{W}_3^{l,m} = \begin{pmatrix} w_{0,0}^3 & \cdots & w_{0,N-1}^3 \\ w_{1,0}^3 & \cdots & w_{1,N-1}^3 \end{pmatrix} = \begin{pmatrix} m_{0,Lf}^{l,m} & \cdots & m_{N-1,Lf}^{l,m} \\ m_{0,Rf}^{l,m} & \cdots & m_{N-1,Rf}^{l,m} \end{pmatrix},$$

$$\mathbf{W}_4^{l,m} = \begin{pmatrix} w_{0,0}^4 & \cdots & w_{0,N-1}^4 \\ w_{1,0}^4 & \cdots & w_{1,N-1}^4 \end{pmatrix} = \begin{pmatrix} m_{0,C}^{l,m} & \cdots & m_{N-1,C}^{l,m} \\ m_{0,Lfe}^{l,m} & \cdots & m_{N-1,Lfe}^{l,m} \end{pmatrix}.$$

The superscript of the matrix elements, e.g. $w_{0,0}^h$, refers to the OTT element index.

7.6.2.3 Estimation of power and cross power terms

Incorporating index h denoting the OTT element, the power and cross power terms can be estimated by:

$$p_{h,0}^2 = \sum_{i=0}^{N-1}\left(\sum_{j=0}^{N-1} w_{0,i}^h w_{0,j}^h e_{i,j}\right), \quad p_{h,1}^2 = \sum_{i=0}^{N-1}\left(\sum_{j=0}^{N-1} w_{1,i}^h w_{1,j}^h e_{i,j}\right), \quad R_h = \sum_{i=0}^{N-1}\left(\sum_{j=0}^{N-1} w_{0,i}^h w_{1,j}^h e_{i,j}\right).$$

7.6.2.4 Derivation of the MPS parameters

Finally, the corresponding CLD and ICC parameters are derived as:

$$CLD_h^{l,m} = 10\log_{10}\left(\max\left(\frac{p_{h,0}^2}{p_{h,1}^2}, \varepsilon^2\right)\right),$$

$$ICC_h^{l,m} = \frac{\max\left(R_h, \varepsilon^2\right)}{\sqrt{\max\left(p_{h,0}, \varepsilon^2\right)\max\left(p_{h,1}, \varepsilon^2\right)}},$$

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

where

$$CLD_h^{l,m} = \mathbf{D}_{\text{CLD}}\left(h,l,m\right) \text{ and } ICC_h^{l,m} = \mathbf{D}_{\text{ICC}}\left(h,l,m\right).$$

In order to achieve not only a repanning of the objects but also an object attenuation/amplification with an MPS renderer, the Arbitrary Downmix Gains (ADGs) can be used for a "virtual" modification of the downmix signal energy. The ADG is a logarithmic measure and based on the rendering matrix $\mathbf{A}^{l,m}$ and the object parameters (OLD, IOC and DMG):

$$ADG^{l,m} = 10\log_{10}\left(\max\left(\frac{f^{l,m}}{v^{l,m}},\varepsilon^2\right)\right).$$

The scalar $f^{l,m}$ is computed as

$$f^{l,m} = \sum_{i=0}^{N-1} f_{i,i}^{l,m}.$$

Matrix $\mathbf{F}^{l,m}$ of size $N_{\text{MPS}} \times N_{\text{MPS}}$ with elements $f_{i,j}^{l,m}$ is given as

$$\mathbf{F}^{l,m} = \mathbf{A}^{l,m}\mathbf{E}^{l,m}\left(\mathbf{A}^{l,m}\right)^*.$$

The scalar $v^{l,m}$ is computed as

$$v^{l,m} = \mathbf{D}^l\mathbf{E}^{l,m}\left(\mathbf{D}^l\right)^* + \varepsilon^2.$$

The obtained CLD, ICC and ADG parameters are quantized, formatted as MPS bitstream and fed into the MPS decoder.

7.6.3 Stereo downmix (“x-2-5”) processing mode

7.6.3.1 Introduction

The following subclauses give a description of the SAOC transcoding mode for the stereo downmix case. The object parameters (OLD, IOC, DMG, DCLD) from the SAOC bitstream are transcoded into spatial parameters (CLD, ICC, CPC) for the MPS bitstream according to the rendering information. The downmix is modified according to object parameters and rendering matrix.

7.6.3.2 Rendering of object energies

The transcoder determines the parameters for the MPS decoder according to the target rendering as described by the rendering matrix $\mathbf{M}_{\text{ren}}$. The six channel target covariance is denoted with $\mathbf{F}$ and given by

$$\mathbf{F} = \mathbf{YY}^* = \mathbf{M}_{\text{ren}}\mathbf{S}(\mathbf{M}_{\text{ren}}\mathbf{S})^* = \mathbf{M}_{\text{ren}}(\mathbf{SS}^*)\mathbf{M}_{\text{ren}}^* = \mathbf{M}_{\text{ren}}\mathbf{E}\mathbf{M}_{\text{ren}}^*.$$

The transcoding process can conceptually be divided into two parts. In one part a three channel rendering is performed to a left, right and center channel. In this stage the parameters for the downmix modification as well as the prediction parameters for the TTT box for the MPS decoder are obtained. In the other part the CLD and ICC parameters for the rendering between the front and surround channels (OTT parameters, left front – left surround, right front – right surround) are determined. This is illustrated in Figure 14.

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

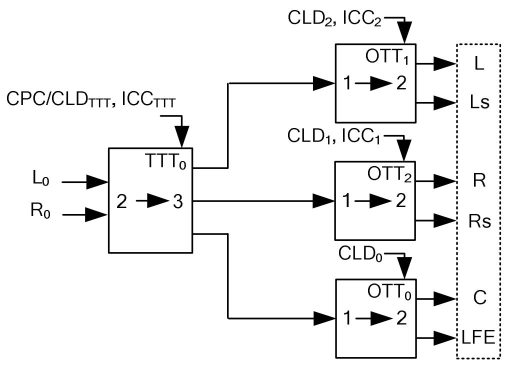


Figure 14 – "5-2-5" tree structure for the MPS decoder (stereo downmix)

7.6.3.2.1 Rendering to left, right and center channel

In this stage the spatial parameters are determined that control the rendering to a left and right channel, consisting of front and surround signals. These parameters describe the prediction matrix of the TTT box for the MPS decoding $\mathbf{C}_{\text{TTT}}$ (CPC parameters for the MPS decoder) and the downmix converter matrix $\mathbf{G}$.

$\mathbf{C}_{\text{TTT}}$ is the prediction matrix to obtain the target rendering from the modifed downmix $\hat{\mathbf{X}} = \mathbf{GX}$:

$$\mathbf{C}_{\text{TTT}}\hat{\mathbf{X}} = \mathbf{C}_{\text{TTT}}\mathbf{GX} \approx \mathbf{A}_3\mathbf{S} .$$

$\mathbf{A}_3$ is a reduced rendering matrix of size $3 \times N$, describing the rendering to the left, right and center channel respectively. It is obtained as $\mathbf{A}_3 = \mathbf{D}_{36}\mathbf{M}_{\text{ren}}$ with the 6 to 3 partial downmix matrix $\mathbf{D}_{36}$ defined by

$$\mathbf{D}_{36} = \begin{pmatrix} w_1 & 0 & 0 & 0 & w_1 & 0 \\ 0 & w_2 & 0 & 0 & 0 & w_2 \\ 0 & 0 & w_3 & w_3 & 0 & 0 \end{pmatrix} .$$

The partial downmix weights w_p , $p = 1,2,3$ are adjusted such that the energy of $w_p(y_{2p-1} + y_{2p})$ is equal to the sum of energies $\|y_{2p-1}\|^2 + \|y_{2p}\|^2$ up to a limit factor.

$$w_1 = \frac{f_{1,1} + f_{5,5}}{f_{1,1} + f_{5,5} + 2f_{1,5}} , \quad w_2 = \frac{f_{2,2} + f_{6,6}}{f_{2,2} + f_{6,6} + 2f_{2,6}} , \quad w_3 = 0.5 ,$$

where $f_{i,j}$ denote the elements of $\mathbf{F}$.

For the estimation of the desired prediction matrix $\mathbf{C}_{\text{TTT}}$ and the downmix preprocessing matrix $\mathbf{G}$ we define a prediction matrix $\mathbf{C}_3$ of size 3×2 , that leads to the target rendering

$$\mathbf{C}_3\mathbf{X} \approx \mathbf{A}_3\mathbf{S} .$$

Such a matrix is derived by considering the normal equations

$$\mathbf{C}_3\left(\mathbf{DED}^*\right) \approx \mathbf{A}_3\mathbf{ED}^* .$$

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

The solution to the normal equations yields the best possible waveform match for the target output given the object covariance model. $\mathbf{G}$ and $\mathbf{C}_{\text{TTT}}$ are now obtained by solving the system of equations

$$\mathbf{C}_{\text{TTT}}\mathbf{G} = \mathbf{C}_3 .$$

To avoid numerical problems when calculating the term $\mathbf{J} = \left(\mathbf{DED}^*\right)^{-1}$, $\mathbf{J}$ is modified. First the eigenvalues $\lambda_{1,2}$ of $\mathbf{J}$ are calculated, solving $\det(\mathbf{J} - \lambda_{1,2}\mathbf{I}) = 0$.

Eigenvalues are sorted in descending ($\lambda_1 \geq \lambda_2$) order and the eigenvector corresponding to the larger eigenvalue is calculated according to the equation above. It is assured to lie in the positive x-plane (first element has to be positive). The second eigenvector is obtained from the first by a – 90 degrees rotation:

$$\mathbf{J} = \left(\mathbf{v}_1\mathbf{v}_2\right)\begin{pmatrix}\lambda_1 & 0\\ 0 & \lambda_2\end{pmatrix}\left(\mathbf{v}_1\mathbf{v}_2\right)^* .$$

A weighting matrix is computed from the downmix matrix $\mathbf{D}$ and the prediction matrix $\mathbf{C}_3$, $\mathbf{W} = \left(\mathbf{D}\ diag(\mathbf{C}_3)\right)$.

Since $\mathbf{C}_{\text{TTT}}$ is a function of the MPS prediction parameters c_1 and c_2 (as defined in ISO/IEC 23003-1:2007), $\mathbf{C}_{\text{TTT}}\mathbf{G} = \mathbf{C}_3$ is rewritten in the following way, to find the stationary point or points of the function,

$$\mathbf{\Gamma}\begin{pmatrix}\tilde{c}_1\\ \tilde{c}_2\end{pmatrix} = \mathbf{b} ,$$

with $\mathbf{\Gamma} = \left(\mathbf{D}_{\text{TTT}}\ \mathbf{C}_3\right)\mathbf{W}\left(\mathbf{D}_{\text{TTT}}\ \mathbf{C}_3\right)^*$ and $\mathbf{b} = \mathbf{GWC}_3\mathbf{v}$,

where $\mathbf{D}_{\text{TTT}} = \begin{pmatrix}1 & 0 & 1\\ 0 & 1 & 1\end{pmatrix}$ and $\mathbf{v} = \begin{pmatrix}1 & 1 & -1\end{pmatrix}$.

If $\mathbf{\Gamma}$ does not provide a unique solution ($\det(\mathbf{\Gamma}) < 10^{-3}$), the point is chosen that lies closest to the point resulting in a TTT pass through. As a first step, the row i of $\mathbf{\Gamma}$ is chosen $\boldsymbol{\gamma} = [\gamma_{i,1} \quad \gamma_{i,2}]$ where the elements contain most energy, thus $\gamma_{i,1}{}^2 + \gamma_{i,2}{}^2 \geq \gamma_{j,1}{}^2 + \gamma_{j,2}{}^2, \quad j = 1,2$. Then a solution is determined such that

$$\begin{pmatrix}\tilde{c}_1\\ \tilde{c}_2\end{pmatrix} = \begin{pmatrix}1\\ 1\end{pmatrix} - 3\mathbf{y} \text{ with } \mathbf{y} = \frac{b_{i,3}}{\left(\sum_{j=1,2}\left(\gamma_{i,j}\right)^2\right) + \varepsilon}\boldsymbol{\gamma}^T .$$

If the obtained solution for $\tilde{c}_1$ and $\tilde{c}_2$ is outside the allowed range for prediction coefficients that is defined as $-2 \leq \tilde{c}_j \leq 3$ (as defined in ISO/IEC 23003-1:2007), $\tilde{c}_j$ shall be calculated according to below.

First define the set of points, $\mathbf{x}_{\text{p}}$ as:

$$\mathbf{x}_{\text{p}} \in \begin{bmatrix} \begin{pmatrix}\min\left(3, \max\left(-2, -\dfrac{-2\gamma_{1,2} - b_1}{\gamma_{1,1} + \varepsilon}\right)\right)\\ -2\end{pmatrix}, \begin{pmatrix}\min\left(3, \max\left(-2, -\dfrac{3\gamma_{1,2} - b_1}{\gamma_{1,1} + \varepsilon}\right)\right)\\ 3\end{pmatrix}\\ \begin{pmatrix}-2\\ \min\left(3, \max\left(-2, -\dfrac{-2\gamma_{2,1} - b_2}{\gamma_{2,2} + \varepsilon}\right)\right)\end{pmatrix}, \begin{pmatrix}3\\ \min\left(3, \max\left(-2, -\dfrac{3\gamma_{2,1} - b_2}{\gamma_{2,2} + \varepsilon}\right)\right)\end{pmatrix} \end{bmatrix},$$

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

and the distance function,

$$distFunc\left(\mathbf{x}_\mathrm{p}\right)=\mathbf{x}_\mathrm{p}^{*}\mathbf{\Gamma}\mathbf{x}_\mathrm{p1}-2\mathbf{b}\mathbf{x}_\mathrm{p} .$$

Then the prediction parameters are defined according to:

$$\begin{pmatrix}\tilde{c}_1\\ \tilde{c}_2\end{pmatrix}=\arg\min_{\mathbf{x}\in\mathbf{x}_\mathrm{p}}\left(distFunc\left(\mathbf{x}\right)\right).$$

The prediction parameters are constrained according to:

$$c_1=\left(1-\lambda\right)\tilde{c}_1+\lambda\gamma_1,\quad c_2=\left(1-\lambda\right)\tilde{c}_2+\lambda\gamma_2,$$

where λ, γ_1 and γ_2 are defined as

$$\gamma_1=\frac{2f_{1,1}+2f_{5,5}-f_{3,3}+f_{1,3}+f_{5,3}}{2f_{1,1}+2f_{5,5}+2f_{3,3}+4f_{1,3}+4f_{5,3}},\qquad \gamma_2=\frac{2f_{2,2}+2f_{6,6}-f_{3,3}+f_{2,3}+f_{6,3}}{2f_{2,2}+2f_{6,6}+2f_{3,3}+4f_{2,3}+4f_{6,3}},$$

$$\lambda=\left(\frac{\left(f_{1,2}+f_{1,6}+f_{5,2}+f_{5,6}+f_{1,3}+f_{5,3}+f_{2,3}+f_{6,3}+f_{3,3}\right)^2}{\left(f_{1,1}+f_{5,5}+f_{3,3}+2f_{1,3}+2f_{5,3}\right)\left(f_{2,2}+f_{6,6}+f_{3,3}+2f_{2,3}+2f_{6,3}\right)}\right)^8.$$

For the MPS decoder, the CPCs and corresponding $\mathrm{ICC_{TTT}}$ are provided as follows

$$\mathbf{D}_{\mathrm{CPC\_1}}=c_1(l,m),\ \mathbf{D}_{\mathrm{CPC\_2}}=c_2(l,m)\ \text{and}\ \mathbf{D}_{\mathrm{ICC_{TTT}}}=1.$$

7.6.3.2.2 Rendering between front and surround channels

The parameters that determine the rendering between front and surround channels can be estimated directly from the target covariance matrix $\mathbf{F}$

$$CLD_{a,b}=10\log_{10}\left(\frac{\max\left(f_{a,a},\varepsilon^2\right)}{\max\left(f_{b,b},\varepsilon^2\right)}\right),\quad ICC_{a,b}=\frac{\max\left(f_{a,b},\varepsilon^2\right)}{\sqrt{\max\left(f_{a,a},\varepsilon^2\right)\max\left(f_{b,b},\varepsilon^2\right)}},$$

with $\left(a,b\right)$ = $\left(1,2\right)$ and $\left(3,4\right)$.

The MPS parameters are provided in the form

$$CLD_h^{l,m}=\mathbf{D}_{\mathrm{CLD}}\left(h,l,m\right)\ \text{and}\ ICC_h^{l,m}=\mathbf{D}_{\mathrm{ICC}}\left(h,l,m\right),$$

for every OTT box h.

7.6.3.3 Stereo Preprocessing

The stereo downmix $\mathbf{X}$ is processed into the modified downmix signal $\hat{\mathbf{X}}$:

$$\hat{\mathbf{X}}=\mathbf{G}\mathbf{X},$$

where

$$\mathbf{G}=\mathbf{D}_{\mathrm{TTT}}\mathbf{C}_3=\mathbf{D}_{\mathrm{TTT}}\mathbf{M}_{\mathrm{ren}}\mathbf{E}\mathbf{D}^{*}\mathbf{J}.$$

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

The final stereo output from the SAOC transcoder $\hat{\mathbf{X}}$ is produced by mixing $\mathbf{X}$ with a decorrelated signal component according to:

$$\hat{\mathbf{X}} = \mathbf{G}_{\text{Mod}}\mathbf{X} + \mathbf{P}_2\mathbf{X}_{\text{d}},$$

where the decorrelated signal $\mathbf{X}_{\text{d}}$ is calculated according to 7.6.3.4, and the mix matrices $\mathbf{G}_{\text{Mod}}$ and $\mathbf{P}_2$ acording to below.

First, define the render upmix error matrix as

$$\mathbf{R} = \mathbf{A}_{\text{diff}}\mathbf{E}\mathbf{A}_{\text{diff}}^*,$$

where

$$\mathbf{A}_{\text{diff}} = \mathbf{D}_{\text{TTT}}\mathbf{A}_3 - \mathbf{G}\mathbf{D},$$

and moreover define the covariance matrix of the predicted signal $\hat{\mathbf{R}}$ as

$$\hat{\mathbf{R}} = \begin{pmatrix} \hat{r}_{1,1} & \hat{r}_{1,2} \\ \hat{r}_{2,1} & \hat{r}_{2,2} \end{pmatrix} = \mathbf{G}\mathbf{D}\mathbf{E}\mathbf{D}^*\mathbf{G}^*.$$

The gain vector $\mathbf{g}_{vec}$ can subsequently be calculated as:

$$\mathbf{g}_{vec} = \left(\min\left(\sqrt{\max\left(\frac{\hat{r}_{1,1} + r_{1,1} + \varepsilon^2}{r_{1,1} + \varepsilon^2}, 0 \right)}, 1.5 \right) \quad \min\left(\sqrt{\max\left(\frac{\hat{r}_{2,2} + r_{2,2} + \varepsilon^2}{r_{2,2} + \varepsilon^2}, 0 \right)}, 1.5 \right) \right),$$

and the mix matrix $\mathbf{G}_{\text{Mod}}$ is given as:

$$\mathbf{G}_{\text{Mod}} = \begin{cases} diag\left(\mathbf{g}_{\text{vec}}\right)\mathbf{G}, & r_{1,2} > 0, \\ \mathbf{G}, & otherwise. \end{cases}$$

Similarly, the mix matrix $\mathbf{P}_2$ is given as:

$$\mathbf{P}_2 = \begin{cases} \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} & , r_{1,2} > 0, \\ \mathbf{v}_{\text{R}} diag\left(\mathbf{W}_{\text{d}}\right) & , otherwise. \end{cases}$$

To derive $\mathbf{v}_{\text{R}}$ and $\mathbf{W}_{\text{d}}$, the characteristic equation of $\mathbf{R}$ needs to be solved:

$\det(\mathbf{R} - \lambda_{1,2}\mathbf{I}) = 0$, giving the eigenvalues, λ_1 and λ_2.

The corresponding eigenvectors $\mathbf{v}_{\text{R1}}$ and $\mathbf{v}_{\text{R2}}$ of $\mathbf{R}$ can be calculated solving the equation system:

$$(\mathbf{R} - \lambda_{1,2}\mathbf{I})\mathbf{v}_{\text{R1,R2}} = 0.$$

Eigenvalues are sorted in descending ($\lambda_1 \geq \lambda_2$) order and the eigenvector corresponding to the larger eigenvalue is calculated according to the equation above. It is assured to lie in the positive x-plane (first element has to be positive). The second eigenvector is obtained from the first by a –90 degrees rotation:

$$\mathbf{R} = \left(\mathbf{v}_{\text{R1}}\mathbf{v}_{\text{R2}}\right)\begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}\left(\mathbf{v}_{\text{R1}}\mathbf{v}_{\text{R2}}\right)^*.$$

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

Incorporating $\mathbf{P}_1 = (1 \quad 1)\mathbf{G}$, $\mathbf{R}_\text{d}$ can be calculated according to:

$$\mathbf{R}_\text{d} = \begin{pmatrix} \mathbf{r}_{\text{d}11} & \mathbf{r}_{\text{d}12} \\ \mathbf{r}_{\text{d}21} & \mathbf{r}_{\text{d}22} \end{pmatrix} = diag\left(\mathbf{P}_1\left(\mathbf{DED}^*\right)\mathbf{P}_1^*\right),$$

which gives

$$\mathbf{w}_{\text{d}1} = \min\left(\sqrt{\frac{\lambda_1}{r_{\text{d}1}+\varepsilon}}, 2\right), \quad \mathbf{w}_{\text{d}2} = \min\left(\sqrt{\frac{\lambda_2}{r_{\text{d}2}+\varepsilon}}, 2\right),$$

and finally the mix matrix,

$$\mathbf{P}_2 = \left(\mathbf{v}_{\text{R}1} \quad \mathbf{v}_{\text{R}2}\right)\begin{pmatrix} \mathbf{w}_{\text{d}1} & 0 \\ 0 & \mathbf{w}_{\text{d}2} \end{pmatrix}.$$

7.6.3.4 Decorrelation

The decorrelated signals $\mathbf{X}_\text{d}$ are created from the decorrelator described in 6.6.2 of ISO/IEC 23003-1:2007. Following this scheme, the bsDecorrConfig == 0 configuration should be used with a decorrelator index, $X = 8$, according to Table A.26 to Table A.29 in ISO/IEC 23003-1:2007. Hence, the $decorrFunc(\ )$ denotes the decorrelation process:

$$\mathbf{X}_\text{d} = \begin{pmatrix} x_{1d} \\ x_{2d} \end{pmatrix} = \begin{pmatrix} decorrFunc\left((1 \quad 0)\mathbf{P}_1\mathbf{X}\right) \\ decorrFunc\left((0 \quad 1)\mathbf{P}_1\mathbf{X}\right) \end{pmatrix}.$$

7.6.3.5 Dual mode

The SAOC transcoder can let the mix matrices $\mathbf{P}_1$, $\mathbf{P}_2$ and the prediction matrix $\mathbf{C}_3$ be calculated according to an alternative scheme for the upper frequency range. This alternative scheme is particularly useful for downmix signals where the upper frequency range is coded by a non-waveform preserving coding algorithm e.g. SBR in High Efficiency AAC.

For the upper parameter bands, defined by $\mathbf{bsTttBandsLow} \le pb < numBands$, $\mathbf{P}_1$, $\mathbf{P}_2$ and $\mathbf{C}_3$ should be calculated according to the alternative scheme described below:

$$\begin{cases} \mathbf{P}_1 = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}, \\ \mathbf{P}_2 = \mathbf{G}. \end{cases}$$

Define the energy downmix and energy target vectors, respectively:

$$\begin{cases} \mathbf{e}_\text{dmx} = \begin{pmatrix} e_\text{dmx1} \\ e_\text{dmx2} \end{pmatrix} = diag\left(\mathbf{DED}^*\right) + \varepsilon\mathbf{I}, \\ \mathbf{e}_\text{tar} = \begin{pmatrix} e_\text{tar1} \\ e_\text{tar2} \\ e_\text{tar3} \end{pmatrix} = diag\left(\mathbf{A}_3\mathbf{E}\mathbf{A}_3^*\right), \end{cases}$$

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

and the help matrix

$$\mathbf{T} = \begin{pmatrix} t_{1,1} & t_{1,2} \\ t_{2,1} & t_{2,2} \\ t_{3,1} & t_{3,2} \end{pmatrix} = \mathbf{A}_3\mathbf{D}^* + \varepsilon\mathbf{I} \,.$$

Then calculate the gain vector

$$\mathbf{g} = \begin{pmatrix} g_1 \\ g_2 \\ g_3 \end{pmatrix} = \begin{pmatrix} \sqrt{\dfrac{e_{\text{tar1}}}{t_{1,1}^2 e_{\text{dmx1}} + t_{1,2}^2 e_{\text{dmx2}}}} \\ \sqrt{\dfrac{e_{\text{tar2}}}{t_{2,1}^2 e_{\text{dmx1}} + t_{2,2}^2 e_{\text{dmx2}}}} \\ \sqrt{\dfrac{e_{\text{tar3}}}{t_{3,1}^2 e_{\text{dmx1}} + t_{3,2}^2 e_{\text{dmx2}}}} \end{pmatrix},$$

which finally gives the new prediction matrix

$$\mathbf{C}_3 = \begin{pmatrix} g_1 t_{1,1} & g_1 t_{1,2} \\ g_2 t_{2,1} & g_2 t_{2,2} \\ g_3 t_{3,1} & g_3 t_{3,2} \end{pmatrix}.$$

7.7 Decoding modes

7.7.1 Introduction

In this Subclause the method for obtaining an output signal using SAOC parameters and panning information associated with each audio object is explained. The SAOC decoder is depicted in Figure 2 (right) and consists of the SAOC parameter processor and downmix processor.

7.7.2 Downmix processor

For the decoder mode of the SAOC system, the output signal of the downmix processor (represented in the hybrid QMF domain) is fed into the corresponding synthesis filterbank as described in ISO/IEC 23003-1:2007 yielding the final output PCM signal. The downmix processing incorporates the mono, stereo and, if required, subsequent binaural processing.

The output signal $\hat{\mathbf{X}}$ is computed from the mono downmix signal $\mathbf{X}$ and the decorrelated mono downmix signal $\mathbf{X}_\text{d}$ as

$$\hat{\mathbf{X}} = \mathbf{G}\mathbf{X} + \mathbf{P}_2\mathbf{X}_\text{d} \,.$$

The decorrelated mono downmix signal $\mathbf{X}_\text{d}$ is computed according to 7.6.3.4 as

$$\mathbf{X}_\text{d} = decorrFunc(\mathbf{X}) \,.$$

In case of binaural output the upmix parameters $\mathbf{G}$ and $\mathbf{P}_2$ derived from the SAOC data, rendering information $\mathbf{M}_{\text{ren}}^{l,m}$ and HRTF parameters are applied to the downmix signal $\mathbf{X}$ (and $\mathbf{X}_\text{d}$) yielding the binaural output $\hat{\mathbf{X}}$, see Figure 15.

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

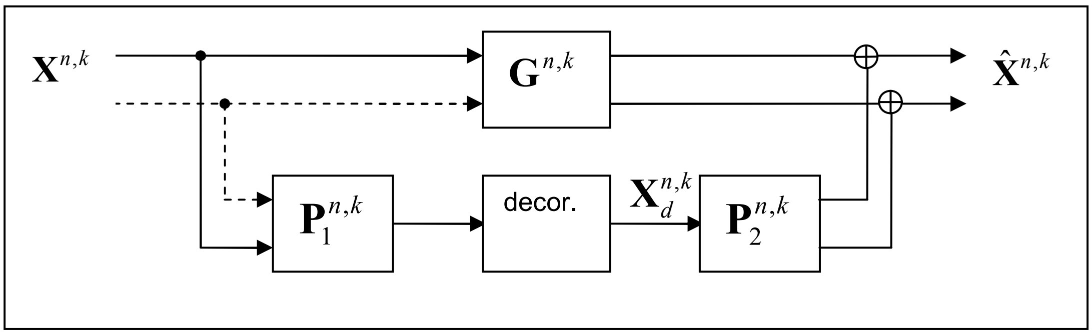


Figure 15 – Basic structure of the downmix processor

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

The target binaural rendering matrix $\mathbf{A}^{l,m}$ of size $2 \times N$ consists of the elements $a_{x,y}^{l,m}$. Each element $a_{x,y}^{l,m}$ is derived from HRTF parameters and rendering matrix $\mathbf{M}_{\text{ren}}^{l,m}$ with elements $m_{y,i}^{l,m}$. The target binaural rendering matrix $\mathbf{A}^{l,m}$ represents the relation between all audio input objects y and the desired binaural output.

$$a_{y,1}^{l,m} = \sum_{i=0}^{N_{\text{HRTF}}-1} m_{y,i}^{l,m} H_{i,\text{L}}^{m} \exp\left(j \frac{\phi_i^m}{2} \right), \qquad a_{y,2}^{l,m} = \sum_{i=0}^{N_{\text{HRTF}}-1} m_{y,i}^{l,m} H_{i,\text{R}}^{m} \exp\left(-j \frac{\phi_i^m}{2} \right).$$

The HRTF parameters are given by $H_{i,L}^{m}$, $H_{i,R}^{m}$ and ϕ_i^m for each processing band m. The spatial positions for which HRTF parameters are available are characterized by the index i. These parameters are described in ISO/IEC 23003-1:2007.

7.7.2.1 Mono to binaural "x-1-b" processing mode

The upmix parameters $\mathbf{G}^{l,m}$ and $\mathbf{P}_2^{l,m}$ are computed as

$$\mathbf{G}^{l,m} = \begin{pmatrix} P_L^{l,m} \exp\left(j \frac{\phi_C^{l,m}}{2} \right) \cos\left(\beta^{l,m} + \alpha^{l,m} \right) \\ P_R^{l,m} \exp\left(-j \frac{\phi_C^{l,m}}{2} \right) \cos\left(\beta^{l,m} - \alpha^{l,m} \right) \end{pmatrix},$$

$$\mathbf{P}_2^{l,m} = \begin{pmatrix} P_L^{l,m} \exp\left(j \frac{\phi_C^{l,m}}{2} \right) \sin\left(\beta^{l,m} + \alpha^{l,m} \right) \\ P_R^{l,m} \exp\left(-j \frac{\phi_C^{l,m}}{2} \right) \sin\left(\beta^{l,m} - \alpha^{l,m} \right) \end{pmatrix}.$$

The gains $P_L^{l,m}$ and $P_R^{l,m}$ for the left and right output channels are

$$P_L^{l,m} = \sqrt{\max\left(\frac{f_{1,1}^{l,m}}{v^{l,m}}, \varepsilon^2 \right)}, \qquad P_R^{l,m} = \sqrt{\max\left(\frac{f_{2,2}^{l,m}}{v^{l,m}}, \varepsilon^2 \right)}.$$

The desired covariance matrix $\mathbf{F}^{l,m}$ of size 2×2 with elements $f_{i,j}^{l,m}$ is given as

$$\mathbf{F}^{l,m} = \mathbf{A}^{l,m} \mathbf{E}^{l,m} \left(\mathbf{A}^{l,m} \right)^{*}.$$

© ISO/IEC 2010 – All rights reserved

The scalar $v^{l,m}$ is computed as

$$v^{l,m} = \mathbf{D}^l \mathbf{E}^{l,m} \left(\mathbf{D}^l\right)^* + \varepsilon^2 .$$

The inter channel phase difference $\phi_C^{l,m}$ is given as

$$\phi_C^{l,m} = \begin{cases} \arg\left(f_{1,2}^{l,m}\right), & 0 \le m \le 11, \quad \rho_C^{l,m} \ge 0.6, \\ 0, & otherwise. \end{cases}$$

The inter channel coherence $\rho_C^{l,m}$ is computed as

$$\rho_C^{l,m} = \min\left(\frac{\left|f_{1,2}^{l,m}\right|}{\sqrt{\max\left(f_{1,1}^{l,m} f_{2,2}^{l,m}, \varepsilon^2\right)}}, 1 \right).$$

The rotation angles $\alpha^{l,m}$ and $\beta^{l,m}$ are given as

$$\alpha^{l,m} = \begin{cases} \frac{1}{2}\arccos\left(\rho_C^{l,m} \cos\left(\arg\left(f_{1,2}^{l,m}\right)\right)\right), & 0 \le m \le 11, \quad \rho_C^{l,m} < 0.6, \\ \frac{1}{2}\arccos\left(\rho_C^{l,m}\right), & otherwise. \end{cases}$$

$$\beta^{l,m} = \arctan\left(\tan\left(\alpha^{l,m}\right) \frac{P_R^{l,m} - P_L^{l,m}}{P_L^{l,m} + P_R^{l,m} + \varepsilon} \right).$$

7.7.2.2 Mono to stereo "x-1-2" processing mode

In case of stereo output the "x-1-b" processing mode can be applied without using HRTF information. This can be done by deriving all elements $a_{x,y}^{l,m}$ of the rendering matrix $\mathbf{A}$, yielding:

$$a_{1,y}^{l,m} = m_{Lf,y}^{l,m}, \qquad a_{2,y}^{l,m} = m_{Rf,y}^{l,m}.$$

7.7.2.3 Mono to mono "x-1-1" processing mode

In case of mono output the "x-1-2" processing mode can be applied with the following entries:

$$a_{1,y}^{l,m} = m_{C,y}^{l,m}, \qquad a_{2,y}^{l,m} = 0.$$

7.7.2.4 Stereo to binaural "x-2-b" processing mode

The upmix parameters $\mathbf{G}^{l,m}$ and $\mathbf{P}_2^{l,m}$ are computed as

$$\mathbf{G}^{l,m} = \begin{pmatrix} P_L^{l,m,1} \exp\left(j\frac{\phi^{l,m,1}}{2}\right)\cos\left(\beta^{l,m}+\alpha^{l,m}\right) & P_L^{l,m,2} \exp\left(j\frac{\phi^{l,m,2}}{2}\right)\cos\left(\beta^{l,m}+\alpha^{l,m}\right) \\ P_R^{l,m,1} \exp\left(-j\frac{\phi^{l,m,1}}{2}\right)\cos\left(\beta^{l,m}-\alpha^{l,m}\right) & P_R^{l,m,2} \exp\left(-j\frac{\phi^{l,m,2}}{2}\right)\cos\left(\beta^{l,m}-\alpha^{l,m}\right) \end{pmatrix},$$

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

$$\mathbf{P}_2^{l,m} = \begin{pmatrix} P_L^{l,m} \exp\left(j\frac{\arg\left(c_{1,2}^{l,m}\right)}{2}\right)\sin\left(\beta^{l,m}+\alpha^{l,m}\right) \\ P_R^{l,m} \exp\left(-j\frac{\arg\left(c_{1,2}^{l,m}\right)}{2}\right)\sin\left(\beta^{l,m}-\alpha^{l,m}\right) \end{pmatrix}.$$

The corresponding gains $P_L^{l,m,x}$, $P_R^{l,m,x}$ and $P_L^{l,m}$, $P_R^{l,m}$ for the left and right output channels are

$$P_L^{l,m,x} = \sqrt{\max\left(\frac{f_{1,1}^{l,m,x}}{v^{l,m,x}},\varepsilon^2\right)},\quad P_R^{l,m,x} = \sqrt{\max\left(\frac{f_{2,2}^{l,m,x}}{v^{l,m,x}},\varepsilon^2\right)},$$

$$P_L^{l,m} = \sqrt{\max\left(\frac{c_{1,1}^{l,m}}{v^{l,m}},\varepsilon^2\right)},\ P_R^{l,m} = \sqrt{\max\left(\frac{c_{2,2}^{l,m}}{v^{l,m}},\varepsilon^2\right)}.$$

The desired covariance matrix $\mathbf{F}^{l,m,x}$ of size 2×2 with elements $f_{u,v}^{l,m,x}$ is given as

$$\mathbf{F}^{l,m,x} = \mathbf{A}^{l,m}\mathbf{E}^{l,m,x}\left(\mathbf{A}^{l,m}\right)^*.$$

The covariance matrix $\mathbf{C}^{l,m}$ of size 2×2 with elements $c_{u,v}^{l,m}$ of the “dry” binaural signal is estimated as

$$\mathbf{C}^{l,m} = \tilde{\mathbf{G}}^{l,m}\mathbf{D}^l\mathbf{E}^{l,m}\left(\mathbf{D}^l\right)^*\left(\tilde{\mathbf{G}}^{l,m}\right)^*,$$

where

$$\tilde{\mathbf{G}}^{l,m} = \begin{pmatrix} P_L^{l,m,1}\exp\left(j\frac{\phi^{l,m,1}}{2}\right) & P_L^{l,m,2}\exp\left(j\frac{\phi^{l,m,2}}{2}\right) \\ P_R^{l,m,1}\exp\left(-j\frac{\phi^{l,m,1}}{2}\right) & P_R^{l,m,2}\exp\left(-j\frac{\phi^{l,m,2}}{2}\right) \end{pmatrix}.$$

The corresponding scalars $v^{l,m,x}$ and $v^{l,m}$ are computed as

$$v^{l,m,x} = \mathbf{D}^{l,x}\mathbf{E}^{l,m}\left(\mathbf{D}^{l,x}\right)^* + \varepsilon^2,\quad v^{l,m} = \left(\mathbf{D}^{l,1}+\mathbf{D}^{l,2}\right)\mathbf{E}^{l,m}\left(\mathbf{D}^{l,1}+\mathbf{D}^{l,2}\right)^* + \varepsilon^2.$$

The downmix matrix $\mathbf{D}^{l,x}$ of size $1\times N$ with elements $d_i^{l,x}$ can be found as

$$d_i^{l,1} = 10^{0.05DMG_i^l}\sqrt{\frac{10^{0.1DCLD_i^l}}{1+10^{0.1DCLD_i^l}}},\qquad d_i^{l,2} = 10^{0.05DMG_i^l}\sqrt{\frac{1}{1+10^{0.1DCLD_i^l}}}.$$

The stereo downmix matrix $\mathbf{D}^l$ of size $2\times N$ with elements $d_{x,i}^l$ can be found as

$$d_{x,i}^l = d_i^{l,x}.$$

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

The matrix $\mathbf{E}^{l,m,x}$ with elements $e_{i,j}^{l,m,x}$ are derived from the following relationship

$$e_{i,j}^{l,m,x} = e_{i,j}^{l,m} \left(\frac{d_i^{l,x}}{d_i^{l,1} + d_i^{l,2}} \right) \left(\frac{d_j^{l,x}}{d_j^{l,1} + d_j^{l,2}} \right).$$

The inter channel phase differences $\phi_C^{l,m}$ are given as

$$\phi^{l,m,x} = \begin{cases} \arg\left(f_{1,2}^{l,m,x} \right), & 0 \le m \le 11, \quad \rho_C^{l,m} > 0.6, \\ 0, & otherwise. \end{cases}$$

The ICCs $\rho_C^{l,m}$ and $\rho_T^{l,m}$ are computed as

$$\rho_T^{l,m} = \min \left(\frac{\left| f_{1,2}^{l,m} \right|}{\sqrt{\max\left(f_{1,1}^{l,m} f_{2,2}^{l,m}, \varepsilon^2 \right)}}, 1 \right), \quad \rho_C^{l,m} = \min \left(\frac{\left| c_{1,2}^{l,m} \right|}{\sqrt{\max\left(c_{1,1}^{l,m} c_{2,2}^{l,m}, \varepsilon^2 \right)}}, 1 \right).$$

The rotation angles $\alpha^{l,m}$ and $\beta^{l,m}$ are given as

$$\alpha^{l,m} = \frac{1}{2}\left(\arccos\left(\rho_T^{l,m} \right) - \arccos\left(\rho_C^{l,m} \right) \right), \quad \beta^{l,m} = \arctan\left(\tan(\alpha^{l,m}) \frac{P_R^{l,m} - P_L^{l,m}}{P_L^{l,m} + P_R^{l,m}} \right).$$

7.7.2.5 Stereo to stereo "x-2-2" processing mode

In case of stereo output the stereo preprocessing is directly applied as described in 7.6.3.3.

7.7.2.6 Stereo to mono "x-2-1" processing mode

In case of mono output the stereo preprocessing is applied with a single active rendering matrix entry as described in 7.6.3.3.

7.8 EAO processing

7.8.1 Introduction

The SAOC technology allows the individual manipulation of a number of audio objects in terms of their level amplification/attenuation without significant decrease in the resulting sound quality only in a very limited way. A special "Karaoke-type" application scenario requires a total suppression of the specific objects, typically the lead vocal, keeping the perceptual quality of the background sound scene unharmed. A typical application case contains up to four Enhanced Audio Object (EAO) signals, which can, for example, represent two independent stereo objects.

7.8.2 SAOC architecture supporting EAO

The EAO processing incorporates the OTN or TTN units, depending on the SAOC downmix mode. The OTN processing unit is dedicated to a mono and the TTN to a stereo downmix signal. Both these units represent a generalized and enhanced modification of the TTT box known from ISO/IEC 23003-1:2007. In the encoder, regular and EAO signals are combined into the downmix. The OTN^{-1} /TTN^{-1} processing units are employed to produce and encode the corresponding residual signals. The EAO and regular signals are recovered from the downmix by the OTN/TTN units using the SAOC side information and incorporated residual signals. The recovered EAOs are fed into the rendering unit which represents the product of the corresponding rendering matrix and the resulting output of the OTN/TTN unit. The regular audio objects are delivered to the SAOC downmix pre-processor for further processing. Figure 16 depicts the general structure of the residual processor.

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

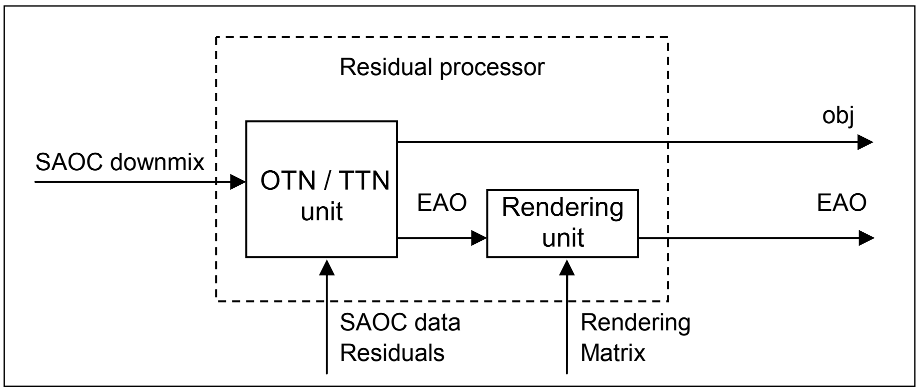


Figure 16 – Architecture of the residual processor

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

An MBO is treated the same way as explained in 7.10.

7.8.3 Dimensionality of signals and parameters

The audio signals are defined for every time slot n and every hybrid subband k. The corresponding SAOC parameters are defined for each parameter time slot l and processing band m. The subsequent mapping between the hybrid and parameter domain is specified by Table A.31, ISO/IEC 23003-1:2007. Hence, all calculations are performed with respect to the certain time/band indices and the corresponding dimensionalities are implied for each introduced variable.

7.8.4 Calculation of the OTN/TTN elements

The OTN/TTN upmix process is represented either by matrix $\mathbf{M}$ for the prediction mode or $\mathbf{M}_{Energy}$ for the energy mode. In the first case $\mathbf{M}$ is the product of two matrices exploiting the downmix information and the CPCs for each EAO channel. It is expressed in parameter-domain by

$$\mathbf{M} = \mathbf{A}\tilde{\mathbf{D}}^{-1}\mathbf{C},$$

where $\tilde{\mathbf{D}}^{-1}$ is the inverse of the extended downmix matrix $\tilde{\mathbf{D}}$ and $\mathbf{C}$ implies the CPCs, which can be calculated as

$$\tilde{\mathbf{D}}^{-1} = \frac{\tilde{d}_{i,j}}{den}.$$

The matrix $\mathbf{A}$ is defined in 7.7. The elements of the inverse of the extended downmix matrix $\tilde{\mathbf{D}}^{-1}$ of size 6×6 are derived using the following values:

$$\tilde{d}_{1,1} = 1 + \sum_{j=1}^{4} n_j^2,$$

$$\tilde{d}_{1,2} = -\left(\sum_{j=1}^{4} m_j n_j\right),$$

$$\tilde{d}_{1,3} = m_1 + m_1 n_2^2 + m_1 n_3^2 + m_1 n_4^2 - m_2 n_1 n_2 - m_3 n_1 n_3 - m_4 n_1 n_4,$$

$$\tilde{d}_{1,4} = m_2 + m_2 n_1^2 + m_2 n_3^2 + m_2 n_4^2 - m_1 n_2 n_1 - m_3 n_2 n_3 - m_4 n_2 n_4,$$

© ISO/IEC 2010 – All rights reserved

$$\tilde{d}_{1,5} = m_3 + m_3 n_1^2 + m_3 n_2^2 + m_3 n_4^2 - m_1 n_3 n_1 - m_2 n_3 n_2 - m_4 n_3 n_4 \,,$$

$$\tilde{d}_{1,6} = m_4 + m_4 n_1^2 + m_4 n_2^2 + m_4 n_3^2 - m_1 n_4 n_1 - m_2 n_4 n_2 - m_3 n_4 n_3 \,,$$

$$\tilde{d}_{2,2} = 1 + \sum_{j=1}^{4} m_j^2 \,,$$

$$\tilde{d}_{2,3} = n_1 + n_1 m_2^2 + n_1 m_3^2 + n_1 m_4^2 - m_1 m_2 n_2 - m_1 m_3 n_3 - m_1 m_4 n_4 \,,$$

$$\tilde{d}_{2,4} = n_2 + n_2 m_1^2 + n_2 m_3^2 + n_2 m_4^2 - m_2 m_1 n_1 - m_2 m_3 n_3 - m_2 m_4 n_4 \,,$$

$$\tilde{d}_{2,5} = n_3 + n_3 m_1^2 + n_3 m_2^2 + n_3 m_4^2 - m_3 m_1 n_1 - m_3 m_2 n_2 - m_3 m_4 n_4 \,,$$

$$\tilde{d}_{2,6} = n_4 + n_4 m_1^2 + n_4 m_2^2 + n_4 m_3^2 - m_4 m_1 n_1 - m_4 m_2 n_2 - m_4 m_3 n_3 \,,$$

$$\tilde{d}_{3,3} = -1 - \sum_{j=2}^{4} m_j^2 - \sum_{j=2}^{4} n_j^2 - m_3^2 n_2^2 - m_4^2 n_2^2 - m_2^2 n_3^2 - m_4^2 n_3^2 - m_2^2 n_4^2 - m_3^2 n_4^2 + 2 m_2 m_3 n_2 n_3 + 2 m_2 m_4 n_2 n_4 + 2 m_3 m_4 n_3 n_4 \quad ,$$

$$\tilde{d}_{3,4} = m_1 m_2 + n_1 n_2 + m_3^2 n_1 n_2 + m_4^2 n_1 n_2 + m_1 m_2 n_3^2 + m_1 m_2 n_4^2 - m_2 m_3 n_1 n_3 - m_1 m_3 n_2 n_3 - m_2 m_4 n_1 n_4 - m_1 m_4 n_2 n_4 \,,$$

$$\tilde{d}_{3,5} = m_1 m_3 + n_1 n_3 + m_2^2 n_1 n_3 + m_4^2 n_1 n_3 + m_1 m_3 n_2^2 + m_1 m_3 n_4^2 - m_2 m_3 n_1 n_2 - m_1 m_2 n_2 n_3 - m_3 m_4 n_1 n_4 - m_1 m_4 n_3 n_4 \,,$$

$$\tilde{d}_{3,6} = m_1 m_4 + n_1 n_4 + m_2^2 n_1 n_4 + m_3^2 n_1 n_4 + m_1 m_4 n_2^2 + m_1 m_4 n_3^2 - m_2 m_4 n_1 n_2 - m_3 m_4 n_1 n_3 - m_1 m_2 n_2 n_4 - m_1 m_3 n_4 n_3 \,,$$

$$\tilde{d}_{4,4} = -1 - \sum_{\substack{j=1 \\ j \neq 2}}^{4} m_j^2 - \sum_{\substack{j=1 \\ j \neq 2}}^{4} n_j^2 - m_3^2 n_1^2 - m_4^2 n_1^2 - m_1^2 n_3^2 - m_4^2 n_3^2 - m_1^2 n_4^2 - m_3^2 n_4^2 + 2 m_1 m_3 n_1 n_3 + 2 m_1 m_4 n_1 n_4 + 2 m_3 m_4 n_3 n_4 \,,$$

$$\tilde{d}_{4,5} = m_2 m_3 + n_2 n_3 + m_1^2 n_2 n_3 + m_4^2 n_2 n_3 + m_2 m_3 n_1^2 + m_2 m_3 n_4^2 - m_1 m_3 n_1 n_2 - m_1 m_2 n_1 n_3 - m_3 m_4 n_2 n_4 - m_2 m_4 n_3 n_4 \,,$$

$$\tilde{d}_{4,6} = m_2 m_4 + n_2 n_4 + m_1^2 n_2 n_4 + m_3^2 n_2 n_4 + m_2 m_4 n_1^2 + m_2 m_4 n_3^2 - m_1 m_4 n_1 n_2 - m_3 m_4 n_2 n_3 - m_1 m_2 n_1 n_4 - m_2 m_3 n_3 n_4 \,,$$

$$\tilde{d}_{5,5} = -1 - \sum_{\substack{j=1 \\ j \neq 3}}^{4} m_j^2 - \sum_{\substack{j=1 \\ j \neq 3}}^{4} n_j^2 - m_2^2 n_1^2 - m_4^2 n_1^2 - m_1^2 n_2^2 - m_4^2 n_2^2 - m_1^2 n_4^2 - m_2^2 n_4^2 + 2 m_1 m_2 n_1 n_2 + 2 m_1 m_4 n_1 n_4 + 2 m_2 m_4 n_2 n_4 \,,$$

$$\tilde{d}_{5,6} = m_3 m_4 + n_3 n_4 + m_1^2 n_3 n_4 + m_2^2 n_3 n_4 + m_3 m_4 n_1^2 + m_3 m_4 n_2^2 - m_1 m_4 n_1 n_3 - m_2 m_4 n_2 n_3 - m_1 m_3 n_1 n_4 - m_2 m_3 n_2 n_4 \,,$$

$$\tilde{d}_{6,6} = -1 - \sum_{j=1}^{3} m_j^2 - \sum_{j=1}^{3} n_j^2 - m_2^2 n_1^2 - m_3^2 n_1^2 - m_1^2 n_2^2 - m_3^2 n_2^2 - m_1^2 n_3^2 - m_2^2 n_3^2 + 2 m_1 m_2 n_1 n_2 + 2 m_1 m_3 n_1 n_3 + 2 m_2 m_3 n_2 n_3 \,,$$

$$den = 1 + \sum_{j=1}^{4} m_j^2 + \sum_{j=1}^{4} n_j^2 + m_2^2 n_1^2 + m_3^2 n_1^2 + m_4^2 n_1^2 + m_1^2 n_2^2 + m_3^2 n_2^2 + m_4^2 n_2^2 + m_1^2 n_3^2 + m_2^2 n_3^2 + m_4^2 n_3^2 + m_1^2 n_4^2 + m_2^2 n_4^2 +$$
$$+ m_3^2 n_4^2 - 2 m_1 m_2 n_1 n_2 - 2 m_1 m_3 n_1 n_3 - 2 m_2 m_3 n_2 n_3 - 2 m_1 m_4 n_1 n_4 - 2 m_2 m_4 n_2 n_4 - 2 m_3 m_4 n_3 n_4 .$$

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

The coefficients m_j and n_j of the extended downmix matrix $\tilde{\mathbf{D}}$ denote the downmix values for every EAO j for the right and left downmix channel as

$$m_j = d_{1,EAO(j)}, \quad n_j = d_{2,EAO(j)}.$$

The elements $d_{i,j}$ of the downmix matrix $\mathbf{D}$ are obtained as described in 6.5.3.2.

The function $EAO(j)$ determines mapping between indices of input audio object channels and EAO signals:

$$EAO(j) = N - 1 - j, \qquad j = 0, \ldots, N_{EAO} - 1.$$

The energy based encoding/decoding procedure is designed for non-waveform preserving coding of the downmix signal. Thus the OTN/TTN upmix matrix for the corresponding energy mode does not rely on specific waveforms, but only describe the relative energy distribution of the input audio objects.

7.8.4.1 TTN matrix using prediction mode

In case of a stereo, the extended downmix matrix $\tilde{\mathbf{D}}$ is

$$\tilde{\mathbf{D}} = \left(\begin{array}{cc:ccc} 1 & 0 & m_0 & \cdots & m_{N_{EAO}-1} \\ 0 & 1 & n_0 & \cdots & n_{N_{EAO}-1} \\ \hdashline m_0 & n_0 & -1 & \cdots & 0 \\ \vdots & \vdots & 0 & \ddots & \vdots \\ m_{N_{EAO}-1} & n_{N_{EAO}-1} & 0 & \cdots & -1 \end{array} \right),$$

and for a mono, it becomes

$$\tilde{\mathbf{D}} = \left(\begin{array}{c:ccc} 1 & m_0 & \cdots & m_{N_{EAO}-1} \\ 1 & n_0 & \cdots & n_{N_{EAO}-1} \\ \hdashline m_0 + n_0 & -1 & \cdots & 0 \\ \vdots & 0 & \ddots & \vdots \\ m_{N_{EAO}-1} + n_{N_{EAO}-1} & 0 & \cdots & -1 \end{array} \right).$$

With a stereo downmix, each EAO j holds two CPCs $c_{j,0}$ and $c_{j,1}$ yielding matrix $\mathbf{C}$

$$\mathbf{C} = \left(\begin{array}{cc:ccc} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ \hdashline c_{0,0} & c_{0,1} & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ c_{N_{EAO}-1,0} & c_{N_{EAO}-1,1} & 0 & \cdots & 1 \end{array} \right).$$

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

The CPCs are derived from the transmitted SAOC parameters, i.e. the OLDs, IOCs, DMGs and DCLDs. For one specific EAO channel $j = 0 \ldots N_{EAO} - 1$ the CPCs can be estimated by

$$\tilde{c}_{j,0} = \frac{P_{LoCo,j} P_{Ro} - P_{RoCo,j} P_{LoRo}}{P_{Lo} P_{Ro} - P_{LoRo}^2}, \qquad \tilde{c}_{j,1} = \frac{P_{RoCo,j} P_{Lo} - P_{LoCo,j} P_{LoRo}}{P_{Lo} P_{Ro} - P_{LoRo}^2}.$$

The energy quantities P_{Lo}, P_{Ro}, P_{LoRo}, $P_{LoCo,j}$ and $P_{RoCo,j}$ are computed as

$$P_{Lo} = OLD_L + \sum_{j=0}^{N_{EAO}-1} \sum_{k=0}^{N_{EAO}-1} m_j m_k e_{j,k},$$

$$P_{Ro} = OLD_R + \sum_{j=0}^{N_{EAO}-1} \sum_{k=0}^{N_{EAO}-1} n_j n_k e_{j,k},$$

$$P_{LoRo} = e_{L,R} + \sum_{j=0}^{N_{EAO}-1} \sum_{k=0}^{N_{EAO}-1} m_j n_k e_{j,k},$$

$$P_{LoCo,j} = m_j OLD_L + n_j e_{L,R} - m_j OLD_j - \sum_{\substack{i=0 \\ i \neq j}}^{N_{EAO}-1} m_i e_{i,j},$$

$$P_{RoCo,j} = n_j OLD_R + m_j e_{L,R} - n_j OLD_j - \sum_{\substack{i=0 \\ i \neq j}}^{N_{EAO}-1} n_i e_{i,j}.$$

The parameters OLD_L, OLD_R and IOC_{LR} correspond to the regular objects and can be derived using downmix information:

$$OLD_L = \sum_{i=0}^{N-N_{EAO}-1} d_{0,i}^2 OLD_i,$$
$$OLD_R = \sum_{i=0}^{N-N_{EAO}-1} d_{1,i}^2 OLD_i,$$
$$IOC_{L,R} = \begin{cases} IOC_{0,1}, & N - N_{EAO} = 2, \\ 0, & otherwise. \end{cases}$$

The covariance matrix $e_{i,j}$ (and $e_{L,R}$) is defined in 7.5.3.

The CPCs are constrained by the subsequent limiting functions:

$$\gamma_{j,1} = \frac{m_j OLD_L + n_j e_{L,R} - \sum_{i=0}^{N_{EAO}-1} m_i e_{i,j}}{2\left(OLD_L + \sum_{i=0}^{N_{EAO}-1} \sum_{k=0}^{N_{EAO}-1} m_i m_k e_{i,k}\right)}, \qquad \gamma_{j,2} = \frac{n_j OLD_R + m_j e_{L,R} - \sum_{i=0}^{N_{EAO}-1} n_i e_{i,j}}{2\left(OLD_R + \sum_{i=0}^{N_{EAO}-1} \sum_{k=0}^{N_{EAO}-1} n_i n_k e_{i,k}\right)}.$$

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

With the weighting factor

$$\lambda = \left(\frac{P_{LoRo}^2}{P_{Lo} P_{Ro}} \right)^8 .$$

The constrained CPCs become

$$c_{j,0} = (1-\lambda)\tilde{c}_{j,0} + \lambda\gamma_{j,0}, \qquad c_{j,1} = (1-\lambda)\tilde{c}_{j,1} + \lambda\gamma_{j,1} .$$

The output of the TTN element yields

$$\mathbf{Y} = \left(\begin{array}{c} \mathbf{y}_L \\ \mathbf{y}_R \\ \hline \mathbf{y}_{0,EAO} \\ \vdots \\ \mathbf{y}_{N_{EAO}-1,EAO} \end{array} \right) = \mathbf{MX} = \mathbf{A}\tilde{\mathbf{D}}^{-1}\mathbf{C} \left(\begin{array}{c} \mathbf{l}_0 \\ \mathbf{r}_0 \\ \hline \mathbf{res}_0 \\ \vdots \\ \mathbf{res}_{N_{EAO}-1} \end{array} \right),$$

where $\mathbf{X}$ represents the input signal to the SAOC decoder/transcoder.

7.8.4.2 OTN matrix using prediction mode

In case of a stereo, the extended downmix matrix $\tilde{\mathbf{D}}$ matrix is

$$\tilde{\mathbf{D}} = \left(\begin{array}{cc|ccc} 1 & 1 & m_0 & \dots & m_{N_{EAO}-1} \\ \hline m_0/2 & m_0/2 & -1 & \dots & 0 \\ \vdots & \vdots & 0 & \ddots & \vdots \\ m_{N_{EAO}-1}/2 & m_{N_{EAO}-1}/2 & 0 & \dots & -1 \end{array} \right),$$

and for a mono, it becomes

$$\tilde{\mathbf{D}} = \left(\begin{array}{c|ccc} 1 & m_0 & \dots & m_{N_{EAO}-1} \\ \hline m_0 & -1 & \dots & 0 \\ \vdots & 0 & \ddots & \vdots \\ m_{N_{EAO}-1} & 0 & \dots & -1 \end{array} \right).$$

With a mono downmix, one EAO j is predicted by only one coefficient c_j yielding

$$\mathbf{C} = \left(\begin{array}{c|ccc} 1 & 0 & \dots & 0 \\ \hline c_0 & 1 & \dots & 0 \\ \vdots & 0 & \ddots & \vdots \\ c_{N_{EAO}-1} & 0 & \dots & 1 \end{array} \right).$$

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

All matrix elements c_j are obtained from the SAOC parameters according to the relationships provided in the previous subclause.

For the mono downmix case the output signal $\mathbf{Y}$ of the OTN element yields

$$\mathbf{Y} = \mathbf{M}\begin{pmatrix} \mathbf{d}_0 \\ \hdashline \mathbf{res}_0 \\ \vdots \\ \mathbf{res}_{N_{EAO}-1} \end{pmatrix}.$$

7.8.4.3 TTN matrix using energy mode

In case of a stereo, the matrix $\mathbf{M}_{\text{Energy}}$ are obtained from the corresponding OLDs according to

$$\mathbf{M}_{\text{Energy}} = \mathbf{A}\begin{pmatrix} \sqrt{\dfrac{OLD_L}{OLD_L + \sum\limits_{i=0}^{N_{EAO}-1} m_i^2 OLD_i}} & 0 \\ 0 & \sqrt{\dfrac{OLD_R}{OLD_R + \sum\limits_{i=0}^{N_{EAO}-1} n_i^2 OLD_i}} \\ \hdashline \sqrt{\dfrac{m_0^2 OLD_0}{OLD_L + \sum\limits_{i=0}^{N_{EAO}-1} m_i^2 OLD_i}} & \sqrt{\dfrac{n_0^2 OLD_0}{OLD_R + \sum\limits_{i=0}^{N_{EAO}-1} n_i^2 OLD_i}} \\ \vdots & \vdots \\ \sqrt{\dfrac{m_{N_{EAO}-1}^2 OLD_{N_{EAO}-1}}{OLD_L + \sum\limits_{i=0}^{N_{EAO}-1} m_i^2 OLD_i}} & \sqrt{\dfrac{n_{N_{EAO}-1}^2 OLD_{N_{EAO}-1}}{OLD_R + \sum\limits_{i=0}^{N_{EAO}-1} n_i^2 OLD_i}} \end{pmatrix}.$$

The output of the TTN element yields

$$\mathbf{Y} = \begin{pmatrix} \mathbf{y}_L \\ \mathbf{y}_R \\ \hdashline \mathbf{y}_{0,EAO} \\ \vdots \\ \mathbf{y}_{N_{EAO}-1,EAO} \end{pmatrix} = \mathbf{M}_{\text{Energy}}\mathbf{X} = \mathbf{M}_{\text{Energy}}\begin{pmatrix} \mathbf{l}_0 \\ \mathbf{r}_0 \end{pmatrix}.$$

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

The adaptation of the equations for the mono signal results in

$$\mathbf{M}_{\text{Energy}} = \mathbf{A}\begin{pmatrix} \sqrt{\dfrac{OLD_L}{OLD_L + \sum_{i=0}^{N_{EAO}-1} m_i^2 OLD_i}} & \sqrt{\dfrac{OLD_L}{OLD_L + \sum_{i=0}^{N_{EAO}-1} n_i^2 OLD_i}} \\ \hdashline \sqrt{\dfrac{m_0^2 OLD_0}{OLD_L + \sum_{i=0}^{N_{EAO}-1} m_i^2 OLD_i}} & \sqrt{\dfrac{n_0^2 OLD_0}{OLD_L + \sum_{i=0}^{N_{EAO}-1} n_i^2 OLD_i}} \\ \vdots & \vdots \\ \sqrt{\dfrac{m_{N_{EAO}-1}^2 OLD_{N_{EAO}-1}}{OLD_L + \sum_{i=0}^{N_{EAO}-1} m_i^2 OLD_i}} & \sqrt{\dfrac{n_{N_{EAO}-1}^2 OLD_{N_{EAO}-1}}{OLD_L + \sum_{i=0}^{N_{EAO}-1} n_i^2 OLD_i}} \end{pmatrix}.$$

The output of the TTN element yields

$$\mathbf{Y} = \begin{pmatrix} \mathbf{y}_L \\ \hdashline \mathbf{y}_{0,EAO} \\ \vdots \\ \mathbf{y}_{N_{EAO}-1,EAO} \end{pmatrix} = \mathbf{M}_{\text{Energy}}\mathbf{X} = \mathbf{M}_{\text{Energy}}\begin{pmatrix} \mathbf{l}_0 \\ \mathbf{r}_0 \end{pmatrix}.$$

7.8.4.4 OTN matrix using energy mode

The corresponding OTN matrix $\mathbf{M}_{\text{Energy}}$ for the stereo case can be derived as

$$\mathbf{M}_{\text{Energy}} = \mathbf{A}\left(\frac{1}{\sqrt{OLD_L + \sum_{i=0}^{N_{EAO}-1} m_i^2 OLD_i}} + \frac{1}{\sqrt{OLD_R + \sum_{i=0}^{N_{EAO}-1} n_i^2 OLD_i}}\right)\begin{pmatrix} \sqrt{OLD_L} \\ \sqrt{OLD_R} \\ \hdashline \sqrt{m_0^2 OLD_0} + \sqrt{n_0^2 OLD_0} \\ \vdots \\ \sqrt{m_{N_{EAO}-1}^2 OLD_{N_{EAO}-1}} + \sqrt{n_{N_{EAO}-1}^2 OLD_{N_{EAO}-1}} \end{pmatrix},$$

hence the output signal $\mathbf{Y}$ of the OTN element yields

$$\mathbf{Y} = \mathbf{M}_{\text{Energy}}\mathbf{d}_0 .$$

For the mono case the OTN matrix $\mathbf{M}_{\text{Energy}}$ reduces to

$$\mathbf{M}_{\text{Energy}} = \mathbf{A}\frac{1}{\sqrt{OLD_L + \sum_{i=0}^{N_{EAO}-1} m_i^2 OLD_i}}\begin{pmatrix} \sqrt{OLD_L} \\ \hdashline \sqrt{m_0^2 OLD_0} \\ \vdots \\ \sqrt{m_{N_{EAO}-1}^2 OLD_{N_{EAO}-1}} \end{pmatrix}.$$

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

7.9 DCU processing

7.9.1 Introduction

Within the overall SAOC system, the Distortion Control Unit (DCU) is incorporated into the SAOC decoder/transcoder processing chain between the rendering interface and the actual SAOC decoding/transcoding unit. It provides a modified rendering matrix using the information from the rendering interface and SAOC data, see Figure 17. The modified rendering matrix can be accessed by the application, reflecting the actually effective rendering settings.

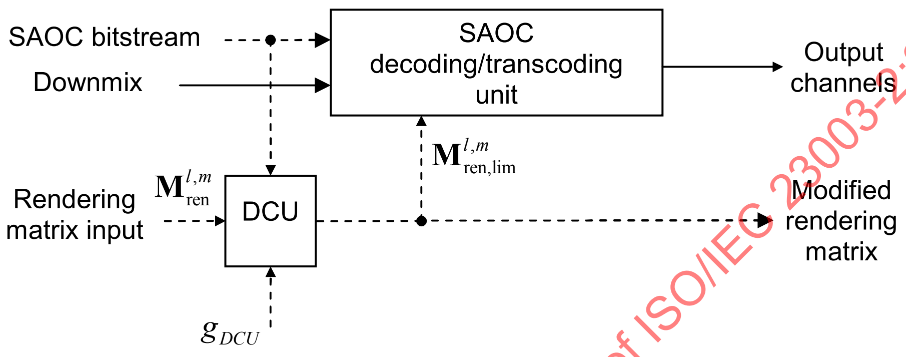

SAOC bitstream
Downmix
SAOC decoding/transcoding unit
Output channels
$\mathbf{M}_{\text{ren,lim}}^{l,m}$
Rendering matrix input
$\mathbf{M}_{\text{ren}}^{l,m}$
DCU
Modified rendering matrix
g_{DCU}

Figure 17 – DCU within the overall SAOC system

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

Based on the user specified rendering scenario represented by rendering matrix $\mathbf{M}_{\text{ren}}^{l,m}$ with elements $m_{i,j}^{l,m}$, the DCU prevents extreme rendering settings by producing a modified matrix $\mathbf{M}_{\text{ren,lim}}^{l,m}$ comprising limited rendering coefficients, which shall be used by the SAOC rendering engine. For all operational modes of SAOC the final (DCU processed) rendering coefficients shall be calculated according to:

$$\mathbf{M}_{\text{ren,lim}}^{l,m} = \left(1 - g_{DCU}\right)\mathbf{M}_{\text{ren}}^{l,m} + g_{DCU}\mathbf{M}_{\text{ren,tar}}^{l,m} .$$

The parameter $g_{DCU} \in \left[0,1\right]$ is used to define the degree of transition from the user specified rendering matrix $\mathbf{M}_{\text{ren}}^{l,m}$ towards the distortion-free target matrix $\mathbf{M}_{\text{ren,tar}}^{l,m}$. The parameter g_{DCU} is derived from the bitstream element **bsDcuParam** according to:

g_{DCU} = DcuParam[**bsDcuParam**].

There are two possible versions of the distortion-free target matrix $\mathbf{M}_{\text{ren,tar}}^{l,m}$, suited for different applications. It is controlled by the bitstream element **bsDcuMode:**

- (**bsDcuMode** = 0): The "downmix-similar" rendering, where $\mathbf{M}_{\text{ren,tar}}^{l,m}$ corresponds to the energy normalized downmix matrix.

- (**bsDcuMode** = 1): The "best effort" rendering, where $\mathbf{M}_{\text{ren,tar}}^{l,m}$ is defined as a function of both downmix and target rendering matrix.

7.9.2 DCU application for EAO

For SAOC decoders that decode residual coding data and thus support the handling of EAOs, it can be meaningful to provide a second parameterization of the DCU which allows to take advantage of the enhanced audio quality provided by the use of EAOs. This is achieved by decoding and using a second alternate set of DCU parameters (i.e. bsDcuMode2 and bsDcuParam2) which is transmitted as part of the data structures containing residual data (i.e. SAOCExtensionConfigData() and SAOCExtensionFrameData()). An application can make use of this second parameter set if it decodes residual coding data and operates in strict EAO mode which is defined by the condition that only EAOs can be modified arbitrarily while all non-EAOs only undergo a single common modification. Specifically, this strict EAO mode requires fulfillment of two following conditions:

- The downmix matrix and rendering matrix have the same dimensions (implying that the number of rendering channels is equal to the number of downmix channels).
- The application only employs rendering coefficients for each of the regular objects (i.e. non-EAOs) that are related to their corresponding downmix coefficients by a single common scaling factor.

7.9.3 "Downmix-similar" rendering

7.9.3.1 Introduction

The "downmix-similar" rendering method can typically be used in cases where the downmix is an important reference of artistic high quality.

The "downmix-similar" rendering matrix $\mathbf{M}_{\text{ren,DS}}^{l}$ is computed as

$$\mathbf{M}_{\text{ren,DS}}^{l} = \mathbf{M}_{\text{ren,tar}}^{l} = \sqrt{N_{DS}^{l}}\mathbf{D}_{DS}^{l},$$

where N_{DS}^{l} represents the energy normalization scalar (for each parameter slot l) and $\mathbf{D}_{DS}^{l}$ is the downmix matrix extended by rows of zero elements such that number and order of the rows of $\mathbf{D}_{DS}^{l}$ correspond to the constellation of $\mathbf{M}_{\text{ren}}^{l,m}$. For example, in the SAOC stereo to multichannel transcoding mode ("x-2-5" processing mode) $M = 2$ and $N_{MPS} = 6$. Accordingly $\mathbf{D}_{DS}^{l}$ is of size $N_{MPS} \times M$ and its rows representing the front left and right output channels equal $\mathbf{D}^{l}$.

7.9.3.2 All decoding/transcoding SAOC modes

For all decoding/transcoding SAOC modes the energy normalization scalar N_{DS}^{l} is computed using the following equation:

$$N_{DS}^{l} = \frac{trace\left(\mathbf{M}_{\text{ren}}^{l,m}\left(\mathbf{M}_{\text{ren}}^{l,m}\right)^{*}\right)+\varepsilon}{trace\left(\mathbf{D}^{l}\left(\mathbf{D}^{l}\right)^{*}\right)+\varepsilon}.$$

7.9.4 "Best effort" rendering

7.9.4.1 Introduction

The "best effort" rendering method can typically be used in cases where the target rendering is an important reference.

The "best effort" rendering matrix describes a target rendering matrix, which depends on the downmix and rendering information. The energy normalization is represented by a matrix $\mathbf{N}_{BE}^{l,m}$ of size $N_{MPS} \times M$, hence it

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

provides individual values for each output channel. This requires different calculations of $\mathbf{N}_{BE}^{l,m}$ for the different SAOC operation modes, which are outlined in the following. The "best effort" rendering matrix is computed as

$\mathbf{M}_{\text{ren,BE}}^{l} = \mathbf{M}_{\text{ren,tar}}^{l} = \sqrt{\mathbf{N}_{BE}^{l}}\mathbf{D}^{l}$, for the following SAOC modes "x-1-1/2/5/b", "x-2-1/b",

$\mathbf{M}_{\text{ren,BE}}^{l} = \mathbf{M}_{\text{ren,tar}}^{l} = \mathbf{N}_{BE}^{l}\mathbf{D}^{l}$, for the following SAOC modes "x-2-2/5".

where $\mathbf{D}^{l}$ is the downmix matrix and $\mathbf{N}_{BE}^{l,m}$ represents the energy normalization matrix.

7.9.4.2 SAOC mono-to-mono ("x-1-1") decoding mode

For the "x-1-1" SAOC mode the energy normalization scalar $\mathbf{N}_{BE}^{l,m}$ is computed using the following equation

$$\mathbf{N}_{BE}^{l,m} = \frac{\sum_{j=0}^{N-1}\left(m_{j,0}^{l,m}\right)^{2} + \varepsilon}{\sum_{j=0}^{N-1}\left(d_{j}^{l}\right)^{2} + \varepsilon}.$$

7.9.4.3 SAOC mono-to-stereo ("x-1-2") decoding mode

For the "x-1-2" SAOC mode the energy normalization matrix $\mathbf{N}_{BE}^{l,m}$ of size 2×1 is computed using the following equation

$$\mathbf{N}_{BE}^{l,m} = \left(\frac{\sum_{j=0}^{N-1}\left(m_{j,0}^{l,m}\right)^{2} + \varepsilon}{\sum_{j=0}^{N-1}\left(d_{j}^{l}\right)^{2} + \varepsilon}, \quad \frac{\sum_{j=0}^{N-1}\left(m_{j,1}^{l,m}\right)^{2} + \varepsilon}{\sum_{j=0}^{N-1}\left(d_{j}^{l}\right)^{2} + \varepsilon} \right)^{T}.$$

7.9.4.4 SAOC mono-to-binaural ("x-1-b") decoding mode

For the "x-1-b" SAOC mode the energy normalization matrix $\mathbf{N}_{BE}^{l,m}$ of size $N_{HRTF} \times 1$ is computed using the following equation

$$\mathbf{N}_{BE}^{l,m} = \left(\frac{\sum_{j=0}^{N-1} a_{j,0}^{l,m}\left(a_{j,0}^{l,m}\right)^{*} + \varepsilon}{\sum_{j=0}^{N-1}\left(d_{j}^{l}\right)^{2} + \varepsilon}, \quad \ldots, \quad \frac{\sum_{j=0}^{N-1} a_{j,N_{HRTF}-1}^{l,m}\left(a_{j,N_{HRTF}-1}^{l,m}\right)^{*} + \varepsilon}{\sum_{j=0}^{N-1}\left(d_{j}^{l}\right)^{2} + \varepsilon} \right)^{T}.$$

The elements $a_{x,y}^{l,m}$ comprise the target binaural rendering matrix $\mathbf{A}^{l,m}$.

7.9.4.5 SAOC stereo-to-mono ("x-2-1") decoding mode

For the "x-2-1" SAOC mode the energy normalization matrix $\mathbf{N}_{BE}^{l,m}$ of size 1×2 is computed using the following equation

$$\mathbf{N}_{BE}^{l,m} = \mathbf{M}_{\text{ren}}^{l,m}\left(\mathbf{D}^{l}\right)^{*}\mathbf{J}^{l},$$

where $\mathbf{M}_{\text{ren}}^{l,m}$ is mono rendering matrix of size $1 \times N$.

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

7.9.4.6 SAOC stereo-to-stereo ("x-2-2") decoding mode

For the "x-2-2" SAOC mode the energy normalization matrix $\mathbf{N}_{BE}^{l,m}$ of size 2×2 is computed using the following equation

$$\mathbf{N}_{BE}^{l,m} = \mathbf{M}_{\text{ren}}^{l,m} \left(\mathbf{D}^{l}\right)^{*} \mathbf{J}^{l},$$

where $\mathbf{M}_{\text{ren}}^{l,m}$ is stereo rendering matrix of size $2 \times N$.

7.9.4.7 SAOC mono-to-binaural ("x-2-b") decoding mode

For the "x-2-b" SAOC mode the energy normalization matrix $\mathbf{N}_{BE}^{l,m}$ of size $N_{HRTF} \times 2$ is computed using the following equation

$$\mathbf{N}_{BE}^{l,m} = \mathbf{A}^{l,m} \left(\mathbf{D}^{l}\right)^{*} \mathbf{J}^{l},$$

where $\mathbf{A}^{l,m}$ is binaural rendering matrix of size $2 \times N$.

7.9.4.8 SAOC mono-to-multichannel ("x-1-5") transcoding mode

For the "x-1-5" SAOC mode the energy normalization matrix $\mathbf{N}_{BE}^{l,m}$ of size $N_{MPS} \times 1$ is computed using the following equation

$$\mathbf{N}_{BE}^{l,m} = \left(\frac{\sum_{j=0}^{N-1}\left(m_{j,0}^{l,m}\right)^2 + \varepsilon}{\sum_{j=0}^{N-1}\left(d_j^l\right)^2 + \varepsilon}, \quad \ldots, \quad \frac{\sum_{j=0}^{N-1}\left(m_{j,N_{MPS}-1}^{l,m}\right)^2 + \varepsilon}{\sum_{j=0}^{N-1}\left(d_j^l\right)^2 + \varepsilon} \right)^T.$$

7.9.4.9 SAOC stereo-to-multichannel ("x-2-5") transcoding mode

For the "x-2-5" SAOC mode the energy normalization matrix $\mathbf{N}_{BE}^{l,m}$ of size $N_{MPS} \times 2$ is computed using the following equation

$$\mathbf{N}_{BE}^{l,m} = \mathbf{M}_{\text{ren}}^{l,m} \left(\mathbf{D}^{l}\right)^{*} \mathbf{J}^{l}.$$

To avoid numerical problems when calculating the term $\mathbf{J}^{l} = \left(\mathbf{D}^{l}\left(\mathbf{D}^{l}\right)^{*}\right)^{-1}$ in 7.9.4.5, 7.9.4.6, 7.9.4.7 and 7.9.4.9, $\mathbf{J}^{l}$ is modified. First the eigenvalues $\lambda_{1,2}$ of $\mathbf{J}^{l}$ are calculated, solving $\det(\mathbf{J} - \lambda_{1,2}\mathbf{I}) = 0$.

Eigenvalues are sorted in descending ($\lambda_1 \geq \lambda_2$) order and the eigenvector corresponding to the larger eigenvalue is calculated according to the equation above. It is assured to lie in the positive x-plane (first element has to be positive). The second eigenvector is obtained from the first by a – 90 degrees rotation:

$$\mathbf{J} = \left(\mathbf{v}_1 \mathbf{v}_2\right) \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix} \left(\mathbf{v}_1 \mathbf{v}_2\right)^{*}.$$

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

7.10 MBO processing

7.10.1 Introduction

A Multi-channel Background Object (MBO) is an MPS mono or stereo downmix that is part of the SAOC downmix. The associated MPS bitstream can be conveyed either parallel to the SAOC bitstream, i.e., according to the MPS standard (ISO/IEC 23003-1:2007), or it can be embedded in the SAOC bitstream using the bsSaocExtType==2 mechanism. As opposed to using individual SAOC objects for each channel in a multi-channel signal, an MBO can be used enabling SAOC to more efficiently handle a multi-channel object. In the MBO case, the SAOC overhead gets lower as the MBO's SAOC parameters only are related to the downmix channels rather than all the upmix channels.

7.10.2 Decoding process

The following decoding process shall be applied in the case where MPS data is embedded in the SAOC bitstream (i.e., carried by means of the bsSaocExtType==2 mechanism):

- If the SAOC decoder supports multi-channel rendering (by means of a subsequent MPS decoder) and if the number of output channels of the SAOC rendering matrix is the same as the number of downmix channels of the MPS bitstream, then the embedded MPS bitstream is send "as is" to the subsequent MPS decoder, and SAOC operates as a normal SAOC decoder (as shown in the right part of Figure 2), sending the SAOC-processed output signal as input into the MPS decoder. Note that SAOC and MPS have to be connected directly in the QMF domain in order to ensure proper time-alignment of the MPS and SAOC side information.
- If the SAOC decoder does not support multi-channel rendering or if the number of output channels of the SAOC rendering matrix is different from the number of downmix channels of the MPS bitstream, then the embedded MPS data is discarded and the SAOC decoding or transcoding (depending upon the number of channels of the rendering matrix) is carried out according to the standard decoding process.

The following decoding process shall be applied in the case where MPS data and SAOC data are both present independently and in parallel, i.e., when MPS data is not embedded in the SAOC bitstream:

- If a decoder only implements SAOC, then the standard SAOC decoding process is carried out.
- If a decoder only implements MPS, then the standard MPS decoding process is carried out.
- If a decoder implements both SAOC and MPS, then, if the number of output channels of the SAOC rendering matrix is the same as the number of downmix channels, first the SAOC decoding process is applied, directly followed by the MPS decoding process (connected in the QMF domain), i.e., this is exactly the same decoding process as defined for the case where MPS data is embedded in the SAOC bitstream). If the number of output channels of the SAOC rendering matrix is the different from number of downmix channels, then either SAOC decoding (discarding the MPS data) or MPS decoding (discarding the SAOC data) can be applied.

To illustrate the decoding process defined above, an example scenario can be considered, where a stereo SAOC downmix includes the stereo downmix of an MBO and a mono vocal object, where the MBO MPS data is embedded in the SAOC bitstream, and where the SAOC decoder supports multi-channel rendering (by means of a subsequent MPS decoder), To obtain a multi-channel rendering of only the MBO ("karaoke mode"), a stereo SAOC rendering matrix can be used that attenuates the vocal object. To obtain a multi-channel rendering of only the vocal object ("solo mode"), a multi-channel SAOC rendering matrix can be used that routes the vocal object to the desired output channels and attenuates the MBO.

7.10.3 MPS bitstream settings for MBO

The MPS parameters for MBO shall be specified according to those of the corresponding SAOC data in order to be compatible in the transcoding process. The following table contains the list of variables with the corresponding references to tables defined in ISO/IEC 23003-1:2007.

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

Table 51 – List of the MPS bitstream variables

MBO MPS configuration parameters	SAOC configuration parameters	Reference*)
bsSamplingFrequencyIndex	**bsSamplingFrequencyIndex**	Table 5
bsSamplingFrequency	**bsSamplingFrequency**	Table 5
bsFrameLength	**bsFrameLength**	Table 5
*) defined in ISO/IEC 23003-1:2007		

7.11 MCU Combiner

7.11.1 Introduction

To use the SAOC concept for teleconferencing applications a so-called Multipoint Control Unit (MCU) is required, which is able to combine the signals of several communication partners without decoding/re-encoding the corresponding audio objects. As illustrated in Figure 18 the MCU combines the input SAOC side information streams into one common SAOC bitstream in a way that the parameters representing all audio objects from the input bitstreams are included in the resulting output bitstream. These calculations can be performed in the parameter domain without the need to analyze the downmix signals and introducing no additional delay in the signal processing chain.

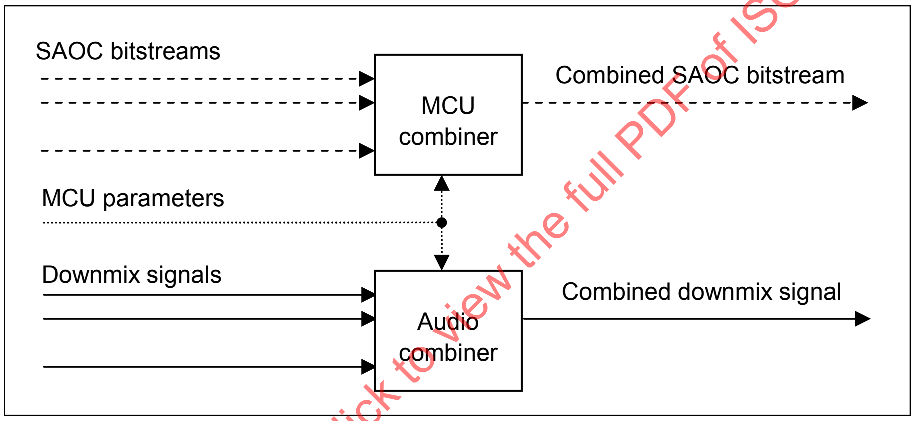


Figure 18 – Outline of the MCU combiner

7.11.2 MCU interface and SAOC parameter estimation

The MCU combiner interface is defined by the input, output and control data. The SAOC parameters (i.e. OLD, IOC and NRG) of one or several SAOC bitstreams represent the input data for the MCU. The output SAOC bitstream contains the resulting parameters (i.e. OLD, IOC and NRG) of the combined SAOC bitstream calculated as follows:

$$OLD_j^{comb} = \frac{\mathbf{E}_j}{NRG^{comb}},$$

$$NRG^{comb} = \max_j \left(\mathbf{E}_j \right),$$

$$IOC_{k,j}^{comb} = IOC_{N^{comb}(l,n) N^{comb}(i,n)}^{bs_n},$$

where

$$\mathbf{E}_j = \sum_{N^{comb}(i,n)} \mathbf{F}_{i,n}, \qquad \mathbf{F}_{i,n} = \mathbf{G}_n^{comb} OLD_i^{bs_n} NRG^{bs_n}, \qquad j = N^{comb}(i,n),$$

$$OLD_i = \mathbf{D}_{\text{OLD}}(i,l,m), \qquad NRG = \mathbf{D}_{\text{NRG}}(l,m), \qquad IOC_{i,j} = \mathbf{D}_{\text{IOC}}(i,j,l,m).$$

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

The MCU control parameters contain the gains $\mathbf{G}_n^{comb}$ describing the downmix combination process for the weighted mixing, the mapping function $j = N^{comb}(i,n)$ determining a set and order of audio objects in the combined SAOC bitstream, the parameter for activating the NRG parameter transmission for the combined bitstream, the control parameter describing the resulting downmix mode (i.e. mono or stereo) and the corresponding parameters $\mathbf{D}_{\mathrm{DMG}}^{comb}$ (and $\mathbf{D}_{\mathrm{DCLD}}^{comb}$) describing the downmix.

7.11.3 Energy estimation in the SAOC encoder

All combined SAOC bitstreams must contain parameters specifying the absolute energy of the object with the highest energy level for a certain time/frequency band (i.e. bsTransmitAbsNrg=1).

The absolute energy parameter (NRG) is calculated in the SAOC encoder as product of the object signals with the highest energy (normalized according the corresponding time/frequency tiles) according to the following formula:

$$NRG^{l,m} = \max_i \left(\frac{\sum\limits_{n\in l} \sum\limits_{k\in m} x_i^{n,k} \left(x_i^{n,k} \right)^*}{\sum\limits_{n\in l} \sum\limits_{k\in m} 1} + \varepsilon \right),$$

where the signal $x_i^{n,k}$ corresponds to the filterbank processing described in 7.3 of ISO/IEC 23003-1:2007 for the regular SAOC processing mode and in 7.13.2 for the LP processing mode.

7.11.4 Parameters of the combined bitstreams

The SAOC configuration parameters specifying the frame size and sampling frequency (i.e. bsFrameLength, bsSamplingFrequencyIndex, bsSamplingFrequency, bsFreqRes) should be identical for all combined (and resulting) SAOC bitstreams.

7.12 Effects

7.12.1 Introduction

The SAOC effects interface operates on the downmix and therefore is part of the downmix processor of the SAOC transcoder or decoder as shown in Figure 19. The effects interface allows objects or linear combinations of objects to be extracted from the downmix for effects processing. Two types of effects processing interfaces are supported. The first type, referred to as the insert effects interface, allows effects processing to individual objects in the downmix.

The second type, referred to as the send effects interface, allows effect processing on individual objects or linear combinations thereof. For this type, the resulting “wet” signals can be directly added to the SAOC decoder output (see Annex C).

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

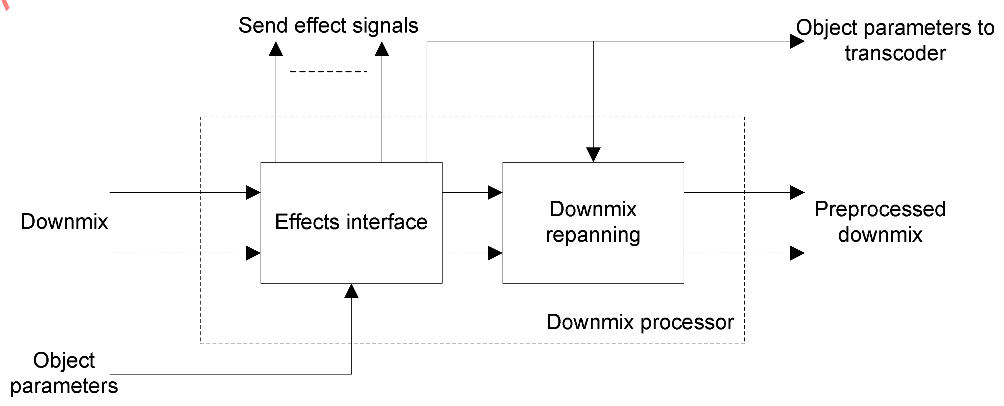


Figure 19 – The effects interface is a part of the downmix processor

A block diagram of the SAOC effects interface is shown in Figure 20. The object splitter, combiner and extractor are described in the following subclauses. The application of the effects on the effect signals (i.e. the signals consisting of objects or linear combinations thereof to which effects processing is to be applied), and the insertion of the send effects signals (i.e. the signals consisting of objects or linear combinations thereof to which send effects processing has been applied), are discussed in Annex C.

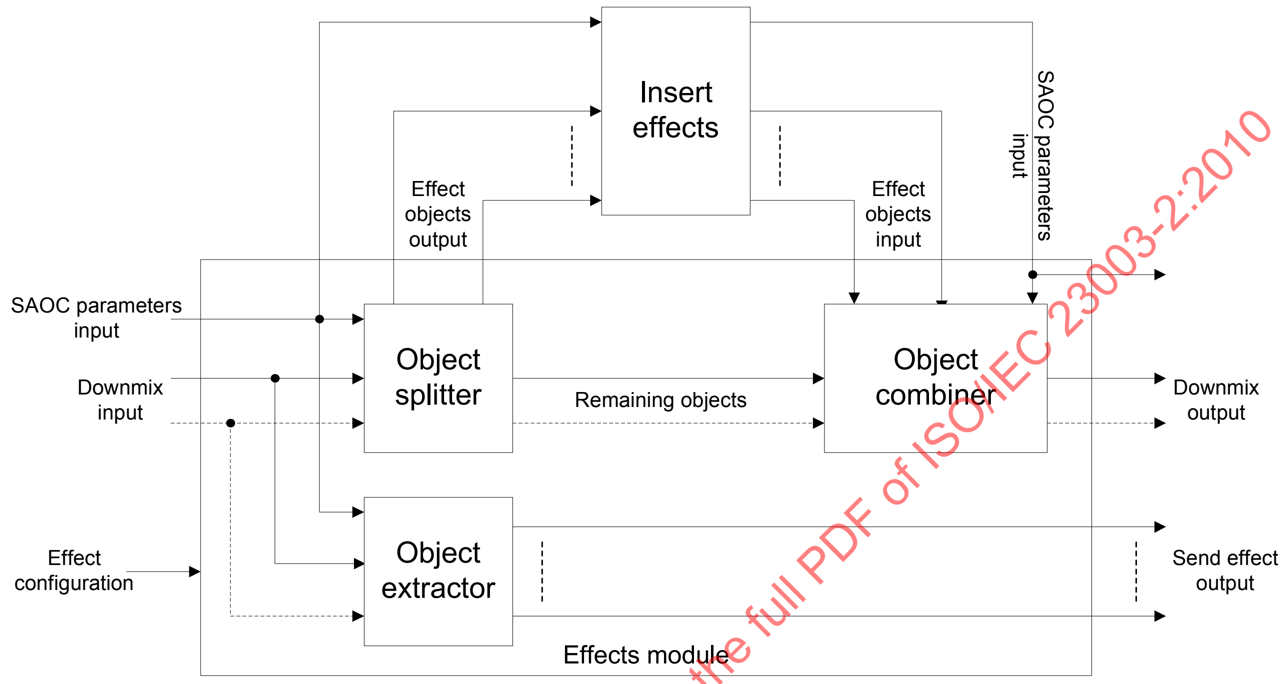


Figure 20 – Effects module in SAOC framework.

7.12.2 Insert effects

The processing block for the insert effects interface consists of an *object splitter* isolating certain objects from the downmix for effect processing, and an *object combiner* inserting the effect processed objects back into the downmix. The effect configuration data that is input to the effects module contains an object split vector $\mathbf{r}^{insert}$ indicating which objects should be separated from the other objects in the downmix. The object splitter (Figure 20) splits object $y \in [0,\ldots,N-1]$ off of the downmix when r_y^{insert} = 1. All entries corresponding to objects that are not input to an insert effect are zero. The effect objects are represented by signal vector $\mathbf{x}_{insert}^{n,k}$ with $numEffObj$ elements, defined as:

$$x_{insert,y_e}^{n,k} = w_{insert_L,y_e}^{n,k} x_{L0}^{n,k} + w_{insert_R,y_e}^{n,k} x_{R0}^{n,k},$$

where

$$y_e = f_{obj\rightarrow eff}(y),$$

$$w_{insert_L,y_e}^{l,m} = \frac{\hat{d}_{1,y}^2 OLD_y}{\sqrt{d_{1,y} OLD_y \sum_{i=0}^{N-1} d_{1,i} OLD_i\, IOC_{i,y}}}, \qquad w_{insert_R,y_e}^{l,m} = \frac{\hat{d}_{2,y}^2 OLD_y}{\sqrt{d_{2,y} OLD_y \sum_{i=0}^{N-1} d_{2,i} OLD_i\, IOC_{i,y}}},$$

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

$$\hat{d}_{i,j} = \frac{d_{i,j}}{10^{0.05 DMG_j}},$$

$$numEffObj = \sum_{i=0}^{N-1} r_i^{insert} .$$

Function $f_{obj \to eff}$ maps object indices to insert effect object indices according to:

$$f_{obj \to eff}(y) = \sum_{i=0}^{y} r_i^{insert} .$$

For a mono downmix case the downmix matrix $\mathbf{D}$ is defined as $d_j = d_{1,j} = d_{2,j} = 10^{0.05 DMG_j} / \sqrt{2}$.

The remaining objects are kept in the same format as the downmix and form an additional output of the object splitter. This residual downmix is calculated as follows:

$$x_{res_L}^{n,k} = \left(1 - \sum_{i=0}^{numEffObj-1} w_{insert_L,i}^{l,m}\right) x_{L0}^{n,k},$$

$$x_{res_R}^{n,k} = \left(1 - \sum_{i=0}^{numEffObj-1} w_{insert_R,i}^{l,m}\right) x_{R0}^{n,k}.$$

The effects applied to the effect objects in $\mathbf{x}_{insert}$ may change the spectro-temporal properties of the objects. Using the SAOC parameters in the bitstream for further processing (transcoding and downmix preprocessing) would then yield suboptimal performance. Therefore, the effects interface provides a SAOC parameter output and input allowing the effects functions to update the parameters to reflect the spectro-temporal properties of the objects after effects processing. These parameters shall be used by the subsequent transcoding and downmix preprocessing.

After effects have been applied, the resulting effect objects $\hat{\mathbf{x}}_{insert}$ are fed back into the object combiner. This functional block combines the effect objects with the other objects in the residual downmix $\left(x_{res_L}^{n,k}, x_{res_R}^{n,k}\right)$ which it receives from the object splitter

$$x_{L0_{eff}}^{n,k} = x_{res_L}^{n,k} + \sum_{i=0}^{numEffObj-1} d_{1 f_{eff \to obj}(i)}^2 \hat{x}_{insert,i}^{n,k}, \qquad x_{R0_{eff}}^{n,k} = x_{res_R}^{n,k} + \sum_{i=1}^{numEffObj} d_{2 f_{eff \to obj}(i)}^2 \hat{x}_{insert,i}^{n,k},$$

where

$$f_{eff \to obj}(y_e) = y \Big|_{\sum_{i=1}^{y} r_i^{insert} = y_e} .$$

7.12.3 Send effects

The preparative processing for the send effects consists of an *object extractor* extracting objects or linear combinations of objects from the downmix that are output as send effect channels. As part of the effect configuration object extraction matrix $\mathbf{r}^{send}$ (which may be time- and/or frequency variant) is fed into the effects module indicating the gains per object where each row represents a send effect channel. With this matrix, effect channel $c_e \in [0, \ldots, numEffCh - 1]$ is calculated by:

$$x_{send,c_e}^{n,k} = w_{send_L,c_e}^{n,k} x_{L0}^{n,k} + w_{send_R,c_e}^{n,k} x_{R0}^{n,k},$$

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

where

$$w_{send_L,c_e}^{l,m}=\sqrt{\frac{\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}\hat{d}_{1i}^2\hat{d}_{1j}^2 r_{c_e,i}^{send} r_{c_e,j}^{send} e_{i,j}}{\sum_{i=0}^{N-1}\sum_{j=0}^{N-1} d_{1,i} d_{1,j} e_{i,j}}},\quad w_{send_R,c_e}^{l,m}=\sqrt{\frac{\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}\hat{d}_{2i}^2\hat{d}_{2j}^2 r_{c_e,i}^{send} r_{c_e,j}^{send} e_{i,j}}{\sum_{i=0}^{N-1}\sum_{j=0}^{N-1} d_{2,i} d_{2,j} e_{i,j}}},$$

$$\hat{d}_{i,j}=\frac{d_{i,j}}{10^{0.05 DMG_j}}.$$

For a mono downmix case the downmix matrix $\mathbf{D}$ is defined as $d_j = d_{1,j} = d_{2,j} = 10^{0.05 DMG_j} / \sqrt{2}$.

7.13 Low Power SAOC processing

7.13.1 Introduction

The SAOC Low Power (LP) processing mode includes LP tools from MPS such as its partially real valued filter bank, alias reduction and LP decorrelator. In SAOC LP processing mode the decorrelator is disabled in stereo downmix mode.

7.13.2 Time/frequency transform

In SAOC LP processing mode the filterbank described in 6.10.2 of ISO/IEC 23003-1:2007 shall be used.

7.13.3 Alias reduction

In SAOC LP processing mode the alias reduction tool described in 6.10.3 of ISO/IEC 23003-1:2007 shall be used. However, in 6.10.3.3 of ISO/IEC 23003-1:2007 the alias equalization applied to $\mathbf{W}_1^{l,k}$ and $\mathbf{W}_2^{l,k}$ should instead be applied to:

- $\mathbf{G}_{\text{Mod}}$, in the stereo-to-mono, stereo-to-stereo and stereo-to-transcoding case.
- $\mathbf{G}$ and $\mathbf{P}_2$, in the mono-to-mono, mono-to-stereo and mono-to-binaural case.
- $\mathbf{G}$, in the stereo-to-binaural case.

7.13.4 Low Power decorrelator

In the mono downmix cases the MPS LP decorrelator should be use according to 6.10.7.2 in ISO/IEC 23003-1:2007.

For the stereo downmix cases, the decorrelator is disabled. This should be implemented by:

- in the stereo downmix transcoding (and stereo output) case, always forcing $r_{1,2} > 0$ in section 7.6.3.3.
- in the stereo to binaural case, always force $\alpha^{l,m} = \beta^{l,m} = 0$ in 7.7.2.4.

7.13.5 Residual coding

Residual coding is limited according to 6.10.8 of ISO/IEC 23003-1:2007.

7.14 Low Delay SAOC processing

7.14.1 Overview

The following subclauses outline the differences of the implementation of the Low Delay (LD) version of the SAOC decoder/transcoder compared to the regular version outlined in the previous subclauses. The QMF bank is replaced with a LD QMF bank and the additional oddly-modulated Nyquist filterbank is omitted. As result the number of processing bands is reduced. Furthermore, the decorrelator filters are adapted to the LD operation.

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

7.14.2 Time/frequency transforms

7.14.2.1 Overview

The LD SAOC decoder/transcoder employs LD QMF banks to convert time domain signals into subband domain signals and vice versa. Contrary to the regular SAOC decoding mode, the oddly-modulated Nyquist filter bank is omitted for LD SAOC decoding, see Figure 21.

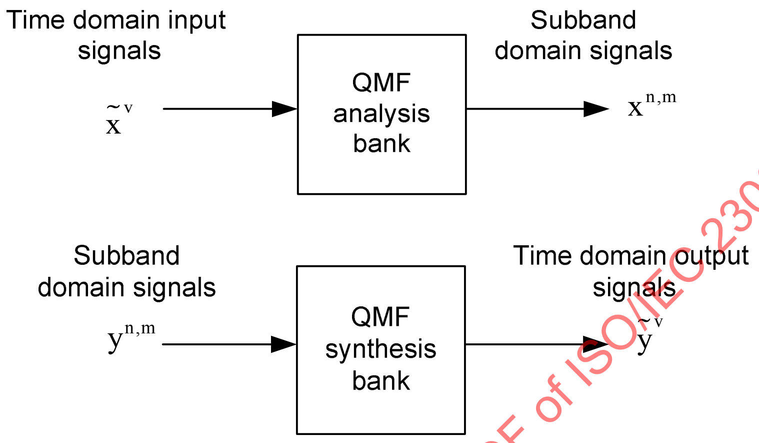


Figure 21 – Time/frequency transforms in LD SAOC

The LD QMF banks used for LD SAOC decoding can be derived from the QMF banks defined in 6.3.2.1 and 6.3.3 of ISO/IEC 23003-1:2007 by changing the window function and the modulation as described in the following subclauses.

7.14.2.2 Analysis filterbank

The 64 band LD analysis QMF bank is defined by using the QMF LD 256 window function c[i] specified in Table A.21 and by replacing the modulation with

$$\mathrm{M}(k,n)=\exp\left(\frac{i\pi(k+0.5)(2n+129)}{128}\right),\qquad \begin{cases}0\le k<64\\0\le n<128\end{cases}$$

7.14.2.3 Synthesis filterbank

The 64 band LD synthesis QMF bank is defined by using the QMF LD 256 window function c[i] specified in Table A.21 and by replacing the modulation with

$$\mathrm{M}(k,n)=\frac{1}{64}\exp\left(\frac{i\pi(k+0.5)(2n-255)}{128}\right),\qquad \begin{cases}0\le k<64\\0\le n<128\end{cases}$$

7.14.2.4 Downsampled analysis filterbank

The 32 band LD analysis QMF bank is defined by using window coefficients $c_i[i]$ obtained by linear interpolation

$$c_i(i)=\frac{c(2i+1)+c(2i)}{2},\qquad 0\le i<320$$

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

of the QMF LD 256 window function c[i] specified in Table A.21 and by replacing the modulation with

$$\mathrm{M}(k,n) = 2\exp\left(\frac{i\pi(k+0.5)(2n+65)}{64}\right), \qquad \begin{cases} 0 \le k < 32 \\ 0 \le n < 64 \end{cases}$$

7.14.2.5 Downsampled synthesis filterbank

The 32 band LD synthesis QMF bank is defined by using window coefficients c$_i$[i] obtained by linear interpolation

$$c_i(i) = \frac{c(2i+1)+c(2i)}{2}, \qquad 0 \le i < 320$$

of the QMF LD 256 window function c[i] specified in Table A.21 and by replacing the modulation with

$$\mathrm{M}(k,n) = \frac{1}{64}\exp\left(\frac{i\pi(k+0.5)(2n-127)}{64}\right), \qquad \begin{cases} 0 \le k < 32 \\ 0 \le n < 64 \end{cases}$$

7.14.2.6 Tables

The window function c[i] used for the LD QMF banks is specified in Table A.21.

7.14.2.7 Processing frequency resolution

Due to the omission of the Nyquist filterbank the number of parameter and processing bands is adapted accordingly. The number of hybrid subbands *K* is reduced to 64 and the resulting number of processing bands M_{proc} is 23. The number of parameter bands is also adapted accordingly, as described in Table 33.

Table A.22 specifies the mapping from parameter to processing bands and is used instead of Table 86 of ISO/IEC 23003-1:2007.

The hybrid to processing band mapping is specified in Table A.23 and is used instead of Table A.31 of ISO/IEC 23003-1:2007.

7.14.2.8 Binaural decoding mode

The processing follows the description of 7.7.2.1 and 7.7.2.4. The HRTF parameters are given by $H_{i,L}^m$, $H_{i,R}^m$ and ϕ_i^m for each processing band m. The spatial positions for which HRTF parameters are available are characterized by the index i. These parameters are described in ISO/IEC 23003-1:2007, where the index mapping $q(m)$ from Table A.32 in ISO/IEC 23003-1:2007 is replaced by $q(m) = m$ as no hybrid filter is applied.

As the number of processing bands is different for the LD decoding mode, the expression $0 \le m \le 11$ in the formulas for inter channel phase difference $\phi_C^{l,m}$ and rotation angle $\alpha^{l,m}$ is replaced by $0 \le m \le 6$.

7.14.2.9 Decorrelation

The decorrelated signals $\mathbf{X}_{\mathrm{d}} = \begin{pmatrix} x_{1d} \\ x_{2d} \end{pmatrix}$ are created from the decorrelator described in 6.6.2 of ISO/IEC 23003-1:2007. Following this scheme, the bsDecorrConfig=0 configuration shall be used with a decorrelator index X = 2. For region k_1, as defined in Table B.13, the lattice coefficients are defined in Table B.14. For region k_2 and k_3 the coefficients are given in Table B.15 and Table B.16.

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

8 Transport of SAOC side information

8.1 Overview

Due to its basic principle of operation, SAOC technology requires two data streams for its operation, namely a stream of downmix data (the coded downmix audio signal) and as side information a stream of SAOC data (the parametric spatial audio data guiding the SAOC transcoding/decoding process). Depending on the application scenario, both data streams may or may not be conveyed to the SAOC transcoder/decoder as part of a single stream, and may or may not be conveyed in a synchronized way. The subsequent subclauses define mechanisms for the transport of spatial data for SAOC for different environments employing MPEG Audio and Systems to convey a coded downmix audio signal.

8.2 Transport and signalling in an MPEG environment

8.2.1 Time alignment of SAOC frames and downmix coder frames

The SAOC frame length is preferred to be an integer multiple of the frame length of the underlying downmix coder. Asynchronous framing of SAOC data and the downmix data (i.e., different frame lengths) is possible. However, in this case, additional buffering of the SAOC data in the decoder might be needed.

In general, SAOC data is preferably conveyed in such a manner that it is available to the SAOC decoder in time when it is required to process the decoded downmix signals, assuming the most efficient connection of downmix decoder to the SAOC decoder. This is a direct connection of HE-AAC and SAOC in the QMF domain in the case that both use the same number of OMF bands (see 5.4), and a connection in the PCM time domain in all other cases. When HE-AAC and SAOC are connected in the time domain even though the most efficient connection would have been in the QMF domain, the SAOC parameters have to be delayed accordingly in order to maintain the time alignment between SAOC data and downmix data. Information about this delay is given in 4.5.

In the case that the SAOC data is embedded in the downmix data stream (see 8.2.2, 8.2.3, and 8.2.4), the temporal relationship between SAOC frames and downmix frames is indicated by the value of saocTimeAlign (see 8.2.5). If saocTimeAlign has the value 0, this indicates that the SAOC data is conveyed in the preferred manner outlined above.

In the case that the downmix data and the SAOC data are conveyed in separate streams, the temporal relationship between SAOC frames and downmix frames is indicated by the time stamps of the access units of the corresponding streams. If a downmix coder other than HE-AAC is used, the time stamp of an access unit carrying an SAOC frame identifies the first PCM sample of the corresponding time domain downmix signal frame that is input to the SAOC decoder. If HE-AAC is used as downmix coder, the time stamp of the SAOC frame identifies the first PCM sample of the corresponding time domain downmix signal frame at the output of the AAC core decoder.

If the transport layer does not provide time stamps, the temporal relationship between the data of these both streams needs to be defined by other means. In case of LATM (see ISO/IEC 14496-3), the first SAOC access unit and the first downmix coder access unit in an AudioMuxElement() are considered to have the same time stamp.

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

8.2.2 Transport and Signalling in an MPEG-4 Audio/Systems Environment

The signalling of the availability of SAOC data is possible by means of an SAOC elementary stream that depends on the elementary stream containing the related coded audio downmix data (as, e.g., indicated by the dependsOn_ES_ID field defined in 7.2.6.5.2, ISO/IEC 14496-1:2010). The actual SAOC data is either conveyed in the SAOC elementary stream or multiplexed into the downmix data stored in the elementary stream upon which the SAOC elementary stream, if present, depends. The latter is specified for MPEG-2/4 AAC payloads (see 8.2.3) and for MPEG-1/2 Layer I/II/III payloads (see 8.2.4).

Backwards compatibility with decoders that can decode the coded audio downmix data but not the SAOC data is achieved in both scenarios.

The interface to ISO/IEC 14496-1 is in line with the specification given in 1.6 of ISO/IEC 14496-3:2009. An elementary stream carrying SAOC data is identified by the Audio Object Type "SAOC" (Object Type ID 43). The AudioSpecificConfig() for this object carries the SaocSpecificConfig() data and a saocPayloadEmbedding flag that indicates whether the SaocFrame() payload is conveyed as an elementary stream or embedded into the downmix data, as defined in 1.6.3.18 of ISO/IEC 14496-3:2009.

8.2.3 Embedding SAOC data in MPEG-2/4 AAC payloads

SAOC data can be conveyed in the AAC extension_payload() mechanism using extension_type EXT_SAOC_DATA ("1010"), as defined in 8.8 of ISO/IEC 13818-7:2006 and 4.5.2.9 of ISO/IEC 14496-3:2009. The extension_payload() for type EXT_SAOC_DATA comprises the saoc_extension_data(), as defined in 6.3 of ISO/IEC 13818-7:2006 and 4.4.27 of ISO/IEC 14496-3:2009, which is used to carry a SaocDataFrame(), complete or split into several fragments, using the same syntax elements ancType, ancStart, ancStop , and ancDataSegmentByte as defined in 7.2.4, and where in the semantics of the syntax element ancDataSegmentByte, the term AncDataElement is to be replaced by saoc_extension_data.

This mechanism is backwards compatible with existing AAC decoders and provides implicit signalling. Explicit signalling is possible in an MPEG-4 environment if the SaocSpecificConfig() is conveyed out-of-band, as e.g. described in the previous subclause. In that case, any in-band SaocSpecificConfig() shall be identical to the out-of-band one. There must be at least one extension_payload() element with extension_type==EXT_SAOC_DATA in each AAC frame in order to enable immediate implicit signalling. An saoc_extension_data() element can have an empty payload, i.e. cnt==1.

8.2.4 Embedding SAOC data in MPEG-1/2 Layer I/II/III bitstreams

SAOC data can be inserted as ancillary data into an MPEG-1/2 Layer I/II/III bitstream as defined in ISO/IEC 11172-3 and ISO/IEC 13818-3. The AncDataElement() is conveyed in the ancillary data part of an MPEG-1/2 frame. It does not have to be located immediately after the audio_data of that frame. The first bit of the ancSyncword must be byte-aligned with respect to the first bit of the 0xFFF syncword of the MPEG-1/2 frame header. The AncDataElement() must be completely included in the ancillary data of a single MPEG-1/2 frame. There must be at least one AncDataElement() in the ancillary data of each MPEG-1/2 frame in order to enable immediate implicit signalling. An AncDataElement() can have an empty payload, i.e. ancLenBytes==0. If there is more than one AncDataElement() in the ancillary data of an MPEG-1/2 frame, all these AncDataElements() must directly follow each other.

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

Table 52 – Syntax of AncDataElement()

Syntax	No. of bits	Mnemonic
AncDataElement()		
{		
ancSyncword; /* 0x473 */	**11**	**bslbf**
ancCrcLen;	**1**	**uimsbf**
ancType;	**2**	**uimsbf**
ancStart;	**1**	**uimsbf**
ancStop;	**1**	**uimsbf**
cnt = **ancLenBytes**;	**8**	**uimsbf**
if (cnt==255) {		
cnt += **ancLenBytesAdd**;	**16**	**uimsbf**
}		
if (**ancCrcLen**==0) {		
ancCrcWord;	**8**	**uimsbf**
} else {		
ancCrcWord;	**16**	**uimsbf**
}		
ancCrcWord;	**8**	**uimsbf**
for (i=0; i<cnt; i++) {		
ancDataSegmentByte[i];	**8**	**bslbf**
}		
}		

ancSyncword	Identification syncword. Shall be 0x473.
ancCrcLen	Indicates the length of ancCrcWord: 0: 8 bit, 1: 16 bit
ancType	Indicates type of ancillary data, see following table:

Table 53 – ancType

ancType	**Meaning**
0x0	SaocDataFrame(0) (SAOC frame)
0x1	SaocDataFrame(1) (SAOC header and SAOC frame)
0x2	reserved
0x3	reserved

ancStart	Indicates if data segment begins a data block.
ancStop	Indicates if data segment ends a data block.
ancLenBytes	Number of bytes in data segment.
ancLenBytesAdd	Additional number of bytes in data segment, needed if the data segments contain 255 or more bytes.
ancCrcWord	ancCrcWord is defined by the generator polynomial G(x) = x^8+x^2+x+1 and the initial value for the CRC calculation is 0xFF if an 8 bit CRC is signalled by ancCrcLen. For the 16 bit CRC, the generator polynomial is G(x) = x^16 + x^15 + x^2 + 1 and the initial value is 0xFFFF. The CRC covers all bits in the AncDataElement() excluding ancSyncword, ancCrcLen and the ancCrcWord itself.
ancDataSegmentByte	The concatenation of all ancDataSegmentByte from consecutive AncDataElement(), starting from the AncDataElement with ancStart==1 up to and including the AncDataElement() with ancStop==1 forms one data block. In case a complete data block is contained in one AncDataElement(), it has ancStart==1 and ancStop==1. If ancType==0x0 or ancType==0x1 then this data block constitutes one SaocDataFrame() syntax element, padded at the end to obtain an integer number of bytes.

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

8.2.5 SaocDataFrame including optional header

The syntax of “SaocDataFrame()” is given in the table below:

Table 54 – Syntax of SaocDataFrame()

Syntax	No. of bits	Mnemonic
SaocDataFrame(saocHeaderFlag)		
{		
numSaocBits = 0;		
if (saocHeaderFlag) {		
saocTimeAlignFlag;	**1**	**uimsbf**
numSaocBits += 1;		
cnt = **saocHeaderLen**;	**7**	**uimsbf**
numSaocBits += 7;		
if (cnt==127) {		
cnt += **saocHeaderLenAdd**;	**16**	**uimsbf**
numSaocBits += 16;		
}		
tmpBits = SaocSpecificConfig();		Note 1
numFillBits = (8*cnt)-tmpBits;		
bsFillBits;	**numFillBits**	**bslbf**
numSaocBits += tmpBits+numFillBits;		
if (saocTimeAlignFlag) {		
saocTimeAlign;	**16**	**simsbf**
numSaocBits += 16;		
}		
}		
numSaocBits += SaocFrame();		Note 1
return (numSaocBits);		
}		
Note 1: SaocSpecificConfig() and SaocFrame() return the number of bits read.		

The following semantics apply:

saocHeaderFlag	Indicates the presence of a SaocSpecificConfig() element.
saocHeaderLen	Indicates the length in bytes of SaocSpecificConfig() plus the following fill bits.
saocHeaderLenAdd	Additional length in bytes of SaocSpecificConfig() plus the following fill bits, needed if this length is 127 or more bytes.
bsFillBits	Fill bits, to be ignored.
saocTimeAlignFlag	Indicates the presence of a saocTimeAlign element. If saocTimeAlignFlag==0 then saocTimeAlign is set to the default value 0.
saocTimeAlign	Signed integer in the range -32768...32767, identifying the PCM sample in the time domain output frame of the downmix decoder that corresponds to the beginning of the present SAOC frame (i.e., the first sample of the time domain input signal that is consumed by the SAOC decoding process for the present SAOC frame). The position of the first sample of the output frame is represented as 0. The present SAOC frame is the first SAOC frame that is completed (i.e., ancStop==1) in the present downmix decoder frame. If HE-AAC is used as downmix coder, the time domain output frame of the AAC core decoder (delay-free upsampled by a factor of two in case of normal operation of SAOC with 64 QMF bands) is considered here.

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

8.3 Transport of SAOC data over PCM channels

8.3.1 Introduction

SAOC data can be conveyed over traditional PCM audio channels through adding a noise signal with specific characteristics to the PCM data. This noise signal, which actually is a randomized version of the SAOC data, can be rendered inaudible by employing subtractive dithered noise shaping controlled by the masked threshold.

8.3.2 Syntax

According to 7.3.2 of ISO/IEC 23003-1:2007

8.3.3 Semantics

According to 7.3.3 of ISO/IEC 23003-1:2007

With the following changes:

bsBDType Indicates the type of buried data according to 7.3.3 in ISO/IEC 23003-1:2007, with the addition of the two types 4 and 5 as specified in the following table.

Table 55 – bsBDType

bsBDType	Type of data
0 ... 3	see ISO/IEC 23003-1:2007
4	SAOC frame, i.e., SaocDataFrame(0)
5	SAOC header+frame, i.e., SaocDataFrame(1)
6 … 7	see ISO/IEC 23003-1:2007

bsBDBytes[b] One byte of the payload of BuriedDataElement(). The concatenation of all bsBDBytes from one BuriedDataElement() forms one data block. If bsBDType==0x4 or bsBDType==0x5 then this data block constitutes one SaocDataFrame() syntax element, padded at the end to obtain an integer number of bytes.

8.3.4 Decoding process

8.3.4.1 Introduction

The decoding process consists of two steps. In the first step the payload from the buried data channel is decoded. In the second step the payload is converted into a complete SAOC audio bitstream.

In the next subclauses the decoding of the payload is described.

8.3.4.2 Extraction of buried data payload (including header information)

According to 7.3.4.2 of ISO/IEC 23003-1:2007

8.3.4.3 Synchronization

According to 7.3.4.3 of ISO/IEC 23003-1:2007

8.3.4.4 Allocation

According to 7.3.4.4 of ISO/IEC 23003-1:2007

8.3.4.5 Retrieval of buried data payload

According to 7.3.4.5 of ISO/IEC 23003-1:2007

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

8.3.4.6 CRC check

According to 7.3.4.6 of ISO/IEC 23003-1:2007

8.3.4.7 SAOC decoding

In the case that the buried data payload is of type SAOC frame or SAOC header+frame, the SaocDataFrame data buried in one PCM buried data frame shall be applied to a PCM frame having the same length as the PCM buried data frame and having an offset in PCM samples specified by the value of saocTimeAlign (see 7.2.5). Furthermore the first frame shall contain the SaocSpecificConfig, i.e. bsBDType shall have the value 5, and it is recommended, for an encoder, to add the SaocSpecificConfig to a BuriedDataFrame(), at regular time intervals, such that a decoder is also able to start decoding from another position in the stream.

9 Transport of predefined rendering information

9.1 Introduction

The preset rendering concept allows to start playback with some initial predefined settings and (if more than one set of the rendering data is available) instantaneously switch between them. A single configuration data structure can contain several independent descriptions of the predefined output audio scenes. The number of the available rendering options is defined by the bitstream syntax variable **bsNumPresets**. Depending on the application specific scenario the preset information specification can also be omitted. Each preset element contains a simple text string label stored in the text variable **bsPresetLabel** and rendering data, which include the relative output reproduction level and panning position. In view of the fact that the SAOC encoded audio objects can be represented by the mono, stereo or multi-channel signals, the preset rendering information relates to each individual channel of the audio object.

The SAOC bitstream variable **bsPresetMatrix** determines which type of preset data format is applied. More precisely, **bsPresetMatrix** = 1 means that the default matrix-based representation format is applied, whereas **bsPresetMatrix** = 0 means that the preset data have a specific format. The matrix-based format represents the rendering information in the native form used for the SAOC transcoding/decoding. The dequantized rendering parameters obtained from the variable **bsPresetMatrixElements**[i][j] can be unambiguously fed into the rendering interface of the SAOC transcoder/decoder using the following mapping information, see Table 56.

Table 56 – Matrix-based preset parameter quantization table

idx	0	1	2	3	4	5	6	7
bsPresetMatrixElements[i][idx]	-150 dB	-24 dB	-20 dB	-18 dB	-16 dB	-14 dB	-12 dB	-10 dB
idx	8	9	10	11	12	13	14	15
bsPresetMatrixElements[i][idx]	-8 dB	-7 dB	-6 dB	-5 dB	-4 dB	-3 dB	-2 dB	-1 dB
idx	16	17	18	19	20	21	22	23
bsPresetMatrixElements[i][idx]	0 dB	1 dB	2 dB	3 dB	4 dB	5 dB	6 dB	7 dB
idx	24	25	26	27	28	29	30	31
bsPresetMatrixElements[i][idx]	8 dB	10 dB	12 dB	14 dB	16 dB	18 dB	20 dB	24 dB
idx	0	1	2	3	4	5	6	7
bsPresetMatrixElements[i][idx]	-inf	-20 dB	-16 dB	-12 dB	-9 dB	-6 dB	-4 dB	-2 dB
idx	8	9	10	11	12	13	14	15
bsPresetMatrixElements[i][idx]	0 dB	2 dB	4 dB	6 dB	9 dB	12 dB	16 dB	20 dB

The proposed bitstream structure permits a support of any other alternative data formats containing preset information. Description of such upmix representation approaches remains informative. The specific preset description formats are recognized by the particular identifier label **bsPresetUserDataIdentifier**. The variable **bsPresetUserDataLen** contains the length information for the preset data included into PresetUserDataContainer() bitstream syntax element. This allows the SAOC transcoder/decoder to skip the corresponding part of the preset configuration data if this specific description format is not relevant (not supported) by the transcoder/decoder.

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

9.2 Rendering information description file format

The preset rendering concept allows to store and exchange user-defined rendering information. The binary file structure is based on the bitstream syntax elements ObjectMetaData() and PresetConfig(). The bitstream variable **bsNumObjects** provides an information about the number of audio objects presented in the described scene and used in ObjectMetaData() and PresetConfig().

The information stored in the variable bsNumObjects and in the element ObjectMetaData() can be also used for identification and compatibility check. For example, the SAOC transcoder/decoder can use the number of audio objects, metadata and other corresponding information obtained from the SAOC bitstream for these purposes.

If it is necessary, the bitstream syntax element RenderingInformation() determining the binary file structure can be encapsulated into a higher level structure depending on the specific realization approach.

Table 57 – Syntax of RenderingInformation()

Syntax	No. of bits	Mnemonic
RenderingInformation()		
{		
bsNumObjects;	**5**	**uimsbf** Note 1
ByteAlign();		
ObjectMetaData();		Note 1
ByteAlign();		
PresetConfig();		Note 1
ByteAlign();		
}		
Note 1: Described in 6.1.		

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

Annex A
(normative)

Tables

A.1 Huffman tables

The function *1Dhuff_dec()* is used as:

data = 1Dhuff_dec(hcod_huff, codeword),

where *hcod_huff* is the selected Huffman table and *codeword* is the word read from the bitstream. The returned value *data,* is a Huffman table index corresponding to a specific code word, with the Huffman table offset value subtracted. The offset value is specified for each Huffman table in Table A.1.

Similarly the function *2Dhuff_dec()* is used as:

(data0, data1) = 2Dhuff_dec(hcod_huff, codeword),

where *hcod_huff* is the selected Huffman table and *codeword* is the word read from the bitstream. The returned values *data0* and *data1* is the corresponding Huffman table index Idx0 and Idx1 corresponding to a specific code word with the Huffman table offset value subtracted. The offset value is specified for each Huffman table in Table A.1.

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

Table A.1 – Huffman table overview

Table name	Offset	LAV	Notes
hcodFirstBand_OLD	15	15	
hcodFirstBand_NRG	0	12	Note 1
hcod1D_OLD_YY	0	15	Note 1
hcod1D_NRG_YY	0	12	Note 1
hcod2D_NRG_DT_ZZ_LL_escape	N/A	N/A	
hcod2D_NRG_DT_ZZ_LL_escape	N/A	N/A	
hcod2D_NRG_DT_ZZ_LL_escape	N/A	N/A	
hcod2D_NRG_DF_ZZ_03	0	3	Note 1
hcod2D_NRG_DF_ZZ_05	0	5	Note 1
hcod2D_NRG_DF_ZZ_07	0	7	Note 1
hcod2D_NRG_DF_ZZ_09	0	9	Note 1
hcod2D_OLD_YY_ZZ_01	0	1	Note 1
hcod2D_OLD_YY_ZZ_03	0	3	Note 1
hcod2D_OLD_YY_ZZ_05	0	5	Note 1
hcod2D_OLD_YY_ZZ_07	0	7	Note 1
hcod2D_NRG_DT_ZZ_03	0	3	Note 1
hcod2D_NRG_DT_ZZ_06	0	6	Note 1
hcod2D_NRG_DT_ZZ_09	0	9	Note 1
hcod2D_NRG_DT_ZZ_12	0	12	Note 1
hcod1D_ICC_Diff	0	7	Note 1
hcodLavIdx	0	N/A	
Note 1: Data can only have non-negative values for this table.			

© ISO/IEC 2010 – All rights reserved

Table A.2 – hcodFirstBand_OLD

Index	length	codeword (hexadecimal)	Index	length	codeword (hexadecimal)
0	1	`0x00`	8	6	`0x22`
1	5	`0x1e`	9	5	`0x10`
2	5	`0x12`	10	6	`0x2e`
3	5	`0x14`	11	6	`0x2f`
4	5	`0x15`	12	6	`0x23`
5	5	`0x16`	13	6	`0x26`
6	5	`0x1f`	14	6	`0x27`
7	3	`0x06`	15	4	`0x0e`

Table A.3 – hcodFirstBand_NRG

Index	length	codeword (hexadecimal)	Index	length	codeword (hexadecimal)
0	11	`0x67e`	32	8	`0x0ce`
1	12	`0x8fe`	33	7	`0x007`
2	12	`0x8ff`	34	7	`0x046`
3	11	`0x67f`	35	7	`0x05e`
4	11	`0x2fe`	36	7	`0x05f`
5	11	`0x47e`	37	7	`0x077`
6	10	`0x17e`	38	7	`0x066`
7	11	`0x2ff`	39	6	`0x002`
8	10	`0x23e`	40	6	`0x00e`
9	10	`0x33e`	41	6	`0x027`
10	9	`0x0be`	42	5	`0x002`
11	9	`0x11e`	43	5	`0x00a`
12	9	`0x19e`	44	5	`0x012`
13	8	`0x05e`	45	5	`0x018`
14	8	`0x08e`	46	5	`0x01c`
15	7	`0x006`	47	5	`0x016`
16	7	`0x02e`	48	5	`0x00f`
17	7	`0x076`	49	5	`0x000`
18	6	`0x006`	50	6	`0x037`
19	6	`0x012`	51	5	`0x004`
20	6	`0x036`	52	5	`0x00d`
21	5	`0x00e`	53	5	`0x01a`
22	5	`0x014`	54	5	`0x010`
23	5	`0x00c`	55	5	`0x005`
24	6	`0x03e`	56	5	`0x008`
25	6	`0x03f`	57	6	`0x032`
26	5	`0x015`	58	6	`0x007`
27	5	`0x006`	59	6	`0x022`
28	6	`0x03c`	60	6	`0x013`
29	6	`0x016`	61	6	`0x03a`
30	6	`0x026`	62	6	`0x00f`
31	6	`0x02e`	63	6	`0x03d`

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

Table A.4 – hcod1D_OLD_YY

Index	DF		DT	
	length	codeword	length	codeword
0	1	0x000	1	0x000
1	3	0x006	2	0x002
2	3	0x004	4	0x00e
3	4	0x00e	4	0x00c
4	4	0x00a	5	0x01e
5	5	0x01e	5	0x01a
6	5	0x016	6	0x03e
7	5	0x01f	5	0x01b
8	7	0x05c	8	0x0fc
9	8	0x0be	9	0x1fc
10	7	0x05d	9	0x1fa
11	8	0x0bc	9	0x1fb
12	9	0x17e	10	0x3fe
13	9	0x17a	10	0x3ff
14	9	0x17b	9	0x1fd
15	9	0x17f	9	0x1fe

Table A.5 – hcod1D_NRG_YY

Index	DF		DT	
	length	codeword	length	codeword
0	3	0x000006	2	0x000000
1	2	0x000000	2	0x000002
2	3	0x000007	3	0x000006
3	3	0x000004	3	0x000002
4	3	0x000002	4	0x00000e
5	4	0x00000a	5	0x00001e
6	4	0x000006	5	0x00000e
7	5	0x000016	5	0x00000c
8	5	0x00000e	6	0x00003e
9	6	0x00002e	6	0x00001e
10	6	0x00001e	6	0x00001a
11	7	0x00005e	7	0x00007e
12	7	0x00003e	7	0x00003e
13	8	0x0000be	7	0x000036
14	9	0x00017e	8	0x0000fe
15	9	0x0000fe	8	0x00007e
16	9	0x0000fc	8	0x00006e
17	10	0x0002fe	9	0x0000fe
18	10	0x0001fe	9	0x0000de
19	11	0x0005fe	10	0x0003fe
20	10	0x0001fa	10	0x0003fc
21	11	0x0003f6	10	0x0001fe
22	11	0x0003fe	10	0x0001be
23	11	0x0003f7	11	0x0007fa
24	12	0x000bfe	11	0x0003fe
25	12	0x0007fe	11	0x00037e
26	13	0x0017fe	11	0x00037f
27	14	0x001ffe	12	0x000ffc
28	14	0x001fff	12	0x000ffe
29	14	0x001ffc	12	0x000ff6
30	15	0x005ffe	12	0x0007fe
31	15	0x003ffa	13	0x000ffe
32	17	0x00ffee	13	0x000fff

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

Table A.5 (*continued*)

Index	DF		DT	
	length	codeword	length	codeword
33	15	0x005fff	14	0x003ffc
34	22	0x1ffde8	15	0x007ffc
35	16	0x007ff6	15	0x007ffd
36	14	0x002ffe	14	0x003ff6
37	22	0x1ffde9	15	0x007ffe
38	22	0x1ffdea	15	0x007fea
39	22	0x1ffdeb	14	0x003fdc
40	22	0x1ffdec	16	0x00fff6
41	22	0x1ffded	14	0x003ff7
42	22	0x1ffdee	16	0x00fffe
43	22	0x1ffdef	15	0x007feb
44	22	0x1ffdf0	17	0x01fffe
45	22	0x1ffdf1	14	0x003fde
46	22	0x1ffdf2	14	0x003fdd
47	22	0x1ffdf3	14	0x003ff4
48	22	0x1ffdf4	18	0x03fffe
49	22	0x1ffdf5	24	0xfffffe
50	22	0x1ffdf6	14	0x003fdf
51	22	0x1ffdf7	16	0x00fff4
52	22	0x1ffdf8	16	0x00fff5
53	22	0x1ffdf9	24	0xffffff
54	22	0x1ffdfa	16	0x00fff7
55	22	0x1ffdfb	20	0x0ffffe
56	22	0x1ffdfc	19	0x07fffe
57	22	0x1ffdfd	23	0x7ffff8
58	22	0x1ffdfe	23	0x7ffff9
59	22	0x1ffdff	23	0x7ffffa
60	21	0x0ffef0	23	0x7ffffb
61	21	0x0ffef1	23	0x7ffffc
62	21	0x0ffef2	23	0x7ffffd
63	21	0x0ffef3	23	0x7ffffe

Table A.6 – hcod2D_OLD_YY_ZZ_LL_escape

LL	DF/FP		DT/FP	
	length	codeword	length	codeword
03		N/A		N/A
06		N/A		N/A
09		N/A		N/A
12		N/A		N/A

Table A.7 – hcod2D_NRG_DF_ZZ_LL_escape

LL	DF/FP	
	length	codeword
03		N/A
05		N/A
07		N/A
09		N/A

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

Table A.8 – hcod2D_NRG_DT_ZZ_LL_escape

LL	DT/FP	
	length	codeword
03		N/A
06		N/A
09		N/A
12		N/A

Table A.9 – hcod2D_OLD_YY_ZZ_03

Idx0	Idx1	DF/FP		DT/FP	
		length	codeword	length	codeword
0	0	2	0x002	1	0x000
0	1	3	0x006	6	0x03e
0	2	3	0x002	8	0x0de
0	3	3	0x003	8	0x0dc
1	0	3	0x000	5	0x01e
1	1	4	0x00e	4	0x00e
1	2	4	0x002	6	0x036
1	3	5	0x01e	7	0x07e
2	0	6	0x03e	6	0x032
2	1	6	0x00e	7	0x07f
2	2	6	0x00f	5	0x018
2	3	7	0x07e	5	0x01a
3	0	9	0x1fe	8	0x0df
3	1	9	0x1ff	8	0x0dd
3	2	8	0x0fe	6	0x033
3	3	5	0x006	2	0x002

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

Table A.10 – hcod2D_OLD_YY_ZZ_06

Idx0	Idx1	DF/FP		DT/FP	
		length	codeword	length	codeword
0	0	1	0x0000	1	0x0000
0	1	5	0x001a	7	0x007e
0	2	6	0x0032	9	0x01ea
0	3	7	0x0066	11	0x07ae
0	4	7	0x0056	13	0x1ef6
0	5	8	0x00ce	16	0xf7fe
0	6	9	0x01f4	16	0xf7ff
1	0	6	0x002e	6	0x003e
1	1	4	0x0008	4	0x000c
1	2	6	0x002a	8	0x00f2
1	3	7	0x006e	9	0x015e
1	4	8	0x00f4	12	0x0f7e
1	5	9	0x01ea	14	0x3dfe
1	6	13	0x1f7e	8	0x00da
2	0	7	0x0057	6	0x002c
2	1	6	0x0026	6	0x003a
2	2	5	0x0016	5	0x0014
2	3	8	0x00fe	10	0x03d6
2	4	8	0x00a6	12	0x0f7a
2	5	12	0x0fbc	8	0x00de
2	6	12	0x0fbd	7	0x005a
3	0	9	0x01f5	7	0x006e
3	1	7	0x0052	6	0x002e
3	2	8	0x00f6	8	0x00f6
3	3	6	0x0028	7	0x0056
3	4	8	0x00f7	9	0x01e6
3	5	8	0x00fc	7	0x005b
3	6	8	0x00ff	6	0x002f
4	0	12	0x0fba	8	0x00f4
4	1	11	0x07dc	7	0x006c
4	2	12	0x0fbe	8	0x00ae
4	3	8	0x00a7	11	0x07af
4	4	6	0x003c	6	0x002a
4	5	7	0x006f	7	0x007f
4	6	7	0x005e	6	0x003b
5	0	14	0x3efe	9	0x01ee
5	1	12	0x0fbb	8	0x00df
5	2	9	0x01f6	13	0x1efe
5	3	8	0x00fd	11	0x07be
5	4	7	0x007c	8	0x00db
5	5	5	0x0018	5	0x001c
5	6	6	0x0036	5	0x001a
6	0	14	0x3eff	9	0x015f
6	1	9	0x01eb	15	0x7bfe
6	2	8	0x00cf	13	0x1ef7
6	3	7	0x005f	11	0x07bc
6	4	6	0x0027	9	0x01e7
6	5	5	0x0012	7	0x0078
6	6	4	0x000e	3	0x0004

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

Table A.11 – hcod2D_OLD_YY_ZZ_09

Idx0	Idx1	DF/FP		DT/FP	
		length	codeword	length	codeword
0	0	1	0x00000	1	0x00000
0	1	6	0x00036	7	0x00042
0	2	7	0x0007e	9	0x0015a
0	3	7	0x0005c	11	0x007da
0	4	8	0x000ec	12	0x009ee
0	5	8	0x000c6	14	0x027fe
0	6	7	0x00068	15	0x06f7e
0	7	6	0x0002a	14	0x0279e
0	8	9	0x001d2	19	0x6f7fe
0	9	10	0x003b6	17	0x13dfe
1	0	7	0x00062	6	0x00022
1	1	5	0x00016	5	0x0001e
1	2	7	0x00066	8	0x000ac
1	3	8	0x000ee	10	0x0037e
1	4	8	0x000ae	11	0x004f6
1	5	9	0x0018e	13	0x01bde
1	6	8	0x000d2	13	0x013fe
1	7	10	0x00278	16	0x0defe
1	8	10	0x0031e	19	0x6f7ff
1	9	6	0x00030	11	0x006fa
2	0	7	0x00056	7	0x0005a
2	1	7	0x0006a	6	0x00020
2	2	6	0x00032	6	0x0003a
2	3	8	0x000ce	10	0x003ec
2	4	9	0x001da	11	0x004fa
2	5	8	0x000ea	11	0x004f2
2	6	10	0x0021e	15	0x04f7c
2	7	12	0x00d7e	17	0x1bdfe
2	8	12	0x009f6	11	0x006fb
2	9	11	0x005fe	10	0x0027c
3	0	8	0x0008e	7	0x0004e
3	1	7	0x00040	7	0x0007c
3	2	6	0x00037	8	0x000fe
3	3	6	0x00022	7	0x0006e
3	4	7	0x0004c	10	0x0037c
3	5	10	0x0033e	13	0x013de
3	6	11	0x0077e	14	0x0279f
3	7	11	0x0043e	11	0x0056e
3	8	11	0x005f6	10	0x002be
3	9	11	0x0043f	6	0x00036
4	0	11	0x005f2	8	0x000ff
4	1	10	0x0033f	7	0x0005e
4	2	10	0x003ae	8	0x000fa
4	3	9	0x0017e	7	0x0004a
4	4	9	0x001ae	10	0x0027e
4	5	6	0x00028	14	0x037be
4	6	10	0x003a6	12	0x00fba
4	7	10	0x002fa	10	0x002b6
4	8	10	0x0027a	6	0x00023
4	9	11	0x004fa	7	0x00072
5	0	11	0x0075e	8	0x000e6
5	1	11	0x0074e	7	0x00054
5	2	11	0x0074f	7	0x00076
5	3	10	0x0023e	11	0x007de
5	4	10	0x00279	12	0x00dee
5	5	9	0x0019e	11	0x0056f

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

Table A.11 (*continued*)

Idx0	Idx1	DF/FP		DT/FP	
		length	codeword	length	codeword
5	6	10	0x0035e	11	0x006f6
5	7	9	0x0011e	7	0x0005b
5	8	10	0x002fe	7	0x00043
5	9	10	0x003be	7	0x00077
6	0	11	0x0075f	8	0x000e7
6	1	11	0x0077f	6	0x0002c
6	2	10	0x0027c	10	0x002bf
6	3	11	0x005f7	11	0x004fe
6	4	11	0x005ff	14	0x027ff
6	5	10	0x0023f	13	0x01bfe
6	6	5	0x0001c	6	0x0002e
6	7	7	0x00042	8	0x000ae
6	8	7	0x0004a	7	0x00055
6	9	7	0x0004e	7	0x0007e
7	0	10	0x0027e	5	0x0001a
7	1	11	0x006be	10	0x0037a
7	2	10	0x0027f	11	0x006fe
7	3	10	0x0027b	15	0x04f7d
7	4	10	0x003b7	14	0x037fe
7	5	7	0x00041	11	0x007df
7	6	9	0x001de	10	0x00278
7	7	6	0x00024	6	0x00024
7	8	7	0x00046	7	0x0005f
7	9	7	0x0004d	6	0x00026
8	0	12	0x00d7f	11	0x004fb
8	1	11	0x005f3	11	0x007db
8	2	10	0x002f8	17	0x13dff
8	3	9	0x0010e	15	0x04f7e
8	4	8	0x000e8	12	0x009e6
8	5	8	0x00086	12	0x00dfe
8	6	8	0x000d3	11	0x007dc
8	7	7	0x0004b	9	0x001bc
8	8	6	0x0003e	5	0x00014
8	9	7	0x0007f	6	0x00038
9	0	12	0x009f7	12	0x00fbb
9	1	11	0x0063e	18	0x37bfe
9	2	11	0x0063f	16	0x09efe
9	3	8	0x000d6	14	0x037ff
9	4	9	0x001d6	13	0x013ce
9	5	8	0x000af	12	0x009fe
9	6	7	0x0005d	10	0x0027a
9	7	7	0x0005e	9	0x0015e
9	8	6	0x00029	7	0x0004b
9	9	5	0x0001e	4	0x0000c

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

Table A.12 – hcod2D_OLD_YY_ZZ_12

Idx0	Idx1	DF/FP		DT/FP	
		length	codeword	length	codeword
0	0	4	0x00000a	2	0x000002
0	1	8	0x0000d6	7	0x000070
0	2	8	0x000074	8	0x000036
0	3	3	0x000000	9	0x00006e
0	4	5	0x000004	11	0x00035e
0	5	6	0x00001a	12	0x000d7e
0	6	7	0x000046	13	0x0003fc
0	7	10	0x0002de	16	0x0045fe
0	8	11	0x0006fc	15	0x0022fe
0	9	7	0x000036	22	0x3cfbf4
0	10	4	0x00000e	22	0x3cfbf5
0	11	6	0x000026	22	0x3cfbf6
0	12	7	0x00006a	22	0x3cfbf7
1	0	7	0x00004e	6	0x00001c
1	1	7	0x00005c	4	0x000000
1	2	7	0x00004a	7	0x000028
1	3	6	0x000022	9	0x0001fa
1	4	4	0x00000c	10	0x0001f6
1	5	6	0x000024	11	0x0000fe
1	6	7	0x00003e	13	0x001d76
1	7	8	0x00006e	14	0x0007fa
1	8	8	0x000075	14	0x0007fb
1	9	7	0x00004f	22	0x3cfbf8
1	10	6	0x000036	22	0x3cfbf9
1	11	4	0x00000f	22	0x3cfbfa
1	12	7	0x00005e	9	0x0001be
2	0	8	0x0000de	7	0x00007a
2	1	7	0x00006e	6	0x00001e
2	2	6	0x00000a	6	0x00003e
2	3	6	0x00001e	9	0x0001e6
2	4	5	0x000006	9	0x000066
2	5	4	0x000004	11	0x00022e
2	6	7	0x000016	12	0x0006be
2	7	11	0x00069e	15	0x0079f6
2	8	7	0x00005d	14	0x003cfa
2	9	7	0x000068	22	0x3cfbfb
2	10	6	0x00001c	16	0x00f3ee
2	11	5	0x00000c	9	0x0001fe
2	12	5	0x00000a	8	0x00006e
3	0	10	0x0002fa	7	0x00003a
3	1	9	0x00017c	6	0x00000e
3	2	6	0x00000e	7	0x000068
3	3	6	0x00000f	7	0x000078
3	4	6	0x00002c	10	0x0001f7
3	5	5	0x000010	11	0x00022c
3	6	11	0x00047c	12	0x000eba
3	7	10	0x0000ba	13	0x0008b6
3	8	11	0x00037c	13	0x0003fe
3	9	12	0x0006fa	22	0x3cfbfc
3	10	19	0x02ebf4	9	0x0001d4
3	11	19	0x02ebf5	8	0x0000de
3	12	19	0x02ebf6	7	0x00002a
4	0	11	0x0005ae	7	0x000024
4	1	10	0x0002d6	6	0x00000a
4	2	10	0x0002d4	7	0x00001a
4	3	10	0x0001de	8	0x00007e

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

Table A.12 (*continued*)

Idx0	Idx1	DF/FP		DT/FP	
		length	codeword	length	codeword
4	4	9	0x0000be	8	0x0000f6
4	5	11	0x00047d	11	0x0001be
4	6	10	0x000176	11	0x0001bc
4	7	11	0x0004bc	12	0x0006bf
4	8	12	0x000966	15	0x006bfe
4	9	19	0x02ebf7	9	0x00003e
4	10	19	0x02ebf8	8	0x00005e
4	11	19	0x02ebf9	7	0x000029
4	12	19	0x02ebfa	7	0x000018
5	0	11	0x0002ee	7	0x000026
5	1	10	0x000170	7	0x000074
5	2	11	0x0005fe	7	0x00001e
5	3	11	0x0005f6	8	0x000046
5	4	11	0x000476	9	0x0000be
5	5	9	0x00017e	8	0x0000dc
5	6	11	0x0004b6	11	0x0006be
5	7	12	0x000bee	13	0x001e7c
5	8	11	0x000472	10	0x00038e
5	9	11	0x0005be	9	0x0001de
5	10	11	0x0005af	8	0x0000f7
5	11	12	0x0008fe	8	0x0000f2
5	12	13	0x001a7e	7	0x00002b
6	0	13	0x001a7f	8	0x0000fc
6	1	12	0x000d3e	7	0x000069
6	2	11	0x0002e2	7	0x00000a
6	3	12	0x000dfe	9	0x0001fb
6	4	10	0x0000bb	9	0x0001c6
6	5	11	0x0005aa	10	0x0003ce
6	6	9	0x00005c	10	0x00035e
6	7	12	0x000dff	11	0x0007f6
6	8	11	0x00037a	9	0x000067
6	9	11	0x000474	8	0x000047
6	10	12	0x0006f6	9	0x0001fc
6	11	12	0x0006fb	7	0x00001f
6	12	12	0x0005d6	7	0x00006a
7	0	14	0x00175e	8	0x00006f
7	1	13	0x000bae	7	0x000025
7	2	13	0x0016ae	8	0x00006a
7	3	12	0x000dfc	8	0x0000e6
7	4	12	0x000d36	9	0x0000d6
7	5	12	0x0006fe	11	0x00075c
7	6	12	0x000d37	13	0x001e7e
7	7	9	0x00011e	9	0x0001ff
7	8	10	0x0001df	9	0x0001ae
7	9	11	0x0002ef	9	0x0000f8
7	10	11	0x000473	8	0x0000e2
7	11	11	0x000475	7	0x00002e
7	12	11	0x00047e	6	0x000004
8	0	19	0x02ebfb	8	0x000044
8	1	19	0x02ebfc	7	0x00000e
8	2	19	0x02ebfd	8	0x0000ee
8	3	19	0x02ebfe	8	0x00001e
8	4	19	0x02ebff	10	0x0003fa
8	5	12	0x000bef	14	0x00117e
8	6	11	0x0004be	13	0x001d77
8	7	10	0x0001bc	11	0x00075e

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

Table A.12 (*continued*)

Idx0	Idx1	DF/FP		DT/FP	
		length	codeword	length	codeword
8	8	10	0x00034c	8	0x000032
8	9	11	0x00069a	9	0x0001bf
8	10	10	0x000174	8	0x0000dd
8	11	10	0x00017e	7	0x00003b
8	12	10	0x000258	6	0x00000b
9	0	18	0x0175f0	9	0x0001d5
9	1	18	0x0175f1	7	0x000034
9	2	18	0x0175f2	8	0x00006c
9	3	18	0x0175f3	9	0x0000f9
9	4	12	0x0008ff	18	0x03cfbe
9	5	11	0x0004b7	13	0x0008b7
9	6	10	0x000172	12	0x000ebe
9	7	11	0x000477	12	0x000ebf
9	8	11	0x0006fd	10	0x0001ae
9	9	9	0x00012e	7	0x000027
9	10	10	0x00034e	7	0x00000b
9	11	9	0x0000bc	7	0x000072
9	12	9	0x0000bd	6	0x000016
10	0	18	0x0175f4	9	0x0001d6
10	1	18	0x0175f5	8	0x00007f
10	2	18	0x0175f6	9	0x0001df
10	3	12	0x0006ff	22	0x3cfbfd
10	4	12	0x000dfd	13	0x0008be
10	5	11	0x0004bf	13	0x001e7f
10	6	11	0x0002fe	12	0x00045a
10	7	11	0x0005bf	11	0x0001bd
10	8	10	0x00025a	11	0x0007f7
10	9	9	0x00005e	9	0x0000bf
10	10	8	0x0000b4	6	0x000010
10	11	9	0x0001ae	6	0x000006
10	12	9	0x0001be	6	0x000036
11	0	18	0x0175f7	9	0x0001ac
11	1	18	0x0175f8	9	0x0001ad
11	2	12	0x0006f7	22	0x3cfbfe
11	3	11	0x0002e3	22	0x3cfbff
11	4	11	0x0004b2	14	0x0035fe
11	5	11	0x0002ea	16	0x00d7fe
11	6	11	0x0004bd	13	0x0003ff
11	7	10	0x000173	11	0x0001bf
11	8	10	0x0002fe	10	0x00007e
11	9	9	0x0000ee	9	0x0000fa
11	10	9	0x0001af	8	0x0000e7
11	11	7	0x00003f	5	0x00000c
11	12	8	0x0000b6	5	0x000004
12	0	18	0x0175f9	10	0x00038f
12	1	13	0x0016af	21	0x1e7df8
12	2	12	0x000967	16	0x0045ff
12	3	11	0x0002ff	21	0x1e7df9
12	4	12	0x000b56	16	0x00d7ff
12	5	11	0x00037e	17	0x01e7de
12	6	11	0x0005ff	13	0x001afe
12	7	10	0x000238	12	0x00045e
12	8	9	0x00005f	11	0x00079e
12	9	9	0x00016e	9	0x00008a
12	10	8	0x000076	8	0x00006d
12	11	8	0x0000d2	7	0x000076
12	12	6	0x000016	4	0x00000c

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

Table A.13 – hcod2D_NRG_DF_ZZ_03

Idx0	Idx1	DF/FP	
		length	codeword
0	0	4	0x0c
0	1	4	0x0d
0	2	4	0x08
0	3	4	0x06
1	0	4	0x0e
1	1	3	0x00
1	2	4	0x0a
1	3	4	0x09
2	0	4	0x07
2	1	4	0x02
2	2	5	0x1e
2	3	4	0x0b
3	0	5	0x06
3	1	5	0x07
3	2	5	0x1f
3	3	3	0x02

Table A.14 – hcod2D_NRG_DF_ZZ_05

Idx0	Idx1	DF/FP	
		length	codeword
0	0	4	0x0006
0	1	4	0x0008
0	2	5	0x0006
0	3	7	0x004a
0	4	8	0x0092
0	5	9	0x005e
1	0	4	0x0007
1	1	3	0x0000
1	2	5	0x0016
1	3	6	0x0026
1	4	7	0x0016
1	5	8	0x002e
2	0	5	0x0008
2	1	5	0x0017
2	2	5	0x0007
2	3	7	0x0048
2	4	7	0x002e
2	5	7	0x007e
3	0	7	0x004b
3	1	6	0x0016
3	2	7	0x004e
3	3	7	0x007f
3	4	6	0x003e
3	5	5	0x000a
4	0	8	0x0093
4	1	8	0x009e
4	2	7	0x002f
4	3	5	0x0004
4	4	5	0x001e
4	5	4	0x000a
5	0	9	0x005f
5	1	8	0x009f
5	2	6	0x000a
5	3	5	0x0009
5	4	4	0x000e
5	5	3	0x0006

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

Table A.15 – hcod2D_NRG_DF_ZZ_07

Idx0	Idx1	DF/FP	
		length	codeword
0	0	5	0x00018
0	1	5	0x0001a
0	2	5	0x0000a
0	3	6	0x00002
0	4	7	0x0002e
0	5	8	0x000ce
0	6	6	0x00008
0	7	9	0x000bc
1	0	5	0x0001e
1	1	4	0x00008
1	2	5	0x00014
1	3	6	0x00026
1	4	7	0x0004a
1	5	7	0x0007c
1	6	7	0x00012
1	7	7	0x0001e
2	0	5	0x00002
2	1	5	0x00015
2	2	5	0x00003
2	3	6	0x00016
2	4	7	0x0004e
2	5	8	0x0009e
2	6	9	0x00176
2	7	9	0x001f6
3	0	6	0x00024
3	1	5	0x00000
3	2	6	0x00003
3	3	7	0x0005c
3	4	7	0x0003c
3	5	8	0x000cf
3	6	8	0x000fa
3	7	8	0x000ba
4	0	8	0x000fe
4	1	7	0x0003e
4	2	7	0x0001f
4	3	6	0x0002c
4	4	7	0x00013
4	5	7	0x00066
4	6	7	0x0005e
4	7	6	0x0000e
5	0	9	0x000be
5	1	8	0x00096
5	2	9	0x001f7
5	3	8	0x000ff
5	4	7	0x0005f
5	5	6	0x0000a
5	6	6	0x0002d
5	7	5	0x0000e
6	0	10	0x0017a
6	1	9	0x000bf
6	2	8	0x00097
6	3	7	0x0003d
6	4	7	0x0007e
6	5	6	0x00032
6	6	5	0x0001b
6	7	4	0x00006
7	0	10	0x0017b
7	1	9	0x00177
7	2	8	0x0009f
7	3	7	0x0003f
7	4	6	0x0000b
7	5	5	0x00006
7	6	4	0x00004
7	7	4	0x0000e

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

Table A.16 – hcod2D_NRG_DF_ZZ_09

Idx0	Idx1	DF/FP	
		length	codeword
0	0	5	0x000006
0	1	5	0x000010
0	2	6	0x000032
0	3	7	0x000068
0	4	8	0x0000d6
0	5	8	0x000066
0	6	10	0x0002fe
0	7	10	0x0002de
0	8	11	0x00077e
0	9	12	0x000efe
1	0	5	0x000011
1	1	4	0x000002
1	2	5	0x000012
1	3	6	0x000036
1	4	7	0x00006a
1	5	7	0x000026
1	6	8	0x00004e
1	7	8	0x00006a
1	8	9	0x000166
1	9	11	0x00037e
2	0	6	0x000037
2	1	5	0x00000e
2	2	5	0x00000a
2	3	6	0x00001e
2	4	7	0x000034
2	5	8	0x0000e0
2	6	8	0x0000e2
2	7	9	0x000167
2	8	10	0x0003bc
2	9	9	0x0000ca
3	0	7	0x000072
3	1	6	0x000033
3	2	6	0x000018
3	3	7	0x00005c
3	4	7	0x000008
3	5	7	0x00000e
3	6	8	0x00006e
3	7	9	0x0000de
3	8	9	0x0001da
3	9	9	0x00016e
4	0	8	0x0000e1
4	1	7	0x00003e
4	2	7	0x00001c
4	3	7	0x000006
4	4	7	0x000004
4	5	8	0x000064
4	6	9	0x0001db
4	7	8	0x0000b2
4	8	8	0x000067
4	9	8	0x00006b
5	0	9	0x0001c6
5	1	7	0x00001d
5	2	7	0x000007
5	3	8	0x0000ee
5	4	8	0x0000b6
5	5	8	0x0000ec

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

Table A.16 (*continued*)

Idx0	Idx1	DF/FP	
		length	codeword
5	6	7	0x00003f
5	7	8	0x0000bc
5	8	7	0x00000a
5	9	7	0x000009
6	0	10	0x0002ff
6	1	9	0x0001c7
6	2	8	0x00004f
6	3	8	0x00001e
6	4	8	0x0000bd
6	5	7	0x00001e
6	6	8	0x0000d7
6	7	7	0x000036
6	8	7	0x000073
6	9	6	0x000026
7	0	10	0x0001be
7	1	9	0x0000cb
7	2	10	0x0003bd
7	3	8	0x00003e
7	4	7	0x00000b
7	5	7	0x000005
7	6	7	0x000069
7	7	6	0x000027
7	8	6	0x00003a
7	9	5	0x000008
8	0	11	0x00037f
8	1	10	0x0002df
8	2	9	0x00017e
8	3	8	0x0000ba
8	4	8	0x0000be
8	5	7	0x000058
8	6	6	0x000006
8	7	5	0x000000
8	8	5	0x000018
8	9	5	0x00001e
9	0	12	0x000eff
9	1	10	0x0003be
9	2	8	0x00003f
9	3	8	0x00001f
9	4	8	0x0000bb
9	5	7	0x00005a
9	6	6	0x000012
9	7	5	0x00000b
9	8	5	0x00001f
9	9	4	0x00000a

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

Table A.17 – hcod2D_NRG_DT_ZZ_03

Idx0	Idx1	DT/FP	
		length	codeword
0	0	2	0x000
0	1	3	0x006
0	2	3	0x002
0	3	4	0x00e
1	0	3	0x004
1	1	4	0x006
1	2	5	0x00e
1	3	6	0x03e
2	0	6	0x03f
2	1	7	0x03e
2	2	7	0x03c
2	3	5	0x01e
3	0	8	0x07a
3	1	8	0x07b
3	2	7	0x03f
3	3	3	0x005

Table A.18 – hcod2D_NRG_DT_ZZ_06

Idx0	Idx1	DT/FP	
		length	codeword
0	0	4	0x0000e
0	1	5	0x00012
0	2	8	0x000ba
0	3	10	0x0027e
0	4	14	0x03dae
0	5	17	0x1ed7e
0	6	17	0x1ed7f
1	0	3	0x00002
1	1	4	0x00002
1	2	7	0x0005e
1	3	9	0x00176
1	4	11	0x004fe
1	5	16	0x0f6be
1	6	7	0x0000e
2	0	4	0x00008
2	1	4	0x00000
2	2	7	0x0007e
2	3	9	0x00177
2	4	11	0x004ff
2	5	8	0x000b8
2	6	6	0x00006
3	0	5	0x00006
3	1	5	0x00007
3	2	7	0x0004e
3	3	9	0x0013e
3	4	9	0x001ec
3	5	7	0x0007a
3	6	5	0x00016
4	0	6	0x00026
4	1	7	0x0007f
4	2	9	0x001ee
4	3	11	0x007b6
4	4	8	0x000be
4	5	6	0x0003c

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

Table A.18 (*continued*)

Idx0	Idx1	DT/FP	
		length	codeword
4	6	4	0x0000a
5	0	7	0x0000f
5	1	8	0x000bf
5	2	12	0x00f6a
5	3	11	0x007b4
5	4	8	0x000b9
5	5	6	0x0003e
5	6	3	0x00003
6	0	9	0x001ef
6	1	15	0x07b5e
6	2	13	0x01ed6
6	3	11	0x007b7
6	4	8	0x0009e
6	5	5	0x00002
6	6	3	0x00006

Table A.19 – hcod2D_NRG_DT_ZZ_09

Idx0	Idx1	DT/FP	
		length	codeword
0	0	4	0x00004
0	1	5	0x0000a
0	2	8	0x000f6
0	3	10	0x0032e
0	4	11	0x002de
0	5	12	0x0035c
0	6	18	0x0d77c
0	7	18	0x0d77d
0	8	18	0x0d77e
0	9	14	0x032fa
1	0	4	0x0000a
1	1	4	0x00002
1	2	7	0x00066
1	3	9	0x000fe
1	4	11	0x001be
1	5	12	0x0037e
1	6	14	0x00d76
1	7	14	0x032fe
1	8	18	0x0d77f
1	9	8	0x0007a
2	0	5	0x0001a
2	1	5	0x00016
2	2	7	0x00064
2	3	9	0x001ea
2	4	10	0x0016e
2	5	11	0x0035e
2	6	12	0x0035e
2	7	14	0x032fb
2	8	8	0x0005e
2	9	7	0x0003c
3	0	6	0x0003c
3	1	5	0x00002
3	2	7	0x0005e
3	3	8	0x0005a
3	4	9	0x0006e
3	5	11	0x0065e
3	6	13	0x0197e
3	7	9	0x001ee

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

Table A.19 (*continued*)

Idx0	Idx1	DT/FP	
		length	codeword
3	8	7	0x0002c
3	9	6	0x00018
4	0	6	0x00030
4	1	6	0x00036
4	2	7	0x00034
4	3	8	0x000f4
4	4	9	0x000f6
4	5	11	0x002df
4	6	10	0x003d6
4	7	8	0x0005f
4	8	7	0x0003e
4	9	6	0x0002e
5	0	6	0x0000c
5	1	6	0x0000e
5	2	7	0x00032
5	3	8	0x00036
5	4	10	0x000de
5	5	10	0x000d6
5	6	9	0x001ef
5	7	7	0x00033
5	8	7	0x0005f
5	9	6	0x00037
6	0	7	0x00036
6	1	6	0x00006
6	2	8	0x000ca
6	3	9	0x000ff
6	4	13	0x0197c
6	5	11	0x003de
6	6	8	0x00034
6	7	7	0x0000e
6	8	6	0x0000f
6	9	5	0x0000e
7	0	7	0x00037
7	1	7	0x0002e
7	2	9	0x00196
7	3	13	0x006ba
7	4	12	0x0037f
7	5	10	0x001ee
7	6	9	0x000d6
7	7	7	0x0000f
7	8	6	0x00031
7	9	4	0x00000
8	0	8	0x0006a
8	1	8	0x0007e
8	2	16	0x035de
8	3	13	0x00f7e
8	4	12	0x007be
8	5	10	0x001ae
8	6	9	0x0006a
8	7	8	0x000ce
8	8	6	0x0003e
8	9	4	0x0000e
9	0	9	0x000b6
9	1	15	0x01aee
9	2	13	0x00f7f
9	3	12	0x0035f
9	4	14	0x032ff
9	5	11	0x0035f
9	6	10	0x003d7
9	7	8	0x000cf
9	8	6	0x0003f
9	9	3	0x00004

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

Table A.20 – hcod2D_NRG_DT_ZZ_12

Idx0	Idx1	DT/FP	
		length	codeword
0	0	5	0x00001a
0	1	5	0x000002
0	2	8	0x0000b4
0	3	10	0x0002d6
0	4	10	0x0000fe
0	5	13	0x001dfe
0	6	13	0x0016be
0	7	22	0x3bfef2
0	8	22	0x3bfef3
0	9	22	0x3bfef4
0	10	22	0x3bfef5
0	11	22	0x3bfef6
0	12	22	0x3bfef7
1	0	4	0x000004
1	1	5	0x000014
1	2	6	0x000000
1	3	9	0x00011e
1	4	10	0x000376
1	5	11	0x00055c
1	6	16	0x00effe
1	7	22	0x3bfef8
1	8	22	0x3bfef9
1	9	22	0x3bfefa
1	10	22	0x3bfefb
1	11	22	0x3bfefc
1	12	8	0x00009a
2	0	5	0x000012
2	1	5	0x000010
2	2	7	0x000074
2	3	8	0x00000e
2	4	11	0x0001fe
2	5	12	0x000dfc
2	6	15	0x0013fe
2	7	13	0x001bfe
2	8	22	0x3bfefd
2	9	22	0x3bfefe
2	10	22	0x3bfeff
2	11	8	0x000006
2	12	7	0x000010
3	0	6	0x00003e
3	1	6	0x00003f
3	2	6	0x000002
3	3	8	0x0000e6
3	4	9	0x0001de
3	5	12	0x000aba
3	6	15	0x005afe
3	7	16	0x00effa
3	8	13	0x001bfa
3	9	21	0x1dff70
3	10	9	0x00000e
3	11	8	0x0000b6
3	12	7	0x000054
4	0	6	0x00001a
4	1	6	0x00002e
4	2	7	0x00004e
4	3	8	0x00009e

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

Table A.20 (*continued*)

Idx0	Idx1	DT/FP	
		length	codeword
4	4	9	0x00001e
4	5	11	0x0005ae
4	6	13	0x001bfb
4	7	15	0x0013ff
4	8	12	0x00027e
4	9	10	0x00023a
4	10	8	0x0000aa
4	11	7	0x000036
4	12	7	0x000072
5	0	7	0x000056
5	1	6	0x00001c
5	2	7	0x000046
5	3	8	0x00003e
5	4	9	0x0000de
5	5	11	0x00055e
5	6	11	0x0001ff
5	7	13	0x001bff
5	8	10	0x000238
5	9	9	0x000136
5	10	8	0x0000b7
5	11	7	0x000032
5	12	6	0x00000a
6	0	7	0x000075
6	1	7	0x000076
6	2	7	0x000016
6	3	8	0x00001e
6	4	9	0x000137
6	5	10	0x0003be
6	6	10	0x00003e
6	7	12	0x000efe
6	8	9	0x00000f
6	9	8	0x000026
6	10	8	0x00000c
6	11	7	0x00003a
6	12	6	0x000022
7	0	7	0x000002
7	1	7	0x00005e
7	2	8	0x0000ae
7	3	8	0x0000dc
7	4	9	0x0000df
7	5	11	0x00055f
7	6	12	0x000dfe
7	7	10	0x00037e
7	8	9	0x00007e
7	9	10	0x000377
7	10	8	0x0000e7
7	11	7	0x00004c
7	12	6	0x00001e
8	0	7	0x00000e
8	1	7	0x00001c
8	2	7	0x00001d
8	3	8	0x00006e
8	4	10	0x00009e
8	5	16	0x00effc
8	6	11	0x00007e
8	7	11	0x0000fe

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

Table A.20 (*continued*)

Idx0	Idx1	DT/FP	
		length	codeword
8	8	10	0x000239
8	9	9	0x000156
8	10	7	0x000017
8	11	6	0x000006
8	12	6	0x000036
9	0	8	0x0000af
9	1	7	0x00005f
9	2	8	0x000066
9	3	9	0x00004e
9	4	21	0x1dff71
9	5	14	0x0009fe
9	6	12	0x000b5e
9	7	11	0x0000ff
9	8	11	0x00007f
9	9	9	0x00016a
9	10	7	0x00001e
9	11	6	0x000018
9	12	5	0x000006
10	0	8	0x0000ee
10	1	7	0x000011
10	2	9	0x00011f
10	3	21	0x1dff72
10	4	21	0x1dff73
10	5	15	0x005aff
10	6	17	0x01dff6
10	7	11	0x000476
10	8	9	0x00001a
10	9	9	0x0001ba
10	10	7	0x000012
10	11	6	0x00002c
10	12	5	0x00001e
11	0	8	0x000067
11	1	8	0x00009f
11	2	21	0x1dff74
11	3	21	0x1dff75
11	4	16	0x00effd
11	5	21	0x1dff76
11	6	12	0x000abb
11	7	11	0x000477
11	8	10	0x00007e
11	9	9	0x00003e
11	10	7	0x00003b
11	11	6	0x00001f
11	12	4	0x000005
12	0	9	0x0001be
12	1	21	0x1dff77
12	2	14	0x002d7e
12	3	15	0x0077fc
12	4	21	0x1dff78
12	5	16	0x00efff
12	6	13	0x0004fe
12	7	11	0x00013e
12	8	11	0x00077e
12	9	9	0x00001b
12	10	8	0x0000de
12	11	6	0x000038
12	12	4	0x00000c

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

A.2 Misc. tables

Table A.21 – Coefficients c[i] of the QMF LD 256 window

i	c[i]	i	c[i]
0	-7.949261005955764e-004	320	-1.210755701624524e-001
1	-1.232074328145439e-003	321	-1.185237142283346e-001
2	-1.601053942982895e-003	322	-1.159184450952715e-001
3	-1.980720409470913e-003	323	-1.132654367461266e-001
4	-2.397504953865715e-003	324	-1.105698782276963e-001
5	-2.838709203607079e-003	325	-1.078369135648348e-001
6	-3.314755401090670e-003	326	-1.050716118804287e-001
7	-3.825180949035082e-003	327	-1.022789198651472e-001
8	-4.365307413613105e-003	328	-9.946367410320074e-002
9	-4.937260935539922e-003	329	-9.663069107327295e-002
10	-5.537381514710146e-003	330	-9.378454802679648e-002
11	-6.164241937824271e-003	331	-9.092970207094843e-002
12	-6.816579194002503e-003	332	-8.807051083640835e-002
13	-7.490102145765528e-003	333	-8.521107266503664e-002
14	-8.183711450708110e-003	334	-8.235562752947133e-002
15	-8.894930051379498e-003	335	-7.950789957683559e-002
16	-9.620004581607449e-003	336	-7.667177989755110e-002
17	-1.035696814015217e-002	337	-7.385092587441364e-002
18	-1.110238617202191e-002	338	-7.104866702770536e-002
19	-1.185358556146692e-002	339	-6.826847016140082e-002
20	-1.260769256679562e-002	340	-6.551341011471171e-002
21	-1.336080675156018e-002	341	-6.278658929544248e-002
22	-1.411033176541011e-002	342	-6.009091369370080e-002
23	-1.485316243134798e-002	343	-5.742919825387360e-002
24	-1.558550942227883e-002	344	-5.480383115198150e-002
25	-1.630436835497356e-002	345	-5.221738078737957e-002
26	-1.700613959422392e-002	346	-4.967213638808988e-002
27	-1.768770555992799e-002	347	-4.717023345307148e-002
28	-1.834568069395711e-002	348	-4.471364025371278e-002
29	-1.897612496482356e-002	349	-4.230438144160113e-002
30	-1.957605813345359e-002	350	-3.994384828552555e-002
31	-2.014213322475170e-002	351	-3.763371362431132e-002
32	-2.067061748933033e-002	352	-3.537544041600725e-002
33	-2.115814831921453e-002	353	-3.317035188016126e-002
34	-2.160130854695980e-002	354	-3.101971215825843e-002
35	-2.199696217022438e-002	355	-2.892453070357571e-002
36	-2.234169110698344e-002	356	-2.688575425197388e-002
37	-2.263170795250229e-002	357	-2.490421725219031e-002
38	-2.286416556008894e-002	358	-2.298058501129975e-002
39	-2.303589449043864e-002	359	-2.111545692324888e-002
40	-2.314344724218223e-002	360	-1.930927680100128e-002
41	-2.318352524475873e-002	361	-1.756239270089077e-002
42	-2.315297727620401e-002	362	-1.587511449869362e-002
43	-2.304918234544422e-002	363	-1.424750749465213e-002
44	-2.286864521420490e-002	364	-1.267955527855867e-002
45	-2.260790764376614e-002	365	-1.117125833414906e-002
46	-2.226444264459477e-002	366	-9.722405440999532e-003
47	-2.183518667784246e-002	367	-8.332704660914712e-003
48	-2.131692017682024e-002	368	-7.001789872901951e-003
49	-2.070614962636994e-002	369	-5.729226040772489e-003
50	-1.999981321635736e-002	370	-4.514503359783591e-003

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

Table A.21 (*continued*)

i	c[i]	i	c[i]
51	-1.919566223498554e-002	371	-3.356946762357950e-003
52	-1.828936158524688e-002	372	-2.255849987026407e-003
53	-1.727711874492186e-002	373	-1.210459261524451e-003
54	-1.615648494779686e-002	374	-2.199474640570699e-004
55	-1.492335807272955e-002	375	7.167268627887994e-004
56	-1.357419760297910e-002	376	1.600440185590357e-003
57	-1.210370330110896e-002	377	2.432366605744087e-003
58	-1.050755164953818e-002	378	3.213605482343768e-003
59	-8.785746151726750e-003	379	3.945301462616821e-003
60	-6.927329556345040e-003	380	4.628665378925932e-003
61	-4.929378450735877e-003	381	5.264976586624488e-003
62	-2.800333941149626e-003	382	5.855653555178131e-003
63	-4.685580749545335e-004	383	6.401634331453516e-003
64	2.210315255690887e-003	384	-6.903046246257517e-003
65	5.183294908090526e-003	385	-7.364537203059431e-003
66	8.350964449424035e-003	386	-7.785917436812734e-003
67	1.166118535611788e-002	387	-8.168780818165564e-003
68	1.513166797475777e-002	388	-8.514510536234886e-003
69	1.877264877027943e-002	389	-8.824526581578384e-003
70	2.258899222368603e-002	390	-9.100444687042341e-003
71	2.659061474958830e-002	391	-9.343819821939981e-003
72	3.078087745385930e-002	392	-9.556089247587111e-003
73	3.516391224752870e-002	393	-9.738929904236388e-003
74	3.974674893613862e-002	394	-9.893728065983530e-003
75	4.453308211110493e-002	395	-1.002221842309897e-002
76	4.952626097917320e-002	396	-1.012567516563336e-002
77	5.473026727738295e-002	397	-1.020575952382967e-002
78	6.014835645056577e-002	398	-1.026389875785943e-002
79	6.578414516120631e-002	399	-1.030162959448537e-002
80	7.163950999489413e-002	400	-1.032037849566083e-002
81	7.771656494569829e-002	401	-1.032154667898522e-002
82	8.401794441130064e-002	402	-1.030658039367325e-002
83	9.054515924487507e-002	403	-1.027682791880806e-002
84	9.729889691289549e-002	404	-1.023360327572998e-002
85	1.042804039148369e-001	405	-1.017821017226088e-002
86	1.114900795290448e-001	406	-1.011195224927225e-002
87	1.189284254931251e-001	407	-1.003602653649432e-002
88	1.265947532678997e-001	408	-9.951564927254814e-003
89	1.344885599112251e-001	409	-9.859735321541087e-003
90	1.426090972422485e-001	410	-9.761689935477358e-003
91	1.509550307914161e-001	411	-9.658335268268776e-003
92	1.595243494708706e-001	412	-9.550506541750015e-003
93	1.683151598707939e-001	413	-9.439239790180602e-003
94	1.773250461581686e-001	414	-9.325311662898867e-003
95	1.865511418631904e-001	415	-9.209571052890813e-003
96	1.959902227114119e-001	416	-9.092729858436259e-003
97	2.056386275763479e-001	417	-8.975504153186832e-003
98	2.154925974105375e-001	418	-8.858564024669505e-003
99	2.255475564993390e-001	419	-8.742547510216072e-003
100	2.357989864681126e-001	420	-8.627917215653412e-003
101	2.462418809459464e-001	421	-8.515236113018675e-003
102	2.568709554604541e-001	422	-8.404834686887089e-003

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

© ISO/IEC 2010 – All rights reserved

Table A.21 (*continued*)

i	c[i]	i	c[i]
103	2.676805358910440e-001	423	-8.297046056582970e-003
104	2.786645734207760e-001	424	-8.192181771808344e-003
105	2.898168394038287e-001	425	-8.090558375952284e-003
106	3.011307516871287e-001	426	-7.992340268718087e-003
107	3.125994749246541e-001	427	-7.897787592331651e-003
108	3.242157192666507e-001	428	-7.806979111626161e-003
109	3.359722796803192e-001	429	-7.720005213599928e-003
110	3.478614117031655e-001	430	-7.636899169053526e-003
111	3.598752336287570e-001	431	-7.557692588413262e-003
112	3.720056632072922e-001	432	-7.482361735247336e-003
113	3.842444358173011e-001	433	-7.410882580163479e-003
114	3.965831241942321e-001	434	-7.343084196594709e-003
115	4.090129566893579e-001	435	-7.278918614409016e-003
116	4.215250930838456e-001	436	-7.218206312830178e-003
117	4.341108982328533e-001	437	-7.160843298305507e-003
118	4.467608231633283e-001	438	-7.106600272887440e-003
119	4.594659376709624e-001	439	-7.055249359796239e-003
120	4.722166595058233e-001	440	-7.006591539682229e-003
121	4.850038204075748e-001	441	-6.960450953203489e-003
122	4.978178235802594e-001	442	-6.916554770130135e-003
123	5.106483456192374e-001	443	-6.874623603448978e-003
124	5.234865375971977e-001	444	-6.834443173086539e-003
125	5.363218470709771e-001	445	-6.795786363014294e-003
126	5.491440356706657e-001	446	-6.758476537306303e-003
127	5.619439923555571e-001	447	-6.722125942626111e-003
128	-5.746001351404267e-001	448	-6.686140904391229e-003
129	-5.872559277139351e-001	449	-6.650228698006217e-003
130	-5.998618924353250e-001	450	-6.614354298921371e-003
131	-6.123980151490041e-001	451	-6.578320578669048e-003
132	-6.248504862282382e-001	452	-6.541865503698597e-003
133	-6.372102969387355e-001	453	-6.504729306516950e-003
134	-6.494654463921502e-001	454	-6.466690242148724e-003
135	-6.616044277534099e-001	455	-6.427556828582072e-003
136	-6.736174463977084e-001	456	-6.387124476277924e-003
137	-6.854929931488056e-001	457	-6.345262303711465e-003
138	-6.972201618598393e-001	458	-6.301766582696827e-003
139	-7.087881675504216e-001	459	-6.256542736138121e-003
140	-7.201859881692665e-001	460	-6.209372064970386e-003
141	-7.314035334082558e-001	461	-6.160215935384255e-003
142	-7.424295078874311e-001	462	-6.108902434484468e-003
143	-7.532534422335129e-001	463	-6.055355267266873e-003
144	-7.638649113306198e-001	464	-5.999473903317320e-003
145	-7.742538112450130e-001	465	-5.941211676077848e-003
146	-7.844095212375462e-001	466	-5.880495927392625e-003
147	-7.943222347831999e-001	467	-5.817286139372493e-003
148	-8.039818519286321e-001	468	-5.751536864441650e-003
149	-8.133789939828571e-001	469	-5.683230954033062e-003
150	-8.225037151897938e-001	470	-5.612375999953358e-003
151	-8.313468549324594e-001	471	-5.538957988293047e-003
152	-8.398991600556686e-001	472	-5.462963107291498e-003
153	-8.481519810689574e-001	473	-5.384396217909888e-003
154	-8.560963550316389e-001	474	-5.303337109336215e-003

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010

Table A.21 (*continued*)

i	c[i]	i	c[i]
155	-8.637239863984174e-001	475	-5.219739772898678e-003
156	-8.710266607496513e-001	476	-5.133623037830525e-003
157	-8.779965198108476e-001	477	-5.045046346880483e-003
158	-8.846258145496611e-001	478	-4.954008597884707e-003
159	-8.909071890560218e-001	479	-4.860588885693231e-003
160	-8.968337036455653e-001	480	-4.764720830452409e-003
161	-9.023985431182168e-001	481	-4.666469548192818e-003
162	-9.075955881221292e-001	482	-4.565946029127366e-003
163	-9.124187296760565e-001	483	-4.463150894014690e-003
164	-9.168621399784253e-001	484	-4.358150755039186e-003
165	-9.209204531389191e-001	485	-4.250967471708103e-003
166	-9.245886139655739e-001	486	-4.141634861746089e-003
167	-9.278619263447355e-001	487	-4.030165355928349e-003
168	-9.307362242659798e-001	488	-3.916597675997815e-003
169	-9.332075222986479e-001	489	-3.800994685405442e-003
170	-9.352724511271509e-001	490	-3.683451012833619e-003
171	-9.369278287932853e-001	491	-3.563914929838276e-003
172	-9.381709878904797e-001	492	-3.442490007998456e-003
173	-9.389996917291260e-001	493	-3.319256438897666e-003
174	-9.394121230559878e-001	494	-3.194250476422174e-003
175	-9.394068064126931e-001	495	-3.067525877056119e-003
176	-9.389829174860432e-001	496	-2.939139106182801e-003
177	-9.381397976778112e-001	497	-2.809151898728351e-003
178	-9.368773370086998e-001	498	-2.677703006241942e-003
179	-9.351961242404785e-001	499	-2.544830774162231e-003
180	-9.330966718935136e-001	500	-2.410617950987095e-003
181	-9.305803205049067e-001	501	-2.275190768887402e-003
182	-9.276488080866625e-001	502	-2.138586519570023e-003
183	-9.243040558859498e-001	503	-2.000881763033976e-003
184	-9.205488097488350e-001	504	-1.862161137529843e-003
185	-9.163856478189402e-001	505	-1.722850651410707e-003
186	-9.118180055332041e-001	506	-1.583005323492318e-003
187	-9.068503557855540e-001	507	-1.442635273572746e-003
188	-9.014858673099563e-001	508	-1.301735673138880e-003
189	-8.957295448806664e-001	509	-1.160531184883257e-003
190	-8.895882558527375e-001	510	-1.018710154718430e-003
191	-8.830582442418677e-001	511	-8.753658738743612e-004
192	-8.761259906419252e-001	512	7.250868879948704e-004
193	-8.688044201931157e-001	513	5.901514303345345e-004
194	-8.611140376567749e-001	514	4.571251178344833e-004
195	-8.530684188588082e-001	515	3.254504484897777e-004
196	-8.446723286380624e-001	516	1.951832637892118e-004
197	-8.359322523144003e-001	517	6.661818101906931e-005
198	-8.268555005748937e-001	518	-6.002729636107936e-005
199	-8.174491260941859e-001	519	-1.845163192347697e-004
200	-8.077214932837783e-001	520	-3.065712811761140e-004
201	-7.976809997929416e-001	521	-4.259661821125124e-004
202	-7.873360271773119e-001	522	-5.424773586381941e-004
203	-7.766956604639097e-001	523	-6.558084462274315e-004
204	-7.657692341138960e-001	524	-7.659101269870789e-004
205	-7.545663748526984e-001	525	-8.724859431432570e-004
206	-7.430967641354331e-001	526	-9.753531169034512e-004

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23003-2:2010