INTERNATIONAL STANDARD

ISO/IEC 14496-4

Second edition
2004-12-15

AMENDMENT 8
2005-06-01

Information technology — Coding of audio-visual objects —

Part 4: Conformance testing

AMENDMENT 8: High Efficiency Advanced Audio Coding, audio BIFS, and structured audio conformance

Technologies de l'information — Codage des objets audiovisuels —

Partie 4: Essai de conformité

AMENDEMENT 8: Codage sonore avancé à haute efficacité, BIFS sonore et conformité sonore structurée

Reference number
ISO/IEC 14496-4:2004/Amd.8:2005(E)

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

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.

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

Amendment 8 to ISO/IEC 14496-4:2004 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*.

Introduction

This document specifies the eighth amendment to the ISO/IEC 14496-4:2004 standard. The amendment adds the conformance testing for the SBR audio object type defined in 14496-3. It also specifies conformance sequences for BIFS and Structured Audio.

Information technology — Coding of audio-visual objects —

Part 4:
Conformance testing

AMENDMENT 8: High Efficiency Advanced Audio Coding, audio BIFS, and structured audio conformance

In subclause 6.5.1 File name conventions, insert the following row into Table 29 after the row for AAC LTP:

Table 29 – File name conventions

SBR (+AAC LC)	al_sbr_<tool>_<fs>_<nchan>[_fsaac<fs>][_sig<sig>]	al_sbr_<mode>_<tool>_<fs>_<nchan>[_fsaac<fs>][_sig<sig>][_<chan>]

And add:

<tool> indicates the SBR module mainly targeted by the test sequence. Possible values are "e" for testing the envelope adjuster "s" for testing sine addition, "gh" for testing time-grid transitions in combination with changes of SBR header data, "i" for testing inverse filtering, "qmf" for testing the QMF implementation, “cm” for testing various channel modes, “sig” for testing SBR signaling, "twi" for QMF identification, and “sr” for testing various combinations of sampling rates.

<nchan> corresponds to the number of channels present in the conformance test sequence. It is either a single integer, in which case it refers to the number of main audio channels, or two integers separated by a ‘.’, in which case the first integer equals the number of main audio channels, while the second number equals the number of low frequency enhancement channels.

fsaac<fs> corresponds to the sampling rate of the underlying AAC-LC data. If it is omitted, it is half the sampling rate given as output sampling rate.

<sig> is an integer describing the kind of signalling used according to the table below. If this value is omitted, backwards compatible explicit signalling of SBR is used.

Table 29A – File name conventions

sig	Signalling method used
0	Implicit signalling of SBR
1	Hierarchical explicit signalling of SBR
2	Backwards compatible explicit signalling of SBR

<mode> is either "hq" or "lp" for the high quality or the low power version of the SBR decoding algorithm respectively.

In subclause 6.6.1.2.2 (Test procedure), replace:

...testing can be done by comparing the output of a decoder under test with a reference output also supplied by ISO/IEC JTC 1/SC 29/WG 11.

with:

...testing can be done by comparing the output of a decoder under test with a reference output also supplied by ISO/IEC JTC 1/SC 29/WG 11. In cases where the decoder under test is followed by additional operations (e.g. quantizing a signal to a 16 bit output signal) the conformance point is prior to such additional operations, i.e. it is permitted to use the actual decoder output (e.g. with more than 16 bit) for conformance testing.

In subclause 6.6.3.1.2.3 (Bitstream payload), add:

6.6.3.1.2.3.2 aac_scalable_main_header()

max_sfb: Shall not be smaller than last_max_sfb (helper variable specified in ISO/IEC 14496-3).

6.6.3.1.2.3.3 aac_scalable_extension_header()

max_sfb: Shall not be smaller than last_max_sfb (helper variable specified in ISO/IEC 14496-3).

In subclause 6.6.4.1.2.1.1 AudioSpecificConfig(), add:

extensionAudioObjectType: Shall be the Audio Object Type SBR (AOT == 5).

extensionSamplingFrequency: Shall be encoded with a value listed in Table 34, and the value shall be the same as samplingFrequency, or twice the value of samplingFrequency.

extensionSamplingFrequencyIndex: Shall be encoded with a value listed in Table 34, and the value shall indicate an extensionSamplingFrequency being the same as samplingFrequency as indicated by samplingFrequencyIndex, or the value shall indicate an extensionSamplingFrequency being twice the value of samplingFrequency.

sbrPresentFlag: Shall be encoded with the value zero if no SBR data is contained in the compressed MPEG-4 data. If SBR data is present in the compressed MPEG-4 data the parameter shall be encoded with the value one.

In subclause 6.6.4.1.2.1.1 AudioSpecificConfig(), add the following entries to Table 34:

Table 34 — Specification of samplingFrequencyIndex and samplingFrequency

SamplingFrequencyIndex / SamplingFrequency	**Level 1**	**Level 2**	**Level 3**	**Level 4**	**Level 5**
AAC Profile	0x6..0xc, 0xf / <= 24000	0x3..0xc, 0xf / <= 48000	NA	0x3..0xc, 0xf / <= 48000	0x0..0xc, 0xf / <= 96000

samplingFrequencyIndex / samplingFrequency		**Level 1**	**Level 2**	**Level 3**	**Level 4**	**Level 5**
High Efficiency AAC Profile	**SBR present**	NA	0x6..0xc, 0xf / <= 24000	0x3..0xc, 0xf / <= 48000	0x3..0xc, 0xf / <= 48000 (Note 1)	0x3..0xc, 0xf / <= 48000
	SBR not present	NA	0x3..0xc, 0xf / <= 48000	0x3..0xc, 0xf / <= 48000	0x3..0xc, 0xf / <= 48000	0x0..0xc, 0xf / <= 96000
Note 1: For Level 4, for one or two channels the maximum AAC sampling rate, with SBR present, is 48 kHz. For more than two channels the maximum AAC sampling rate, with SBR present, is 24 kHz. (0x6..0xc, 0xf / <= 24000)						

extensionSamplingFrequencyIndex / extensionSamplingFrequency	**Level 1**	**Level 2**	**Level 3,4**	**Level 5**
High Efficiency AAC Profile	NA	0x6..0xc, 0xf / <= 24000	0x3..0xc, 0xf / <= 48000	0x0..0xc, 0xf / <= 96000

In subclause 6.6.4.1.2.1.1 AudioSpecificConfig(), add the following entries to Table 35:

Table 35 — Specification of ChannelConfiguration

ChannelConfiguration	**Level 1**	**Level 2**	**Level 3**	**Level 4**	**Level 5**
AAC Profile	0..2	0..2	NA	0..6	0..6
High Efficiency AAC Profile	NA	0..2	0..2	0..6	0..6

In subclause 6.6.4.1.2.2.4 (ics_info()), replace:

Test bitstreams al03 and as17 are provided respectively for Main and Low-Complexity profiles to test decoder performance on non-meaningful transitions

with:

Test sequences al03 and as17 are provided respectively for AOT 2 (AAC LC) and AOT 3 (AAC SSR) to test decoder performance on non-meaningful window sequence transitions (note that AOT 1 (AAC Main) and AOT 4 (AAC LTP) decoders also need to fulfil conformance for AOT 2)

In subclause 6.6.4.1.2.2.9 fill_element(), add:

Fill elements containing an extension_payload with an extension_type of EXT_SBR_DATA or EXT_SBR_DATA_CRC shall not contain any other extension_payload of any other extension_type. For fill elements containing an extension_payload with an extension_type of EXT_SBR_DATA or EXT_SBR_DATA_CRC, the fill_element count field shall be set equal to the total length in bytes, including the SBR enhancement data plus the extension_type field.

In subclause 6.6.15.5 Procedure to Test Decoder Conformance:

At the end of the second paragraph, replace the last sentence:

Conformant decoders must use the RMS Measurement criterion for sequences SY001 through SY004

with:

Conformant decoders must use the RMS Measurement criterion for sequences SY001 through SY004 and SY016 through SY019

In the same subclause, before the last paragraph (Testing of non-normative effects...) add:

Testing of the bus width calculation and send/route mechanism shall be performed using test sequence SY014 and SY015. A reference output is provided by ISO/IEC as example. To be called an ISO/IEC 14496-3 audio decoder, the decoder shall provide two identical outputs for the two sequences, and in particular this output shall be composed of two channels characterized by a reverberated sound in the first and a dry sound in the second

In subclause 6.6.15.6 Description of Conformance sequences, add the following sequence descriptions after SY013:

Sequence SY014 "clarinet1.mp4"

A clarinet synthesized at 44.1 kHz in FM with linear interpolation is routed to a reverberation instrument. The bus width is not set in the send statement, instead the sound is output twice to the bus, creating a two-channel bus. In the reverberation instrument only the first channel is reverberated, the second is output as is.

Sequence SY015 "clarinet2.mp4"

A clarinet synthesized at 44.1 kHz in FM with linear interpolation is routed to a reverberation instrument. The bus width is set to 2 in the send statement, the sound is output in mono to the bus, and then it must be replicated identical on the second channel. In the reverberation instrument only the first channel is reverberated, the second is output as is.

Sequence SY016 "fir.mp4"

In this sequence a sound synthesized at 32kHz is low pass filtered using both fir and firt core opcodes. The two filters have 16 coefficients and normalized cutoff frequency of 0.5 and 0.25 respectively. They are used to filter the left (fir) and right (firt) channels of the sound. This sequence also test pitch converters and other mathematical operators.

Sequence SY017 "vtone.mp4"

A sequence of monophonic sinusoidal tones with sampling rate frequency at 44.1 kHz is shaped according to attack and release time. This test sequence experiences mathematical operators, pitch converters and kline signal generator. Interpolation is linear.

Sequence SY018 "ttone.mp4"

A sequence of monophonic sinusoidal tones with sampling rate frequency at 44.1 kHz is shaped according to attack and release time. It is similar to SY017 but in this case tones are generated through tables instead that by mathematical functions. This test sequence experiences table generators, mathematical operators, table access and oscillators. Interpolation is linear.

Sequence SY019 "otone.mp4"

A sine tone is played in the left channel, with its octave tone in the right channel. Interpolation is linear, sampling rate is 44.1 kHz. This sequence especially experiences user defined core opcodes (including rate polymorphic), parameter passing, array variables.

In the same subclause, after Table 92, add:

Table AMD8-1 — Algorithmic Synthesis and Audio Fx Object Type Test Sequences (continued)

File Name	SY014	SY015	SY016	SY017	SY018	SY019
Content	clarinet1	clarinet2	fir	vtone	ttone	otone
Processing level	All	All	≥Med	All	All	All
RCU - RAM (KB)	< 4	< 4	< 4	< 4	< 4	< 4

After subclause 6.6.16 Main Synthetic, add the following subclauses:

6.6.17 SBR

6.6.17.1 Compressed data

6.6.17.1.1 Characteristics

For all applicable Audio Object Types the SBR extension_payload() elements should be placed last among the extension_payload() elements, i.e. if another type of extension_payload() element is present it should be placed prior to the SBR extension_payload() elements.

If the Audio Object Type SBR is used in combination with either of the Audio Object Types AAC main, AAC LC, AC SSR or AAC LTP, the compressed data shall be stored as outlined in ISO/IEC 14496-3, subclause 4.5.2.8.2.2 SBR Extension Payload for the Audio Object Types AAC main, AAC SSR, AAC LC and AAC LTP.

If the Audio Object Type SBR is used in combination with either of the Audio Object Types ER AAC LC or ER AAC LTP, the compressed data shall be stored as outlined in ISO/IEC 14496-3, subclause 4.5.2.8.2.3 SBR Extension Payload for the Audio Object Types ER AAC LC and ER AAC LTP. For these AOTs, DRC extension_payload() elements are not permitted simultaneously with SBR extension_payload() elements within one er_raw_data_block(). Moreover, SBR extension_payload() elements of the type EXT_SBR_DATA_CRC shall not be used with these AOTs.

For the scalable AOTs (AAC scalable and ER AAC scalable), the SBR data should be transmitted and devised according to ISO/IEC 14496-3, subclause 4.5.2.8.2.4 SBR extension payload for the Audio Object Types AAC scalable and ER AAC scalable. Restrictions are here put on the frequency range of the SBR data and in what layers of the scalable stream the SBR data is stored. Furthermore, SBR extension_payload() elements of the type EXT_SBR_DATA_CRC shall not be used with the audio object types ER AAC scalable.

6.6.17.1.2 Test procedure

Each compressed data shall meet the syntactic and semantic requirements specified in ISO/IEC 14496-3. The decoded data shall also meet the requirements defined in ISO/IEC 14496-3 subclause 4.6.18.3.6 Requirements. If a syntactic element is not listed below, no restrictions apply to that element. The **bs_reserved** elements shall be encoded with the value zero.

6.6.17.1.2.1 Compressed MPEG-4 data payload

6.6.17.1.2.1.1 sbr_header()

The following parameters shall be encoded with values subsequently used in defining a frequency range, a number of noise bands, a number of limiter bands, and a number of patches:

bs_start_freq
bs_stop_freq
bs_xover_band
bs_alter_scale
bs_noise_bands
bs_limiter_bands

The above parameters are used (in ISO/IEC 14496-3) to calculate the variables below:

k_2
k_0
k_x
M
N_Q
numPatches
numBands
numBands0
vDk0
vDk1

Conformant compressed MPEG-4 data shall have values for the above parameters that subsequently evaluate to values of the above variables that satisfy the requirements outlined in ISO/IEC 14496-3, subclause 4.6.18.3.6 Requirements.

6.6.17.1.2.1.2 sbr_channel_pair_base_element()

bs_coupling: Shall be encoded with the value of 1

6.6.17.1.2.1.3 sbr_grid()

The following compressed MPEG-4 data elements shall be encoded so that a value of the number of SBR envelopes for a SBR frame, for a given frame class, is within the limits defined in ISO/IEC 14496-3, subclause 4.6.18.3.6 Requirements:

bs_rel_bord_0
bs_rel_bord_1
bs_num_env
bs_var_bord
bs_num_rel_0
bs_var_bord_0
bs_var_bord_1
bs_num_rel_0
bs_num_rel_1

Conformant compressed MPEG-4 data shall have the above parameters chosen so that the leading border of a given SBR frame (the frame boundary) coincides with the trailing border of the previous SBR frame (the frame boundary of the previous frame). Furthermore, the above parameters shall be chosen so that the envelope borders of the SBR envelopes in a given frame fall within the boundaries of the SBR frame. The above parameters shall also be chosen so that every SBR envelope within the SBR frame has a duration larger than zero.

6.6.17.1.2.1.4 sbr_dtdf()

bs_df_env[]: Shall be encoded with the value 0 for the first envelope of the present frame, if the compressed MPEG-4 data element bs_header_flag has the value one (i.e. a new sbr_header is available), or if the amp_res value has changed from the previous frame due to the rule specifying amp_res = 0 for a frame of frame class FIXFIX with only one envelope.

bs_df_noise[]: Shall be encoded with the value 0 for the first noise floor of the present frame, if the compressed MPEG-4 data element bs_header_flag has the value one, i.e. a new sbr_header is available.

6.6.17.1.2.1.5 sbr_envelope()

bs_codeword: Shall be encoded with the values listed in the corresponding Huffman table, defined in ISO/IEC 14496-3, Annex 4.A.6.1

Conformant compressed MPEG-4 data shall have coded envelope scalefactors based on quantized envelopes scalefactors that satisfy the requirements outlined in ISO/IEC 14496-3, subclause 4.6.18.3.6 Requirements.

The quantised envelope scale factors $\mathbf{E}$ for single channel elements and $\mathbf{E}_0$ and $\mathbf{E}_1$ for channel pair elements shall be encoded with values that are within the following limits:

- For single channel elements: $0 \le \mathbf{E}(i,l) < 2^{7-amp\_res}$
- For channel pair elements: $\begin{cases} 0 \le \mathbf{E}_0(i,l) < 2^{7-amp\_res} \\ 0 \le \mathbf{E}_1(i,l) < 2^{7-amp\_res-bs\_coupling} \end{cases}$

where

$$amp\_res = \begin{cases} 0 & ,\text{if } bs\_num\_env = 1 \text{ and } frame\_class = FIXFIX \\ bs\_amp\_res & ,otherwise \end{cases}$$

where subscript zero indicates the firstly encoded channel in the channel pair element and subscript one indicates the secondly encoded channel in the channel pair element.

6.6.17.1.2.1.6 sbr_noise()

bs_codeword: Shall be encoded with the values listed in the corresponding Huffman table, defined in ISO/IEC 14496-3, Annex 4.A.6.1

Conformant compressed MPEG-4 data shall have coded noise floor scalefactors based on quantised noise floor scalefactors that satisfy the requirements outlined in ISO/IEC 14496-3, subclause 4.6.18.3.6 Requirements.

6.6.17.2 Decoders

6.6.17.2.1 Characteristics

The object type SBR has the Object Type ID 5, and the compressed MPEG-4 data syntax is defined in ISO/IEC 14496-3. The Audio Object Type SBR contains the SBR Tool. The SBR Tool can be implemented in two different versions:

- High-Quality SBR Tool
- Low-Power SBR Tool

The different versions can also be operated in down-sampled SBR-mode.

The internal sampling rate of the SBR Tool shall always be twice the sampling rate indicated by samplingFrequency or samplingFrequencyIndex in the AudioSpecificConfig().

A conformant implementation of the SBR tool that receives an SBR enhanced data stream shall operate in upsampling mode only, until an sbr_header is received, ensuring that the SBR data can be decoded correctly.

6.6.17.2.1.1 HE-AAC

The ability to do down-sampled SBR is mandatory for levels 3 and 4 of the High Efficiency AAC Profile.

A decoder conforming to that profile and level shall operate the down-sampled SBR tool if one of the following conditions is fulfilled:

- extensionSamplingFrequency is the same as samplingFrequency, or
- extensionSamplingFrequencyIndex is the same as samplingFrequencyIndex, or

the output sampling rate would otherwise exceed the maximum allowed output sample rate for the given level.

An HE-AAC profile decoder shall support implicit SBR signaling, as outlined in ISO/IEC 14496-3, subclause 1.6.5.3 HE AAC Profile Decoder Behavior in Case of Implicit Signaling.

An HE-AAC profile decoder shall support explicit SBR signaling as outlined in ISO/IEC 14496-3, subclause 1.6.5.4 HE AAC Profile Decoder Behavior in Case of Explicit Signaling.

6.6.17.2.2 SBR conformance test procedure

For the sake of simplicity, conformance testing of the AOT SBR is carried out in conjunction with AAC LC.

The conformance test procedure for the SBR tool internally creates a reference for comparison, given an input compressed MPEG-4 data file and the output from the decoder under test. In order to accomplish this, apart from the *_twi_* and *_qmf_* types, every SBR conformance compressed MPEG-4 data file is divided into two parts as outlined in Figure AMD8-1, where the AAC data for the two parts is identical but the SBR header does not arrive until the second part. This ensures that in the case of implicit signaling a conformant decoder will recognise the SBR extension element at the beginning of the compressed MPEG-4 data, or in the case of explicit signalling, by parsing the audioSpecificConfig(). Since no SBR header is present it cannot start SBR decoding and will hence do up-sampling using the SBR QMF filterbank in anticipation of the SBR header.

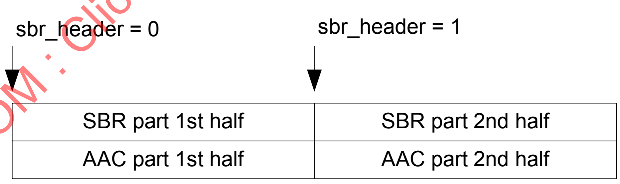

Figure AMD8-1 — The disposition of the SBR conformance compressed MPEG-4 data files

The conformance test procedure stipulate:

- reading the compressed MPEG-4 data file;
- while no SBR header is present taking the input decoded file (the output from the decoder under test) and down-sample the signal;
- store the down-sampled signal.

Since this signal is just an up-sampled version of the output-signal from the AAC and hence the input signal to the SBR decoder under test, it can be down-sampled, and by means of a polyphase correction filter, be approximated to be the same signal as was used by the SBR Tool in the decoder under test.

In parallel to storing the signal it shall also be fed to the reference SBR decoder where, since no SBR header is available, up-sampling is performed. The, in the reference SBR decoder, upsampled signal shall be compared to the input signal, i.e. the output from the decoder under test. This serves as a QMF test of the first half of the conformance file.

When the SBR header arrives, a SBR processed reference signal based on the stored lowband signal (that is a very close approximation of the signal that the SBR Tool in the decoder under test used) shall be calculated.

This means that it is possible to test the accuracy of the SBR part of the implementation, without having to deal with the differences between the AAC implementation used in the decoder under test, and the AAC implementation used for producing reference waveforms. Furthermore, the accuracy of the QMF implementation of the decoder under test is tested separately for every conformance sequence.

If a complex QMF filter bank with a modified internal phase angle (hereinafter referred to as twiddles) is used in the decoder under test, the al_sbr_twi_* sequences shall be tested first. Contrary to the other al_sbr_* test-sequences, these sequences consist of zero signal AAC-data and SBR elements that trigger sine-addition in the entire frequency range covered by SBR. The output-signal of this test sequence will contain sinusoids with phase angles that depend on the implementation of the synthesis filterbank in the decoder under test. Based on the obtained output signal from the decoder under test, the two parameters φ and β describing the phase characteristics of the filter banks in the decoder under test are identified. Since different phase characteristics may be used, depending on whether downsampled SBR is used or not, both al_sbr_twi_* sequences must be run prior to conformance testing. The file sequence is only used to perform an examination and identification of the filter bank. It does not test conformance of the QMF bank. The performance of the total QMF (analysis – synthesis) shall be tested with the *_qmf_* sequences and the other informative QMF bank tests.

In order to ensure that the QMF is implemented correctly, the output from the QMF test specific sequences *_qmf_* are compared to the internal reference without down-sampling and storing the first half of the test sequence. This is since it is possible, however very unlikely, to introduce errors in the QMF implementation that from the SBR conformance test procedure point of view look like differences in the AAC implementation. By, for the QMF specific sequences, omitting the parts of the tool that are designed to neglect differences in the AAC implementation it is ensured that the QMF is implemented correctly.

If the decoder under test passes the conformance criteria for the dedicated QMF test sequences, this is a good indication that the QMF implementation is accurate. However, it is no definite guarantee, and hence it could happen that a QMF implementation that barely passes the conformance for the QMF test, does not pass conformance for other parts of the system due to the QMF implementation. Therefore, it is useful to observe the result from the QMF test for the first half for any of the conformance sequences. This can give a good indication of the origin of a potential error.

Figure AMD8-2 outlines the SBR conformance test procedure.

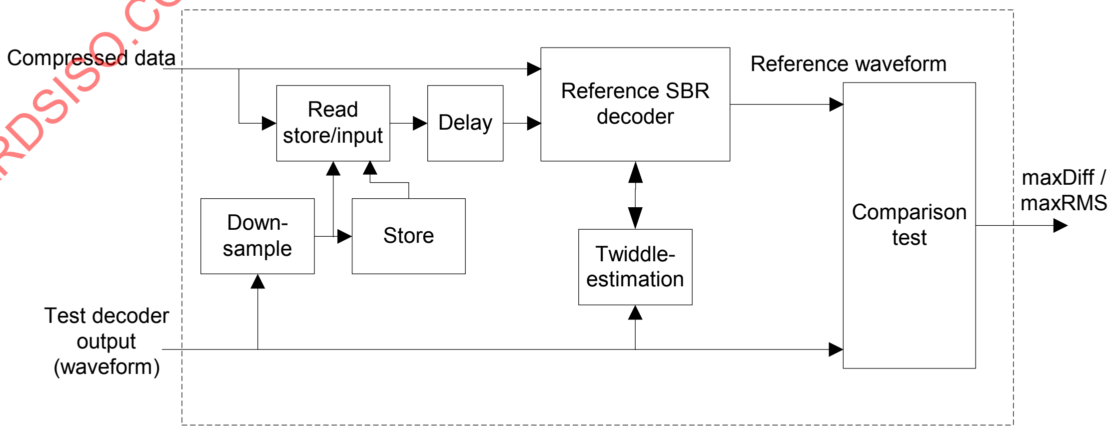

Figure AMD8-2 — Block diagram of the SBR conformance test procedure

The essential modules are:

- **Read store/input**: This module parses the SBR part of the compressed MPEG-4 data and searches for the SBR header. When no SBR header is available, the downsampled Test decoder output is routed to both the Storage Module and through the Delay module to the Reference SBR decoder. When an SBR header is available, the data stored in the Storage module is routed through the Delay module to the Reference SBR decoder. The Test decoder output is routed to the comparison test module.

- **Down-sample**: This module downsamples the Test decoder output signal, by decimation, and applies a polyphase filter that approximates the inverse of the equivalent polyphase filter of the QMF-upsampler. The delay imposed by the downsampler is given by: $delay = 32 \cdot \left(\frac{K-1}{2} \right)$, where $K = 25$ is the length of the polyphase filter. If down-sampled SBR is used, the down-sampler omits the decimation and does only the polyphase filtering. The polyphase filter matrix $\mathbf{H}(k,l)$ of size $32 \times K$ is tabulated in Table AMD8-2. The polyphase filtering step consists of the operation which maps a time signal $x(n)$ to $y(n)$, where $y(k+32i) = \sum_{l=0}^{K-1} \mathbf{H}(k,l)\, x\big(k+32(i-l)\big), \quad k = 0,1,\ldots,31.$

- **Twiddle-estimation:** This module identifies the twiddles used in the synthesis filter bank in the codec under test, and based on this computes the analysis filterbank twiddles in the codec under test. This information is passed on to the reference SBR decoder where the QMF filter bank is given the same distribution of the twiddle factors as used in the decoder under test.

- **Reference SBR decoder** is a reference SBR decoder according to the ISO specification. It generates a reference signal based on the stored low-band signal and the compressed MPEG-4 SBR data. The delay of the reference SBR decoder (at the input sampling rate) is 481 samples.

- **Comparison test,** this module calculates the difference signals between the output from the decoder under test and the internal reference. The maximum amplitude of the difference signal as well as the RMS of the difference signal are calculated. These conformance criteria are specified with respect to PCM-sample in the range –32768 .. 32767.

The process of identifying the QMF filter bank is outlined in the following steps:

- Decode the SBR compressed data so that the SBR start band k_x and number of SBR bands M are determined.

- Set the following variables:

  Upsampling factor $upS = 2 - bDownSampledSbr$,
  DFT analysis length $fftLen = 16384\, upS$,
  Number of synthesis QMF channels $L = 32\, upS$,
  Prototype filter order $N = 320\, upS$, and
  SBR stop band $k_y = \min\left(k_x + M, L\right)$,
  where $bDownSampledSbr$ is 1 if down-sampled SBR is used, otherwise zero.

- Read at least $fftLen$ samples from the mono/left channel output of the decoder under test and store them in the array $inputSignal(n)$.

- Compute DFT with frequency offset $H(k)$ as

$$H(k)=\sum_{n=0}^{fftLen-1} inputSignal(n)\exp\left(-i\frac{\pi}{L}(k+0.5)n\right),\quad k_x \le k < k_y$$

- Compute the scaled unwrapped phase $P(k)$ according to the following pseudo code:

$P(k_x\text{-}1) = 0$

for (k = k_x; k < k_y; k++) {

    P(k) = $ATAN2$($IMAG$(H(k)), $REAL$(H(k))) / π

    if(P(k) > P(k-1))

        while(P(k) - P(k-1) > 1)

            P(k) = P(k) - 2

    else

        while(P(k-1) - P(k) > 1)

            P(k) = P(k) + 2

}

where $ATAN2(\ )$ is the arc tangent in four quadrants, and $REAL(\ )$ and $IMAG(\ )$ are operators to extract the real and the imaginary parts of the argument respectively.

- Perform linear curve fitting using least square error minimization as

$$K=\sum_{k=k_x}^{k_y-1} k^2 \qquad R=\sum_{k=k_x}^{k_y-1} k$$

$$S=\sum_{k=k_x}^{k_y-1} k\,P(k) \qquad T=\sum_{k=k_x}^{k_y-1} P(k)$$

$$f(k)=ak+b,\quad where$$

$$a=\frac{(k_y-k_x)\,S-R\,T}{K\,(k_y-k_x)-R^2}\quad and\quad b=\frac{S-K\,a}{R}$$

- Identification of the terms of the complex exponent

$-j\,\pi\,f(k)=-j\,\pi\,(ak+b)$ with

$-j\frac{\pi}{L}\{(k+0.5)\varphi+\beta\}$ makes

$$\varphi = L\,a$$

$$\beta = L\,b-\frac{\varphi}{2}$$

- Quantize φ in $1/L$ steps and β in steps of one as

$$\varphi = \frac{NINT(L\varphi)}{L}$$
$$\beta = NINT(\beta)$$

- The synthesis QMF bank can subsequently be identified as using the modulation matrix $N(k,n)$ as

$$N(k,n) = \frac{1}{64}\exp\left(i\frac{\pi}{L}\{(k+0.5)\varphi+\beta\}\right)\exp\left(i\frac{\pi}{L}(k+0.5)n\right) =$$
$$= \frac{1}{64}\exp\left(i\frac{\pi}{L}\{(k+0.5)(n+\varphi)+\beta\}\right) \quad ,\begin{cases}0 \le k < L \\ 0 \le n < 2L\end{cases}$$

- From the synthesis modulation expression, the analysis modulation matrix $M(k,n)$ may be obtained as

$$M(k,n) = \exp\left(-i\frac{\pi}{L}\{(k+0.5)(N+\varphi)+\beta\}\right)\exp\left(i\frac{\pi}{L_A}(k+0.5)n\right) =$$
$$= \exp\left(i\frac{\pi}{L_A}\left\{(k+0.5)\left(n-N_A-\frac{\varphi}{upS}\right)-\frac{\beta}{upS}\right\}\right) \quad ,\begin{cases}0 \le k < L_A \\ 0 \le n < 2L_A\end{cases}$$

where

L_A is the number of analysis QMF channels (32), and
N_A is the prototype filter order for the analysis QMF bank (320).

The above calculated analysis and synthesis matrices should be used when calculating the reference SBR decoded signal that is to be compared with the output signal of the decoder under test.

Table AMD8-2 — Filter coefficients for the polyphase downsample filter

```
H[32][25] = {
   {3.476697953131265e-008, 0, -6.763157773587183e-008, 0, -6.974973600673098e-006, 0,
   -1.924219571106937e-004, 0,  1.629822651831791e-004, 0,  3.540528761884519e-004, 0,
    9.999940276586996e-001, 0,  3.540528761884519e-004, 0,  1.629822651831791e-004, 0,
   -1.924219571106937e-004, 0, -6.974973600673098e-006, 0, -6.763157773587183e-008, 0,
    3.476697953131265e-008},
   {1.396471916031115e-007, 0, -2.974245852183672e-007, 0, -1.318138373054617e-005, 0,
   -3.865443699577471e-004, 0,  3.721410827846135e-004, 0,  3.666791540864976e-004, 0,
    9.999958594813760e-001, 0,  3.666791540864976e-004, 0,  3.721410827846135e-004, 0,
   -3.865443699577471e-004, 0, -1.318138373054617e-005, 0, -2.974245852183672e-007, 0,
    1.396471916031115e-007},
   {1.224963549817200e-007, 0, -2.646404681011980e-007, 0, -1.108178601327391e-005, 0,
   -3.610222352290030e-004, 0,  3.554111749102114e-004, 0,  3.596349479868729e-004, 0,
    1.000005611329237e+000, 0,  3.596349479868729e-004, 0,  3.554111749102114e-004, 0,
   -3.610222352290030e-004, 0, -1.108178601327391e-005, 0, -2.646404681011980e-007, 0,
    1.224963549817200e-007},
   {1.326399319916421e-007, 0, -2.700352437567778e-007, 0, -9.882252437042349e-006, 0,
   -3.735656036780282e-004, 0,  3.517646637852679e-004, 0,  3.633285653067748e-004, 0,
    1.000010164947417e+000, 0,  3.633285653067748e-004, 0,  3.517646637852679e-004, 0,
   -3.735656036780282e-004, 0, -9.882252437042349e-006, 0, -2.700352437567778e-007, 0,
    1.326399319916421e-007},
   {1.522562232650754e-007, 0, -2.984765811933622e-007, 0, -8.632246089961839e-006, 0,
   -3.981722856143535e-004, 0,  3.668277251617622e-004, 0,  3.652031799326229e-004, 0,
    1.000002336408856e+000, 0,  3.652031799326229e-004, 0,  3.668277251617622e-004, 0,
   -3.981722856143535e-004, 0, -8.632246089961839e-006, 0, -2.984765811933622e-007, 0,
    1.522562232650754e-007},
   {1.770077632287240e-007, 0, -3.191765536892460e-007, 0, -7.252506589625763e-006, 0,
   -4.269398602904086e-004, 0,  3.673910838620092e-004, 0,  3.739888363247150e-004, 0,
    1.000002916620203e+000, 0,  3.739888363247150e-004, 0,  3.673910838620092e-004, 0,
   -4.269398602904086e-004, 0, -7.252506589625763e-006, 0, -3.191765536892460e-007, 0,
    1.770077632287240e-007},
   {2.068765829715687e-007, 0, -3.511660788410439e-007, 0, -5.652904186405254e-006, 0,
   -4.594357796680696e-004, 0,  3.776142052929487e-004, 0,  3.659398368134911e-004, 0,
    1.000009380633999e+000, 0,  3.659398368134911e-004, 0,  3.776142052929487e-004, 0,
   -4.594357796680696e-004, 0, -5.652904186405254e-006, 0, -3.511660788410439e-007, 0,
    2.068765829715687e-007},
   {2.421414005440356e-007, 0, -3.782805603719183e-007, 0, -3.994773586637473e-006, 0,
   -4.950706931314535e-004, 0,  3.790387553154356e-004, 0,  3.671113185055310e-004, 0,
    1.000014455098329e+000, 0,  3.671113185055310e-004, 0,  3.790387553154356e-004, 0,
   -4.950706931314535e-004, 0, -3.994773586637473e-006, 0, -3.782805603719183e-007, 0,
    2.421414005440356e-007},
   {2.788460966259352e-007, 0, -4.079961401854086e-007, 0, -2.260958885598900e-006, 0,
   -5.296088142178838e-004, 0,  3.835318699545681e-004, 0,  3.763419839421675e-004, 0,
    9.999940885015929e-001, 0,  3.763419839421675e-004, 0,  3.835318699545681e-004, 0,
   -5.296088142178838e-004, 0, -2.260958885598900e-006, 0, -4.079961401854086e-007, 0,
    2.788460966259352e-007},
   {3.161911779625002e-007, 0, -4.313203422567221e-007, 0, -5.273303865290991e-007, 0,
   -5.625765678906092e-004, 0,  3.829529676084116e-004, 0,  3.673391073119114e-004, 0,
    9.999951684002906e-001, 0,  3.673391073119114e-004, 0,  3.829529676084116e-004, 0,
   -5.625765678906092e-004, 0, -5.273303865290991e-007, 0, -4.313203422567221e-007, 0,
    3.161911779625002e-007},
   {3.543122532697675e-007, 0, -4.608625488256222e-007, 0,  1.184078570054111e-006, 0,
   -5.943681506843667e-004, 0,  3.884059372906167e-004, 0,  3.792552922175888e-004, 0,
    9.999972692390483e-001, 0,  3.792552922175888e-004, 0,  3.884059372906167e-004, 0,
   -5.943681506843667e-004, 0,  1.184078570054111e-006, 0, -4.608625488256222e-007, 0,
    3.543122532697675e-007},
   {3.909402775633086e-007, 0, -4.739340938681313e-007, 0,  2.706078490008691e-006, 0,
   -6.234965573461278e-004, 0,  3.816668617870812e-004, 0,  3.772857914767589e-004, 0,
    1.000002886171420e+000, 0,  3.772857914767589e-004, 0,  3.816668617870812e-004, 0,
   -6.234965573461278e-004, 0,  2.706078490008691e-006, 0, -4.739340938681313e-007, 0,
    3.909402775633086e-007},
```

```
{4.222809367831911e-007, 0, -4.878510064125678e-007, 0,  4.036795191477909e-006, 0,
-6.473621895606138e-004, 0,  3.791705819779010e-004, 0,  3.847083857331159e-004, 0,
 9.999969286735556e-001, 0,  3.847083857331159e-004, 0,  3.791705819779010e-004, 0,
-6.473621895606138e-004, 0,  4.036795191477909e-006, 0, -4.878510064125678e-007, 0,
 4.222809367831911e-007},
{4.502138847524999e-007, 0, -4.999326042535363e-007, 0,  5.135058417623972e-006, 0,
-6.679743577822745e-004, 0,  3.771635262404024e-004, 0,  3.866181819880556e-004, 0,
 9.999972973017242e-001, 0,  3.866181819880556e-004, 0,  3.771635262404024e-004, 0,
-6.679743577822745e-004, 0,  5.135058417623972e-006, 0, -4.999326042535363e-007, 0,
 4.502138847524999e-007},
{4.709334223508883e-007, 0, -5.078830669590950e-007, 0,  5.962080730857686e-006, 0,
-6.828692913148293e-004, 0,  3.751956167593989e-004, 0,  3.827003698106720e-004, 0,
 1.000008336100520e+000, 0,  3.827003698106720e-004, 0,  3.751956167593989e-004, 0,
-6.828692913148293e-004, 0,  5.962080730857686e-006, 0, -5.078830669590950e-007, 0,
 4.709334223508883e-007},
{4.843159761574267e-007, 0, -5.062025248855524e-007, 0,  6.458113652318982e-006, 0,
-6.923844198753944e-004, 0,  3.691031022061020e-004, 0,  3.826743784427686e-004, 0,
 1.000007446899661e+000, 0,  3.826743784427686e-004, 0,  3.691031022061020e-004, 0,
-6.923844198753944e-004, 0,  6.458113652318982e-006, 0, -5.062025248855524e-007, 0,
 4.843159761574267e-007},
{4.881790944646534e-007, 0, -5.091494784777248e-007, 0,  6.616136460265565e-006, 0,
-6.950689516769845e-004, 0,  3.698758977096631e-004, 0,  3.826493196713532e-004, 0,
 9.999860218829778e-001, 0,  3.826493196713532e-004, 0,  3.698758977096631e-004, 0,
-6.950689516769845e-004, 0,  6.616136460265565e-006, 0, -5.091494784777248e-007, 0,
 4.881790944646534e-007},
{4.843159761574265e-007, 0, -5.062025248858790e-007, 0,  6.458113652319081e-006, 0,
-6.923844198753948e-004, 0,  3.691031022060960e-004, 0,  3.826743784426941e-004, 0,
 1.000007446899662e+000, 0,  3.826743784426941e-004, 0,  3.691031022060960e-004, 0,
-6.923844198753948e-004, 0,  6.458113652319081e-006, 0, -5.062025248858790e-007, 0,
 4.843159761574265e-007},
{4.709334223509752e-007, 0, -5.078830668471703e-007, 0,  5.962080730857398e-006, 0,
-6.828692913148301e-004, 0,  3.751956167594234e-004, 0,  3.827003698109648e-004, 0,
 1.000008336100521e+000, 0,  3.827003698109648e-004, 0,  3.751956167594234e-004, 0,
-6.828692913148301e-004, 0,  5.962080730857398e-006, 0, -5.078830668471703e-007, 0,
 4.709334223509752e-007},
{4.502138847525010e-007, 0, -4.999326042533778e-007, 0,  5.135058417624179e-006, 0,
-6.679743577822758e-004, 0,  3.771635262403889e-004, 0,  3.866181819878970e-004, 0,
 9.999972973017255e-001, 0,  3.866181819878970e-004, 0,  3.771635262403889e-004, 0,
-6.679743577822758e-004, 0,  5.135058417624179e-006, 0, -4.999326042533778e-007, 0,
 4.502138847525010e-007},
{4.222809367831921e-007, 0, -4.878510064126016e-007, 0,  4.036795191477679e-006, 0,
-6.473621895606155e-004, 0,  3.791705819779300e-004, 0,  3.847083857334735e-004, 0,
 9.999969286735594e-001, 0,  3.847083857334735e-004, 0,  3.791705819779300e-004, 0,
-6.473621895606155e-004, 0,  4.036795191477679e-006, 0, -4.878510064126016e-007, 0,
 4.222809367831921e-007},
{3.909402775633092e-007, 0, -4.739340938680947e-007, 0,  2.706078490009059e-006, 0,
-6.234965573461287e-004, 0,  3.816668617870499e-004, 0,  3.772857914763712e-004, 0,
 1.000002886171421e+000, 0,  3.772857914763712e-004, 0,  3.816668617870499e-004, 0,
-6.234965573461287e-004, 0,  2.706078490009059e-006, 0, -4.739340938680947e-007, 0,
 3.909402775633092e-007},
{3.543122532697675e-007, 0, -4.608625488256092e-007, 0,  1.184078570054243e-006, 0,
-5.943681506843663e-004, 0,  3.884059372906052e-004, 0,  3.792552922174683e-004, 0,
 9.999972692390470e-001, 0,  3.792552922174683e-004, 0,  3.884059372906052e-004, 0,
-5.943681506843663e-004, 0,  1.184078570054243e-006, 0, -4.608625488256092e-007, 0,
 3.543122532697675e-007},
{3.161911779624999e-007, 0, -4.313203422567522e-007, 0, -5.273303865294305e-007, 0,
-5.625765678906092e-004, 0,  3.829529676084380e-004, 0,  3.673391073122271e-004, 0,
 9.999951684002906e-001, 0,  3.673391073122271e-004, 0,  3.829529676084380e-004, 0,
-5.625765678906092e-004, 0, -5.273303865294305e-007, 0, -4.313203422567522e-007, 0,
 3.161911779624999e-007},
{2.788460966259361e-007, 0, -4.079961401853398e-007, 0, -2.260958885598140e-006, 0,
-5.296088142178834e-004, 0,  3.835318699545062e-004, 0,  3.763419839414048e-004, 0,
 9.999940885015904e-001, 0,  3.763419839414048e-004, 0,  3.835318699545062e-004, 0,
-5.296088142178834e-004, 0, -2.260958885598140e-006, 0, -4.079961401853398e-007, 0,
 2.788460966259361e-007},
```

```
    {2.421414005440362e-007, 0, -3.782805603718790e-007, 0, -3.994773586636954e-006, 0,
    -4.950706931314529e-004, 0,  3.790387553153991e-004, 0,  3.671113185050744e-004, 0,
     1.000014455098326e+000, 0,  3.671113185050744e-004, 0,  3.790387553153991e-004, 0,
    -4.950706931314529e-004, 0, -3.994773586636954e-006, 0, -3.782805603718790e-007, 0,
     2.421414005440362e-007},
    {2.068765829715683e-007, 0, -3.511660788410130e-007, 0, -5.652904186404924e-006, 0,
    -4.594357796680679e-004, 0,  3.776142052929195e-004, 0,  3.659398368131350e-004, 0,
     1.000009380633996e+000, 0,  3.659398368131350e-004, 0,  3.776142052929195e-004, 0,
    -4.594357796680679e-004, 0, -5.652904186404924e-006, 0, -3.511660788410130e-007, 0,
     2.068765829715683e-007},
    {1.770077632288873e-007, 0, -3.191765536890821e-007, 0, -7.252506589403714e-006, 0,
    -4.269398602904083e-004, 0,  3.673910838620111e-004, 0,  3.739888363247353e-004, 0,
     1.000002916620202e+000, 0,  3.739888363247353e-004, 0,  3.673910838620111e-004, 0,
    -4.269398602904083e-004, 0, -7.252506589403714e-006, 0, -3.191765536890821e-007, 0,
     1.770077632288873e-007},
    {1.522562232650743e-007, 0, -2.984765811933783e-007, 0, -8.632246089962066e-006, 0,
    -3.981722856143523e-004, 0,  3.668277251617779e-004, 0,  3.652031799328445e-004, 0,
     1.000002336408854e+000, 0,  3.652031799328445e-004, 0,  3.668277251617779e-004, 0,
    -3.981722856143523e-004, 0, -8.632246089962066e-006, 0, -2.984765811933783e-007, 0,
     1.522562232650743e-007},
    {1.326399319916421e-007, 0, -2.700352437567779e-007, 0, -9.882252437042350e-006, 0,
    -3.735656036780283e-004, 0,  3.517646637852678e-004, 0,  3.633285653067912e-004, 0,
     1.000010164947417e+000, 0,  3.633285653067912e-004, 0,  3.517646637852678e-004, 0,
    -3.735656036780283e-004, 0, -9.882252437042350e-006, 0, -2.700352437567779e-007, 0,
     1.326399319916421e-007},
    {1.224963549817213e-007, 0, -2.646404681008590e-007, 0, -1.108178601327352e-005, 0,
    -3.610222352290024e-004, 0,  3.554111749101667e-004, 0,  3.596349479863394e-004, 0,
     1.000005611329234e+000, 0,  3.596349479863394e-004, 0,  3.554111749101667e-004, 0,
    -3.610222352290024e-004, 0, -1.108178601327352e-005, 0, -2.646404681008590e-007, 0,
     1.224963549817213e-007},
    {1.396471916031121e-007, 0, -2.974245852182784e-007, 0, -1.318138373054608e-005, 0,
    -3.865443699577474e-004, 0,  3.721410827846046e-004, 0,  3.666791540863757e-004, 0,
     9.999958594813758e-001, 0,  3.666791540863757e-004, 0,  3.721410827846046e-004, 0,
    -3.865443699577474e-004, 0, -1.318138373054608e-005, 0, -2.974245852182784e-007, 0,
     1.396471916031121e-007},
};
```

The test procedures specified in Table AMD8-3 have to be applied. The testing shall be done using the test procedure defined above. The conformance of the underlying AAC decoder shall be tested before conformance testing is done for the SBR Tool. The SBR Tool is based on a pseudo QMF filterbank. This is the most critical part of the SBR Tool in terms of precision. Hence, in order to simplify conformance testing, it is recommended to first check the accuracy of the QMF implementation, after proper identification of the filterbank twiddling in the decoder under test.

The Low Power SBR Tool has an aliasing detection algorithm that includes threshold binary decisions which can give rise to large output differences due to small rounding errors. In the aliasing detection algorithm a decision is made on how to modify gain calculation in the envelope adjuster. If potentially strong aliasing is detected, a clear indication on how to modify the gain values will be obtained, and no rounding problems will occur. However, if no strong aliasing is detected, the gain values will be modified anyway, albeit to a much smaller extent. Since no strong aliasing is detected, the modification decision could vary due to rounding effects in the aliasing detection algorithm. A small rounding error will then cause the gain calculation algorithm to modify different gain values, and hence an inaudible, but clearly measurable difference is observed in the output. The conformance sequences are designed to avoid the above problem, but differences can appear for other sequences.

The inverse filtering in the HF generator is another module that is prone to give output differences due to rounding effects. Similarly to the differences due to rounding in the aliasing detection, these are inaudible differences.

Reference waveform output files from decoding the compressed MPEG-4 data conformance sequences with the MPEG-4 reference software are also available. These files are available for reference only, and are not part of the official conformance testing procedure.

Notes: The reference software implementation of the conformance test procedure will behave differently for the test sequences named al_sbr_twi_*, since for these test sequences the QMF implementation is identified with respect to the twiddle factors used on the analysis and synthesis side of the QMF.

The reference software implementation of the conformance test procedure will behave differently for the test sequences named al_sbr_qmf_*, since for these test sequences only the QMF implementation is tested, as outlined above in the tool description. For these sequences only the output from the QMF test is displayed. This QMF test is not informative as for the other sequences, it is normative and shall be passed in order for the device under test to be conformant.

6.6.17.2.3 Test sequences

Table AMD8-3 — SBR test sequences

file base name	content	bitrate (kbit/s)	QMF Identification	QMF Accuracy	Envelope Adjuster Accuracy	Grid control tests	Header Change Tests	Inverse Filtering Tests	Additional Sines Tests	CRC	Diff max	RMS max (linear value)	test procedure
al_sbr_twi	none	24	y	y	-	-	-	-	y	-	-	-	-
al_sbr_qmf	Sine Sweep	24	-	Y	-	-	-	-	-	-	5	1.4	maxDiff/RMS
al_sbr_e	rectangle * 10Hz sine	24/48	-	-	y	-	-	-	-	y[A]	90	2.0	maxDiff/RMS
al_sbr_gh	rectangle * 10Hz sine	24/48	-	-	-	y	y	-	-	-	51	1.5	maxDiff/RMS
al_sbr_i	rectangle + noise	24/48	-	-	-	-	-	y	-	y[A]	36	3.4	maxDiff/RMS
al_sbr_s	noise	24	-	-	-	-	-	-	y	-	120	1.9	maxDiff/RMS
al_sbr_cm	music	24-128	-	-	-	-	-	-	-	-	-	-	-
al_sbr_sig	music	48	-	-	-	-	-	-	-	-	-	-	-
al_sbr_sr	music	24-56	-	-	-	-	-	-	-	-	-	-	-

[A] CRC enabled for 32 kHz testvectors

After subclause 6.8.4 AudioSource and Sound, add the following subclause:

6.8.5 AudioClip

6.8.5.1 BIFS fields Characteristic

None.

6.8.5.2 Procedure to Test AudioClip Node

Testing the AudioClip node shall be performed by comparing the output of a decoder under test with a reference output supplied by the electronic attachment to this part of ISO/IEC 14496 using the procedure RMS measurement (subclause 5.6.1.2.2.2). (Test scenes AB021 to AB024).

6.8.5.3 Audio BIFS Test Scenes

AB021 One AudioClip node, using CU1_Yx as input, connected to one Sound2D node with default fields, except spatialize = FALSE.

AB022 The same as AB021, with CU2_Yx as input.

AB023 One AudioClip node, using CU3_Yx as input, connected to one Sound node with default fields, except spatialize = FALSE.

AB024 The same as AB023, with CU4_Yx as input.

For sequences AB021 to AB024 the electronic attachment to this part of ISO/IEC 14496 provides a normative MP4 file, a reference output and a textual parametric source, to be encoded by the decoder provider using the specific input CU and either the reference encoder or an equivalent.

Table AMD8-4 — AudioBIFS Test Sequence

File Name	AB021	AB022	AB023	AB024
Content	BIFS	BIFS	BIFS	BIFS
Sequence from a source (url)	CU1	CU2	CU3	CU4

Subclause 6.8.5 becomes 6.8.6. Replace the sentence at the beginning of this subclause with:

See also subclause 6.8.2.2. Conformance Test of the click free capability of the AudioSwitch node is not required for decoders at Level 1 (sequence AB031).

At the beginning of current 6.8.5.2 (6.8.6.2 if new numbering), replace the words:

Testing the AudioSwitch Scene

with:

Testing the AudioSwitch Scene for click free capability

At the end of the same subclause add:

Testing the AudioSwitch node shall be performed by comparing the output of a decoder under test with a reference output supplied by the electronic attachment to this part of ISO/IEC 14496 using the procedure RMS measurement (subclause 6.6.1.2.2.2). (Test scenes AB032 to AB033).

In the following subclause (6.8.5.3 in current numbering), after the description of sequence AB031, add:

AB032 Two AudioSource nodes connected to one AudioSwitch node with default fields, whichChoice initially set to [1 0], and to a Sound2D with default fields (except spatialize at FALSE) using as inputs CU1_Yx and CU2_Yx. Switching whichChoice to [0 1] is performed after 5.5 seconds.

AB033 The same as AB032, with CU3_Yx and CU4_Yx as inputs.

At the end of the same subclause add:

For sequences AB032 and AB033 the electronic attachment to this part of ISO/IEC 14496 provides a normative MP4 file, a reference output and a textual parametric source, to be encoded by the decoder provider using the specific input CU and either the reference encoder or an equivalent.

Replace Table 100 with:

Table 100

File Name	AB031	AB032	AB033
Content	BIFS	BIFS	BIFS
Sequence 1 from a source (url)	CU1	CU1	CU3
Sequence 2 from a source (url)	CU1	CU2	CU4

At the beginning of the AudioMix and Sampling Rate conversion subclause (currently 6.8.6) replace:

See also subclause 6.8.2.2.

with:

See also subclause 6.8.2.2. Testing of the Sampling Rate conversion is not required for Level 1 decoders. Testing of non integer ratios for Sampling Rate conversion is not required for Level 1 and Level 2 decoders

After the AudioMix and Sampling Rate conversion subclause add:

6.8.8 AudioDelay and AudioMix

6.8.8.1 BIFS fields Characteristic

None.

6.8.8.2 Procedure to Test AudioDelay + AudioMix scenes

Testing the AudioDelay node shall be performed by comparing the output of a decoder under test with a reference output supplied by the electronic attachment to this part of ISO/IEC 14496 using the procedure RMS measurement (subclause 6.6.1.2.2.2). (Test scenes AB051 and AB052).

6.8.8.3 Audio BIFS Test Scenes

AB051 One AudioSource node using CU1_Yx as input connected to an AudioDelay node with delay field set to 1 second, feeding one AudioMix node with matrix elements equal to 0.5; the AudioMix node receives a second AudioSource, using CU4_Yx as input, and it feeds a Sound node with default fields except spatialize = FALSE. The sources are stopped after 6 seconds.

AB052 The same as AB051, with the delay field of the AudioDelay node set to 1.5 seconds.

For sequences AB051 and AB052 the electronic attachment to this part of ISO/IEC 14496 provides a normative MP4 file, a reference output and a textual parametric source, to be encoded by the decoder provider using the specific input CU and either the reference encoder or an equivalent.

Table AMD8-5 — AudioBIFS Test Sequence

File Name	AB051	AB052
Content	BIFS	BIFS
Sequence from a source (url)	CU1 CU4	CU1 CU4

