INTERNATIONAL STANDARD ISO/IEC 14496-3:2005
TECHNICAL CORRIGENDUM 4

Published 2008-04-15

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION • МЕЖДУНАРОДНАЯ ОРГАНИЗАЦИЯ ПО СТАНДАРТИЗАЦИИ • ORGANISATION INTERNATIONALE DE NORMALISATION
INTERNATIONAL ELECTROTECHNICAL COMMISSION • МЕЖДУНАРОДНАЯ ЭЛЕКТРОТЕХНИЧЕСКАЯ КОМИССИЯ • COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

Information technology — Coding of audio-visual objects —

Part 3: **Audio**

TECHNICAL CORRIGENDUM 4

Technologies de l'information — Codage des objets audiovisuels —

Partie 3: Codage audio

RECTIFICATIF TECHNIQUE 4

Technical Corrigendum 4 to ISO/IEC 14496-3:2005 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information.*

ICS 35.040 **Ref. No. ISO/IEC 14496-3:2005/Cor.4:2008(E)**

Published in Switzerland

Replace subclause 1.6.5 with the following:

1.6.5 Signaling of SBR

1.6.5.1 Generating and signaling AAC+SBR content

The SBR tool in combination with the AAC coder provides a significant increase of audio compression efficiency. At the same time it allows for compatibility with existing AAC-only decoders. However, the audio quality for decoders without the SBR tool will of course be significantly lower than for those supporting the SBR tool. Therefore, depending on the application, a content provider or content creator may wish to choose between the two alternatives given below. In general, the SBR data is always embedded in the AAC stream in an AAC compatible way (in the extension_payload), and SBR is a pure post processing step in the decoder. Therefore, compatibility can be achieved. However, by means of different signaling the content creator can select between the full-quality mode and the backward compatibility mode as follows.

1.6.5.1.1 Ensuring full audio quality of AAC+SBR for the listener

To ensure that all listeners get the full audio quality of AAC+SBR, the stream should indicate the HE AAC profile and use the explicit, hierarchical signalling (signaling 2.A. as described below), so that it is played by HE AAC Profile decoders. With regard to AAC-only streams, an HE AAC Profile decoder will decode all AAC Profile streams of the appropriate level, as the HE AAC Profile is a superset of the AAC Profile.

1.6.5.1.2 Achieving backward compatibility with existing AAC-only decoders

The aim of this mode is to get all AAC-based decoders to play the stream, even if they don't support the SBR tool. Compatible streams can be created using the following two signaling methods:

a) indicating a profile containing AAC (e.g. the AAC Profile), except the HE AAC Profile, and using the explicit backward compatible signalling (2.B. as described below). This method is recommended for all MPEG-4 based systems in which the length of the AudioSpecificConfig() is known in the decoder. As this is not the case for LATM with audioMuxVersion==0 (see subclause 1.7), this method cannot be used for LATM with audioMuxVersion==0. In explicit backward compatible signaling, SBR-specific configuration data is added at the end of the AudioSpecificConfig(). Decoders that do not know about SBR will ignore these parts, while HE AAC Profile decoders will detect its presence and configure the decoder accordingly.

b) indicating a profile containing AAC (e.g. the AAC Profile, or an MPEG-2 AAC profile), except the HE AAC Profile, and using implicit signalling. In this mode, there is no explicit indication of the presence of SBR data. Instead, decoders check the presence while decoding the stream and use the SBR tool if SBR data is found. This is possible because SBR can be decoded without SBR-specific configuration data if a certain way of handling decoder output sample rate is obeyed, as described below for HE AAC Profile decoders.

Both methods lead to the result that the AAC part of an AAC+SBR streams will be decoded by AAC-only decoders. AAC+SBR decoders will detect the presence of SBR and decode the full quality AAC+SBR stream.

1.6.5.2 Implicit and explicit signaling of SBR

This subclause outlines the different signaling methods of SBR, and the decoder behavior for different types of signaling.

There are several ways to signal the presence of SBR data:

1. **implicit signaling:** If EXT_SBR_DATA or EXT_SBR_DATA_CRC extension_payload() elements are detected in the bitstream payload, this implicitly signals the presence of SBR data. The ability to detect and decode implicitly signaled SBR is mandatory for all High Efficiency AAC Profile (HE AAC Profile) decoders.

2. **explicit signaling:** The presence of SBR data is signaled explicitly by means of the SBR Audio Object Type in the AudioSpecificConfig(). When explicit signaling is used, implicit signaling shall not occur. Two different types of explicit signaling are available:

2.A. **hierarchical signaling:** If the first audioObjectType (AOT) signaled is the SBR AOT, a second audio object type is signaled which indicates the underlying audio object type. This signaling method is not backward compatible. This method may be needed in systems that do not convey the length of the AudioSpecificConfig(), such as LATM with audioMuxVersion==0, and content authors are encouraged to use it only when thus needed.

2.B. **backward compatible signaling:** The extensionAudioObjectType is signaled at the end of the AudioSpecificConfig(). This method shall only be used in systems that convey the length of the AudioSpecificConfig(). Hence, it shall not be used for LATM with audioMuxVersion==0.

Table 1.22 shows the decoder behavior depending on profile and audio object type indication when implicit or explicit signaling is used.

Table 1.22 — SBR Signaling and Corresponding Decoder Behavior

Bitstream payload characteristics				**Decoder behavior**	
Profile indication	**extension AudioObjectType**	**sbrPresent Flag**	**raw_data_block**	**AAC Profile decoders**	**HE AAC Profile decoders**
Profiles with AAC support other than High Efficiency AAC Profile	!= SBR (signaling 1)	-1	AAC	Play AAC	Play AAC (Note 1)
			AAC+SBR	Play AAC	Play at least AAC, should play AAC+SBR (Note 1)
	== SBR (signaling 2.B)	0	AAC	Play AAC	Play AAC (Note 2)
		1	AAC+SBR	Play AAC	Play at least AAC, should play AAC+SBR (Note 3)
High Efficiency AAC Profile	== SBR (signaling 2.A or 2.B)	1	AAC+SBR	Undefined	Play AAC+SBR (Note 3)
Note 1: Implicit signaling, check payload in order to determine output sampling frequency, or assume the presence of SBR data in the payload, giving an output sampling frequency of twice the sampling frequency indicated by samplingFrequency in the AudioSpecificConfig() (unless the down sampled SBR Tool is operated, or twice the sampling frequency indicated by samplingFrequency exceeds the maximum allowed output sampling frequency of the current level, in which case the output sampling frequency is the same as indicated by samplingFrequency). Note 2: Explicitly signals that there is no SBR data, hence no implicit signaling is present, and the output sampling frequency is given by samplingFrequency in the AudioSpecificConfig(). Note 3: Output sampling frequency is the extensionSamplingFrequency in AudioSpecificConfig().					

The upper part of Table 1.22 displays bitstream payload characteristics and decoder behavior if the profile indication is any profile with AAC, apart from the High Efficiency AAC Profile. The lower part displays bitstream payload characteristics and decoder behavior if the profile indication is the High Efficiency AAC Profile.

1.6.5.3 HE AAC profile decoder behavior in case of implicit signaling

If the presence of SBR data is backward compatible implicitly signaled (signaling 1 in the list above) the extensionAudioObjectType is not the SBR AOT, and the sbrPresentFlag is set to –1, indicating that implicit signaling may occur.

Since the HE AAC Profile decoder is a dual rate system, with the SBR Tool operating at twice the sample rate of the underlying AAC decoder, the output sample rate cannot be assumed to be that of the AAC decoder just because SBR is not explicitly signaled. The decoder shall determine the output sample rate by either of the following two methods:

- Check for the presence of SBR data in the bitstream payload prior to decoding. If no SBR data is found, the output sample rate is equal to that signaled as samplingFrequency in the AudioSpecificConfig(). If SBR data is found the output sample rate is twice that signaled as samplingFrequency in the AudioSpecificConfig

- Assume that the SBR data is available and decide the output sample rate to be twice that signaled in the AudioSpecificConfig(). If no SBR data is found once the decoding process has started, the SBR Tool can be used for upsampling only, as described in subclause 4.6.18.5.

The above only applies if twice the sample rate signaled in the AudioSpecificConfig() does not exceed the maximum output sample rate allowed for the current level. Hence, for a HE AAC Profile decoder of levels 2, 3, or 4, the output sample rate is equal to the sample rate signaled in the AudioSpecificConfig() if the latter exceeds 24kHz.

The down sampled SBR Tool shall be used when needed to ensure that the output sample rate does not exceed the maximum allowed sample rate of the present level of the High Efficiency AAC Profile decoder.

1.6.5.4 HE AAC profile decoder behavior in case of explicit signaling

If the presence of SBR data is explicitly signaled (signaling 2, in the list above) the presence of SBR data is backward compatible explicitly signaled (signaling 2.B) or non-backward explicitly signaled (signaling 2.A).

For the backward compatible explicit signaling (signaling 2.B) the extensionAudioObjectType signaled is the SBR AOT. For this backward compatible explicit signaling the sbrPresentFlag is transmitted and can be either zero or one. If the sbrPresentFlag is zero, this indicates that SBR data is not present, and hence the HE AAC Profile decoder does not have to check the Fill-element for the presence of SBR data or make assumptions on the output sample rate in anticipation of SBR data. If the sbrPresentFlag is one, SBR data is present and the HE AAC Profile decoder shall operate the SBR Tool.

For the non-backward compatible explicit signaling of SBR (signaling 2.A) the extensionAudioObjectType signaled is the SBR AOT. For this hierarchical explicit signaling, the sbrPresentFlag is set to one if the extensionAudioObjectType is SBR. The sbrPresentFlag is not transmitted and hence it is not possible to explicitly signal the absence of implicit signaling. Hence, for the hierarchical explicit signaling, SBR data is always present and the HE AAC Profile decoder shall operate the SBR Tool.

The down sampled SBR Tool shall be operated if the output sample rate would otherwise exceed the maximum allowed output sample rate for the present level, or if the extensionSamplingFrequency is the same as the samplingFrequency.

After subclause 1.6.5, insert new subclause 1.6.6 "Interface between Audio and Systems":

1.6.6 Interface between Audio and Systems

1.6.6.1 Introduction

This subclause clarifies the interface between MPEG-4 Audio and MPEG-4 Systems.

Every access unit delivered to the audio decoder from the systems interface shall result in a corresponding composition unit delivered from the audio decoder to the systems interface, i.e., the compositor. This shall include start-up and shut-down conditions, i.e., when the access unit is the first or the last in a finite sequence of access units.

1.6.6.2 Handling of composition time stamps

For an audio composition unit, ISO/IEC 14496-1 subclause 7.1.3.5 *Composition Time Stamp (CTS)* specifies that the composition time applies to the *n*-th audio sample within the composition unit. The value of *n* is 1 unless specified differently in the remainder of this subclause.

For compressed data, like HE-AAC coded audio, which can be decoded by different decoder configurations, special attention is needed. In this case, decoding can be done in a backward-compatible fashion (AAC only) as well as in an enhanced fashion (AAC+SBR). In order to insure that composition time stamps are handled correctly (so that audio remains synchronized with other media), the following applies:

- If compressed data permits both backward-compatible and enhanced decoding, and if the decoder is operating in a backwards-compatible fashion, then the decoder does not have to take any special action. In this case, the value of *n* is 1.
- If compressed data permits both backward-compatible and enhanced decoding, and if the decoder is operating in enhanced fashion such that it is using a post-processor that inserts some additional delay (e.g., the SBR post-processor in HE-AAC), then it must insure that this additional time delay incurred relative to the backwards-compatible mode, as described by a corresponding value of *n*, is taken into account when presenting the composition unit. The value of *n* is specified in Table 1.22B.

Table 1.22B – Composition time stamp handling for decoder operation modes

Value of *n*	Addional delay (Note 1)	Decoder operation mode
1	0	A) All operation modes not listed elsewhere in this table.
963	962	B1) HE-AAC or HE-AAC v2 decoder with SBR operated in dual-rate mode; decoding HE-AAC or HE-AAC v2 compressed audio.
482	481	B2) Same as B1), but with SBR operated in downsampled mode.
Note 1: The delay introduced by the post-processing is given in number of samples (per audio channel) at the output sample rate for the given decoder operation mode.		

In subclause 4.6.18.4.3 Down sampled synthesis filterbank, replace equation:

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

with:

$$\mathbf{N}(k,n)=\frac{1}{64}\cdot\exp\left(\frac{i\cdot\pi\cdot(k+0.5)\cdot(2\cdot n-127.5)}{64}\right)\quad,\begin{cases}0\le k<32\\0\le n<64\end{cases}$$

After subclause 4.6.18.4.3 Down sampled synthesis filterbank, add the following subclause:

4.6.18.4.4 Complex-exponential phase-shifting in combination with the SBR QMF banks

The following subclause clarifies the allowed usage of phase-shifts in combination with the QMF analysis and synthesis filterbanks outlined in the previous subclauses. The restrictions of the phase shifts are defined in subclause 6.6.17.2.2 of ISO/IEC 14496-4:2004 *Conformance Testing*. The phase-shifts do not affect audio quality, and are allowed in order to facilitate efficient implementations. A given filterbank implementation operating in a certain sample-rate mode, i.e., normal dual-rate operation or down-sampled SBR mode, shall not have phase-shifts that are bitstream dependent.

The theory for the QMF banks used in SBR is a complex-exponential extension of the theory of cosine modulated filter banks. In cosine modulated filter banks the analysis $h_k(n)$ and synthesis $f_k(n)$ filters are given by

$$h_k(n) = p_0(n)\cos\left(\frac{\pi}{M}(k+0.5)\left(n-\frac{N}{2}\right)+\theta_k\right) \quad ,\begin{cases}0 \le k < M \\ 0 \le n \le N\end{cases}$$

and

$$f_k(n) = p_0(n)\cos\left(\frac{\pi}{M}(k+0.5)\left(n-\frac{N}{2}\right)-\theta_k\right) \quad ,\begin{cases}0 \le k < M \\ 0 \le n \le N\end{cases}$$

where $p_0(n)$ is a real-valued symmetric low-pass prototype filter, M denotes the number of channels and N is the prototype filter order. θ_k are channel dependent factors needed for the cancellation of the main alias terms. It can be shown that the alias cancellation constraints become obsolete when extending the cosine modulated filterbank with complex-exponential modulation. Thus, for the SBR QMF banks, both the analysis and synthesis filter coefficients are

$$h_k(n) = f_k(n) = p_0(n)\exp\left(i\frac{\pi}{M}(k+0.5)\left(n-\frac{N}{2}\right)\right) \quad ,\begin{cases}0 \le k < M \\ 0 \le n \le N\end{cases}$$

Since the subband samples from the filter bank are complex-valued, an additive possibly channel-dependent phase-shift step may be appended to the analysis filterbank. These extra phase-shifts need to be compensated for before the synthesis filter bank. This is illustrated in Fig. 4.40A where $\alpha(k), 0 \le k < M$ denotes the channel dependent phase-shifts.

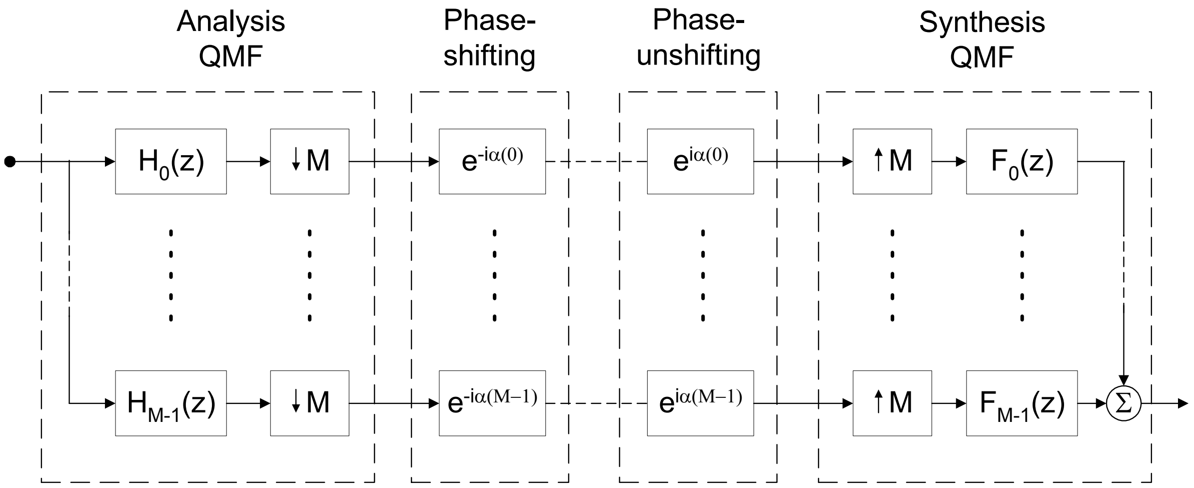

Figure 4.40A - Phase-shifting subband samples of complex-exponential modulated QMF banks

While the phase-shifting terms in principle can be of arbitrary values without destroying the operation of the QMF analysis / synthesis-chain, they are constrained to certain values by the conformance criterias. This is since the SBR signal will be affected by the choice of the phase factors while the low pass signal coming from the AAC decoder will not. The audio quality of the output signal is not affected. In subclause 6.6.17.2.2 of ISO/IEC 14496-4:2004 *Conformance Testing*, the constraints for the channel dependent phase-shifts are outlined. More specifically, the channel dependent phase-shifts $\alpha(k)$ shall be

$$\alpha(k) = \frac{\pi}{M}\left\{(k+0.5)\left(\varphi + \frac{N}{2}\right) + \beta\right\}$$

where φ shall be limited to a value being an integer multiple of $1/M$ and β shall be constrained to be an integer. Further, it is possible to identify the values of $\alpha(k)$ for the complex QMF banks outlined in the previous subclauses. Given the modulation matrix $\mathbf{M}$ or $\mathbf{N}$, for any of the QMF banks in subclauses 4.6.18.4.1 to 4.6.18.4.3, and the expression for the filter coefficients above, the phase-shifting factors are

$$\alpha(k) = -\frac{\pi}{32}(k+0.5)\cdot 31.75$$

This corresponds to $\varphi = 0.25$ and $\beta = 32$ for $M = 32$ in the subclause 6.6.17.2.2 of ISO/IEC 14496-4:2004 *Conformance Testing*.

In subclause 4.4.2.8 Payloads for the audio object type SBR, remove the following sentence from the footnote in table 4.58, table 4.59, and table 4.60:

The sbr_extension() element is reserved for future use.

In subclause 4.5.2.8.1 Payloads for the audio object type SBR - definitions, replace:

Table 4.96 – bs_extension_id

bs_extension_id	Meaning
0	reserved
1	reserved
2	reserved
3	reserved

with:

Table 4.96 – bs_extension_id

bs_extension_id	Meaning	Notes
0	reserved	
1	reserved	
2	EXTENSION_ID_PS (Note 1)	Note 1
3	reserved	
Note 1: See subclause 8.A		

and remove the then obsolete Table 8.A.2 from ISO/IEC 14496-3:2005, subpart 8, Annex 8.A, subclause 8.A.2.