INTERNATIONAL ORGANISATION FOR STANDARDISATION
ORGANISATION INTERNATIONALE DE NORMALISATION
ISO/IEC JTC1/SC29/WG11
CODING OF MOVING PICTURES AND AUDIO

ISO/IEC JTC1/SC29
WG11 N15073
February 2015, Geneva, Switzerland

Title **Technical Note: AAC-ELD v2 IMPLEMENTATION GUIDE**

Source **Communication**

AhG on Responding to Industry Needs on Adoption of MPEG Audio

The purpose of this document is to describe principles of explicit AAC-ELD v2 signaling and filterbank set-up. The intention is to support those implementing AAC-ELD codecs with an overview of the relevant standards as well as some implementation tips and hints. It is in no way intended to replace intensive study of MPEG reference software and standard documents.

1. Introduction

The AAC-ELD family includes Low Delay AAC (AAC-LD), Enhanced Low Delay AAC (AAC-ELD) and AAC-ELD v2. The order of the three family members represents an evolution in AAC low delay audio codec technology. These codecs share the same core coder while each descendant adds new coding tools *(Figure 1)*. Decoders of the AAC-ELD codec family can be expected to be fully backwards compatible. The codecs can handle mono, stereo and multichannel signals with latencies as low as 15ms at 48 kHz and support for a wide range of bit rates.

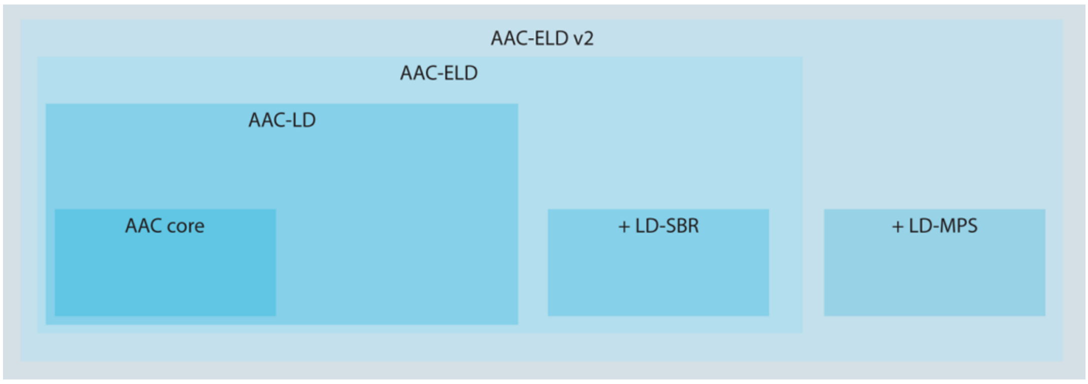

Figure 1: AAC-ELD family. AAC-LD and AAC-ELD share all major tools, only minor differences of the bit stream syntax and filterbank separate AAC-LD from AAC-ELD core. AAC-ELD v2 is based on AAC-ELD and is extended by the Low Delay MPEG Surround tool. (Image source: Fraunhofer IIS)

AAC-ELD v2 is the latest member of the AAC-ELD codec family. It is based on AAC-ELD and is extended with the “Low Delay MPEG Surround” (LD-MPS) tool, which provides improved audio quality for stereo and multichannel at low bit rates.

Instead of discrete channel coding, the AAC-ELD v2 encoder extracts spatial parameters and encodes a downmix stream *(Figure 2).* The AAC-ELD core decoder reconstructs the downmix signal while the LD-MPS decoder module recreates the spatial image by applying the LD-MPS parameters to the downmix signal *(see[1][2]).*

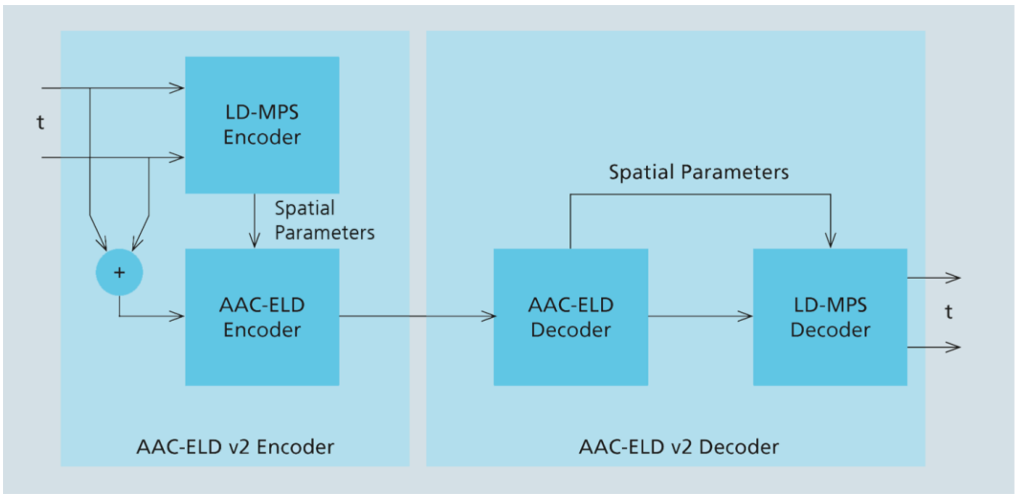

Figure 2: Block diagram of AAC-ELD v2 (Image source: Fraunhofer IIS)

2. Background

2.1 List of Abbreviations

AAC	Advanced Audio Coding
AAC-ELD	Enhanced Low Delay Advanced Audio Coding
AAC-LD	Low Delay Advanced Audio Coding
AAC-(E)LD	AAC-LD and AAC-ELD
AAC-ELD v2	Enhanced Low Delay Advanced Audio Coding Version 2
AOT	Audio Object Type
ASC	Audio Specific Configuration
CLDFB	Complex Low Delay Filterbank
DEMUX	De-Multiplexer
iMDCT	Inverse Modified Discrete Cosine Transform
LD-MDCT	Low Delay Modified Discrete Cosine Transform
LD-MPS	Low Delay MPEG Surround
LD-QMF	Low Delay Quadrature Mirror Filter
LD-SBR	Low Delay Spectral Band Replication
MDCT	Modified Discrete Cosine Transform
MPEG	Motion Picture Experts Group
MUX	Multiplexer
PCM	Pulse Code Modulation
PNS	Perceptual Noise Substitution
QMF	Quadrature Mirror Filter
SBR	Spectral Band Replication
SIP	Session Initialization Protocol
TNS	Temporal Noise Shaping

2.2 Reference Documents

As the AAC-ELD v2 coding scheme is based on its predecessors, AAC-LD and AAC-ELD, there is no single document in the MPEG standard describing all AAC-ELD v2 coding tools in one place. For AAC-ELD v2, the relevant MPEG standard documents are:

- **ISO/IEC 14496-3:2009**. This MPEG-4 standard defines the basic AAC-LD and AAC-ELD coding schemes (AOTs 23 and 39) and its signaling via the `AudioSpecificConfig()` and the `ELDSpecificConfig()` [3].
- **ISO/IEC 14496-3:2009/Amd 2:2010.** This amendment to the 14496-3 standard contains signaling of LD-MPS (AOT 44) via the `LDSpatialSpecificConfig()` [4].
- **ISO/IEC 14496-3:2009/Amd 3:2012.** This amendment to the 14496-3 standard defines the Low Delay AAC v2 Profile [7].
- **ISO/IEC 23003-1:2007.** Basic tools and bit stream elements are defined in the MPEG Surround standard [6].
- **ISO/IEC 23003-2:2010.** This standard is part of MPEG-D and describes the syntax of the `LDSpatialSpecificConfig()` and the LD-MPS payload [5].

2.3 Terminology

Coding Tools. Each MPEG codec is a composition of one or more coding tools/modules. The important tools are Low Delay Window, Filterbank, Quantization&Coding – AAC, MS

Stereo, SBR and others. For a complete list, please see table 1.1 “Audio Object Type definition based on Tools/Modules” in [3].

Audio Object Type, referred to as AOT. An Audio Object Type is a self-contained codec which consists of a defined set of coding tools. For instance, AOT “ER AAC-ELD” has the ID 39 which includes the tools Low Delay Window, Filterbank, TNS, Intensity Stereo, MS Stereo, PNS, Quantization&Coding, Low Delay SBR and others. Table 1 describes AOTs 23, 39 and 44. For a complete overview of all AOTs, please see table 1.1 “Audio Object Type definition based on Tools/Modules” in [3] and [7].

AOT	**Description**
23	Low Delay AAC (AAC-LD)
39	Enhanced Low Delay AAC (AAC-ELD) Enhancement of AAC-LD, with an delay optimized filterbank and the Low Delay Spectral Band Replication tool (LD-SBR) for low bit rate encoding
44	Low Delay MPEG Surround (LD MPEG Surround) for parametric coding of stereo and multichannel signals.

Table 1: Description of AOT 23, 39 and 44

- **Audio Profiles.** An Audio Profile provides an application-oriented set of one or more AOTs and may also restrict tools. Audio Profiles are signaled during the codec negotiation and capability exchange amidst session setup (e.g. via Session Initialization Protocol (SIP)). Table 2 shows relevant Audio profiles for the AAC-ELD family. For a complete overview of all MPEG Audio Profiles, please see section 1.5.2.1 “Profiles” in [3] and [7].

MPEG Audio Profile	**Associated AOTs**
Low Delay	23 (AAC-LD) and others
Low Delay AAC	23 (AAC-LD)
Low Delay AAC v2	23 (AAC-LD) 39 (AAC-ELD) 44 (LD MPEG Surround)

Table 2: MPEG Audio Profiles relevant for the AAC-ELD family

- **Audio Profile Levels.** These levels allow signaling of restrictions within an Audio Profile. The Low Delay AAC v2 profile consists of 4 levels which are differentiated in the number of output channels and in the low delay MPEG surround channel configuration (Table 3). For a complete overview of all Audio Profile Levels, please see section 1.5.2.3 “Levels within the profiles” in [3] and [7].

Level	**AOT of core coder**	**Max. no. output channels**	**Max. sampling rate [kHz]**	**MPEG Surround**
1	23, 39	1.0	48	n/a
2	23, 39	2.0	48	LD-MPS 2-1-2
3	23, 39	5.1	48	LD-MPS 2-1-2
4	23, 39	5.1	48	LD-MPS 5-x-5

Table 3: Levels of the Low Delay AAC v2 profile

- **Payload.** The payload contains the coded audio data including tools like SBR or LD-MPS. Different AOTs use different bit stream payload syntax.

3. Signaling of AAC-ELD v2

3.1 Signaling MPEG-4 Codecs

A decoder requires configuration information to set up its internal configuration for proper parsing and decoding of the bit stream. The configuration information is packed into the Audio Specific Configuration (ASC) and sent to the decoder before the audio data bit stream. A complete description of the ASC syntax can be found in section 1.6.2.1 “AudioSpecificConfig” in [3].

3.2 The Audio Specific Configuration for AAC-LD and AAC-ELD

In case of AAC-ELD (AOT 39) the ASC includes the `ELDSpecificConfig()` element. The exact syntax can be found in table 4.180 “Syntax of ELDSpecificConfig ()” in [3].

In case of AAC-LD (AOT 23) the ASC includes the `GASpecificConfig()`. The exact syntax can be found in section 4.4.1 “Decoder configuration (GASpecificConfig)” in [3].

3.3 The Audio Specific Configuration for LD-MPS

The configuration of the LD-MPS tool is stored in the `LDSpatialSpecificConfig()` as described in section B.2.1 “Payloads for LD MPS” in [5]. It is embedded into the ASC but, depending on the underlying core codec (AAC-ELD or AAC-LD), the `LDSpatialSpecific-Config()` has to be accessed in two different ways that are explained in the following sections 3.3.1 and 3.3.2.

3.3.1 Signaling AAC-ELD v2 (AAC-ELD and LD-MPS)

When using AAC-ELD (AOT 39) as core codec, the `LDSpatialSpecificConfig()` is embedded in the `ELDSpecificConfig()` (see table 4.180 “Syntax of ELDSpecificConfig ()” in [4]).

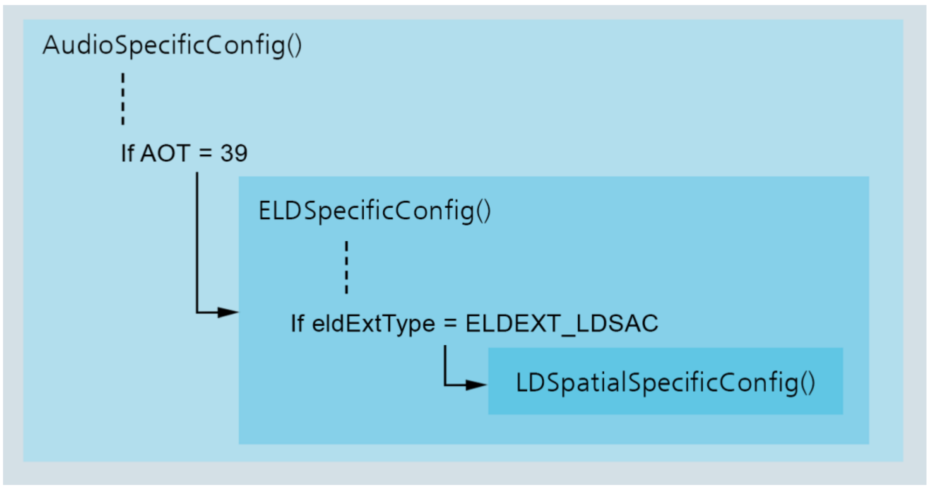

Figure 3: Signaling AAC-ELD v2 (Image source: Fraunhofer IIS)

3.3.2 Signaling AAC-LD and LD-MPS

When using AAC-LD (AOT 23) as core codec, the `LDSpatialSpecificConfig()` is embedded in the `AudioSpecificConfig()` (see table 1.15 "Syntax of AudioSpecificConfig()" in [4]).

Figure 4 shows the hierarchical structure of the ASC when signaling AAC-LD and LD-MPS.

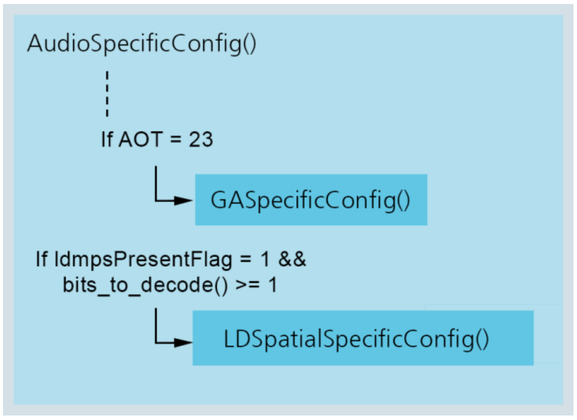

Figure 4: Signaling AAC-LD and LD-MPS (Image source: Fraunhofer IIS)

3.4 The LD-MPS Payload within the AAC Bit Stream

Every AAC frame consists of the AAC-(E)LD core data and, if applicable, also the SBR data and the extension payload. The SBR data is stored directly after the core data while the LD-MPS payload is embedded in the `extension_payload()` and classified as extension type `EXT_LDSAC_DATA` (see table 4.57 "Syntax of extension_payload()" in [4]).

To ensure forward compatibility the extension type `EXT_DATA_LENGTH` has been introduced together with LD-MPS. This extension type defines a payload that contains the explicit length of the subsequent payload container and thus enables skipping of unknown/future extension payloads.

Figure 5 shows the structure of an AAC-ELD bit stream with embedded LD-MPS payload.

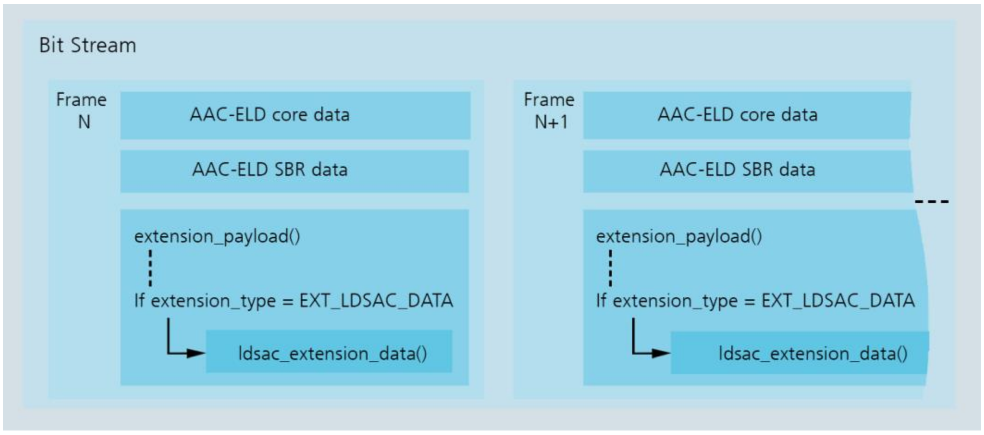

Figure 5: The LD-MPS payload within the AAC-ELD bit stream (Image source: Fraunhofer IIS)

Note: For AAC-LD the payload structure is identical, although it contains no SBR data.

4. AAC -ELD v2 Filterbank dependencies

As shown in Figure 2, an AAC-ELD v2 codec includes the AAC-ELD core codec, the LD-SBR and the LD-MPS tool. According to the MPEG standard, each of these components uses another type of filterbank as listed in Table 4.

Tool	**Used type of filterbank**
AAC-ELD core codec	LD-MDCT
LD-SBR	CLDFB
LD-MPS	LD-QMF

Table 4: The different types of filterbanks used in the AAC-ELD v2 codec

To save workload it is possible to employ the LD-QMF also for the LD-SBR tool and to omit the CLDFB. In this section there is a full explanation of the different variants of sequential arrangements of AAC-ELD v2 filterbanks for the encoder and decoder.

4.1 Filterbank Sequence of the AAC-ELD v2 Encoder

4.1.1 Extending a Legacy Encoder: Each Tool Has Its Own Filterbank

This implementation option shows how a legacy AAC-ELD encoder can be extended by the LD-MPS tool without any modification to the legacy part. As Figure 6 shows, the LD-MPS and LD-SBR tools do not exchange any frequency domain data. Two separate filterbanks - LD QMF analysis in the LD-MPS tool and CLDFB analysis in the LD-SBR tool - must be processed.

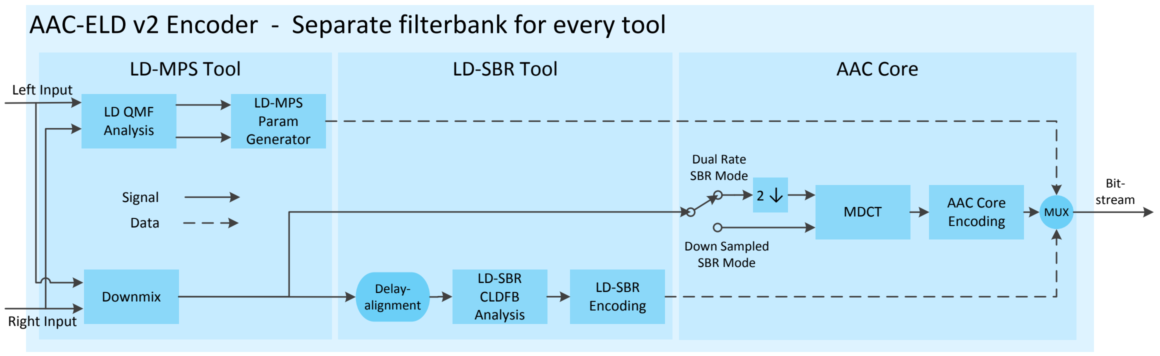

Figure 6: Extending a legacy AAC-ELD encoder by the LD-MPS tool (Image source: Fraunhofer IIS)

The LD-QMF causes a higher delay than the CLDFB. When LD-SBR operates in Dual Rate mode the LD-SBR path has to be delayed by 96 samples to align the LD-SBR to the LD-MPS tool. If the SBR tool runs in Down Sampled SBR mode[1], the LD-SBR path must be delayed by 48 samples.

4.1.2 Workload Efficient implementation: Sharing LD-QMF Data between tools

As depicted in Figure 7 the LD-MPS tool directly shares its LD-QMF filterbank output data with the LD-SBR encoding block. Therefore, it is possible to omit the LD-SBR CLDFB analysis block that is usually part of the LD-SBR encoder. In this way workload of the encoder can be reduced compared to the implementation explained in section 4.1.1.

[1] An explanation of Down Sampled LD-SBR can be found in [1] and in 4.6.18.2.1.3 “Down Sampled SBR” in [3].

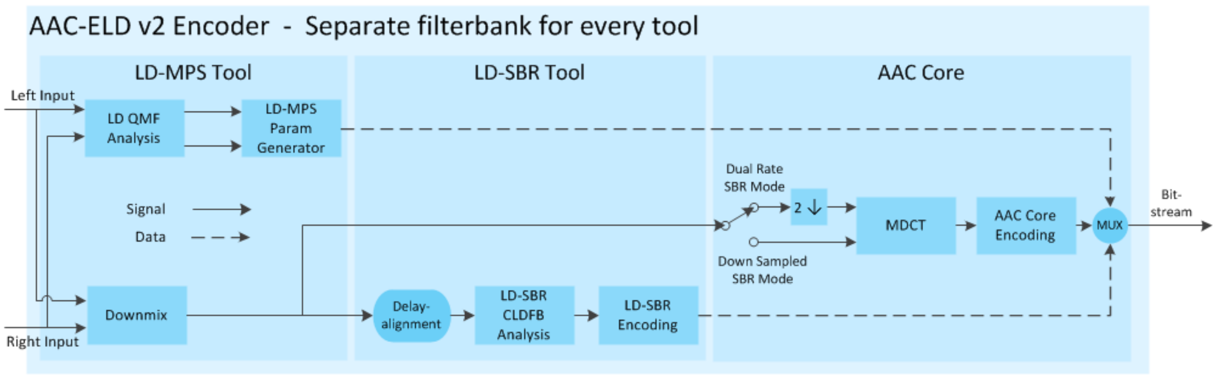

Figure 7: Workload efficient implementation achieved by sharing the output of the LD-QMF with the LD-SBR encoder (Image source: Fraunhofer IIS)

Here the LD-SBR and LD-MPS signal paths share the same LD-QMF filterbank, and therefore no delay alignment between the LD-SBR and LD-MPS tool is required.

4.2 Filterbank Sequence of the AAC-ELD v2 Decoder

When adding LD-MPS to an AAC-ELD v1 decoder a LD-QMF filterbank has to be implemented. This filterbank is used for LD-MPS processing and also for LD-SBR processing whenever LD-MPS has been activated on encoder side. The CLDFB is exclusively active for plain LD-SBR decoding, i.e. when LD-MPS is inactive. The structure of the AAC-ELD v2 decoder is depicted in Figure 8, where we have to differentiate between the following cases:

- **LD-MPS off / LD-SBR off.** If no tool is signaled as active, the iMDCT of the AAC core directly outputs the decoded signal. This case is not explicitly shown in Figure 8.
- **LD-MPS off / LD-SBR on.** If LD-SBR is signaled as active and if `ELDEXT_LDSAC` is not present in the `ELDSpecificConfig()` (see also section 3.3.1 and 4.6.20.4 “Decoding of ELDSpecificConfig” in [4]), the CLDFB is used for LD-SBR analysis and synthesis.
- **LD-MPS on / LD-SBR off.** If LD-SBR is not present in the bit stream but `ELDEXT_LDSAC` exists, the LD-QMF is used for LD-MPS analysis and synthesis.
- **LD-MPS on / LD-SBR on.** If both tools are signaled as active in the bit stream, the output of the LD-QMF analysis filterbank is directly used by the LD-SBR decoder followed by the LD-MPS decoder and LD-QMF synthesis filterbank. The CLDFB is not active in this case.

The necessary filterbank depending on the active tool is also shown in Table 5.

	LD-MPS off	**LD-MPS on**
LD-SBR off	N/A	LD-QMF
LD-SBR on	CLDFB	LD-QMF

Table 5: Used filterbank in an AAC-ELD v2 decoder depending on the active tools

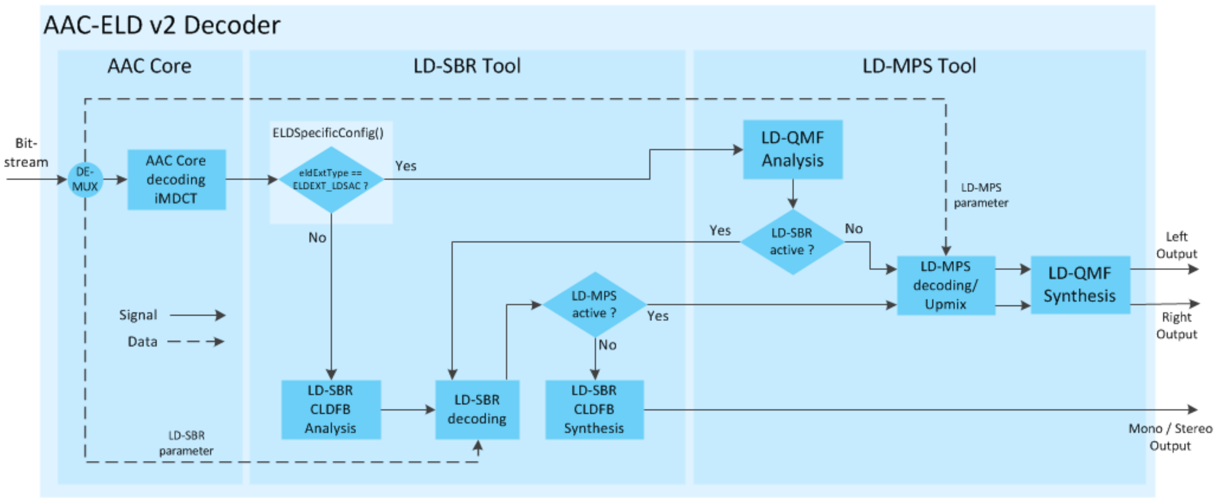

Figure 8: Filterbank sequence of the AAC-ELD v2 decoder (Image source: Fraunhofer IIS)

4.3 Forward Compatibility of AAC-ELD v1 Decoders

AAC-ELD v1 is forward compatible to AAC-ELD v2, i.e. when receiving an AAC-ELD v2 bit stream an AAC-ELD v1 decoder does not process the LD-MPS payload but outputs the decoded mono AAC core and LD-SBR parts of the signal. As a result, the LD-SBR signal lags slightly behind the AAC core signal[2] due to the AAC-ELD v2 encoder LD-SBR delay alignment of 48 or 96 samples (*see section 4.1.1*). The alignment has been done under the assumption that the decoder utilizes the LD-QMF for LD-SBR processing. Hence, the time domain input of the CLDFB analysis should be delayed by 96 samples in Dual Rate SBR mode or 48 samples in Down Sampled SBR mode to compensate for the AAC-ELD v2 encoder delay alignment.

It is recommended to update AAC-ELD v1 decoders as follows (see Figure 9):

- Adapt the ASC parser of the legacy decoder to detect the eldExtType `ELDEXT_LDSAC` in the `ELDSpecificConfig()`. If `ELDEXT_LDSAC` exists, set the flag `useLDQMFTimeAligment`[3] to 1.
- Implement a delay shift of the time domain input buffer for the LD-SBR CLDFB analysis by the appropriate number of samples (dual rate SBR: 96 samples; Down Sampled SBR: 48 samples). This alignment has to be triggered by the `useLDQMFTimeAligment` flag. For details see 4.6.20.4 “Decoding of ELDSpecificConfig” in [4].
- It is also recommended to support the extension payload `EXT_DATA_LENGTH` that enables skipping of unknown/future extension payloads (see section 3.4).

[2] Listening tests during the standardization process showed that this delay shift has almost no impact on the audio quality.
[3] In the spelling of “useLDQMFTimeAligment” the “n” lacks by intention. You will find the same typo in the standard.

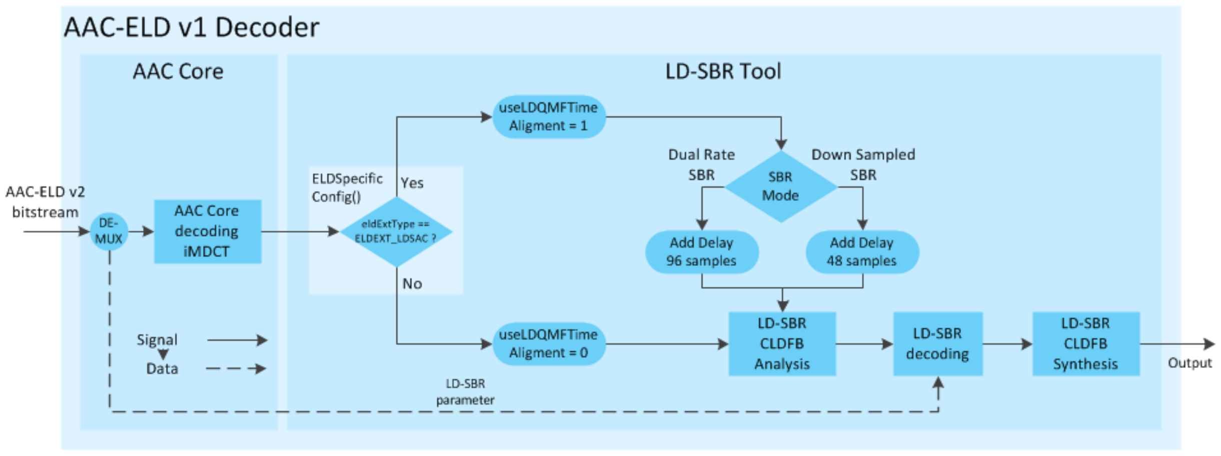

Figure 9: AAC-ELD v1 decoder dealing with AAC-ELD v2 bit streams (Image source: Fraunhofer IIS)

5. Conclusion

When implementing AAC-ELD v2 codec based on the AAC-ELD v1 codec the following modifications have to be conducted:

- **ASC writer / parser.** The configuration of the LD-MPS tool has to be stored in the `LDSpatialSpecificConfig()` that is embedded in the `ELDSpecificConfig()` for AOT 39, respectively in the ASC for AOT 23.
- **LD-MPS payload.** The LD-MPS data has to be stored in an extension payload container of the type `EXT_LDSAC_DATA`. Additionally, the payload `EXT_DATA_LENGTH` should be added to facilitate forward compatibility to possibly “unknown” extension payloads.
- **Filterbank sequence of the AAC-ELD v2 encoder.** Option one: Every tool uses its own filterbank. In this case the LD-SBR signal part has to be delayed to be in sync with the LD-MPS tool. Option two: The LD-MPS tool shares its LD-QMF data with the SBR tool. This is the recommended implementation as the workload is reduced compared to option one.
- **Filterbank sequence of the AAC-ELD v2 decoder.** Whenever LD-MPS is active the LD-QMF has to be used for processing LD-MPS as well as for LD-SBR. The CLDFB is exclusively used for plain LD-SBR decoding when LD-MPS is inactive.
- **AAC-ELD v1 decoders.** Legacy AAC-ELD v1 decoders are forward compatible to AAC-ELD v2 bit streams but cause a slight delay shift between the LD-MPS and the LD-SBR signal parts. For compensation it is recommended to update the ASC parser to detect the presence of LD-MPS and to introduce the appropriate delay to the LD-SBR signal part.

Further references

[1] Mpeg-audio.org White Paper: The AAC-ELD family for High Quality Communication Services. http://www.mpeg-audio.org/docs/w14936_(AAC-ELD-Family_WhitePaper).pdf

[2] Maria Luis Valero, Andreas Hoelzer, Markus Schnell , Johannes Hilpert, Manfred Lutzky, Jonas Engdegard, Heiko Purnhagen, Per Ekstrand, Kristofer Kjoerling. A new parametric Stereo and Multi Channel Extension for MPEG4 Enhanced Low Delay AAC (AAC-ELD v2). In 128th AES convention, London, UK, May 2010.

[3] ISO/IEC 14496-3:2009. Information technology -- Coding of audio-visual objects – Part 3: Audio.

[4] ISO/IEC 14496-3:2009/Amd 2:2010. ALS simple profile and transport of SAOC.

[5] ISO/IEC 23003-2:2010. Information technology -- MPEG audio technologies - Part 2: Spatial Audio Object Coding (SAOC).

[6] ISO/IEC 23003-1:2007. Information technology -- MPEG audio technologies -- Part 1: MPEG Surround.

[7] ISO/IEC 14496-3:2009/Amd 3:2012. Transport of unified speech and audio coding (USAC)