INTERNATIONAL STANDARD

ISO/IEC 14496-3

Third edition
2005-12-01

AMENDMENT 3
2006-06-01

Information technology — Coding of audio-visual objects —

Part 3: **Audio**

AMENDMENT 3: Scalable Lossless Coding (SLS)

Technologies de l'information — Codage des objets audiovisuels —

Partie 3: Codage audio

AMENDEMENT 3: Codage extensible sans perte (SLS)

Reference number
ISO/IEC 14496-3:2005/Amd.3:2006(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 3 to ISO/IEC 14496-3:2005/Amd. 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*.

This Amendment specifies Audio Scalable Lossless Coding (SLS).

Information technology — Coding of audio-visual objects —

Part 3: **Audio**

AMENDMENT 3: Scalable Lossless Coding (SLS)

In ISO/IEC 14496-3, Introduction, add the following to the end of the subclause "MPEG-4 general audio coding tools":

MPEG-4 SLS (Scalable Lossless Coding) is a tool used in combination with optional MPEG-4 General Audio coding tools to provide fine-grain scalable to numerical lossless coding of digital audio waveform.

In Part 3: Audio, Subpart 1, in subclause 1.3 Terms and Definitions, add:

SLS: Audio Scalable to Lossless Coding

and increase the index-number of subsequent entries.

In Part 3: Audio, Subpart 1, in subclause 1.5.1.1 Audio object type definition, amend table 1.1 with the updates in the table below:

Tools/ Modules Audio Object Type	Error Mapping (*)	Integer TNS (*)	Integer M/S (*)	IntMDCT (*)	BPGC/CBAC/LEMC (*)	Remark	Object Type ID
…							
(escape)						X	31
…							
SLS	X	X	X	X	X		37
SLS non-core				X	X		38
…							

Note: (*) marks new columns

In Part 3: Audio, Subpart 1, subclause 1.4 (Symbols and Abbreviations) add the following subclause:

1.4.9 Arithmetic data types

INT32 32 bit signed integer using two's complement

INT64 64 bit signed integer using two's complement

In Part 3: Audio, Subpart 1, subclause 1.5 add the following subclauses:

1.5.1.2.31 SLS object type

The SLS object is supported by the scalable to lossless tool which provides fine-grain scalable to lossless enhancement of MPEG perceptual audio codecs, such as AAC, allowing multiple enhancement steps from the audio quality of the core codec up to near-lossless and lossless signal representation. It also provides stand-alone lossless audio coding when the core audio codec is omitted.

1.5.1.2.32 SLS Non-Core object type

The SLS non-core object is supported by the scalable to lossless tool. It is similar to the SLS object type but the core audio codec is omitted.

In Part 3: Audio, Subpart 1, in subclause 1.6.2.1 AudioSpecificConfig, amend table 1.8 with the updates in the table below:

Syntax	No. of bits	Mnemonic

AudioSpecificConfig ()
{
…
    switch (audioObjectType) {
    case 37:
    case 38:
        SLSSpecificConfig();
        break;
…
    }
…
}		

In Part 3: Audio, Subpart 1, in subclause 1.6.2.1 add the following subclause:

1.6.2.1.13 SLSSpecificConfig

Defined in ISO/IEC 14496-3 subpart 12.

In Part 3: Audio, Subpart 1, in subclause 1.6.2.2.1 Overview, add the following to table 1.14:

Audio Object Type	Object Type ID	Definition of elementary stream payloads and detailed syntax	Mapping of audio payloads to access units and elementary streams
…			
SLS	37	ISO/IEC 14496-3 subpart 12	
SLS non_core	38	ISO/IEC 14496-3 subpart 12	

Create Part 3: Audio, Subpart 12:

Subpart 12: Technical description of scalable lossless coding

12.1 Scope

This subpart of ISO/IEC 14496-3 describes the MPEG-4 scalable lossless coding algorithm for audio signals. This description partially relies on the specification as given in subpart 4.

12.2 Terms and definitions

12.2.1 Definitions

The following definitions are used in this subpart.

Core Layer	The MPEG-4 GA T/F coder used as the first layer in SLS . The audio object types AAC LC, AAC Scalable (without LTP), ER AAC LC, ER AAC Scalable and ER BSAC are supported.
LLE Layer	Lossless enhancement layer used in SLS to enhance the quality of the core layer towards lossless coding.
Bit-Plane	Position of specific bit in binary data word, starting with 0 as the position of the least significant bit (LSB). For example, the binary bit-plane symbols from bit-plane 0, 1, 2, and 3 of data word 0x0011 1101 (0x3d) are 1, 0, 1, and 1 respectively.
BPGC	Bit-Plane Golomb Code
CBAC	Context Based Arithmetic Code
LEMC	Low Energy Mode Code
Implicit Band	A scale factor band for which the quantized spectral data presented in the core layer bit-stream will be used in determining part of the necessary side information for the LLE layer.
Explicit Band	A scale factor band for which the quantized spectral data presented in the core layer bit-stream will not be used in determining the necessary side information for the LLE layer. All the side information will be coded explicitly in the LLE payload.
Oversampling Factor (osf)	Ratio between sampling rates of LLE Layer and Core Layer, possible values are 1, 2 and 4.
Oversampling Range	High frequency range covered only by the LLE Layer, comprises (osf-1)*1024 resp. (osf-1)*128 frequency values per window.
Reserved	All fields labelled *Reserved* are reserved for future standardization. All Reserved fields must be set to zero.

12.2.2 Notations

In order to make the description stringent, the following notations are used in this subpart:

- 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 italics, e.g. *variable*.
- Functions are indicated as *func*(*x*)

12.2.3 Definitions

DIV(m,n) Integer division with truncation of the result of m/n to an integer value towards $-\infty$.

$\lfloor \bullet \rfloor$ The floor operation. Returns the largest integer that is less than or equal to the real-valued argument.

12.3 Payloads for the audio object

Table 12.1 – Syntax of SLSSpecificConfig

Syntax	No. of bits	Mnemonics
SLSSpecificConfig(samplingFrequencyIndex,		
channelConfiguration,		
audioObjectType)		
{		
pcmWordLength;	3	uimsbf
aac_core_present;	1	uimsbf
lle_main_stream;	1	uimsbf
reserved_bit;	1	uimsbf
frameLength;	3	uimsbf
if (!channelConfiguration){		
program_config_element();		
}		
}		

Table 12.2 – Top layer payload for lle stream

Syntax	No. of bits	Mnemonics
lle_element()		
{		
for (ch=0;ch<channel_number;) {		
if (is_channel_pair(ch)) {		
lle_channel_pair_element();		
ch += 2;		
} else {		
lle_single channel_element();		
ch++;		
}		
}		
}		

Table 12.3 – Syntax of lle_single_channel_element

Syntax	No. of bits	Mnemonics
lle_single_channel_element()		
{		
lle_individual_channel_stream(1);		
}		

Table 12.4 – Syntax of lle_channel_pair_element

Syntax	No. of bits	Mnemonics
lle_channel_pair_element()		
{		
lle_individual_channel_stream(1);		
lle_individual_channel_stream(0);		
}		

Table 12.5 – Syntax of lle_individual_channel_stream

Syntax	No. of bits	Mnemonics
lle_individual_channel_stream(is_first_channel)		
{		
lle_ics_length;	16	uimsbf
if (is_first_channel) {		
element_instance_tag;	4	uimsbf
}		
lle_reserved_bit;	1	uimsbf
if (lle_main_stream) {		
lle_header(is_first_channel);		
lle_side_info();		
}		
lle_data();		
byte_align();		
}		

Table 12.6 – Syntax of lle_header()

Syntax	No. of bits	Mnemonics
lle_header(is_first_channel)		
{		
if (lle_channel_pair_element && common_window && is_first_channel) {		
use_stereo_intmdct;	1	uimsbf
}		
if (aac_core_present) {		
band_type_signaling;	2	uimsbf
if (**band_type_signaling**==1) {		
for(g=0;g<num_window_groups;g++) {		
for(sfb=0;sfb<max_sfb;sfb++) {		
band_type[g][sfb];	1	uimsbf
}		
}		
}		
} else {		
if (is_first channel) {		
windows_sequence;	2	uimsbf
}		
}		
}		

Table 12.7 – Syntax of lle_side_info

Syntax	No. of bits	Mnemonics
lle_side_info()		
{		
For(g=0;g<num_window_groups;g++) {		
for(sfb=0;sfb<num_sfb+num_osf_sfb;sfb++) {		
if (band_type[g][sfb]==Explicit_Band) {		
vcod_dpcm_max_bp[g][sfb];	1...17	bslbf
}		
if (max_bp[g][sfb] != -1) {		
vcod_lazy_bp[g][sfb];	1... 2	bslbf
}		
}		
}		
cb_cbac;	1	uimsbf
}		

Table 12.8 – Syntax of lle_data

Syntax	No. of bits	Mnemonics
lle_data()		
{		
BPGC/CBAC data;	varies	bslbf
LEMC data;	varies	bslbf
}		

12.4 Semantics

Data elements:

aac_core_present Indicates, whether the lossless enhancement operates on top of an MPEG-4 GA T/F core (*aac_core_present*=1) or in non-core mode (*aac_core_present*=0).

lle_main_stream Indicates, whether the current stream represents an LLE main stream including all the necessary side information or an LLE extension stream that extends the previous LLE stream.

pcmWordlength Quantization word length of the original PCM waveform.

Table 12.9 – Word length of original PCM waveform

pcmWordlength	Word length of original PCM waveform
0	8
1	16
2	20
3	24
4 – 7	Reserved

frameLength Length of the IntMDCT frame in the LLE layer.

Table 12.10 – Length of the IntMDCT frame

frameLength	Length of the IntMDCT frame	Oversampling factor of the IntMDCT filterbank (osf)
0	1024	1
1	2048	2
2	4096	4
3-7	Reserved	Reserved

element_instance_tag Unique instance tag for syntactic elements. All syntactic elements containing instance tags may occur more than once, but must have a unique element_instance_tag in each audio frame. When the MPEG-4 GA T/F core is present, syntactic elements of SLS and MPEG-4 GA T/F from the same audio channel use the same element_instance_tag.

lle_ics_length Length of LLE individual channel stream (LLE_ICS) for the current frame; in bytes.

band_type_signaling By default, the band type for a scale factor band is defined as follows: A scale factor band that is in a section coded with the zero codebook (ZERO_HCB), Intensity Stereo (IS) coded, or Perceptual Noise Substitution (PNS) coded is an Explicit_Band. Otherwise it is an Implicit_Band.

Scale factor bands above max_sfb and in the oversampling range are always Explicit_Band.

This default band type can by overwritten by band_type_signaling in the following way:

Table 12.11 – Band type signaling

Value of band_type_signaling	band type
00	Use default
01	Band type signaling for each sfb follows
10	All sfb are Explicit_Band
11	Reserved

band_type[g][sfb] Band type signaling for each scale factor band when band_type_signaling==01. A scale factor band is set to Explicit_Band if band_type[g][sfb] is 0.

Table 12.12 –Band type

Value	Band type
0	Explicit_Band
1	Default

vcod_dpcm_max_bp[g][sfb] The variable length coded maximum bit-plane for scale factor band *sfb* and group *g*.

vcod_lazy_bp[g][sfb] The variable length coded lazy bit-plane for non-zero scale factor band *sfb* and group *g*.

cb_cbac Indication of frequency table that will be used in the LLE decoding process.

Table 12.13 – cb_cbac table

cb_cbac	Frequency table
0	BPGC
1	CBAC

bpgc/cbac_data The binary bit-stream of the bpgc/cbac coded residual spectrum data

low_energy_mode_data The binary bit-stream of the LEMC mode coded residual spectrum data

12.5 SLS decoder tool

12.5.1 Overview

The block diagram of the scalable lossless (SLS) decoder is given in Figure 12.1. The core layer MPEG-4 GA stream is decoded by a deterministic Core Layer decoder. Its output, which is a deterministic spectrum in the MDCT domain, is sent to the inverse error mapping process. Meanwhile, the residual IntMDCT spectrum, carried in the LLE layer streams, is decoded and sent to the inverse error mapping process to reconstruct the IntMDCT spectrum. An inverse integer Mid/Side (M/S) and an inverse integer TNS process are then invoked and performed on the IntMDCT coefficients if necessary. Finally, its output is inversely transformed by using the inverse IntMDCT process to produce the PCM audio samples. A detailed description of each process is given in the subsequent sections.

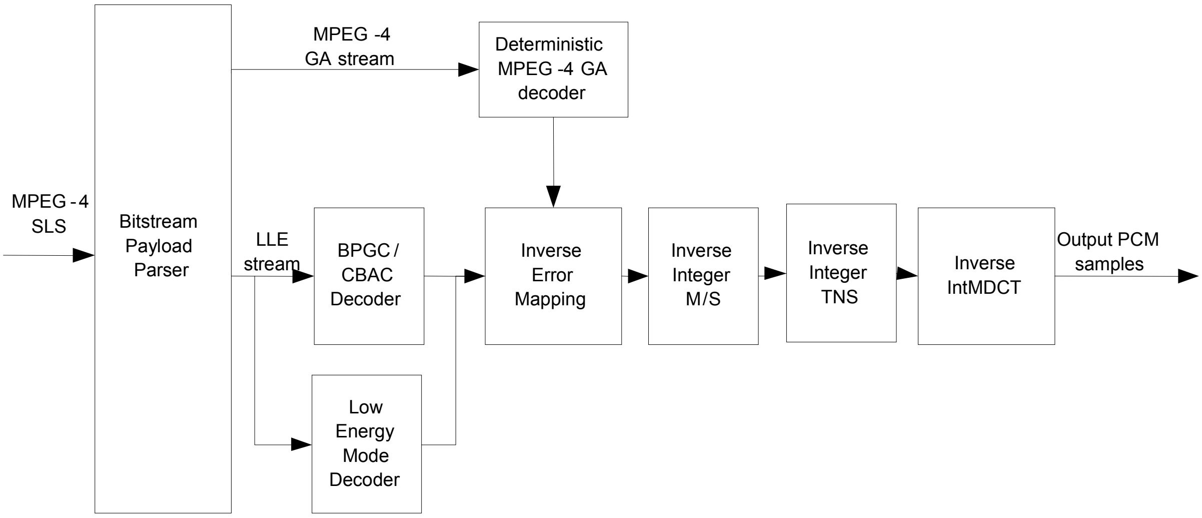

Figure 12.1 – SLS decoder block diagram

12.5.1.1 Non-core Mode

In the non-core mode SLS works as a stand-alone codec without AAC core. In case of the SLS audio object type this is signalled by **aac_core_present=0** for the non-core mode and **aac_core_present=1** for the core-based mode. In case of the SLS non-core audio object type it is always **aac_core_present=0.**

In the non-core mode the following default values are used:

- window_shape = 0 (sine window)
- if (lle_channel_pair_element) common_window = 1 (on)
- if (use_stereo_intmdct) all M/S flags are on, else all M/S flags are off
- if (window_sequence == EIGHT_SHORT_SEQUENCE) grouping = {2,2,2,2}

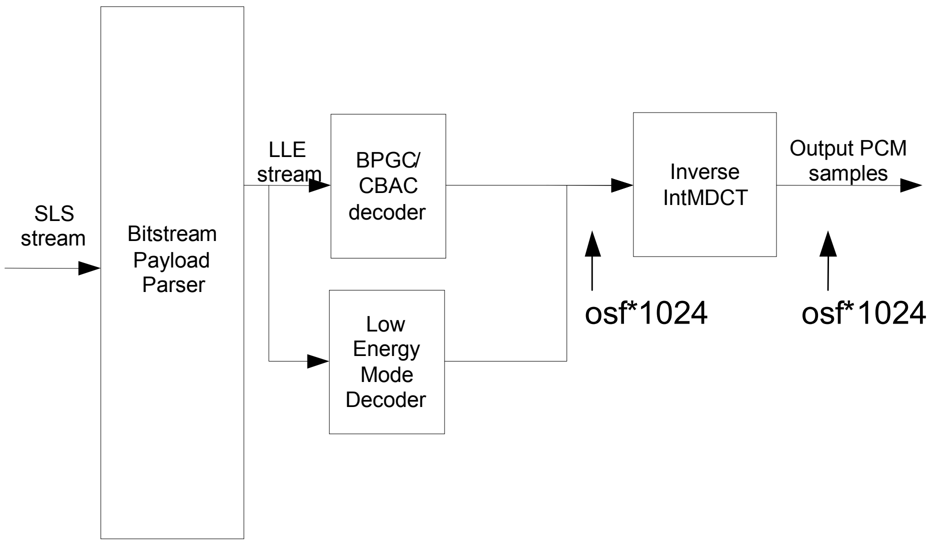

Figure 12.2 – SLS non-core decoder block diagram

12.5.2 Oversampling technique

The core layer is allowed to operate at a lower sampling rate than the LLE layers. The following table shows some possible sampling rate combinations.

Table 12.14 – Example combinations of sampling rates for Core and LLE layers

	Core@ 48 kHz	Core@ 96 kHz	Core@ 192 kHz
LLE@ 48 kHz	X (osf = 1)		
LLE@ 96 kHz	X (osf = 2)	X (osf = 1)	
LLE@ 192 kHz	X (osf = 4)	X (osf = 2)	X (osf = 1)

This technique is referred to as “Oversampling” in the following.

The scalability of the codec using different sampling rates is achieved by changing the length of the inverse IntMDCT in the decoder accordingly. While the AAC core processes 1024 values in each frame, the SLS codec needs to process osf*1024 values per frame. This is achieved by extending the length of the inverse IntMDCT in the decoder to osf*1024 spectral lines. The 1024 inverse quantized spectral values from the AAC core are added to the 1024 low-frequency values of the SLS residual spectrum. This is illustrated in **Figure 12.3**.

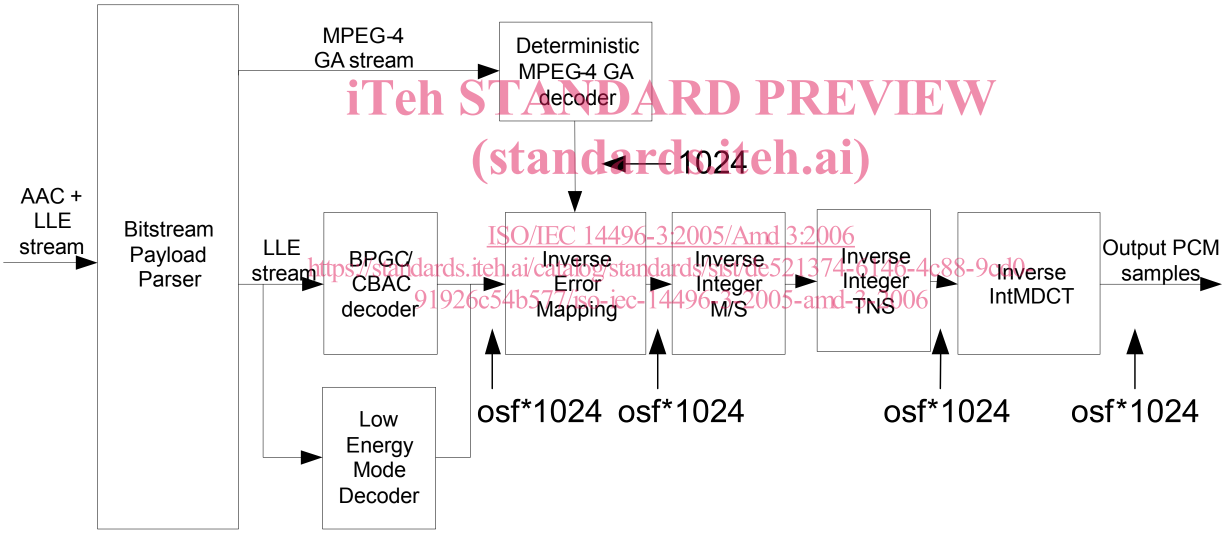

Figure 12.3 – Structure of SLS decoder with oversampling

12.5.3 SLS with Scalable AAC Core

If the core layer is AAC Scalable, the spectral data decoded from the SLS layers are added to the spectral data decoded from the AAC Scalable streams with a deterministic inverse AAC quantizer. The resulting spectral data is then processed with inverse integer M/S and inverse integer TNS process if necessary. Finally, the output is transformed by the inverse IntMDCT to produce the PCM audio samples. The decoding process is illustrated in Figure 12.4.

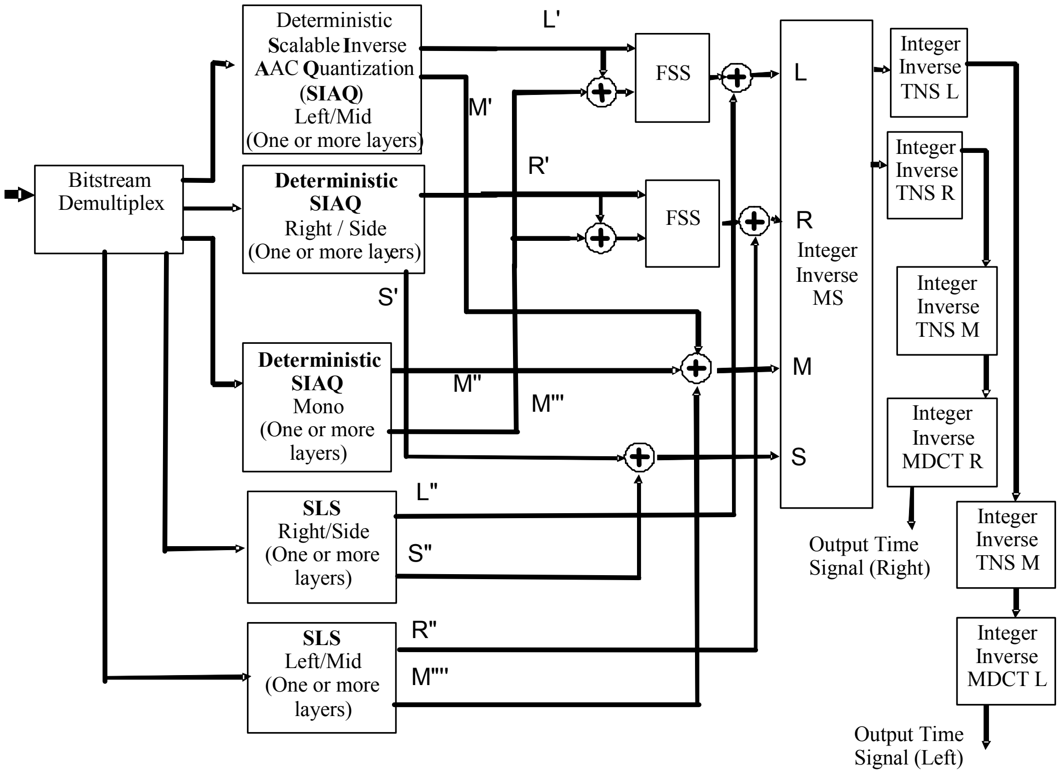

Figure 12.4 – Structure of SLS decoder with Scalable AAC core layer streams

12.5.4 Decoding of lle_single_channel_element (LLE_SCE) and lle_channel_pair_element (LLE_CPE)

12.5.4.1 Definitions

lle_ics_length Length of LLE individual channel stream (LLE_ICS) in bytes.

vcod_dpcm_max_bp[g][sfb] The variable length coded maximum bit-plane for scale factor band *sfb* and group *g***.** This element is only present for insignificant scale factor bands.

vcod_lazy_bp[g][sfb] The variable length coded lazy bit-plane for non-zero scale factor band *sfb* and group *g***.**

g Group index.

sfb Scale factor band within group.

win Window index.

bin Frequency bin index.

num_window_groups Number of groups of windows which share one set of scale factors.