ETSI TS 102 114 V1.2.1 (2002-12)

Technical Specification

DTS Coherent Acoustics;
Core and Extensions

European Broadcasting Union

Union Européenne de Radio-Télévision

EBU·UER

ETSI

Reference
RTS/JTC-DTS-R1

Keywords
acoustic, audio, codec, coding, digital

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Important notice

Individual copies of the present document can be downloaded from:
http://www.etsi.org

The present document may be made available in more than one electronic version or in print. In any case of existing or perceived difference in contents between such versions, the reference version is the Portable Document Format (PDF). In case of dispute, the reference shall be the printing on ETSI printers of the PDF version kept on a specific network drive within ETSI Secretariat.

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at
http://portal.etsi.org/tb/status/status.asp

If you find errors in the present document, send your comment to:
editor@etsi.org

Copyright Notification

No part may be reproduced except as authorized by written permission.
The copyright and the foregoing restriction extend to reproduction in all media.

© European Telecommunications Standards Institute 2002.
© European Broadcasting Union 2002.
All rights reserved.

DECTTM, **PLUGTESTS**TM and **UMTS**TM are Trade Marks of ETSI registered for the benefit of its Members.
TIPHONTM and the **TIPHON logo** are Trade Marks currently being registered by ETSI for the benefit of its Members.
3GPPTM is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners.

Contents

Intellectual Property Rights 5
Foreword 5
1 Scope 6
2 References 6
3 Definitions and abbreviations 6
3.1 Definitions 6
3.2 Abbreviations 6
4 Summary 7
5 Core Audio 7
5.1 Frame structure and decoding procedure 8
5.2 Error classification 9
5.3 Synchronization 10
5.4 Frame header 10
5.4.1 Bit stream header 10
6 Extension to more than 5.1 channels (XCh) 18
6.1 Synchronization 18
6.2 Frame header 18
7 Extension to sampling frequencies of up to 96 kHz and/or higher resolution (X96k) 18
7.1 DTS Core+96 kHz-Extension encoder 19
7.2 DTS Core+96 kHz Extension decoder 20
7.3 Synchronization 21
7.4 X96k frame header 22
Annex A (informative): Bibliography 23
Annex B (normative): Example Pseudocode 24
B.1 Overview of main function calls 24
B.2 Unpack Frame Header Routine 25
B.3 Audio Decoding 27
B.3.1 Primary Audio Coding Header 27
B.3.2 Unpack Subframes 31
B.3.2.1 Primary Audio Coding Side Information 31
B.3.3 Primary Audio Data Arrays 35
B.3.4 Unpack Optional Information 37
Annex C (normative): Decoding Algorithms 39
C.1 Block Code 39
C.2 CRC Error Detection 40
C.3 Inverse ADPCM 41
C.4 Joint Subband Coding 41
C.5 Sum/Difference Decoding 41
C.6 Filter Bank Reconstruction 42
C.7 Interpolation of LFE Channel 43
Annex D (normative): Large Tables 44
D.1 Scale Factor Quantization Tables 44

D.1.1 6-bit Quantization (Nominal 2,2 dB Step)....44
D.1.2 7-bit Quantization (Nominal 1,1 dB Step)....45

D.2 Quantization Step Size....47
D.2.1 Lossy Quantization....47
D.2.2 Lossless Quantization....48

D.3 Scale Factor for Joint Intensity Coding....48

D.4 Dynamic Range Control....50

D.5 Huffman Code Books....54
D.5.1 3 Levels....54
D.5.2 4 Levels (For TMODE)....54
D.5.3 5 Levels....54
D.5.4 7 Levels....55
D.5.5 9 Levels....56
D.5.6 12 Levels (for BHUFF)....56
D.5.7 13 Levels....58
D.5.8 17 Levels....59
D.5.9 25 Levels....61
D.5.10 33 Levels....65
D.5.11 65 Levels....72
D.5.12 129 Levels....79

D.6 Block Code Books....104
D.6.1 3 Levels....104
D.6.2 5 Levels....104
D.6.3 7 Levels....105
D.6.4 9 Levels....106
D.6.5 13 Levels....107
D.6.6 17 Levels....108
D.6.7 25 Levels....110

D.7 Interpolation FIR....112
D.7.1 2 x Interpolation....112
D.7.2 4 x Interpolation....113

D.8 32-Band Interpolation FIR....114
D.8.1 Perfect Reconstruction....114
D.8.2 Non-Perfect Reconstruction....119

D.9 LFE Interpolation FIR....124
D.9.1 64 x Interpolation....124
D.9.2 128 x Interpolation....129

D.10 VQ Tables....135
D.10.1 ADPCM Coefficients....135

History....139

Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (http://webapp.etsi.org/IPR/home.asp).

All published ETSI deliverables shall include information which directs the reader to the above source of information.

Foreword

This Technical Specification (TS) has been produced by Joint Technical Committee (JTC) Broadcast of the European Broadcasting Union (EBU), Comité Européen de Normalisation ELECtrotechnique (CENELEC) and the European Telecommunications Standards Institute (ETSI).

NOTE: The EBU/ETSI JTC Broadcast was established in 1990 to co-ordinate the drafting of standards in the specific field of broadcasting and related fields. Since 1995 the JTC Broadcast became a tripartite body by including in the Memorandum of Understanding also CENELEC, which is responsible for the standardization of radio and television receivers. The EBU is a professional association of broadcasting organizations whose work includes the co-ordination of its members' activities in the technical, legal, programme-making and programme-exchange domains. The EBU has active members in about 60 countries in the European broadcasting area; its headquarters is in Geneva.

European Broadcasting Union
CH-1218 GRAND SACONNEX (Geneva)
Switzerland
Tel: +41 22 717 21 11
Fax: +41 22 717 24 81

1 Scope

The present document describes the key components of the DTS Coherent Acoustics technology. The document also includes the lists of all frame header parameters in the DTS core and extension (XCh and X96k) streams. The information about the remaining parameters of the DTS bit streams is considered confidential and DTS proprietary and as such it is not described in the present document.

2 References

Void.

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the following terms and definitions apply:

DTS Core Audio Stream: which carries the coding parameters of up to 5.1 channels of the original LPCM audio at up to 24 bits per sample with the sampling frequency of up to 48 kHz

DTS Extended Audio Stream: which delivers possible extended frequency bands of the primary audio channels as well as all frequency components of channels beyond 5.1

NOTE: The extended audio stream must always have the accompanying core stream.

DTS XCh Stream: one of DTS extended streams that carries the coding parameters obtained from encoding of up to 2 additional channels of original LPCM audio at up to 24 bits per sample with the sampling frequency of up to 48 kHz

DTS X96k Stream: DTS extended audio stream that enables encoding of original LPCM audio at up to 24 bits per sample with the sampling frequency of up to 96 kHz

NOTE: The stream carries the coding parameters used for the representation of all remaining audio components that are present in the original LPCM audio and are not represented in the core audio stream.

Linear Pulse Code Modulated (LPCM): sequence of digital audio samples

QMF bank: specific filtering structure that provides the means of translating the time domain signal into the multiple sub-band domain signals

vector quantization: joint quantization of a block of signal samples or a block of signal parameters

3.2 Abbreviations

For the purposes of the present document, the following abbreviations apply:

DTS	Digital Theater Systems
LFE	Low Frequency Effect channel
LPCM	Linear Pulse Code Modulation
QMF	Quadrature Mirror Filter
VQ	Vector Quantization

4 Summary

DTS Coherent Acoustics is designed to deliver digital audio reproduction in the home at studio quality level in terms of fidelity and sound stage imagery. Specifically, it delivers up to eight discrete channels of multiplexed audio at sampling frequencies of 8 kHz to 192 kHz at bit rates of 32 kbit/s to 6 144 kbit/s. The encoding algorithm works at 24 bits per sample and can deliver compression rate of 3:1 up to 40:1.

Due to the popularity of the 5.1 channel sound tracks in the movie industry and in the emerging multichannel home audio market, DTS Coherent Acoustics is delivered in the form of a core audio (for the 5.1 channels) plus optional extended audio (for the rest of the DTS Coherent Acoustics). The 5.1 channel audio consists of up to five primary audio channels with frequencies lower than 24 kHz plus a possible low frequency effect (LFE) channel (the 0.1 channel). This implies that the frequency components higher than 24 kHz for the five primary audio channels and all frequency components of the remaining two channels are carried in the extended audio. This structure is illustrated in table 4.1 and as follows:

- Core Audio:
    - Up to 5 primary audio channels (frequency components below 24 kHz).
    - Up to 1 low frequency effect (LFE) channel.
    - Optional information such as time stamps and user information.
- Extended Audio:
    - Up to 2 additional full bandwidth channels (frequency components below 24 kHz).
    - Frequency components above 24 kHz for the primary and extended audio channels.

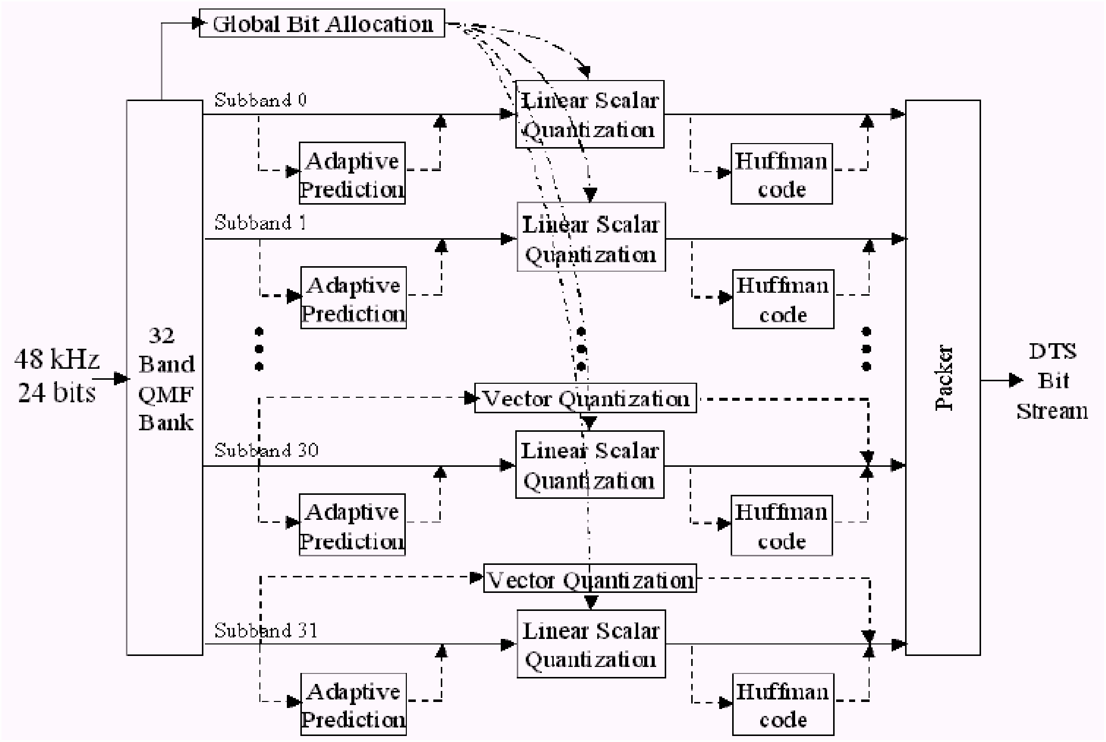
Under this structure, a basic DTS decoder can decode 5.1 channel core audio bits only and does not need to know even the existence of extended audio bits in the bit stream. A sophisticated decoder, however, can first decode the 5.1 core audio bits and then proceed to decode the extended audio bits if they exist.

Table 4.1: DTS Coherent Acoustics is optimized for 5.1 channel applications, but is extensible to deliver 8 channels with sampling frequency up to 192 kHz

Core Audio			**Extended Audio**	
Primary Audio Channels (< 24 kHz)	Low Frequency Effect Channel	Optional Information	Primary and Extended Audio Channels (> 24 kHz)	Channel 7 and 8

5 Core Audio

DTS core encoder delivers 5.1 channel audio at 24 bits per sample with a sampling frequency of up to 48 kHz. As shown in figure 5.1, the audio samples of a primary channel are split and decimated by a 32-band QMF bank into 32 sub-bands. The samples of each sub-band goes through an adaptive prediction process to check if the resultant prediction gain is large enough to justify the overhead of transferring the coefficients of prediction filter. The prediction gain is obtained by comparing the variance of the prediction residual to that of the sub-band samples. If the prediction gain is big enough, the prediction residual is quantified using mid-tread scalar quantization and the prediction coefficients are Vector-Quantized (VQ). Otherwise, the sub-band samples themselves are quantized using mid-tread scalar quantization. In the case of low bit rate applications, the scalar quantization indexes of the residual or sub-band samples are further encoded using Huffman code. When the bit rate is low, Vector Quantization (VQ) may also be used to quantize samples of the high-frequency sub-bands for which the adaptive prediction is disabled. In very low bit rate applications, joint intensity coding and sum/difference coding may be employed to further improve audio quality. The optional LFE channel is compressed by: low-pass filtering, decimation and mid-tread scalar quantization.



NOTE: The dotted lines indicate optional operations and dash dot lines bit allocation control.

Figure 5.1: Compression of a primary audio channel

5.1 Frame structure and decoding procedure

DTS bit stream is a sequence of synchronized frames, each consisting of the following fields (see figure 5.2). A pseudocode overview of the main function calls is listed in clause B.1.

- **Synchronization Word:** Synchronize the decoder to the bit stream.
- **Frame Header:** Carries information about frame construction, encoder configuration, audio data arrangement, and various operational features. See clause B.2 for pseudocode examples illustrating unpacking of the Frame Header.
- **Sub-frames:** Carries core audio data for the 5.1 channels. Each frame may have up to 16 sub-frames. See clause B.3 for pseudocode examples illustrating the primary audio coding header routines.
- **Optional Information:** Carries auxiliary data such as time code, which is not intrinsic to the operation of the decoder but may be used for post processing routines.
- **Extended Audio:** Carries possible extended frequency bands of the primary audio channels as well as all frequency components of channels beyond 5.1.

Each sub-frame contains data for audio samples of the 5.1 channels covering a time duration of up to that of the sub-band analysis window and can be decoded entirely without reference to any other sub-frames. A sub-frame consists of the following fields (see figure 5.3):

- **Side Information:** Relays information about how to decode the 5.1 channel audio data. Information for joint intensity coding is also included here.
- **High Frequency VQ:** Some and a small number of high frequency sub-bands of the primary channels may be encoded using VQ. In this case, the samples of each of those sub-bands within the sub-frame are encoded as a single VQ address.
- **Low Frequency Effect Channel:** The decimated samples of the LFE channel are carried as 8-bit words.

- **Sub-sub-frames:** All sub-bands, except those high-frequency VQ encoded ones, are encoded here in up to 4 sub-sub-frames.

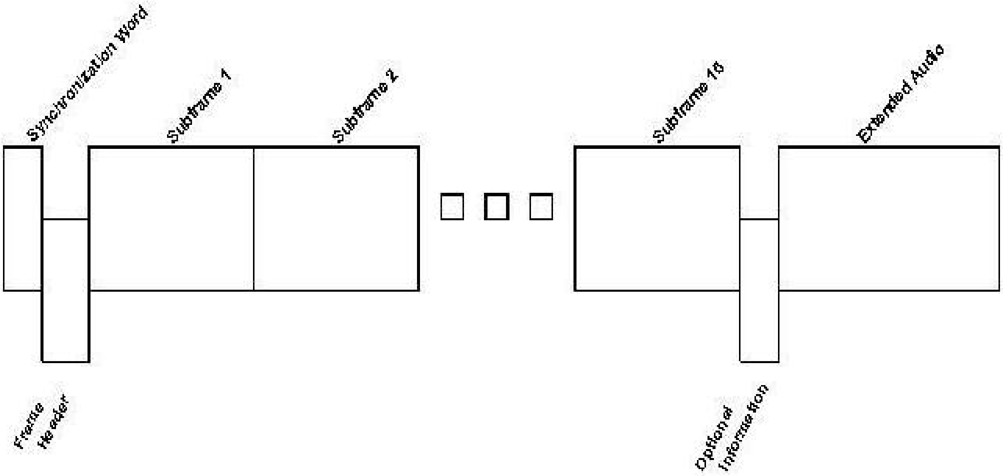


Figure 5.2: DTS frame structure

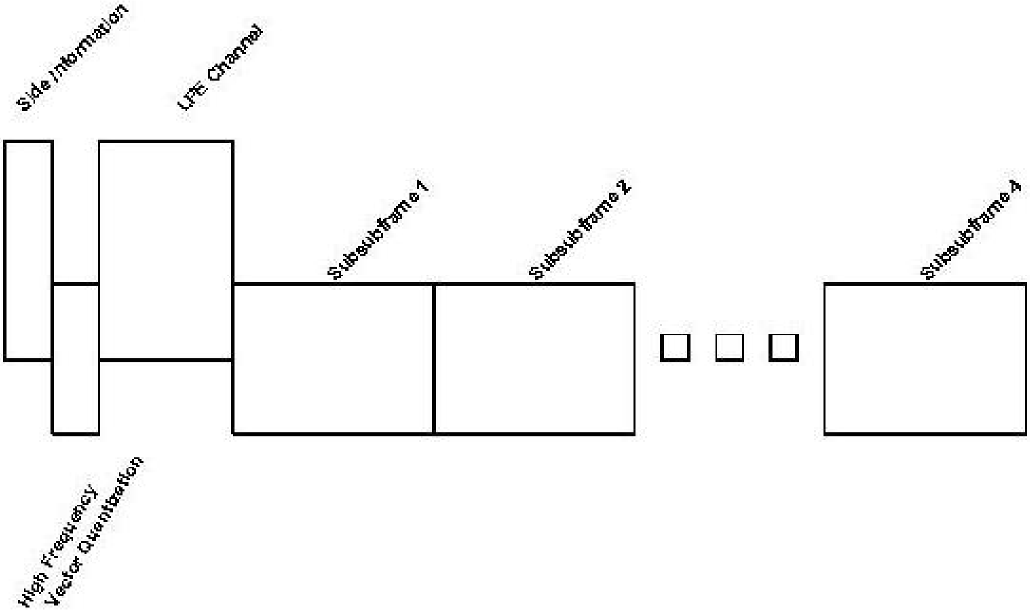


Figure 5.3: Sub-frame structure

5.2 Error classification

Each element in the bit stream carries either a piece of the audio data or the information to decode them. A corrupted bit stream element will cause an error in the decoder and its consequences depend on the information that element carries. In order to control decoded audio quality, the consequence of a corrupted element is categorized as:

V Vital: The element is designed to change from frame to frame and its corruption is likely to lead to failure in the decoding process and instability in decoded PCM outputs.

ACC Corruption could cause failure. Since the element usually does not change from frame to frame, the error may be compensated for by a majority vote over consecutive frames.

NV Non-vital: corruption will degrade the quality of PCM outputs, but the degradation will be graceful.

5.3 Synchronization

DTS bit stream consists of a sequence of audio frames of equal size, each begins with a 32-bit synchronization word:

SYNC = 0x7ffe8001 **V** **32 bits**

So the first decoding step is to search the input bit stream for SYNC. In order to reduce the probability of false synchronization, 6 bits after SYNC in the bit stream may be further checked, since they usually do not change for normal frames (they do carry useful information about frame structure). These 6 bits should be 0x3f (the binary 111111) for normal frames and are called synchronization word extension. Concatenating them with SYNC gives an extended synchronization word (32 + 6 = 38 bits):

SYNC = 0x7ffe8001 + 0x3f for normal frame **V** **38 bits**

which reduces the probability of false synchronization to 10^{-7}. In addition, the fact that SYNC occurs at a fixed interval further reduces the probability of false synchronization to almost zero.

The above search procedure shall be carried out only when the decoder is out of synchronization with the bit stream. After synchronization is established, the decoder checks only if the **SYNC** = 0x7ffe8001 before it begins to decode a frame, because the 6 bits after SYNC may change for abnormal (termination) frames.

The SYNC word appears at the beginning of each DTS data frame in the stream. The length of the DTS data frame is fixed for the entire DTS stream and consequently the SYNC words occur at the fixed intervals within the stream. During the initial synchronization process the decoder shall calculate the distance between the two consecutive SYNC words. While in synchronization with the incoming DTS stream, the decoder shall only look for the SYNC word of a new data frame at the calculated distance from the SYNC word of previously decoded data frame. If the SYNC word is found at the specified distance the decoder shall proceed with the decoding of the new data frame and if not the "out-of-sync" state shall be pronounced.

When DTS bit stream is stored in 16-bit words such as on CD, SYNC will be stored as 0x7ffe and 0x8001. However, when DTS bit stream is viewed on an IBM PC platform, since the high byte and low byte are switched, SYNC will appear like 0xfe7f and x0180.

Note that, in order to make the harsh sound less unpleasant when DTS bit stream is mistakenly played back as PCM format, DTS now provides a 14-bit format that reduces the dynamic range from 16 to 14 bits. In this 14-bit format, DTS bit stream is stored only in the least significant 14 bits of a 16-bit word, the most significant 2 bits are not used, In case of this, SYNC is stored in three words: 0x1fff, 0xe800, and 0x07f.

5.4 Frame header

The frame header consists of a bit stream header and a primary audio coding header. The bit stream header provides information about the construction of the frame, the encoder configuration such as core source sampling frequency, and various optional operational features such as embedded dynamic range control. The primary audio coding header specifies the packing arrangement and coding formats used at the encoder to assemble the audio coding side information. Many elements in the headers are repeated for each separate audio channel. For examples of pseudocode illustrating the unpacking of the frame header routine, see clause B.2.

5.4.1 Bit stream header

Frame Type **V** **FTYPE** **1 bit**

It indicates the type of current frame:

Table 5.1: Frame Type

FTYPE	Frame Type
1	Normal frame
0	Termination frame

Termination frames are used when it is necessary to accurately align the end of an audio sequence with a video frame end point. A termination block carries n×32 core audio samples where block length n is adjusted to just fall short of the video end point. Two termination frames may be transmitted sequentially to avoid transmitting one excessively small frame.

Deficit Sample Count V SHORT 5 bits

It defines the number of core samples by which a termination frame falls SHORT of the normal length of a block. A block = 32 PCM core samples per channel, corresponding to the number of PCM core samples that are feed to the core filter bank to generate one sub-band sample for each sub-band. A normal frame consists of blocks of 32 PCM core samples, while a termination frame provides the flexibility of having a frame size precision finer than the 32 PCM core sample block. On completion of a termination frame, (SHORT+1) PCM core samples must be padded to the output buffers of each channel. The padded samples may be zeros or they may be copies of adjacent samples.

Table 5.2: Deficit Sample Count

SHORT	Valid Value or Range of SHORT
1	[0,30]
0	31 (indicating a normal frame).

CRC Present Flag V CPF 1 bit

A flag that indicates if CRC (cyclic redundancy check) bits present in the bit stream.

Table 5.3: CRC Present Flag

CPF	CRC
1	Present
0	Not Present

Number of PCM Sample Blocks V NBLKS 7 bits

It indicates that there are (NBLKS+1) blocks (a block = 32 PCM core samples per channel, corresponding to the number of PCM samples that are fed to the core filter bank to generate one sub-band sample for each sub-band) in the current frame (see note). The actual core encoding window size is 32 × (NBLKS+1) PCM samples per channel. Valid range for NBLKS: 5 to 127. Invalid range for NBLKS: 0 to 4. For normal frames, this indicates a window size of either 2 048, 1 024, 512, or 256 samples per channel. For termination frames, NBLKS can take any value in its valid range.

NOTE: When frequency extension stream (X96k) is present, the PCM core samples represent the samples at the output of the decimator that precedes the core encoder. This k-times decimator translates the original PCM source samples with the sampling frequency of Fs_src= k×SFREQ to the core PCM samples (Fs_core=SFREQ) suitable for the encoding by the core encoder. The core encoder can handle sampling frequencies $SFREQ \leq 48$ kHz and consequently:

- $k=2$ for 48 kHz $<$ Fsrc $\leq$ 96 kHz; and
- $k=4$ for 96 kHz $<$ Fsrc $\leq$ 192 kHz

Primary Frame Byte Size V FSIZE 14 bits

(FSIZE+1) is the total byte size of the current frame including primary audio data as well as any extension audio data. Valid range for FSIZE: 95 to 16 383. Invalid range for FSIZE: 0 to 94.

Audio Channel Arrangement ACC AMODE 6 bits

Audio channel arrangement that describes the number of audio channels (CHS) and the audio playback arrangement (see table 5.4). Unspecified modes may be defined at a later date (user defined code) and the control data required to implement them, i.e. channel assignments, down mixing etc, can be uploaded from the player platform.

Table 5.4: Audio channel arrangement

AMODE	CHS	Arrangement
0b000000	1	A
0b000001	2	A + B (dual mono)
0b000010	2	L + R (stereo)
0b000011	2	(L+R) + (L-R) (sum-difference)
0b000100	2	LT +RT (left and right total)
0b000101	3	C + L + R
0b000110	3	L + R+ S
0b000111	4	C + L + R+ S
0b001000	4	L + R+ SL+SR
0b001001	5	C + L + R+ SL+SR
0b001010	6	CL + CR + L + R + SL + SR
0b001011	6	C + L + R+ LR + RR + OV
0b001100	6	CF+ CR+LF+ RF+LR + RR
0b001101	7	CL + C + CR + L + R + SL + SR
0b001110	8	CL + CR + L + R + SL1 + SL2+ SR1 + SR2
0b001111	8	CL + C+ CR + L + R + SL + S+ SR
0b010000 - 0b111111		User defined
NOTE: L = left, R = right, C = center, S = surround, F = front, R = rear, T = total, OV = overhead.		

Core Audio Sampling Frequency ACC SFREQ 4 bits

It specifies the sampling frequency of audio samples in the core encoder, based on table 5.5. When the source sampling frequency is beyond 48 kHz the audio is encoded in up to 3 separate frequency bands. The base-band audio, for example, 0 kHz to 16 kHz, 0 kHz to 22,05 kHz or 0 kHz to 24 kHz, is encoded and packed into the core audio data arrays. The SFREQ corresponds to the sampling frequency of the base-band audio. The audio above the base-band (the extended bands), for example, 16 kHz to 32kHz, 22,05 kHz to 44,1 kHz, 24 kHz to 48 kHz, is encoded and packed into the extended coding arrays which reside at the end of the core audio data arrays. If the decoder is unable to make use of the high sample rate data this information may be ignored and the base-band audio converted normally using a standard sampling rates (32 kHz, 44,1 kHz or 48 kHz). If the decoder is receiving data coded at sampling rates lower than that available from the system then interpolation (2× or 4×) will be required (see table 5.6).

Table 5.5: Core audio sampling frequencies

SFREQ	Core Audio Sampling Frequency
0b0000	Invalid
0b0001	8 kHz
0b0010	16 kHz
0b0011	32 kHz
0b0100	Invalid
0b0101	Invalid
0b0110	11,025 kHz
0b0111	22,05 kHz
0b1000	44,1 kHz
0b1001	Invalid
0b1010	Invalid
0b1011	12 kHz
0b1100	24 kHz
0b1101	48 kHz
0b1110	Invalid
0b1111	Invalid

Table 5.6: Sub-sampled audio decoding for standard sampling rates.

Core Audio Sampling Frequency	Hardware Sampling Frequency	Required Filtering
8 kHz	32 kHz	4 × Interpolation
16 kHz	32 kHz	2 × Interpolation
32 kHz	32 kHz	none
11 kHz	44,1 kHz	4 × Interpolation
22,05 kHz	44,1 kHz	2 × Interpolation
44,1 kHz	44,1 kHz	none
12 kHz	48 kHz	4 × Interpolation
24 kHz	48 kHz	2 × Interpolation
48 kHz	48 kHz	none

Transmission Bit Rate ACC RATE 5 bits

RATE specifies the targeted transmission data rate for the current frame of audio (see table 5.7). The open mode allows for bit rates not defined by the table. Variable and loss-less modes imply that the data rate changes from frame to frame.

Table 5.7: RATE parameter vs. targeted bit-rate

RATE	Targeted Bit Rate [kbit/s]
0b00000	32
0b00001	56
0b00010	64
0b00011	96
0b00100	112
0b00101	128
0b00110	192
0b00111	224
0b01000	256
0b01001	320
0b01010	384
0b01011	448
0b01100	512
0b01101	576
0b01110	640
0b01111	768
0b10000	960
0b10001	1 024
0b10010	1 152
0b10011	1 280
0b10100	1 344
0b10101	1 408
0b10110	1 411,2
0b10111	1 472
0b11000	1 536
0b11001	1 920
0b11010	2 048
0b11011	3 072
0b11100	3 840
0b11101	open
0b11110	Variable
0b11111	Loss-less

Due to the limitations of the transmission medium the actual bit rate may be slightly different from the targeted bit rate, as listed in table 5.8 for the two types of applications. The bit-rates that are not shown in the table 5.8 are not applicable on either of these two applications.

Table 5.8: Targeted and actual bit-rate for the CD and DVD-Video applications

RATE	Targeted Bit Rate [kbit/s]	Actual Bit Rate on DTS CDs [kbit/s]		Actual Bit Rate on DVD-Video Discs [kbit/s]
		14-bit format	16-bit format	
0b01111	768	N/A	N/A	754,50
0b10110	1 411,2	1 234,8	1 411,2	N/A
0b11000	1 536	N/A	N/A	1 509,75

Embedded Down Mix Enabled V MIX 1 bit

This indicates if embedded down mixing coefficients are included at the start of each sub-frame (see table 5.9). Down mixing to stereo may be implemented using these coefficients for the duration of the sub-frame.

Table 5.9: Status of embedded down mixing coefficients

MIX	Mix Parameters
0	not present
1	present

Embedded Dynamic Range Flag V DYNF 1 bit

DYNF indicates if embedded dynamic range coefficients are included at the start of each sub-frame. Dynamic range correction may be implemented on all channels using these coefficients for the duration of the sub-frame.

Table 5.10: Embedded Dynamic Range Flag

DYNF	Dynamic Range Coefficients
0	not present
1	present

Embedded Time Stamp Flag V TIMEF 1 bit

It indicates if embedded time stamps are included at the end of the core audio data.

Table 5.11: Embedded Time Stamp Flag

TIMEF	Time Stamps
0	not present
1	present

Auxiliary Data Flag V AUXF 1 bit

It indicates if auxiliary data bytes are appended at the end of the core audio data.

Table 5.12: Auxiliary Data Flag

AUXF	Auxiliary Data Bytes
0	not present
1	present

HDCD NV HDCD 1 bits

The source material is mastered in HDCD format if HDCD = 1, and otherwise HDCD = 0.

Extension Audio Descriptor Flag ACC EXT_AUDIO_ID 3 bits

This flag has meaning only if the EXT_AUDIO = 1 (see table 5.13) and then it indicates the type of data that has been placed in the extension stream(s).

Table 5.13: Extension Audio Descriptor Flag

EXT_AUDIO_ID	Type of Extension Data
0	Channel Extension (XCh)
1	Reserved
2	Frequency Extension (X96k)
3	XCh and X96k
4	Reserved
5	Reserved
6	Reserved
7	Reserved

Extended Coding Flag ACC EXT_AUDIO 1 bit

It indicates if extended audio coding data are present after the core audio data. Extended audio data will include the data for the extended bands of the 5 normal primary channels as well as all bands of additional audio channels. To simplify the process of implementing a 5.1 ch/48 kHz decoder, the extended coding data arrays are placed at the end of the core audio array.

Table 5.14: Extended Coding Flag

EXT_AUDIO	Extended Audio Data
0	not present
1	present

Audio Sync Word Insertion Flag ACC ASPF 1 bit

It indicates how often the audio data check word DSYNC (0xFFFF Extension Audio Descriptor Flag) occurs in the data stream. DSYNC is used as a simple means of detecting the presence of bit errors in the bit stream and is used as the final data verification stage prior to transmitting the reconstructed PCM words to the DACs.

Table 5.15: Audio Sync Word Insertion Flag

ASPF	DSYNC Placed at End of Each
0	Sub-frame
1	Sub-sub-frame

Low Frequency Effects Flag V LFF 2 bits

Indicates if the LFE channel is present and the choice of the interpolation factor to reconstruct the LFE channel (see table 5.16).

Table 5.16: Flag for LFE channel

LFF	LFE Channel	Interpolation Factor
0	not present	
1	Present	128
2	Present	64
3	Invalid	

Predictor History Flag Switch V HFLAG 1 bit

If frames are to be used as possible entry points into the data stream or as audio sequence\start frames the ADPCM predictor history may not be contiguous. Hence these frames can be coded without the previous frame predictor history, making audio ramp-up faster on entry. When generating ADPCM predictions for current frame, the decoder will use reconstruction history of the previous frame if HFLAG = 1. Otherwise, the history will be ignored.

Header CRC Check Bytes V HCRC 16 bits

This 16-bit CRC check word checks if there are errors from beginning of the current frame up to this point. It is present only if CPF = 1.

Multirate Interpolator Switch NV FILTS 1 bit

This flag indicates which set of 32-band interpolation FIR coefficients is to be used to reconstruct the sub-band audio (see table 5.17).

Table 5.17: Multirate interpolation filter bank switch

FILTS	32-band Interpolation Filter
0	Non-perfect Reconstruction
1	Perfect Reconstruction

Encoder Software Revision ACC/NV VERNUM 4 bits

It indicates of the revision status of the encoder software (see table 5.18). In addition the VERNUM is used to indicate the presence of the dialog normalization parameters (see table 5.22).

Table 5.18: Encoder software revision

VERNUM	Encoder Software Revision
0 to 6	Future revision (compatible with the present document)
7	Current
8 to 15	Future revision (incompatible with the present document)

NOTE: If the decoder encounters the DTS stream with the VERNUM>7 and the decoder is not designed for that specific encoder software revision than it must mute its outputs.

Copy History NV CHIST 2 bits

It indicates the copy history of the audio. Because of the copyright regulations, the exact definition of this field is deliberately omitted.

Source PCM Resolution ACC/NV PCMR 3 bits

It indicates the quantization resolution of source PCM samples (see table 5.19). The left and right surrounding channels of the source material are mastered in DTS ES format if ES = 1, and otherwise if ES = 0.

Table 5.19: Quantization resolution of source PCM samples

PCMR	Source PCM Resolution	ES
0b000	16 bits	0
0b001	16 bits	1
0b010	20 bits	0
0b011	20 bits	1
0b110	24 bits	0
0b101	24 bits	1
Others	Invalid	invalid

Front Sum/Difference Flag V SUMF 1 bit

Indicates if front left and right channels are sum-difference encoded prior to encoding (see table 5.20). If set to zero no decoding post processing is required at the decoder.

Table 5.20: Sum/difference decoding status of front left and right channels

SUMF	Front Sum/Difference Encoding
0	L = L, R = R
1	L = L + R, R = L - R

Surrounds Sum/Difference Flag V SUMS 1 bit

Indicates if left and right surround channels are sum-difference encoded prior to encoding (see table 5.21). If set to zero no decoding post processing is required at the decoder.

Table 5.21: Sum/difference decoding status of left and right surround channels.

SUMS	Surround Sum/Difference Encoding
0	Ls = Ls, Rs = Rs
1	Ls = Ls + Rs, Rs = Ls - Rs

Dialog Normalisation Parameter/Unspecified V DIALNORM/UNSPEC 4 bits

For the values of VERNUM = 6 or 7 this 4-bit field is used to determine the dialog normalization parameter. For all other values of the VERNUM this field is a place holder that is not specified at this time.

The dialog normalization gain (DNG), in dB, is specified by the encoder operator and is used to directly scale the decoder outputs samples. In the DTS stream the information about the DNG value is transmitted by means of combined data in the VERNUM and DIALNORM fields (see table 5.22).

For all other values of the VERNUM (i.e. 0, 1, 2, 3, 4, 5, 8, 9, …15) the UNSPEC 4-bit field should be extracted but ignored by the decoder. In addition, for these VERNUM values, the dialog normalization gain should be set to 0 i.e. DNG=0 -> No Dialog Normalisation.

Table 5.22: Dialog Normalization Parameter

Dialog Normalization Gain (DNG) Applied to the Decoder Outputs [dB]	VERNUM	DIALNORM
0	7	0b0000
-1	7	0b0001
-2	7	0b0010
-3	7	0b0011
-4	7	0b0100
-5	7	0b0101
-6	7	0b0110
-7	7	0b0111
-8	7	0b1000
-9	7	0b1001
-10	7	0b1010
-11	7	0b1011
-12	7	0b1100
-13	7	0b1101
-14	7	0b1110
-15	7	0b1111
-16	6	0b0000
-17	6	0b0001
-18	6	0b0010
-19	6	0b0011
-20	6	0b0100
-21	6	0b0101
-22	6	0b0110
-23	6	0b0111
-24	6	0b1000
-25	6	0b1001
-26	6	0b1010
-27	6	0b1011
-28	6	0b1100
-29	6	0b1101
-30	6	0b1110
-31	6	0b1111

6 Extension to more than 5.1 channels (XCh)

When the need arises to encode more than 5.1 channels, the extended channels are compressed using exactly the same technology as the core audio channels. The audio data representing these extension channels are appended to the end of the DTS stream audio. These extension audio data are automatically ignored by the first generation DTS decoders but can be decoded by the second generation DTS decoders.

6.1 Synchronization

Channel Extension Sync Word	**V**	**XChSYNC**	**32 bits**

The synchronization word XChSYNC = 0x5a5a5a5a for the channel extension audio comes after all other extension streams i.e. in case of multiple extension streams the XCh stream is always the last. For 16-bit streams, XChSYNC is aligned to 32-bit word boundary. For 14-bit streams, it is aligned to both 32 bit and 28-bit word boundaries, meaning that, the sync word appears as 0x1696e5a5 in the 28-bit stream and as 0x5a5a5a5a after this stream is packed into a 32-bit stream.

Since the pseudo sync word might appear in the bit stream, it is MANDATORY to check the distance between this sync and the end of the encoded bit stream. This distance in bytes should be equal to XChFSIZE+1. The parameter XChFSIZE is described below.

NOTE: For compatibility reasons with legacy bit streams the estimated distance in bytes is checked against both the XChFSIZE+1 as well as the XChFSIZE. The XCh synchronization is pronounced only if the distance matches either of these two values.

6.2 Frame header

Primary Frame Byte Size	**V**	**XChFSIZE**	**10 bits**

(XChFSIZE+1) is the distance in bytes from current extension sync word to the end of the current audio frame. Valid range for XChFSIZE: 95 to 1 023. Invalid range for XChFSIZE: 0 to 94.

Extension Channel Arrangement	**ACC**	**AMODE**	**4 bits**

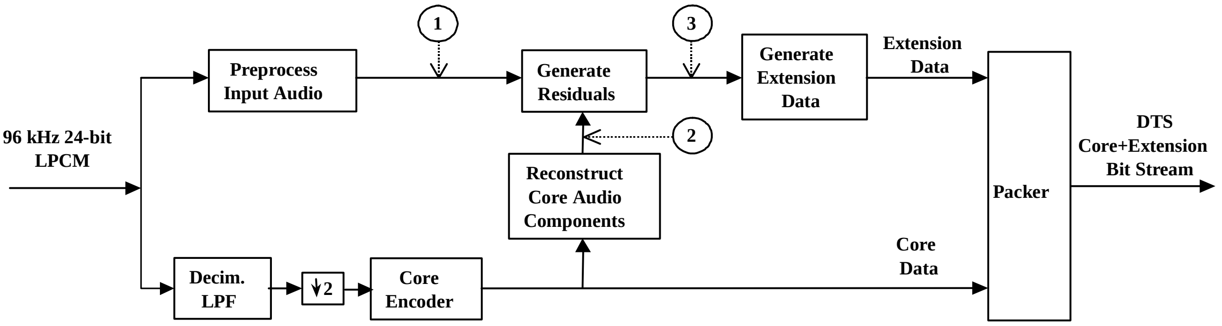
Audio channel arrangement that describes the number of audio channels (CHS) and the audio playback arrangement. It is set to represent the number of extension channels for now. More detail will be added in the future.

7 Extension to sampling frequencies of up to 96 kHz and/or higher resolution (X96k)

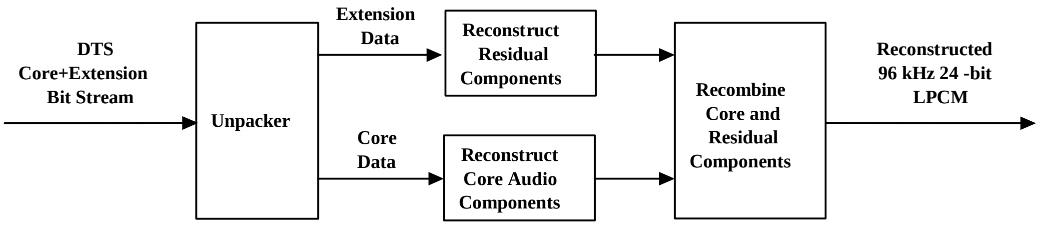
The generalized concept of core+96 kHz-extension coding is illustrated in figure 7.1. To encode 96 kHz LPCM the input audio stream is fed to a 96 kHz to 48 kHz down sampler and the resulting 48 kHz signal is encoded using standard core encoder as in figure 7.1 A). Referring to figure 7.1 A):

- In the "Preprocess Input Audio" block the original 96 kHz/24-bit LPCM audio is first delayed and next passed through the extension 64-band analysis filter bank. Signal "1" in this case consists of the extension sub-band samples @ 96 kHz/64.
- The core data consists of the core audio codes in 32 sub-bands and the side information. In the "Reconstruct Core Audio Components" block the core audio codes are inverse quantized to produce the reconstructed core sub-band samples @ 48 kHz/32. These sub-band samples correspond to signal "2".
- In the "Generate Residuals" block the reconstructed core sub-band samples are subtracted from the extension sub-band samples in the lower 32 sub-bands. The extension sub-band samples in the upper 32 bands remain unaltered. These residual sub-band samples in the 64 bands correspond to signal "3".
- The ("Generate Extension Data" block processes the residual sub-band samples and generates the extension data that, along with the core data, is assembled in a packer to produce a core+extension bit stream.

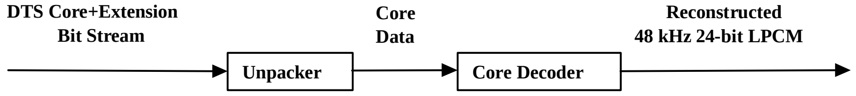
In the 96 kHz decoder, figure 7.1 B), the unpacker first separates the core+extension stream into the core and extension data. The core sub-band decoder, in the "Reconstruct Core Audio Components" block, processes the core data and produces the reconstructed core sub-band samples (same as signal "2" generated in the encoder). Next in the "Reconstruct Residual Components" block, the extension sub-band decoder uses the extension data to generate the reconstructed residual sub-band samples in the 64 bands. In the "Recombine Core and Residual Components" block the core sub-band samples are added to the lower 32 bands of residual sub-band samples to produce the extension sub-band samples in the 64 bands. In the same block the synthesis 64-band filter bank processes the extension sub-band samples and generates the 96 kHz 24-bit LPCM audio. The combining of reconstructed residuals and core signals on the decoder side, figure 7.1 B), is also done in sub-band domain.



A) Backward Compatible 96kHz Encoder



B) 96 kHz Decoder



C) 48 kHz (Legacy) Decoder

Figure 7.1: The concept of Core+Extension coding methodology

When a 48 kHz-only (legacy) decoder is fed the core+extension bit stream, figure 7.1 C), the extension data fields are ignored and only the core data is decoded. This results in 48 kHz core LPCM audio output.

7.1 DTS Core+96 kHz-Extension encoder

The block diagram in figure 7.2 shows the main components of the encoding algorithm. The input digital audio signal with a sampling frequency up to 96 kHz and a word length up to 24 bits is processed in the core branch and extension branch. In the core branch input audio is low-pass filtered to reduce its bandwidth to below 24 kHz, and then decimated by a factor of two, resulting in a 48 kHz sampled audio signal. The purpose of this LPF decimation is to remove signal components that cannot be represented by the core algorithm. The down sampled audio signal is processed in a 32-band analysis cosine modulated filter bank that produces the core sub-band samples. The core bit allocation routine based on the energy contained in each of the sub-bands and configuration of the core encoder determines the desired quantization scheme for each of the sub-bands. The core sub-band encoder performs quantization and encoding after which the audio codes and side information are delivered to the packer. The packer assembles this data into a core bit stream.

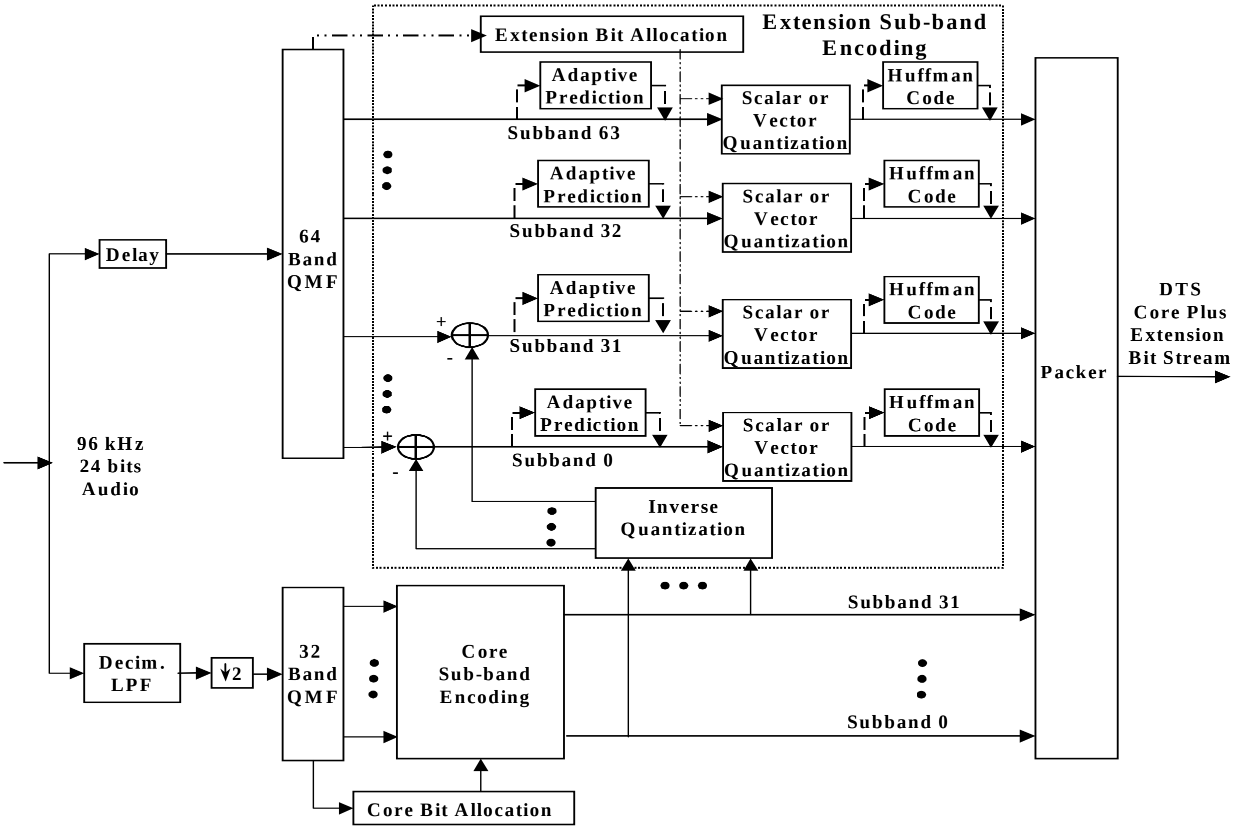


Figure 7.2: The block diagram of DTS Core+Extension encoder

In the extension branch the delayed version of input audio is processed in a 64-band analysis cosine modulated filter bank that produces the extension sub-band samples. Inverse quantization of the core audio codes produces the reconstructed core sub-band samples. Subtracting these samples from the extension sub-band samples in the lower 32 bands generates the residual sub-band samples. The residual signals in the upper 32 sub-bands are unaltered extension sub-band samples in corresponding bands. The delay of input audio is such that reconstructed core sub-band samples and extension sub-band samples in the lower 32 bands are time-aligned before the residual signals are produced i.e.:

$$\text{Delay} = \text{Delay}_{\text{DecimationLPF}} + \text{Delay}_{\text{CoreQMF}} - \text{Delay}_{\text{ExtensionQMF}}$$

The extension bit allocation routine based on the energy of residuals in each of the sub-bands and configuration of the extension encoder determines the desired quantization scheme for each of 64 sub-bands. The residual samples in sub-bands are encoded using a multitude of adaptive prediction, scalar/vector quantization and/or Huffman coding to produce the residual codes and extension side information. The packer assembles this data into an extension bit stream.

7.2 DTS Core+96 kHz Extension decoder

On the decoder side core and extension parts of the encoded bit stream are fed to their respective sub-band decoders. The reconstructed core sub-band samples are added to the corresponding residual sub-band samples in lower 32 bands. The reconstructed residual sub-band samples in the upper 32 bands remain unaltered. Passing the resulting extension sub-band samples through the synthesis 64-band QMF filter bank produces the 96 kHz sampled PCM audio. Figure 7.3 shows the block diagram of the core+extension decoder.

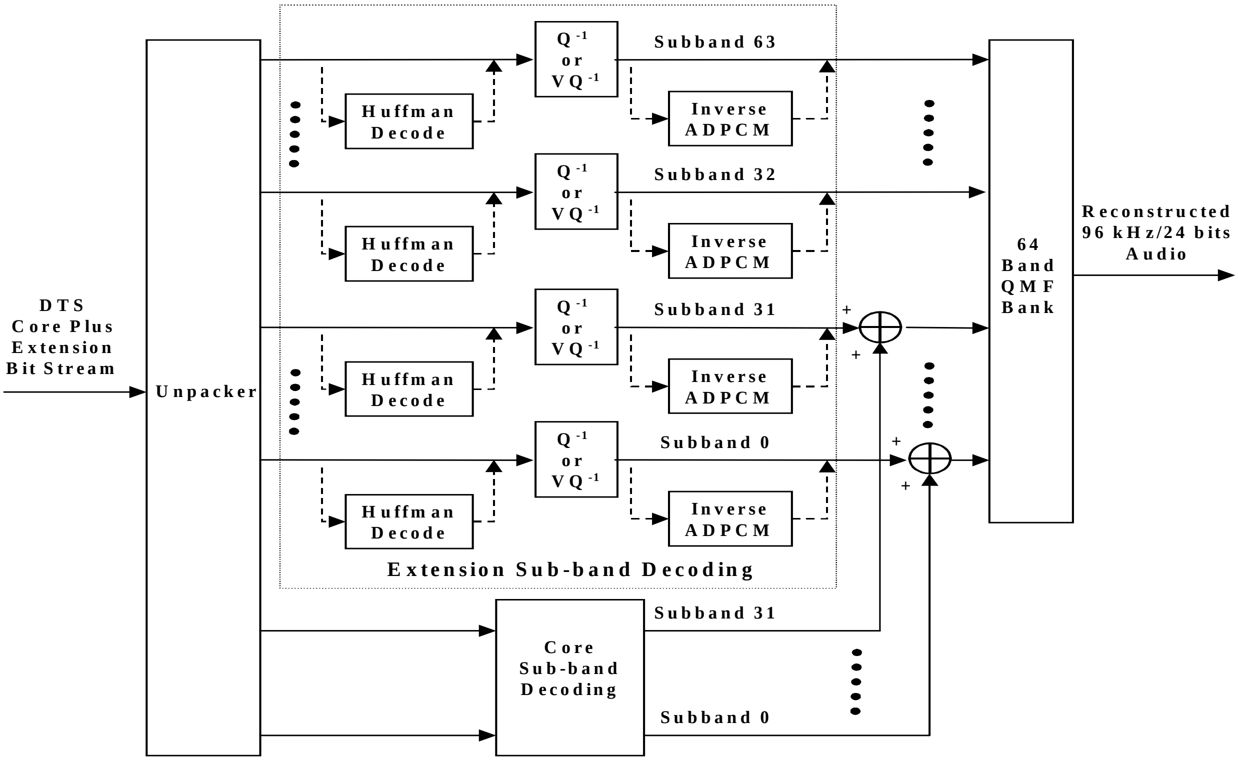


Figure 7.3: The block diagram of DTS Core+Extension decoder

In the case where the encoded bit stream does not contain the extension data, the decoder based on its hardware configuration uses:

a) a 32-band QMF with core sub-band samples as inputs to synthesize the 48 kHz sampled PCM audio;

b) a 64-band QMF with inputs being core sub-band samples in the lower 32 bands and "zero" samples in the upper 32 bands to synthesize the interpolated PCM audio sampled at 96 kHz.

The existing DTS core decoders when receiving the core+extension bit stream will extract and decode the core data to produce the 48 kHz sampled PCM audio. The decoder ignores the extension data by skipping the extraction until the next DTS synchronization word.

7.3 Synchronization

96 kHz Extension Sync Word SYNC96 V 32 bits

The synchronization word SYNC96 = 0x1D95F262 for the 96 kHz extension data comes after the core audio data. Note that if a channel extension is present the X96k extension data is placed before the XCh extension data in the encoded bit stream. For 16-bit streams the sync word is aligned to 32-bit word boundary. In the case of 14-bit streams SYNC96 is aligned to both 32-bit and 28-bit word boundaries meaning that 28 MSB-s of the SYNC96 appear as 0x07651F26.

To reduce the probability of false synchronization caused by the presence of pseudo sync words, it is imperative to check the distance between the detected sync word and the end of current frame (as indicated by FSIZE). This distance in bytes must match the value of FSIZE96 (see below).

After the decoder synchronization is established a flag nX96kPresent is set and the decoder output sampling frequency is selected as:

```
Pseudo Code:    OutSamplingFreq = SFREQ;
        if (nX96kPresent)
                OutSamplingFreq = 2× OutSamplingFreq;
```

Note that SFREQ corresponds to a sampling frequency of reconstructed audio in the core decoder.

7.4 X96k frame header

96 kHz Extension Frame Byte Data Size FSIZE96 V 12 bits

(FSIZE96 + 1) is the byte size of 96 kHz extension data plus any other extension data that appears in between FSIZE96 and the end of current frame. Valid range for FSIZE96: 95 to 4 095; Invalid range: 0 to 94.

Revision Number REVNO ACC/NV 4 bits

Revision number for the high frequency extension processing algorithm.

Table 7.1: X96k Algorithm Revision Number

REVNO	Frequency Extension Encoder Software Revision Number
0	Reserved
1	Current
2 to 7	Future revision (compatible with the original Rev1.0 specification)
8 to 15	Future revision (incompatible with the original Rev1.0 specification)

NOTE: If the decoder is not compatible with some algorithm revisions (REVNO>7) it must ignore the X96k extension stream and reconstruct the core encoded audio components up to 24/22,05 kHz.

Annex A (informative): Bibliography

Zoran Fejzo: "DTS Coherent Acoustics; Core and Extensions, Overview of Technology and Description of DTS Stream Frame Headers"

DTS, Inc. (5171 Clareton Drive Agoura Hills, CA 91301): "DTS Decoder Manual Rev2.1 and it’s Amendment Rev1.1"

Annex B (normative): Example Pseudocode

Scope

Annex B outlines in detail pseudocode examples to clarify the details of the main function calls, unpacking of the frame and primary audio coding headers.

B.1 Overview of main function calls

Based on this subframe structure, the procedure of decoding a subframe may be illustrated by the following pseudocode:

```
DecodeSubframe()
{
// Unpack Side Information.
UnpackSideInformation();
// Inverse VQ to extract high frequency subbands.
for (nChannel=0; nChannel<nNumPrimaryChannels; nChannel++) {
    for (nSubband=nHFreqVQBegin; nSubband<nHFreqVQEnd; nSubband++) {
        VQIndex = ExtractVQIndex();
        InverseVQ(VQIndex);     // One index looks up 32 samples
                                // in one subband analysis window.
    }
}
// Unpack the LFE channel
ExtractLFEDecimatedSamples();   // Extract the decimated samples.
InterpolateLFESamples();        // Interpolate for all LFE samples.
// Unpack subsubframes.
for (nSubsubframe=0; nSubsubframe<nNumOfSubsubframes; nSubsubframe++) {
    UnpackSubsubframe();
}
// Reconstruct all primary channels through filter bank interpolation
for (nChannel=0; nChannel<nNumPrimaryChannels; nChannel++) {
    ReconstructChannel();
    }
}
```

A subsubframe consists of eight subband samples (a subband analysis subwindow) for each subband of all primary channels, so its decoding procedure may be described as:

```
UnpackSubsubframe()
{
for (nChannel=0; nChannel<nNumPrimaryChannels; nChannel++) {
    for (nSubband=0; nSubband<nHFreqVQBegin; nSubband++) {
        UnpackOneSubwindow(); // Get 8 subband samples.
    }
}
```

}

An example of synchronization and decoding procedure may be described as follows:

```
START_SYNC:     InSyncFlag = 0;

    // Search for extend sync word (38-bit sync word + extension)

SearchForExtSync();

// Search for another sync word (32-bit sync word)

SearchForSync();

// Count the distance between the two sync words and check if it is within the

// limits. The next sync word is expected at this distance.

InSyncFlag = CountSyncDist();

    If (InSyncFlag==1)
    // Decode the received frame
        DecodeOneFrame();
    Else
        Goto START_SYNC;
// Decode the remaining frames
while (NotEndOfBitStream) {

        // Check if sync word occurred at the expected interval
    InSyncFlag = CheckSync();
    If (InSyncFlag==1)
        DecodeOneFrame();
    Else
        Goto START_SYNC;
}
```

B.2 Unpack Frame Header Routine

See clause 5.4 for a full description of the variables outlined in this clause.

Frame Type	**V**	**FTYPE**	**1 bit**

FTYPE may be extracted by the following pseudocode:

```
FTYPE = ExtractBits(1);
```

where ExtractBits(NumBits) is a general function which simply reads NumBits of bits from the input bit stream.

Deficit Sample Count	**V**	**SHORT**	**5 bits**

```
SHORT = ExtractBits(5);
```

CRC Present Flag	**V**	**CPF**	**1 bit**

```
    CPF = ExtractBits(1);
```

Number of PCM Sample Blocks	**V**	**NBLKS**	**7 bits**

```
    NBLKS = ExtractBits(7);
```

Primary Frame Byte Size	**V**	**FSIZE**	**14 bits**

```
FSIZE = ExtractBits(14);
```

Audio Channel Arrangement	**ACC**	**AMODE**	**6 bits**

```
AMODE = ExtractBits(6);
```

Core Audio Sampling Frequency	**ACC**	**SFREQ**	**4 bits**

```
    SFREQ = ExtractBits(4);
```

Transmission Bit Rate ACC RATE 5 bits

```
RATE = ExtractBits(5);
```

Embedded Down Mix Enabled V MIX 1 bit

```
MIX = ExtractBits(1);
```

Embedded Dynamic Range Flag V DYNF 1 bit

```
DYNF = ExtractBits(1);
```

Embedded Time Stamp Flag V TIMEF 1 bit

```
TIMEF = ExtractBits(1);
```

Auxiliary Data Flag V AUXF 1 bit

```
AUXF = ExtractBits(1);
```

HDCD NV HDCD 1 bits

```
HDCD = ExtractBits(1);
```

Extension Audio Descriptor Flag ACC EXT_AUDIO_ID 3 bits

```
EXT_AUDIO_ID = ExtractBits(3);
```

Extended Coding Flag ACC EXT_AUDIO 1 bit

```
EXT_AUDIO = ExtractBits(1);
```

Audio Sync Word Insertion Flag ACC ASPF 1 bit

```
ASPF = ExtractBits(1);
```

Low Frequency Effects Flag V LFF 2 bits

```
LFF = ExtractBits(2);
```

Predictor History Flag Switch V HFLAG 1 bit

```
HFLAG = ExtractBits(1);
```

Header CRC Check Bytes V HCRC 16 bits

if (CPF == 1) // Present only if CPF=1.

```
HCRC = ExtractBits(16);
```

Multirate Interpolator Switch NV FILTS 1 bit

```
FILTS = ExtractBits(1);
```

Encoder Software Revision ACC/NV VERNUM 4 bits

```
VERNUM = ExtractBits(4);
```

Copy History NV CHIST 2 bits

```
CHIST = ExtractBits(2);
```

Source PCM Resolution ACC/NV PCMR 3 bits

```
PCMR = ExtractBits(3);
```

Front Sum/Difference Flag V SUMF 1 bit

```
SUMF = ExtractBits(1);
```

Surrounds Sum/Difference Flag V SUMS 1 bit

```
    SUMS = ExtractBits(1);
```

Dialog Normalisation Parameter/Unspecified V DIALNORM/UNSPEC 4 bits

```
 switch (VERNUM){
 case 6:
    DIALNORM = ExtractBits(4);
    DNG = - (16+DIALNORM);
    break;
    case7:
    DIALNORM = ExtractBits(4);
    DNG = - DIALNORM;
    break;
    default:
    UNSPEC = ExtractBits(4);
    DNG = DIALNORM = 0;
    break;
    }
```

B.3 Audio Decoding

This clause outlines pseudocode rountines to illustrate Audio Decoding.

B.3.1 Primary Audio Coding Header

Number of Subframes V SUBFS 4 bits

It indicates that there are nSUBFS = SUBFS+1 audio subframes in the core audio frame. SUBFS is valid for all audio channels.

```
SUBFS = ExtractBits(4);
nSUBFS = SUBFS + 1;
```

Number of Primary Audio Channels V PCHS 3 bits

It indicates that there are nPCHS = PCHS+1 ≤ 5 primary audio channels in the current frame. If AMODE flag indicates more than 5 channels apart from LFE, the additional channels are the extended channels and are packed separately in the extended data arrays.

```
PCHS = ExtractBits(3);
nPCHS = PCHS + 1;
```

Subband Activity Count V SUBS 5 bits per channel

It indicates that there are nSUBS[ch] = SUBS[ch]+2 active subbands in the audio channel ch. Samples in subbands above nSUBS[ch] are zero, provided that intensity coding in that subband is disabled.

```
for (ch=0; ch<nPCHS; ch++) {
    SUBS[ch] = ExtractBits(5);
nSUBS[ch] = SUBS[ch] + 2;
    }
```

High Frequency VQ Start Subband V VQSUB 5 bits per channel

It indicates that high frequency samples starting from subband nVQSUB[ch]=VQSUB[ch]+1 re VQ encoded. High frequency VQ is used only for high frequency subbands, but it may go down o low frequency subbands for such audio episodes as silence. In case of insufficient MIPS, the VQs for the highest frequency subbands may be ignored without causing audible distortion.

```
for (ch=0; ch<nPCHS; ch++) {
VQSUB[ch] = ExtractBits(5);
nVQSUB[ch] = VQSUB[ch] + 1;
}
```

Joint Intensity Coding Index V JOINX 3 bits per channel

JOINX[ch] indicates if joint intensity coding is enabled for channel ch and which audio channel is the source channel from which channel ch will copy subband samples (see table B.1). It is assumed that the source channel index is smaller than that of the intensity channel.

Table B.1: Joint subband coding status and source channels

JOINX[ch]	Joint Intensity	Source Channel
0	Disabled	
> 0	Enabled	JOINX[ch]

```
for (ch=0; ch<nPCHS; ch++) {
    JOINX[ch] = ExtractBits(3);
}
```

Transient Mode Code Book V THUFF 2 bits per channel

It indicates which Huffman codebook was used to encode the transient mode data (see table B.2).

Table B.2: Selection of Huffman codebook for encoding the transient mode data TMODE

THUFF[ch]	Huffman Codebook
0	A4
1	B4
2	C4
3	D4

```
for (ch=0; ch<nPCHS; ch++) {
    THUFF[ch] = ExtractBits(2);
}
```

Scale Factor Code Book V SHUFF 3 bits per channel

The scale factors of a channel are quantized nonlinearly using either a 6-bit (64-level, 2,2 dB per step) or a 7-bit (128-level, 1,1 dB per step) square root square table, depending on the application. The quantization indexes may be further compressed by one of the 5 Huffman codes and this information is transmitted to the decoder by SHUFF[ch] (see table B.3).

Table B.3: Code books and square root tables for scale factors

SHUFF[ch]	Code Book	Square Root Table
0	SA129	6 bit (Appendix D.1.1)
1	SB129	6 bit (Appendix D.1.1)
2	SC129	6 bit (Appendix D.1.1)
3	SD129	6 bit (Appendix D.1.1)
4	SE129	6 bit (Appendix D.1.1)
5	6-bit linear	6 bit (Appendix D.1.1)
6	7-bit linear	7 bit (Appendix D.1.2)
7	Invalid	Invalid

```
for (ch=0; ch<nPCHS; ch++) {
    SHUFF[ch] = ExtractBits(3);
}
```

Bit Allocation Quantizer Select BHUFF V 3 bits per channel

It indicates the codebook that was used to encode the bit allocation index ABITS (to be transmitted later) (see table B.4).

Table B.4: Codebooks for encoding bit allocation index ABITS

BHUFF[ch]	Codebook
0	A12
1	B12
2	C12
3	D12
4	E12
5	Linear 4-bit
6	Linear 5-bit
7	Invalid

```
for (ch=0; ch<nPCHS; ch++) {
    BHUFF[ch] = ExtractBits(3);
    }
```

Quantization Index Codebook Select V SEL variable bits

After subband samples are quantized using a mid-tread linear quantizer, the quantization indexes may be further encoded using either entropy (Huffman) or block coding in order to reduce bit rate. So the subband samples may appear in the bit stream as plain quantization indexes (no further encoding), entropy (Huffman) codes, or block codes. For channel ch, the selection of a particular codebook for a mid-tread linear quantizer indexed by ABITS[ch] is transmitted to the decoder as SEL[ch][ABITS[ch]]. No SEL is transmitted for ABITS[ch]≥11, because no further encoding is used for those quantizers. The decoder can find out the particular codebook that was used using ABITS[ch] and SEL[ch][ABITS[ch]] to look up table B.5.

Table B.5: Selection of quantization levels and codebooks

Quantizer index (ABITS)	Number of index Quantization Levels	Qodebook Select (SEL)							
		0	1	2	3	4	5	6	7
0	0	Not transmitted							
1	3	A3	V3						
2	5	A5	B5	C5	V5				
3	7	A7	B7	C7	V7				
4	9	A9	B9	C9	V9				
5	13	A13	B13	C13	V13				
6	17	A17	B17	C17	D17	E17	F17	G17	V17
7	25	A25	B25	C25	D25	E25	F25	G25	V25
8	33 or 32	A33	B33	C33	D33	E33	F33	G33	NFE
9	65 or 64	A65	B65	C65	D65	E65	F65	G65	NFE
10	129 or 128	A129	B129	C129	D129	E129	F129	G129	NFE
11	256	NFE							
12	512	NFE							
13	1 024	NFE							
14	2 048	NFE							
15	4 096	NFE							
16	8 192	NFE							
17	16 384	NFE							
18	32 768	NFE							
19	65 536	NFE							
20	131 072	NFE							
21	262 144	NFE							
22	524 288	NFE							
23	1 048 576	NFE							
24	2 097 152	NFE							
25	4 194 304	NFE							
26	8 388 608	NFE							
27-32	Invalid	Invalid							
27-32	Invalid	Invalid							

```
// ABITS=1:
n=0;
for (ch=0; ch<nPCHS; ch++)
    SEL[ch][n] = ExtractBits(1);
// ABITS = 2 to 5:

for (n=1; n<5; n++)
    for (ch=0; ch<nPCHS; ch++)
    SEL[ch][n] = ExtractBits(2);
// ABITS = 6 to 10:
for (n=5; n<10; n++)
    for (ch=0; ch<nPCHS; ch++)
    SEL[ch][n] = ExtractBits(3);
// ABITS = 11 to 26:
for (n=10; n<26; n++)
    for (ch=0; ch<nPCHS; ch++)
    SEL[ch][n] = 0; // Not transmitted, set to zero.
```

Scale Factor Adjustment Index V ADJ 2 bits per occasion

A scale factor adjustment index is transmitted whenever a SEL value indicates a Huffman codebook. This index points to the adjustment values shown in table B.6. This adjustment value should be multiplied to the scale factor (SCALE).

Table B.6: Scale factor adjustment values if Huffman coding is used to encode the subband quantization indexes

Scale Factor Adjustment index (ADJ)	Adjustment Value
0	1,0000
1	1,1250
2	1,2500
3	1,4375

```
// ABITS = 1:
n = 0;
for (ch=0; ch<nPCHS; ch++)
    if ( SEL[ch][n] == 0 ) { // Transmitted only if SEL=0 (Huffman code used)
// Extract ADJ index
ADJ = ExtractBits(2);
    // Look up ADJ table
    arADJ[ch][n] = AdjTable[ADJ];
    }
// ABITS = 2 to 5:
for (n=1; n<5; n++){
    for (ch=0; ch<nPCHS; ch++){
    if ( SEL[ch][n] < 3 ) { // Transmitted only when SEL<3
    // Extract ADJ index
    ADJ = ExtractBits(2);
    // Look up ADJ table
    arADJ[ch][n] = AdjTable[ADJ];
 }
}

    }
    // ABITS = 6 to 10:
 for (n=5; n<10; n++){
    for (ch=0; ch<nPCHS; ch++){
if ( SEL[ch][n] < 7 ) { // Transmitted only when SEL<7
// Extract ADJ index
ADJ = ExtractBits(2);
// Look up ADJ table
arADJ[ch][n] = AdjTable[ADJ];
}
 }
 }
```

Audio Header CRC Check Word V AHCRC 16 bits

It checks if there is any error in the bit stream from last CRC word (HCRC) up to this point.

```
    if ( CPF==1 ) // Present only if CPF=1.
    AHCRC = ExtractBits(16);
```

B.3.2 Unpack Subframes

B.3.2.1 Primary Audio Coding Side Information

Subsubframe Count	**V**	**SSC**	**2 bit**

It indicates that there are nSSC = SSC+1 sub-subframes in the current audio subframe.

```
SSC = ExtractBits(2);
nSSC = SSC+1;
```

Partial Subsubframe Sample Count	**V**	**PSC**	**3 bit**

It indicates the number of subband samples held in a partial subsubframe for each of the active subbands. A partial subsubframe is one which has less than 8 subband samples. It exists only in a termination frame and is always at the end of last normal subsubframe. A DSYNC word will always occur after a partial subsubframe.

PSC = ExtractBits(3);

Prediction Mode	**V**	**PMODE**	**1 bit per subband**

PMODE[ch][n]=1 indicates that ADPCM prediction is used (active) for subband n of primary audio channel [ch], and PMODE[ch][n]=0 otherwise. ADPCM must be extracted from the bit stream for all subbands, but ADPCM reconstruction can be limited to the lowest 20 subbands if DSP does not have enough MIPS.

```
    for (ch=0; ch<nPCHS; ch++)
for (n=0; n<nSUBS[ch]; n++)
    PMODE[ch][n] = ExtractBits(1);
```

Prediction Coefficients VQ Address	**V**	**PVQ**	**12 bits per occurrence**

It indexes to the vector code book (clause D.10.1) to get the ADPCM prediction coefficients. It is transmitted only for subbands whose ADPCM is active.

```
int nVQIndex;
for (ch=0; ch<nPCHS; ch++)
    for (n=0; n<nSUBS[ch]; n++)
    if ( PMODE[ch][n]>0 ) { // Transmitted only when ADPCM active

// Extract the VQindex
nVQIndex = ExtractBits(12);
// Look up the VQ table for prediction coefficients.
ADPCMCoeffVQ.LookUp(nVQIndex, PVQ[ch][n]) // 4 coefficients
    }
```

Bit Allocation Index	**V**	**ABITS**	**variable bits**

ABITS[ch][n] is the index to the mid-tread linear quantizer that was used to quantize the subband samples for the n-th subband of channel ch. ABITS[ch][n] may be transmitted as either a 4-bit or 5-bit word. In the case of a 4-bit word, it may be further encoded using one of the 5 Huffman codes. This encoding is the same for all subbands of each channel and is conveyed by BHUFF as shown in table B.4. There is obviously no need to allocate bits for the high frequency subbands because they are encoded using VQ.

```
for (ch=0; ch<nPCHS; ch++) {
    // BHUFF tells which codebook was used
nQSelect = BHUFF[ch];
```

// Use this codebook to decode the bit stream for ABITS[ch][n]

```
for (n=0; n<nVQSUB[ch]; n++) // Not for VQ encoded subbands.
    QABITS.ppQ[nQSelect]->InverseQ(InputFrame, ABITS[ch][n])
    }
```

Transition Mode V TMODE variable bits

TMODE[ch][n] indicates if there is a transient inside a subframe (subband analysis window) for subband n of channel ch. If there is a transient (TMODE[ch][n]>0), it further indicates that the transition occurred in subsubframe (subband analysis subwindow) TMODE[ch][n] + 1. TMODE[ch][n] is encoded by one of the 4 Huffman codes and the selection of which is conveyed by THUFF (see table B.2). The decoder assumes that there is no transition (TMODE[ch][n]=0) for all subbands of all channels unless it is told otherwise by the bit stream. Transient does not occur in the following situations, so TMODE is not transmitted:

- Only one subsubframe within the current subframe. This is because the time resolution of transient analysis is a subsubframe (subband analysis subwindow).

VQ encoded high frequency subbands. If there is a transient for a subband, it would not have been VQ encoded.

Subbands without bit allocation. If no bits are allocate for a subband, there is no need for transient.

```
// Always assume no transition unless told
for (ch=0; ch<nPCHS; ch++){
```

for (n=0; n<NumSubband; n++)

```
    TMODE[ch][n] = 0;
    // Decode TMODE[ch][n]
    if ( nSSC>1 ) {// Transient possible only if more than one subsubframe.
    for (ch=0; ch<nPCHS; ch++) {
    // TMODE[ch][n] is encoded by a codebook indexed by THUFF[ch]
    nQSelect = THUFF[ch];
    for (n=0; n<nVQSUB[ch]; n++)    // No VQ encoded subbands
    if ( ABITS[ch][n] >0 )  // Present only if bits allocated
    // Use codebook nQSelect to decode TMODE from the bit stream
    QTMODE.ppQ[nQSelect]->InverseQ(InputFrame,TMODE[ch][n])
    }
    }
 }
```

Scale Factors V SCALES variable bits

One scale factor is transmitted for subbands without transient. Otherwise two are transmitted, one for the episode before the transient and the other for after the transient. The quantization indexes of the scale factors may be encoded by Huffman code as shown in table B.3. If this is the case, they are difference-encoded before Huffman coding. The scale factors are finally obtained by using the quantization indexes to look up either the 6-bit or 7-bit square root quantization table according to table B.3.

```
    for (ch=0; ch<nPCHS; ch++) {
// Clear SCALES

for (n=0; n<NumSubband; n++) {
    SCALES[ch][n][0] = 0;
    SCALES[ch][n][1] = 0;
}
// SHUFF indicates which codebook was used to encode SCALES
nQSelect = SHUFF[ch];
// Select the root square table (SCALES were nonlinearly
// quantized).
if ( nQSelect == 6 )
pScaleTable = &RMS7Bit; // 7-bit root square table
else
pScaleTable = &RMS6Bit; // 6-bit root square table
//
// Clear accumulation (if Huffman code was used, the difference
// of SCALES was encoded).

//
nScaleSum = 0;
//
// Extract SCALES for Subbands up to VQSUB[ch]
//
for (n=0; n<nVQSUB[ch]; n++)
    if ( ABITS[ch][n] >0 ) { // Not present if no bit allocated
//
// First scale factor
//
// Use the (Huffman) code indicated by nQSelect to decode
```

```
// the quantization index of SCALES from the bit stream
QSCALES.ppQ[nQSelect]->InverseQ(InputFrame, nScale);
// Take care of difference encoding
if ( nQSelect < 5 ) // Huffman encoded, nScale is the difference
    nScaleSum += nScale;    // of the quantization indexes of SCALES.
else    // Otherwise, nScale is the quantization
nScaleSum = nScale;     // level of SCALES.
// Look up SCALES from the root square table
pScaleTable->LookUp(nScaleSum, SCALES[ch][n][0])
//
// Two scale factors transmitted if there is a transient
//
if (TMODE[ch][n]>0) {
// Use the (Huffman) code indicated by nQSelect to decode
// the quantization index of SCALES from the bit stream
QSCALES.ppQ[nQSelect]->InverseQ(InputFrame, nScale);
// Take care of difference encoding
if ( nQSelect < 5 )     // Huffman encoded, nScale is the
    nScaleSum += nScale;    // of SCALES.
else    // Otherwise, nScale is SCALES
    nScaleSum = nScale;     // itself.
    // Look up SCALES from the root square table
    pScaleTable->LookUp(nScaleSum, SCALES[ch][n][1])
}
    }
    //
    // High frequency VQ subbands
    //
    for (n=nVQSUB[ch]; n<nSUBS[ch]; n++) {
// Use the code book indicated by nQSelect to decode
// the quantization index of SCALES from the bit stream
QSCALES.ppQ[nQSelect]->InverseQ(InputFrame, nScale);
// Take care of difference encoding
if ( nQSelect < 5 )     // Huffman encoded, nScale is the
    nScaleSum += nScale; // of SCALES.
else    // Otherwise, nScale is SCALES
    nScaleSum = nScale; // itself.
// Look up SCALES from the root square table
pScaleTable->LookUp(nScaleSum, SCALES[ch][n][0])
    }
    }
```

Joint Subband Scale Factor Codebook Select V JOIN SHUFF 3 bits per channel

If joint subband coding is enabled (JOINX[ch]>0), JOIN SHUFF[ch] selects which code book was used to encode the scale factors (JOIN SCALES) which will be used when copying subband samples from the source channel to the current channel ch. For now, these scale factors are encoded in exactly the same way as that for SCALES, so use table B.3 to look up the codebook.

```
    for (ch=0; ch<nPCHS; ch++)
if (JOINX[ch]>0 ) // Transmitted only if joint subband coding enabled.
    JOIN_SHUFF[ch] = ExtractBits(3);
```

Scale Factors for Joint Subband Coding V JOIN SCALES variable bits

The scale factors are used to scale the subband samples copied from the source channel (JOINX[ch]-1) to the current channel. The index of the scale factor is encoded using the code book indexed by JOIN SHUFF[ch]. After this index is decoded, it is used to look up the table in annex D.3 to get the scale factor. No transient is permitted for jointly encoded subbands, so a single scale factor is included. The joint subbands start from the nSUBS of the current channel until the nSUBS of the source channel.

```
int     nSourceCh;
for (ch=0; ch<nPCHS; ch++)

    if (JOINX[ch]>0 ) { // Only if joint subband coding enabled.
    nSourceCh = JOINX[ch]-1; // Get source channel. JOINX counts
    // channels as 1,2,3,4,5, so minus 1.
   nQSelect = JOIN_SHUFF[ch]; // Select code book.
```

```
    for (n=nSUBS[ch]; n<nSUBS[nSourceCh]; n++) {
    // Use the code book indicated by nQSelect to decode
    // the quantization index of JOIN_SCALES
    QSCALES.ppQ[nQSelect]->InverseQ(InputFrame, nJScale);
    // Bias by 64
    nJScale = nJScale + 64;
    // Look up JOIN_SCALES from the joint scale table
    JScaleTbl.LookUp(nJScale, JOIN_SCALES[ch][n]);
    }
    }
```

Stereo Down-Mix Coefficients NV DOWN 7 bits per coefficient

One concern arising from the proliferation of multi-channel audio systems is that most home systems presently have only two channel playback capability. To accommodate this a fixed 2-channel down matrix processes is commonly used following the multi-channel decoding stage. However, for music only applications the image quality etc. of the down matrixed signal may not match that of an equivalent stereo recording found on CD.

The concept of embedded mixing is to allow the producer to dynamically specify the matrixing coefficients within the audio frame itself. In this way the stereo down mix at the decoder may be better matched to a 2-channel playback environment. Two 7-bit down mix indexes (DOWN) are transmitted along with the multi-channel audio in every subframe (if PCHS+1 > 2 and MIX!=0).

```
if ( (MIX!=0) && (nPCHS>2) )
// Extract down mix indexes
for (ch=0; ch<nPCHS; ch++) { // Each primary channel
    DOWN[ch][0] = ExtractBits(7);
    DOWN[ch][1] = ExtractBits(7);
}
```

// Look up down mix coefficients

After all subband samples are decoded, they can be down-mixed to form the left and right stereo channels as follows:

```
for (n=0; n<nSUBS; n++) { // Each active subbands
LeftChannel = 0;
RightChannel = 0;
for (ch=0; ch<nPCHS; ch++) { // Each primary channels
    LeftChannel += DOWN[ch][0]*Sample[Ch];
    RightChannel += DOWN[ch][1]*Sample[Ch];
}
}
```

Down mixing may also be performed on the PCM samples after the filterbank reconstruction.

Dynamic Range Coefficient NV RANGE 8 bits

Dynamic range coefficient is to allow for the convenient compression of the audio dynamic range at the output of the decoder. Dynamic range compression is particularly important in listening environments where high ambient noise levels make it impossible to discriminate low level signals without risking damaging the loudspeakers during loud passages. This problem is further compounded by the growing use of 20-bit PCM audio recordings which exhibit dynamic ranges as high as 110 dB.

Each coefficient is 8-bit signed fractional Q2 binary, and represents a logarithmic gain value as shown in table A.4 giving a range of ±31,75 dB in steps of 0,25 dB. Dynamic range compression is affected by multiplying the decoded audio samples by the linear coefficient.

The degree of compression can be altered with the appropriate adjustment to the coefficient values at the decoder and can be switched off completely by ignoring the coefficients.

```
if ( DYNF != 0 ) {
    nIndex = ExtractBits(8);
```

RANGEtbl.LookUp(nIndex,RANGE);

// The following range adjustment is to be performed

// after QMF reconstruction

```
    for (ch=0; ch<nPCHS; ch++)
    for (n=0; n<nNumSamples; n++)
    AudioCh[ch].ReconstructedSamples[n] *= RANGE;
    }
```

Side Information CRC Check Word V SICRC 16 bits

It checks if there is any error in the bit stream from the beginning of the Primary Audio Coding Side Information (starting with SSC) up to this point.

```
if ( CPF==1 ) // Present only if CPF=1.
    SICRC = ExtractBits(16);
```

B.3.3 Primary Audio Data Arrays

VQ Encoded High Frequency Subbands NV HFRE 10 bits per applicable subbands

At low bit rates, some high frequency subbands are encoded using vector quantization (VQ). Each vector from this code book consists of 32 subband samples, corresponding to the maximum possible subframe (4 normal subsubframes):

4 subsubframe x 8 samples/subsubframe = 32 samples:

If the current subframe is short of 32 samples, the remaining samples are padded with either zeros or "don't care" and then vector-quantized. The vector address is then included in the bit stream. After the decoder picks up the vector address, it looks up the vector code book to get the 32 samples. But the decoder will only pick nSSC x 8 out of the 32 samples and scale them with the scale factor SCALES.

```
for (ch=0; ch<nPCHS; ch++)
for (n=nVQSUB[ch]; n<nSUBS[ch]; n++) {
    // Extract the VQ address from the bit stream
    nVQIndex = ExtractBits(10);
    // Look up the VQ code book for 32 subband samples.
    HFreqVQ.LookUp(nVQIndex, HFREQ[ch][n])
    // Scale and take the samples
    Scale = (real)SCALES[ch][n][0]; // Get the scale factor
    for (m=0; m<nSSC*8; m++, nSample++)
    aPrmCh[ch].aSubband[n].raSample[m] = rScale*HFREQ[ch][n][m];
    }
```

Low Frequency Effect Data V LFE 8 bits per sample

The presence of a LFE channel and its interpolation filter selection are flagged by LFF in the frame header (see table 5.16). The number of decimated LFE samples in the current subframe is 2 x LFF x nSSC, corresponding to the decimation factor and the subframe size. The LFE samples are normalized with a scale factor and then quantized with a step size of 0,035, before being included in the bit stream as 8-bit 2's compliment. This scale factor is nonlinearly quantized using the 7-bit root square and then directly included in the bit stream right after the decimated LFE samples. Therefore, on the decoder side, these decimated LFE samples need to be adjusted by the quantization step size and scale factor. After this adjustment, they are used to interpolate the other samples. The choice of the interpolation filter is indicated by LFF as shown in table 5.16.

```
if ( LFF>0 ) { // Present only if flagged by LFF
// extract LFE samples from the bit stream
for (n=0; n<2*LFF*nSSC; n++)
LFE[n] = (signed int)(signed char)ExtractBits(8);
// Use char to get sign extension because it
// is 8-bit 2's compliment.
// Extract scale factor index from the bit stream
LFEscaleIndex = ExtractBits(8);
// Look up the 7-bit root square quantization table
pLFE_RMS->LookUp(LFEscaleIndex,nScale);
// Account for the quantizer step size which is 0.035
rScale = nScale*0.035;
// Get the actual LFE samples
for (n=0; n<2*LFF*nSSC; n++)
    LFECh.rLFE[k] = LFE[n]*rScale;
// Interpolation LFE samples
LFECh.InterpolationFIR(LFF); // LFF indicates which
    // interpolation filter to use
    }
```

Audio Data V AUDIO variable bits

The audio data are grouped as nSSC subsubframes, each consisting of 8 samples for each subband. Each sample was quantized by a mid-tread linear quantizer indexed by ABITS. The resultant quantization index may further be encoded by either a Huffman or block code. If it is not, it is included in the bit stream as 2's compliment. All this information is indicated by SEL. The (ABITS,SEL) pair then tells how the subband samples should be extracted from the bit stream (table B.5).

The resultant subband samples are then compensated by their respective quantization step sizes and scale factors. Special care must be paid to possible transient in the subframe. If a transient is flagged by TMODE, one scale factor should be used for samples before the transient and the other one for the after the transient.

For some of the subbands that are ADPCM encoded, the samples of these subbands thus far obtained are actually the difference signals. Their real values must be recovered through a reverse ADPCM process.

At end of each subsubframe there may be a synchronization check word DSYNC = 0xffff depending on the flag ASPF in the frame header, but there must be at least a DSYNC at the end of each subframe.

```
//
// Select quantization step size table
//
if ( RATE == 0x1f )
pStepSizeTable = &StepSizeLossLess;     // Lossless quantization
else
pStepSizeTable = &StepSizeLossy;    // Lossy
//
// Unpack the subband samples
//
    for (nSubSubFrame=0; nSubSubFrame<nSSC; nSubSubFrame++) {
 for (ch=0; ch<nPCHS; ch++)

for (n=0; n<nVQSUB[ch]; n++) { // Not high frequency VQ subbands
//
// Select the mid-tread linear quantizer
//
nABITS = ABITS[ch][n];  // Select the mid-tread quantizer
pCQGroup = &pCQGroupAUDIO[nABITS-1];// Select the group of
    // code books corresponding to the
    // the mid-tread linear quantizer.
nNumQ = pCQGroupAUDIO[nABITS-1].nNumQ-1;// Number of code
    // books in this group
//
// Determine quantization index code book and its type
//
// Select quantization index code book
nSEL = SEL[ch][nABITS-1];
// Determine its type
nQType = 1;     // Assume Huffman type by default
if ( nSEL==nNumQ ) {    // Not Huffman type
    if ( nABITS<=7 )
nQType = 3;     // Block code
    else
nQType = 2;     // No further encoding
}
    if ( nABITS==0 )    // No bits allocated
    nQType = 0;
    //
    // Extract bits from the bit stream
    //
    switch ( nQType ) {
    case 0:     // No bits allocated
    for (m=0; m<8; m++)
    AUDIO[m] = 0;
    break;
    case 1:     // Huffman code
    for (m=0; m<8; m++)
    pCQGroup->ppQ[nSEL]->InverseQ(InputFrame,AUDIO[m]);
    break;
    case 2: // No further encoding
    for (m=0; m<8; m++) {
// Extract quantization index from the bit stream
```

```
pCQGroup->ppQ[nSEL]->InverseQ(InputFrame, nCode)
// Take care of 2's compliment
AUDIO[m] = pCQGroup->ppQ[nSEL]->SignExtension(nCode);
    }
    break;
    case 3: // Block code
    pCBQ = &pCBlockQ[nABITS-1]; // Select block code book
m = 0;
    for (nBlock=0; nBlock<2; nBlock++) {
// Extract the block code index from the bit stream
pCQGroup->ppQ[nSEL]->InverseQ(InputFrame, nCode)
// Look up 4 samples from the block code book
pCBQ->LookUp(nCode,&AUDIO[m])
m += 4;
}
break;
    default: // Undefined
    printf("ERROR: Unknown AUDIO quantization index code book.");
    }
    }
//
// Account for quantization step size and scale factor
//
// Look up quantization step size
nABITS = ABITS[ch][n];
pStepSizeTable->LookUp(nABITS, rStepSize);
// Identify transient location
nTmode = TMODE[ch][n];
if ( nTmode == 0 ) // No transient
    nTmode = nSSC;
// Determine proper scale factor
if (nSubSubFrame<nTmode) // Pre-transient
    rScale = rStepSize * SCALES[ch][n][0]; // Use first scale factor
else    // After-transient
    rScale = rStepSize * SCALES[ch][n][1]; // Use second scale factor
// Adjustmemt of scale factor
rScale *= arADJ[ch][SEL[ch][nABITS-1]];     // arADJ[ ][ ] are assumed 1
// unless changed by bit
// stream when SEL indicates
// Huffman code.
// Scale the samples
nSample = 8*nSubSubFrame; // Set sample index
for (m=0; m<8; m++, nSample++)
    aPrmCh[ch].aSubband[n].aSample[nSample] = rScale*AUDIO[m];
//
// Inverse ADPCM
//
if ( PMODE[ch][n] != 0 ) // Only when prediction mode is on.
    aPrmCh[ch].aSubband[n].InverseADPCM();
//
// Check for DSYNC
//
if ( (nSubSubFrame==(nSSC-1)) || (ASPF==1) ) {
DSYNC = ExtractBits(16);
if ( DSYNC != 0xffff )
    printf("DSYNC error at end of subsubframe #%d", nSubSubFrame);
    }
    }
    }
```

B.3.4 Unpack Optional Information

The optional information may be included at the end of the frame following completion of the audio data arrays, depending on the status of the optional header flags. This data is not intrinsic to the operation of the decoder but may be used for post processing routines.

Time Code Stamp	**ACC**	**TIMES**	**32 bit**

Time code may be used to align audio to video.

```
    if ( TIMEF==1 ) // Present only when TIMEF=1.
    TIMES = ExtractBits(32);
```

Auxiliary Data Byte Count V AUXCT 6 bits

The number of auxiliary data bytes to be transmitted in the following AUXD array. It must be in the range of 1-63.

```
if ( AUXF==1 ) // Present only if AUXF=1.
    AUXCT = ExtractBits(6);
    else
    AUXCT = 0; // Clear it.
```

Auxiliary Data Bytes NV AUXD 8*AUXCT bits

```
for (int n=0; n<AUXCT; n++)
    AUXD[n] = ExtractBits(8);
```

Optional CRC Check Bytes V OCRC 16 bits

Optional CRC check bytes will be present if CPF is active and mix, or dynamic range coefficients are present.

```
if ( (CPF==1) && ( (MIX!=0)||(DYNF!=0) ) )
    OCRC = ExtractBits(16);
```

Annex C (normative): Decoding Algorithms

The following annex outlines the decoding routines utilized by Coherent Acoustics.

C.1 Block Code

We will present two versions of the block code decoder based on

- the table look-up method;
- the arithmetic method that requires one modulus division and one integer division per one decoded quantization index.

The table look-up based decoding of a block code may be best illustrated by an example. Suppose a code of 64 is received as a three level block code. This code can be decoded as follows:

1st Element: 64 = 3 x 21 + 1; so quantization index = 0

2nd Element: 21 = 3 x 7 + 0; so quantization index = -1

3rd Element: 7 = 3 x 2 + 1; so quantization index = 0

4th Element: 2 = 3 x 0 + 2; so quantization index = +1

where the quantization indexes are obtained by using the residuals to look up the quantization index table [-1, 0, 1]. In summary, the quantization indexes of the four samples are (0, -1, 0, +1).

The same code can be decoded using the code book of table V.3 in clause D.6.1. In order to facilitate the decoding process, this table is rearranged to give table 4.1. Then this code of 64 is decoded as follows:

4th Element: 64 - 54 = 10 ≥ 0; so quantization index = +1

3rd Element: 10 - 9 = 1 ≥ 0; so quantization index = 0

2nd Element: 1 - 0 = 1 ≥ 0; so quantization index = -1

1st Element: 1 - 1 = 0 ≥ 0; so quantization index = 0

Therefore, the quantization indexes of the four samples are (0, -1, 0, +1). A general decoding procedure is given in the following pseudocode, assuming that the block codes in clause A.6 are rearranged as in table C.1.

Table C.1: 3-level 4-element 7-bit Block Code Book

Quantization Level index		-1	0	+0
Code For	1st Element	0	1	2
	2nd Element 0 3 6	0	3	6
	3rd Element 0 9 18	0	9	18
	4th Element 0 27 54	0	27	54

```
int DecodeBlockCode(int nCode, int *pnValue) {
     // nCode:  Input code to be decoded.
// nNumElement:     Number of elements (samples) encoded
// in a block.
// nNumLevel:   Number of quantization levels.
// *pnValue:    Array of decoded sample values.
// *pnTable:    Pointer to the code book. The code book is
//  organized as an array, each row of which contains
//  the code book for a particular element (sample).
pnValue += 3;
nOffset = (nNumLevel-1)/2;
int *pnEntry; // Pointer to the entries in the code book.
```

```
for (int n=nNumElement; n>0; n--) {
    pnEntry = pnTable + n*nNumLevel; // Point to the last entry
    // in the code book.
    for (int m=0; m<nNumLevel; m++) {
    pnEntry--;
    if ( nCode >= *pnEntry ) {
    nCode -= *pnEntry;
    *pnValue = nOffset-m; // quantization index is calculated.
    if ( nCode<0 ) {
    printf("ERROR: block code look-up fail.\n");
    return NULL;
    }
    break;
    }
    }
    pnValue--;
    }
// Check if look-up successful
if ( nCode == 0 )
    return 1;
    else {
    printf("ERROR: block code lock-up fail.\n");
    return NULL;
    }
}
```

Very compact version of the block code decoder that does not use table look-up can be obtained using the modulus and integer division. The pseudocode that implements this version of the decoder is listed below.

```
int DecodeBlockCode(int nCode, int *pnValue) {
     // nCode:  Input code to be decoded.
// nNumElement:     Number of elements (samples) encoded in a block.
// nNumLevel:   Number of quantization levels.
// *pnValue:    Array of decoded sample values.
nOffset = (nNumLevel-1)>>1;
for (int n=0; n< nNumElement; n++) {
    pnValue[n] = (nCode % nNumLevel) – nOffset;
    nCode /= nNumLevel;
    }
if ( nCode == 0 )
    return 1;
    else {
    printf("ERROR: block code lock-up fail.\n");
    return NULL;
    }
}
```

C.2 CRC Error Detection

DTS Coherent Acoustics has three 16-bit CRC check words: HCRC, AHCRC, and SICRC for bit stream header (from the frame synchronization word up to HCRC word), audio header (from after HCRC up to AHCRC), and side information CRC check (from after AHCRC up to SICRC), respectively. The following generator polynomial is used to generate each of the 16-bit CRC check word:

$$x^{16} + x^{15} + x^2 + 1.$$

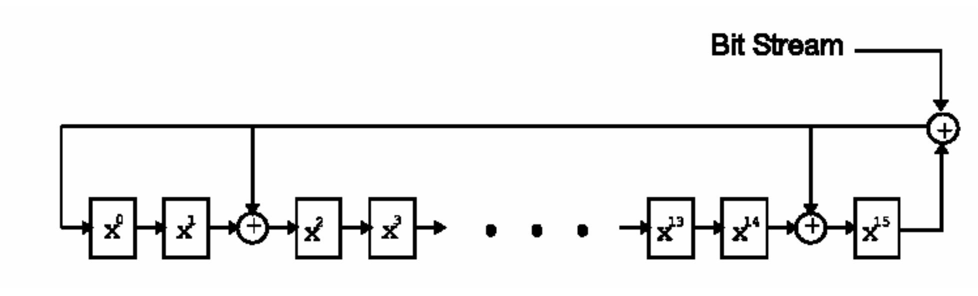


Figure C.1: A linear feedback shift register implementation of CRC calculation

The CRC calculation using this generator polynomial can be implemented by many methods, including the linear feedback shift registers shown in figure C.1. The CRC decoding process consists of the following steps:

- Clear the shift register.
- Shift each bit up to the end of the CRC check word serially into the shift register in the order in which they appear in the bit stream.
- After the last bit of the CRC check word is shifted through, check the shift register. If the shift register is all zero, there is no error in the bit stream up to the CRC check word. Otherwise, an error has occurred and appropriate action such as muting should be taken.

Although CRC check words must be extracted from the bit stream, it is optional to actually implement the error check.

C.3 Inverse ADPCM

Inverse ADPCM process is executed for each sample in a subband whose PMODE=1:

```
void InverseADPCM(void) {
// NumADPCMCoeff =4, the number of ADPCM coefficients.
// raADPCMcoeff[] are the ADPCM coefficients extracted
// from the bit stream.
// raSample[NumADPCMCoeff], ..., raSample[-1] are the
// history from last subframe or subsubframe. It must
// updated each time before reverse ADPCM is run for a
// block of samples for each subband.
for (m=0; m<nNumSample; m++)
    for (n=0; n<NumADPCMCoeff; n++)
    raSample[m] += raADPCMcoeff[n]*raSample[m-n-1];
}
```

C.4 Joint Subband Coding

```
    for (ch=0; ch<nPCHS; ch++)
    if ( JOINX[ch]>0 ) { // Joint subband coding enabled.
    nSourceCh = JOINX[ch]-1;    // Get source channel. JOINX counts
    // channels as 1,2,3,4,5, so minus 1.
    for (n=nSUBS[ch]; n<nSUBS[nSourceCh]; n++)
    for (nSample=0; n<8*nSSC; nSample++)
    aPrmCh[ch].aSubband[n].aSample[nSample] = JOIN_SCALES[ch][n]
    * aPrmCh[nSourceCh].aSubband[n].aSample[nSample];
    }
```

C.5 Sum/Difference Decoding

If flag SUMF is set, the front left and right channels are sum/difference encoded and therefore must be appropriately decoded to produce the correct signals for the front left and right channels. Decoding is achieved by operating on the reconstructed subband samples:

```
    for (n=0; n<nSUBS; n++) // All active subbands.
    for (nSample=0; nSample<8*nSSC; nSample++) { // Samples in all subsubframes
    FrontLeft[nSample] = Fleft[nSample] + Fright[nSample];
    Frontright[nSample] = Fleft[nSample] - Fright[nSample];
    }
```

This decoding is also required when AMODE = 3

Similarly when SUMS is set the reconstructed subband samples of the Left and right surround channels are decoded as:

```
    for (n=0; n<nSUBS; n++) // All active subbands.
    for (nSample=0; nSample<8*nSSC; nSample++) { // Samples in all subsubframes
SurroundLeft[nSample] = Sleft[nSample] + Sright[nSample];
Surroundright[nSample] = Sleft[nSample] - Sright[nSample];
    }
```

C.6 Filter Bank Reconstruction

Having prepared all the subband samples, it is time to go through subband interpolation to reconstruct the PCM samples for each primary channel. As discussed before, there are two filter banks, one for perfect reconstruction and the other for non-perfect. The encoder indicates its choice to the decoder through the FILTS flag in the frame header.

```
for (ch=0; ch<nPCHS; ch++)
    aPrmCh[ch].QMFInterpolation(FILTS, nSUBS[ch]);
\\ FILTS indicates which filter bank to use
\\ nSUBS[ch] indicates the number of active subbands. Subbands
\\ above it are all zeros. For joint intensity coded subbands,
\\ it must be set to that of the source channel, in order to
\\ reflect the true subband activity.
```

There are many methods to efficiently implement the reconstruction filter bank. We present only one of them which we think is fairly efficient. The two sets of 512 FIR coefficients are tabulated in clause D.8.1 (perfect reconstruction) and clause D.8.2 (nonperfect reconstruction), and the selection is flagged by FILTS in the frame header

The first step is to pre-calculate the cosine modulation coefficients:

```
PreCalCosMod() {
    for (j=0,k=0;k<16;k++)
    for (i=0;i<16;i++)
    raCosMod[j++] = (real)cos((2*i+1)*(2*k+1)*Pi/64);
    for (k=0;k<16;k++)
    for (i=0;i<16;i++)
    raCosMod[j++] = (real)cos((i)*(2*k+1)*Pi/32);
    for (k=0;k<16;k++)
    raCosMod[j++] = real(0.25/(2*cos((2*k+1)*Pi/128)));
    for (k=0;k<16;k++)
    raCosMod[j++] = real(-0.25/(2.0*sin((2*k+1)*Pi/128)));
}
```

The filter bank reconstruction is illustrated by the following pseudocode:

```
QMFInterpolation(FILTS, int nSUBS) {
    // Select filter
    if ( FILTS==0 ) // Non-perfect reconstruction
    prCoeff = raCoeffLossy;
    else // Perfect reconstruction
    prCoeff = raCoeffLossLess;
// Interpolation begins
nChIndex = 0;       // Reconstructed channel sample index
for (nSubIndex=nStart; nSubIndex<nEnd; nSubIndex++) { // Subband samples
    // Load in one sample from each subband
    for (i=0; i<nSUBS; i++)
    raXin[i] = aSubband[i].raSample[nSubIndex];
    for (i=nSUBS; i<NumSubband; i++) // Clear inactive subbands
    raXin[i] = 0.0;
//Multiply by cosine modulation coefficients and
// Create temporary arrays SUM and DIFF.
for (j=0,k=0;k<16;k++) {
A[k] = (real)0.0;
for (i=0;i<16;i++)
    A[k]+=(raXin[2*i]+raXin[2*i+1])*raCosMod[j++];
}
for (k=0;k<16;k++) {
    B[k] = (real)0.0;
    for (i=0;i<16;i++) {
if(i>0)
    B[k]+=(raXin[2*i]+raXin[2*i-1])*raCosMod[j++];
    else
    B[k]+=(raXin[2*i])*raCosMod[j++];
    }
SUM[k]=A[k]+B[k];
DIFF[k]=A[k]-B[k];
    }
// Store history
for (k=0;k<16;k++)
    raX[k]=raCosMod[j++]*SUM[k];
for (k=0;k<16;k++)
    raX[32-k-1]=raCosMod[j++]*DIFF[k];
// Multiply by filter coefficients
for(k=31,i=0;i<32;i++,k--)
    for(j=0;j<512;j+=64)
```

```
    raZ[i] += prCoeff[i+j]*(raX[i+j]-raX[j+k]);
for(k=31,i=0;i<32;i++,k--)
    for(j=0;j<512;j+=64)
    raZ[32+i] += prCoeff[32+i+j]*(-raX[i+j]-raX[j+k]);
// Create 32 PCM output samples
for(i=0;i<32;i++)
    naCh[nChIndex++] = int(rScale*raZ[i]);
// Update working arrays
for(i=511;i>=32;i--)
    raX[i] = raX[i-32];
for(i=0;i<NumSubband;i++)
    raZ[i] = raZ[i+32];
for(i=0;i<NumSubband;i++)
    raZ[i+32] = (real)0.0;
    }
}
```

C.7 Interpolation of LFE Channel

```
void InterpolationFIR(int nDecimationSelect) {
// rLFE:    An array holding decimated samples.
//  Samples in current subframe starts from rLFE[0],
//  while rLFE[-1], rLFE[-2], ..., stores samples
//  from last subframe as history.
// naCh:    An array holding interpolated samples
// Select decimation filter
if (nDecimationSelect==1) {// 128 decimation
nDeciFactor = 128;  // Decimation factor = 128
prCoeff = raCoeff128;   // Point to the 128X FIR coefficient array
}
else {  // 64 decimation
nDeciFactor = 64;
prCoeff = raCoeff64;
}
// Interpolation
NumFIRCoef = 512; // Number of FIR coefficients
nInterpIndex = 0; // Index to the interpolated samples
for (nDeciIndex=0; nDeciIndex<nNumDeciSample; nDeciIndex++) {
// One decimated sample generates nDeciFactor interpolated ones.
for (k=0; k<nDeciFactor; k++) {
    // Clear accumulation
rTmp = 0.0;
// Accumulate
for (J=0; J<NumFIRCoef/nDeciFactor; J++)
rTmp += rLFE[nDeciIndex-J]*prCoeff[k+J*nDeciFactor];
    // Save interpolated samples as integer
    naCh[nInterpIndex++] = (int)rTmp;
    }
nDeciIndex++; // Next decimated sample
    }
}
```

Annex D (normative): Large Tables

D.1 Scale Factor Quantization Tables

D.1.1 6-bit Quantization (Nominal 2,2 dB Step)

Index	Quantization level	Quantization level in dB
0	1	0,0
1	2	6,0
2	2	6,0
3	3	9,5
4	3	9,5
5	4	12,0
6	6	15,5
7	7	17,0
8	10	20,0
9	12	21,5
10	16	24,0
11	20	26,0
12	26	28,3
13	34	30,6
14	44	32,8
15	56	35,0
16	72	37,2
17	93	39,4
18	120	41,6
19	155	43,8
20	200	46,0
21	257	48,2
22	331	50,4
23	427	52,6
24	550	54,8
25	708	57,0
26	912	59,2
27	1 175	61,4
28	1 514	63,6
29	1 950	65,8
30	2 512	68,0
31	3 236	70,2
32	4 169	72,4
33	5 370	74,6
34	6 918	76,8
35	8 913	79,0
36	11 482	81,2
37	14 791	83,4
38	19 055	85,6
39	24 547	87,8
40	31 623	90,0
41	40 738	92,2
42	52 481	94,4
43	67 608	96,6
44	87 096	98,8
45	112 202	101,0
46	144 544	103,2
47	186 209	105,4
48	239 883	107,6
49	309 030	109,8
50	398 107	112,0
51	512 861	114,2
52	660 693	116,4
53	851 138	118,6

Index	Quantization level	Quantization level in dB
54	1 096 478	120,8
55	1 412 538	123,0
56	1 819 701	125,2
57	2 344 229	127,4
58	3 019 952	129,6
59	3 890 451	131,8
60	5 011 872	134,0
61	6 456 542	136,2
62	8 317 638	138,4
63	invalid	invalid

D.1.2 7-bit Quantization (Nominal 1,1 dB Step)

Index	Quantization level	Quantization level in dB
0	1	0,0
1	1	0,0
2	2	6,0
3	2	6,0
4	2	6,0
5	2	6,0
6	3	9,5
7	3	9,5
8	3	9,5
9	4	12,0
10	4	12,0
11	5	14,0
12	6	15,5
13	7	17,0
14	7	17,0
15	8	18,0
16	10	20,0
17	11	21,0
18	12	21,5
19	14	23,0
20	16	24,0
21	18	25,1
22	20	26,0
23	23	27,2
24	26	28,3
25	30	29,5
26	34	30,6
27	38	31,6
28	44	32,8
29	50	34,0
30	56	35,0
31	64	36,1
32	72	37,2
33	82	38,3
34	93	39,4
35	106	40,5
36	120	41,6
37	136	42,7
38	155	43,8
39	176	44,9
40	200	46,0
41	226	47,1
42	257	48,2
43	292	49,3
44	331	50,4
45	376	51,5
46	427	52,6
47	484	53,7
48	550	54,8
49	624	55,9
50	708	57,0

Index	Quantization level	Quantization level in dB
51	804	58,1
52	912	59,2
53	1 035	60,3
54	1 175	61,4
55	1 334	62,5
56	1 514	63,6
57	1 718	64,7
58	1 950	65,8
59	2 213	66,9
60	2 512	68,0
61	2 851	69,1
62	3 236	70,2
63	3 673	71,3
64	4 169	72,4
65	4 732	73,5
66	5 370	74,6
67	6 095	75,7
68	6 918	76,8
69	7 852	77,9
70	8 913	79,0
71	10 116	80,1
72	11 482	81,2
73	13 032	82,3
74	14 791	83,4
75	16 788	84,5
76	19 055	85,6
77	21 627	86,7
78	24 547	87,8
79	27 861	88,9
80	31 623	90,0
81	35 892	91,1
82	40 738	92,2
83	46 238	93,3
84	52 481	94,4
85	59 566	95,5
86	67 608	96,6
87	76 736	97,7
88	87 096	98,8
89	98 855	99,9
90	112 202	101,0
91	127 350	102,1
92	144 544	103,2
93	164 059	104,3
94	186 209	105,4
95	211 349	106,5
96	239 883	107,6
97	272 270	108,7
98	309 030	109,8
99	350 752	110,9
100	398 107	112,0
101	451 856	113,1
102	512 861	114,2
103	582 103	115,3
104	660 693	116,4
105	749 894	117,5
106	851 138	118,6
107	966 051	119,7
108	1 096 478	120,8
109	1 244 515	121,9

Index	Quantization level	Quantization level in dB
110	1 412 538	123,0
111	1 603 245	124,1
112	1 819 701	125,2
113	2 065 380	126,3
114	2 344 229	127,4
115	2 660 725	128,5
116	3 019 952	129,6
117	3 427 678	130,7
118	3 890 451	131,8
119	4 415 704	132,9
120	5 011 872	134,0
121	5 688 529	135,1
122	6 456 542	136,2
123	7 328 245	137,3
124	8 317 638	138,4
125	invalid	invalid
126	invalid	invalid
127	invalid	invalid

D.2 Quantization Step Size

D.2.1 Lossy Quantization

ABITS Index	Step-size*2^22	Nominal Step-size
0	0	0,0
1	6 710 886	1,6
2	4 194 304	1,0
3	3 355 443	0,8
4	2 474 639	0,59
5	2 097 152	0,50
6	1 761 608	0,42
7	1 426 063	0,34
8	796 918	0,19
9	461 373	0,11
10	251 658	0,06
11	146 801	0,035
12	79 692	0,019
13	46 137	0,011
14	27 263	0,0065
15	16 777	0,0040
16	10 486	0,0025
17	5 872	0,0014
18	3 355	0,0008
19	1 887	0,00045
20	1 258	0,00030
21	713	0,00017
22	336	0,00008
23	168	0,00004
24	84	0,00002
25	42	0,00001
26	21	0,000005
27	invalid	invalid
28	invalid	invalid
29	invalid	invalid
30	invalid	invalid
31	invalid	invalid

D.2.2 Lossless Quantization

ABITS Index	Step-size *2^22	Nominal Step-size
0	0	0,0
1	4 194 304	1,0
2	2 097 152	0,5
3	1 384 120	0,33
4	1 048 576	0,25
5	696 254	0,166
6	524 288	0,125
7	348 127	0,083
8	262 144	0,0625
9	131 072	0,03125
10	65 431	0,0156
11	33 026	7,874E-3
12	16 450	3,922E-3
13	8 208	1,957E-3
14	4 100	9,775E-4
15	2 049	4,885E-4
16	1 024	2,442E-4
17	512	1,221E-4
18	256	6,104E-5
19	128	3,052E-5
20	64	1,526E-5
21	32	7,629E-6
22	16	3,815E-6
23	8	1,907E-6
24	4	9,537E-7
25	2	4,768E-7
26	1	2,384E-7
27	invalid	invalid
28	invalid	invalid
29	invalid	invalid
30	invalid	invalid
31	invalid	invalid

D.3 Scale Factor for Joint Intensity Coding

0.025088
0.026624
0.02816
0.029824
0.031616
0.033472
0.035456
0.037568
0.039808
0.042176
0.044672
0.047296
0.050112
0.05312
0.056256
0.059584
0.063104
0.066816
0.070784
0.075008
0.079424
0.08416
0.089152
0.0944
0.099968
0.10592
0.112192
0.118848
0.125888
0.133376
0.141248
0.149632
0.158464
0.167872
0.177856
0.188352
0.199552
0.211328
0.223872
0.23712
0.2512
0.266048
0.281856
0.29856
0.316224
0.334976
0.354816
0.375808

0.39808
0.421696
0.446656
0.473152
0.501184
0.53088
0.562368
0.595648
0.630976
0.668352
0.707968
0.749888
0.794304
0.841408
0.891264
0.944064
1
1.05926
1.12205
1.18848
1.25894
1.3335
1.41254
1.49626
1.5849
1.67878
1.7783
1.88365
1.99526
2.11347
2.23872
2.37139
2.51187
2.66074
2.81837
2.98541
3.1623
3.34963
3.54816
3.7584
3.98106
4.21696
4.46682
4.73152
5.0119
5.30886
5.62342
5.95661
6.30957
6.68346
7.07949
7.49894
7.9433
8.41395
8.91251
9.44064
10
10.5925
11.2202
11.885
12.5892
13.3352
14.1254
14.9624
15.849
16.788
17.7828
18.8365
19.9526
21.1349
22.3872
23.7137
25.1188
26.6072
28.1838
29.8538
31.6228
33.4965
35.4813
37.5837
39.8107

D.4 Dynamic Range Control

Index	Q18 binary	Multiplier	Log Multiplier (dB)
0	0,00040394	0,0259	-31,7500
1	0,00041574	0,0266	-31,5000
2	0,00042788	0,0274	-31,2500
3	0,00044037	0,0282	-31,0000
4	0,00045323	0,0290	-30,7500
5	0,00046647	0,0299	-30,5000
6	0,00048009	0,0307	-30,2500
7	0,00049411	0,0316	-30,0000
8	0,00050853	0,0325	-29,7500
9	0,00052338	0,0335	-29,5000
10	0,00053867	0,0345	-29,2500
11	0,00055440	0,0355	-29,0000
12	0,00057058	0,0365	-28,7500
13	0,00058725	0,0376	-28,5000
14	0,00060439	0,0387	-28,2500
15	0,00062204	0,0398	-28,0000
16	0,00064021	0,0410	-27,7500
17	0,00065890	0,0422	-27,5000
18	0,00067814	0,0434	-27,2500
19	0,00069794	0,0447	-27,0000
20	0,00071832	0,0460	-26,7500
21	0,00073930	0,0473	-26,5000
22	0,00076089	0,0487	-26,2500
23	0,00078311	0,0501	-26,0000
24	0,00080597	0,0516	-25,7500
25	0,00082951	0,0531	-25,5000
26	0,00085373	0,0546	-25,2500
27	0,00087866	0,0562	-25,0000
28	0,00090432	0,0579	-24,7500
29	0,00093072	0,0596	-24,5000
30	0,00095790	0,0613	-24,2500
31	0,00098587	0,0631	-24,0000
32	0,00101466	0,0649	-23,7500
33	0,00104429	0,0668	-23,5000
34	0,00107478	0,0688	-23,2500
35	0,00110617	0,0708	-23,0000
36	0,00113847	0,0729	-22,7500
37	0,00117171	0,0750	-22,5000
38	0,00120592	0,0772	-22,2500
39	0,00124114	0,0794	-22,0000
40	0,00127738	0,0818	-21,7500
41	0,00131468	0,0841	-21,5000
42	0,00135307	0,0866	-21,2500
43	0,00139258	0,0891	-21,0000
44	0,00143324	0,0917	-20,7500
45	0,00147510	0,0944	-20,5000
46	0,00151817	0,0972	-20,2500
47	0,00156250	0,1000	-20,0000
48	0,00160813	0,1029	-19,7500
49	0,00165508	0,1059	-19,5000
50	0,00170341	0,1090	-19,2500
51	0,00175315	0,1122	-19,0000
52	0,00180435	0,1155	-18,7500
53	0,00185703	0,1189	-18,5000
54	0,00191126	0,1223	-18,2500
55	0,00196707	0,1259	-18,0000
56	0,00202451	0,1296	-17,7500
57	0,00208363	0,1334	-17,5000
58	0,00214447	0,1372	-17,2500
59	0,00220709	0,1413	-17,0000
60	0,00227154	0,1454	-16,7500
61	0,00233787	0,1496	-16,5000
62	0,00240614	0,1540	-16,2500

Index	Q18 binary	Multiplier	Log Multiplier (dB)
63	0,00247640	0,1585	-16,0000
64	0,00254871	0,1631	-15,7500
65	0,00262313	0,1679	-15,5000
66	0,00269973	0,1728	-15,2500
67	0,00277856	0,1778	-15,0000
68	0,00285970	0,1830	-14,7500
69	0,00294320	0,1884	-14,5000
70	0,00302914	0,1939	-14,2500
71	0,00311760	0,1995	-14,0000
72	0,00320863	0,2054	-13,7500
73	0,00330233	0,2113	-13,5000
74	0,00339876	0,2175	-13,2500
75	0,00349800	0,2239	-13,0000
76	0,00360015	0,2304	-12,7500
77	0,00370527	0,2371	-12,5000
78	0,00381347	0,2441	-12,2500
79	0,00392482	0,2512	-12,0000
80	0,00403943	0,2585	-11,7500
81	0,00415738	0,2661	-11,5000
82	0,00427878	0,2738	-11,2500
83	0,00440372	0,2818	-11,0000
84	0,00453231	0,2901	-10,7500
85	0,00466466	0,2985	-10,5000
86	0,00480087	0,3073	-10,2500
87	0,00494106	0,3162	-10,0000
88	0,00508534	0,3255	-9,7500
89	0,00523383	0,3350	-9,5000
90	0,00538667	0,3447	-9,2500
91	0,00554396	0,3548	-9,0000
92	0,00570585	0,3652	-8,7500
93	0,00587246	0,3758	-8,5000
94	0,00604394	0,3868	-8,2500
95	0,00622042	0,3981	-8,0000
96	0,00640206	0,4097	-7,7500
97	0,00658901	0,4217	-7,5000
98	0,00678141	0,4340	-7,2500
99	0,00697943	0,4467	-7,0000
100	0,00718323	0,4597	-6,7500
101	0,00739299	0,4732	-6,5000
102	0,00760887	0,4870	-6,2500
103	0,00783105	0,5012	-6,0000
104	0,00805972	0,5158	-5,7500
105	0,00829507	0,5309	-5,5000
106	0,00853729	0,5464	-5,2500
107	0,00878658	0,5623	-5,0000
108	0,00904316	0,5788	-4,7500
109	0,00930722	0,5957	-4,5000
110	0,00957900	0,6131	-4,2500
111	0,00985871	0,6310	-4,0000
112	0,01014659	0,6494	-3,7500
113	0,01044287	0,6683	-3,5000
114	0,01074781	0,6879	-3,2500
115	0,01106165	0,7079	-3,0000
116	0,01138466	0,7286	-2,7500
117	0,01171710	0,7499	-2,5000
118	0,01205924	0,7718	-2,2500
119	0,01241138	0,7943	-2,0000
120	0,01277380	0,8175	-1,7500
121	0,01314680	0,8414	-1,5000
122	0,01353069	0,8660	-1,2500
123	0,01392580	0,8913	-1,0000
124	0,01433244	0,9173	-0,7500
125	0,01475095	0,9441	-0,5000
126	0,01518169	0,9716	-0,2500
127	0,01562500	1,0000	0,0000
128	0,01608126	1,0292	0,2500
129	0,01655084	1,0593	0,5000
130	0,01703413	1,0902	0,7500

Index	Q18 binary	Multiplier	Log Multiplier (dB)
131	0,01753154	1,1220	1,0000
132	0,01804347	1,1548	1,2500
133	0,01857035	1,1885	1,5000
134	0,01911261	1,2232	1,7500
135	0,01967071	1,2589	2,0000
136	0,02024510	1,2957	2,2500
137	0,02083627	1,3335	2,5000
138	0,02144470	1,3725	2,7500
139	0,02207090	1,4125	3,0000
140	0,02271538	1,4538	3,2500
141	0,02337868	1,4962	3,5000
142	0,02406135	1,5399	3,7500
143	0,02476396	1,5849	4,0000
144	0,02548708	1,6312	4,2500
145	0,02623131	1,6788	4,5000
146	0,02699728	1,7278	4,7500
147	0,02778562	1,7783	5,0000
148	0,02859697	1,8302	5,2500
149	0,02943202	1,8836	5,5000
150	0,03029145	1,9387	5,7500
151	0,03117597	1,9953	6,0000
152	0,03208633	2,0535	6,2500
153	0,03302327	2,1135	6,5000
154	0,03398756	2,1752	6,7500
155	0,03498002	2,2387	7,0000
156	0,03600145	2,3041	7,2500
157	0,03705271	2,3714	7,5000
158	0,03813467	2,4406	7,7500
159	0,03924823	2,5119	8,0000
160	0,04039429	2,5852	8,2500
161	0,04157383	2,6607	8,5000
162	0,04278781	2,7384	8,7500
163	0,04403723	2,8184	9,0000
164	0,04532314	2,9007	9,2500
165	0,04664660	2,9854	9,5000
166	0,04800871	3,0726	9,7500
167	0,04941059	3,1623	10,0000
168	0,05085340	3,2546	10,2500
169	0,05233835	3,3497	10,5000
170	0,05386666	3,4475	10,7500
171	0,05543959	3,5481	11,0000
172	0,05705846	3,6517	11,2500
173	0,05872459	3,7584	11,5000
174	0,06043938	3,8681	11,7500
175	0,06220425	3,9811	12,0000
176	0,06402064	4,0973	12,2500
177	0,06589008	4,2170	12,5000
178	0,06781410	4,3401	12,7500
179	0,06979431	4,4668	13,0000
180	0,07183234	4,5973	13,2500
181	0,07392988	4,7315	13,5000
182	0,07608868	4,8697	13,7500
183	0,07831051	5,0119	14,0000
184	0,08059721	5,1582	14,2500
185	0,08295069	5,3088	14,5000
186	0,08537290	5,4639	14,7500
187	0,08786583	5,6234	15,0000
188	0,09043156	5,7876	15,2500
189	0,09307221	5,9566	15,5000
190	0,09578997	6,1306	15,7500
191	0,09858709	6,3096	16,0000
192	0,10146588	6,4938	16,2500
193	0,10442874	6,6834	16,5000
194	0,10747811	6,8786	16,7500
195	0,11061653	7,0795	17,0000
196	0,11384659	7,2862	17,2500
197	0,11717097	7,4989	17,5000
198	0,12059242	7,7179	17,7500

Index	**Q18 binary**	**Multiplier**	**Log Multiplier (dB)**
199	0,12411379	7,9433	18,0000
200	0,12773797	8,1752	18,2500
201	0,13146799	8,4140	18,5000
202	0,13530693	8,6596	18,7500
203	0,13925796	8,9125	19,0000
204	0,14332436	9,1728	19,2500
205	0,14750951	9,4406	19,5000
206	0,15181687	9,7163	19,7500
207	0,15625000	10,0000	20,0000
208	0,16081258	10,2920	20,2500
209	0,16550839	10,5925	20,5000
210	0,17034133	10,9018	20,7500
211	0,17531538	11,2202	21,0000
212	0,18043469	11,5478	21,2500
213	0,18570347	11,8850	21,5000
214	0,19112611	12,2321	21,7500
215	0,19670710	12,5893	22,0000
216	0,20245105	12,9569	22,2500
217	0,20836272	13,3352	22,5000
218	0,21444703	13,7246	22,7500
219	0,22070899	14,1254	23,0000
220	0,22715381	14,5378	23,2500
221	0,23378682	14,9624	23,5000
222	0,24061352	15,3993	23,7500
223	0,24763956	15,8489	24,0000
224	0,25487077	16,3117	24,2500
225	0,26231313	16,7880	24,5000
226	0,26997281	17,2783	24,7500
227	0,27785616	17,7828	25,0000
228	0,28596970	18,3021	25,2500
229	0,29432017	18,8365	25,5000
230	0,30291447	19,3865	25,7500
231	0,31175974	19,9526	26,0000
232	0,32086329	20,5353	26,2500
233	0,33023266	21,1349	26,5000
234	0,33987563	21,7520	26,7500
235	0,34980018	22,3872	27,0000
236	0,36001453	23,0409	27,2500
237	0,37052714	23,7137	27,5000
238	0,38134673	24,4062	27,7500
239	0,39248225	25,1189	28,0000
240	0,40394294	25,8523	28,2500
241	0,41573829	26,6073	28,5000
242	0,42787807	27,3842	28,7500
243	0,44037233	28,1838	29,0000
244	0,45323144	29,0068	29,2500
245	0,46646603	29,8538	29,5000
246	0,48008709	30,7256	29,7500
247	0,49410588	31,6228	30,0000
248	0,50853404	32,5462	30,2500
249	0,52338350	33,4965	30,5000
250	0,53866657	34,4747	30,7500
251	0,55439592	35,4813	31,0000
252	0,57058457	36,5174	31,2500
253	0,58724594	37,5837	31,5000
254	0,60439384	38,6812	31,7500
255	0,62204245	39,8107	32,0000

D.5 Huffman Code Books

D.5.1 3 Levels

Table A.3

Quantization level	Code length	Code
0	1	0
1	2	2
-1	2	3

D.5.2 4 Levels (For TMODE)

Table A.4

Quantization level	Code length	Code
0	1	0
1	2	2
2	3	6
3	3	7

Table B.4

Quantization level	Code length	Code
0	2	2
1	3	6
2	3	7
3	1	0

Table C.4

Quantization level	Code length	Code
0	3	6
1	3	7
2	1	0
3	2	2

Table D.4

Quantization level	Code length	Code
0	2	0
1	2	1
2	2	2
3	2	3

D.5.3 5 Levels

Table A.5

Quantization level	Code length	Code
0	1	0
1	2	2
-1	3	6
2	4	14
-2	4	15

Table B.5

Quantization level	Code length	Code
0	2	2
1	2	0
-1	2	1
2	3	6
-2	3	7

Table C.5

Quantization level	Code length	Code
0	1	0
1	3	4
-1	3	5
2	3	6
-2	3	7

D.5.4 7 Levels

Table A.7

Quantization level	Code length	Code
0	1	0
1	3	6
-1	3	5
2	3	4
-2	4	14
3	5	31
-3	5	30

Table B.7

Quantization level	Code length	Code
0	2	3
1	2	1
-1	2	0
2	3	4
-2	4	11
3	5	21
-3	5	20

Table C.7

Quantization level	Code length	Code
0	2	3
1	2	2
-1	2	1
2	4	3
-2	4	2
3	4	1
-3	4	0

D.5.5 9 Levels

Table A.9

Quantization level	Code length	Code
0	1	0
1	3	7
-1	3	5
2	4	13
-2	4	9
3	4	8
-3	5	25
4	6	49
-4	6	48

Table B.9

Quantization level	Code length	Code
0	2	2
1	2	0
-1	3	7
2	3	3
-2	3	2
3	5	27
-3	5	26
4	5	25
-4	5	24

Table C.9

Quantization level	Code length	Code
0	2	2
1	2	0
-1	3	7
2	3	6
-2	3	2
3	4	6
-3	5	15
4	6	29
-4	6	28

D.5.6 12 Levels (for BHUFF)

Table A.12

ABITS	Code length	Code
1	1	0
2	2	2
3	3	6
4	4	14
5	5	30
6	6	62
7	8	255
8	8	254
9	9	507
10	9	506
11	9	505
12	9	504

Table B.12

ABITS	Code length	Code
1	1	1
2	2	0
3	3	2
4	5	15
5	5	12
6	6	29
7	7	57
8	7	56
9	7	55
10	7	54
11	7	53
12	7	52

Table C.12

ABITS	Code length	Code
1	2	0
2	3	7
3	3	5
4	3	4
5	3	2
6	4	13
7	4	12
8	4	6
9	5	15
10	6	29
11	7	57
12	7	56

Table D.12

ABITS	Code length	Code
1	2	3
2	2	2
3	2	0
4	3	2
5	4	6
6	5	14
7	6	30
8	7	62
9	8	126
10	9	254
11	10	511
12	10	510

Table E.12

ABITS	Code length	Code
1	1	1
2	2	0
3	3	2
4	4	6
5	5	14
6	7	63
7	7	61
8	8	124
9	8	121
10	8	120
11	9	251
12	9	250

D.5.7 13 Levels

Table A.13

Quantization level	Code length	Code
0	1	0
1	3	4
-1	4	15
2	4	13
-2	4	12
3	4	10
-3	5	29
4	5	22
-4	6	57
5	6	47
-5	6	46
6	7	113
-6	7	112

Table B.13

Quantization level	Code length	Code
0	2	0
1	3	6
-1	3	5
2	3	2
-2	4	15
3	4	9
-3	4	7
4	4	6
-4	5	29
5	5	17
-5	5	16
6	6	57
-6	6	56

Table C.13

Quantization level	Code length	Code
0	3	5
1	3	4
-1	3	3
2	3	2
-2	3	0
3	4	15
-3	4	14
4	4	12
-4	4	3
5	5	27
-5	5	26
6	5	5
-6	5	4

D.5.8 17 Levels

Table A.17

Quantization level	Code length	Code
0	2	1
1	3	7
-1	3	6
2	3	4
-2	3	1
3	4	11
-3	4	10
4	4	0
-4	5	3
5	6	4
-5	7	11
6	8	20
-6	9	43
7	10	84
-7	11	171
8	12	341
-8	12	340

Table B.17

Quantization level	Code length	Code
0	2	0
1	3	6
-1	3	5
2	3	2
-2	4	15
3	4	9
-3	4	8
4	5	29
-4	5	28
5	5	14
-5	5	13
6	6	30
-6	6	25
7	6	24
-7	7	63
8	8	125
-8	8	124

Table C.17

Quantization level	Code length	Code
0	3	6
1	3	4
-1	3	3
2	3	0
-2	4	15
3	4	11
-3	4	10
4	4	4
-4	4	3
5	5	29
-5	5	28
6	5	10
-6	5	5
7	5	4
-7	6	23
8	7	45
-8	7	44

Table D.17

Quantization level	Code length	Code
0	1	0
1	3	7
-1	3	6
2	4	11
-2	4	10
3	5	19
-3	5	18
4	6	35
-4	6	34
5	7	67
-5	7	66
6	8	131
-6	8	130
7	9	259
-7	9	258
8	9	257
-8	9	256

Table E.17

Quantization level	Code length	Code
0	1	0
1	3	5
-1	3	4
2	4	12
-2	5	31
3	5	28
-3	5	27
4	6	60
-4	6	59
5	6	53
-5	6	52
6	7	122
-6	7	117
7	8	247
-7	8	246
8	8	233
-8	8	232

Table F.17

Quantization level	Code length	Code
0	3	6
1	3	5
-1	3	4
2	3	2
-2	3	1
3	4	15
-3	4	14
4	4	6
-4	4	1
5	5	14
-5	5	1
6	6	31
-6	6	30
7	6	0
-7	7	3
8	8	5
-8	8	4

Table G.17

Quantization level	Code length	Code
0	2	2
1	3	7
-1	3	6
2	3	1
-2	3	0
3	4	5
-3	4	4
4	5	14
-4	5	13
5	6	30
-5	6	25
6	7	62
-6	7	49
7	8	127
-7	8	126
8	8	97
-8	8	96

D.5.9 25 Levels

Table A.25

Quantization level	Code length	Code
0	3	6
1	3	4
-1	3	3
2	3	1
-2	3	0
3	4	15
-3	4	14
4	4	5
-4	4	4
5	5	22
-5	5	21
6	6	47
-6	6	46
7	7	83
-7	7	82
8	8	163
-8	8	162
9	8	160
-9	9	323
10	10	644
-10	11	1 291
11	12	2 580
-11	13	5 163
12	14	10 325
-12	14	10 324

Table B.25

Quantization level	Code length	Code
0	3	5
1	3	2
-1	3	1
2	4	15
-2	4	14
3	4	9
-3	4	8
4	4	6
-4	4	1
5	5	26
-5	5	25
6	5	15
-6	5	14
7	6	55
-7	6	54
8	6	49
-8	6	48
9	6	1
-9	6	0
10	7	6
-10	7	5
11	7	4
-11	8	15
12	9	29
-12	9	28

Table C.25

Quantization level	Code length	Code
0	3	1
1	4	15
-1	4	14
2	4	12
-2	4	11
3	4	9
-3	4	8
4	4	6
-4	4	5
5	4	1
-5	4	0
6	5	26
-6	5	21
7	5	15
-7	5	14
8	5	8
-8	6	55
9	6	41
-9	6	40
10	6	18
-10	7	109
11	7	108
-11	7	39
12	8	77
-12	8	76

Table D.25

Quantization level	Code length	Code
0	2	2
1	3	7
-1	3	6
2	3	1
-2	3	0
3	4	5
-3	4	4
4	5	13
-4	5	12
5	6	29
-5	6	28
6	7	62
-6	7	61
7	8	126
-7	8	121
8	9	255
-8	9	254
9	10	483
-9	10	482
10	11	963
-10	11	962
11	12	1 923
-11	12	1 922
12	12	1 921
-12	12	1 920

Table E.25

Quantization level	Code length	Code
0	2	3
1	3	3
-1	3	2
2	4	11
-2	4	10
3	4	1
-3	4	0
4	5	17
-4	5	16
5	5	5
-5	5	4
6	6	38
-6	6	37
7	6	14
-7	6	13
8	7	79
-8	7	78
9	7	72
-9	7	31
10	7	25
-10	7	24
11	8	147
-11	8	146
12	8	61
-12	8	60

Table F.25

Quantization level	Code length	Code
0	3	1
1	3	0
-1	4	15
2	4	14
-2	4	13
3	4	11
-3	4	10
4	4	8
-4	4	7
5	4	5
-5	4	4
6	5	24
-6	5	19
7	5	13
-7	5	12
8	6	37
-8	6	36
9	7	102
-9	7	101
10	8	207
-10	8	206
11	8	200
-11	9	403
12	10	805
-12	10	804

Table G.25

Quantization level	Code length	Code
0	2	1
1	3	6
-1	3	5
2	3	0
-2	4	15
3	4	8
-3	4	3
4	5	28
-4	5	19
5	5	4
-5	6	59
6	6	36
-6	6	11
7	7	116
-7	7	75
8	7	21
-8	7	20
9	8	149
-9	8	148
10	9	470
-10	9	469
11	10	943
-11	10	942
12	10	937
-12	10	936

D.5.10 33 Levels

Table A.33

Quantization level	Code length	Code
0	3	2
1	3	1
-1	3	0
2	4	14
-2	4	13
3	4	12
-3	4	11
4	4	9
-4	4	8
5	4	6
-5	5	31
6	5	20
-6	5	15
7	6	61
-7	6	60
8	6	29
-8	6	28
9	7	85
-9	7	84
10	8	174
-10	8	173
11	9	351
-11	9	350
12	10	691
-12	10	690
13	11	1 379
-13	11	1 378
14	12	2 755
-14	12	2 754
15	13	5 507
-15	13	5 506
16	13	5 505
-16	13	5 504

Table B.33

Quantization level	Code length	Code
0	3	1
1	4	15
-1	4	14
2	4	11
-2	4	10
3	4	8
-3	4	7
4	4	4
-4	4	1
5	5	27
-5	5	26
6	5	19
-6	5	18
7	5	12
-7	5	11
8	5	1
-8	5	0
9	6	50
-9	6	49
10	6	26
-10	6	21
11	7	103
-11	7	102
12	7	96
-12	7	55
13	7	41
-13	7	40
14	8	194
-14	8	109
15	8	108
-15	9	391
16	10	781
-16	10	780

Table C.33

Quantization level	Code length	Code
0	4	13
1	4	11
-1	4	10
2	4	8
-2	4	7
3	4	4
-3	4	3
4	4	2
-4	4	1
5	5	30
-5	5	29
6	5	25
-6	5	24
7	5	19
-7	5	18
8	5	11
-8	5	10
9	5	0
-9	6	63
10	6	62
-10	6	57
11	6	27
-11	6	26
12	6	24
-12	6	3
13	7	113
-13	7	112
14	7	50
-14	7	5
15	7	4
-15	8	103
16	9	205
-16	9	204

Table D.33

Quantization level	Code length	Code
0	2	1
1	3	6
-1	3	5
2	3	0
-2	4	15
3	4	8
-3	4	3
4	5	28
-4	5	19
5	5	4
-5	6	59
6	6	36
-6	6	11
7	7	116
-7	7	75
8	7	21
-8	7	20
9	8	149
-9	8	148
10	9	469
-10	9	468
11	10	941
-11	10	940
12	11	1 885
-12	11	1 884
13	12	3 773
-13	12	3 772
14	13	7 551
-14	13	7 550
15	14	15 099
-15	14	15 098
16	14	15 097
-16	14	15 096

Table E.33

Quantization level	Code length	Code
0	2	2
1	3	2
-1	3	1
2	4	12
-2	4	7
3	4	0
-3	5	31
4	5	27
-4	5	26
5	5	3
-5	5	2
6	6	59
-6	6	58
7	6	27
-7	6	26
8	7	123
-8	7	122
9	7	120
-9	7	115
10	7	112
-10	7	51
11	7	49
-11	7	48
12	8	242
-12	8	229
13	8	227
-13	8	226
14	8	101
-14	8	100
15	9	487
-15	9	486
16	9	457
-16	9	456

Table F.33

Quantization level	Code length	Code
0	4	13
1	4	12
-1	4	11
2	4	9
-2	4	8
3	4	7
-3	4	6
4	4	4
-4	4	3
5	4	1
-5	4	0
6	5	30
-6	5	29
7	5	21
-7	5	20
8	5	10
-8	5	5
9	6	63
-9	6	62
10	6	56
-10	6	23
11	6	9
-11	6	8
12	7	45
-12	7	44
13	8	230
-13	8	229
14	9	463
-14	9	462
15	9	456
-15	10	915
16	11	1 829
-16	11	1 828

Table G.33

Quantization level	Code length	Code
0	3	6
1	3	3
-1	3	2
2	4	15
-2	4	14
3	4	9
-3	4	8
4	4	1
-4	4	0
5	5	22
-5	5	21
6	5	6
-6	5	5
7	6	46
-7	6	41
8	6	14
-8	6	9
9	7	94
-9	7	81
10	7	30
-10	7	17
11	8	191
-11	8	190
12	8	63
-12	8	62
13	8	32
-13	9	323
14	9	321
-14	9	320
15	9	67
-15	9	66
16	10	645
-16	10	644

D.5.11 65 Levels

Table A.65

Quantization level	Code length	Code
0	4	6
1	4	5
-1	4	4
2	4	2
-2	4	1
3	4	0
-3	5	31
4	5	29
-4	5	28
5	5	27
-5	5	26
6	5	24
-6	5	23
7	5	21
-7	5	20
8	5	18
-8	5	17
9	5	14
-9	5	7
10	5	6
-10	6	61
11	6	50
-11	6	45
12	6	38
-12	6	33
13	6	31
-13	6	30
14	7	120
-14	7	103
15	7	89
-15	7	88
16	7	65
-16	7	64
17	8	205
-17	8	204
18	8	157
-18	8	156
19	9	486
-19	9	485
20	9	318
-20	9	317
21	10	975
-21	10	974
22	10	639
-22	10	638
23	11	1 939
-23	11	1 938
24	11	1 936
-24	11	1 267
25	11	1 264
-25	12	3 875
26	12	2 532
-26	12	2 531
27	13	7 749
-27	13	7 748
28	13	5 061
-28	13	5 060
29	14	10 133
-29	14	10 132
30	15	20 269
-30	15	20 268
31	16	40 543
-31	16	40 542

Quantization level	Code length	Code
32	16	40 541
-32	16	40 540

Table B.65

Quantization level	Code length	Code
0	4	4
1	4	2
-1	4	1
2	5	30
-2	5	29
3	5	26
-3	5	25
4	5	23
-4	5	22
5	5	19
-5	5	18
6	5	16
-6	5	15
7	5	12
-7	5	11
8	5	7
-8	5	6
9	6	63
-9	6	62
10	6	56
-10	6	55
11	6	49
-11	6	48
12	6	41
-12	6	40
13	6	34
-13	6	29
14	6	26
-14	6	21
15	6	20
-15	6	3
16	6	0
-16	7	115
17	7	109
-17	7	108
18	7	86
-18	7	85
19	7	70
-19	7	57
20	7	56
-20	7	55
21	7	4
-21	7	3
22	8	229
-22	8	228
23	8	175
-23	8	174
24	8	143
-24	8	142
25	8	108
-25	8	11
26	8	10
-26	8	5
27	9	339
-27	9	338
28	9	336
-28	9	219
29	9	9
-29	9	8
30	10	674

Quantization level	Code length	Code
-30	10	437
31	10	436
-31	11	1 351
32	12	2 701
-32	12	2 700

Table C.65

Quantization level	Code length	Code
0	5	28
1	5	25
-1	5	24
2	5	23
-2	5	22
3	5	19
-3	5	18
4	5	16
-4	5	15
5	5	13
-5	5	12
6	5	10
-6	5	9
7	5	7
-7	5	6
8	5	4
-8	5	3
9	5	1
-9	5	0
10	6	62
-10	6	61
11	6	59
-11	6	58
12	6	54
-12	6	53
13	6	43
-13	6	42
14	6	40
-14	6	35
15	6	29
-15	6	28
16	6	17
-16	6	16
17	6	11
-17	6	10
18	6	4
-18	7	127
19	7	121
-19	7	120
20	7	110
-20	7	105
21	7	83
-21	7	82
22	7	68
-22	7	47
23	7	46
-23	7	45
24	7	11
-24	7	10
25	8	252
-25	8	223
26	8	209
-26	8	208
27	8	138
-27	8	89
28	8	88
-28	9	507

Quantization level	Code length	Code
29	9	445
-29	9	444
30	9	278
-30	10	1 013
31	10	1 012
-31	10	559
32	11	1 117
-32	11	1 116

Table D.65

Quantization level	Code length	Code
0	3	4
1	3	1
-1	3	0
2	4	13
-2	4	12
3	4	7
-3	4	6
4	5	31
-4	5	30
5	5	23
-5	5	22
6	5	11
-6	5	10
7	6	59
-7	6	58
8	6	43
-8	6	42
9	6	19
-9	6	18
10	7	115
-10	7	114
11	7	83
-11	7	82
12	7	35
-12	7	34
13	8	227
-13	8	226
14	8	163
-14	8	162
15	8	160
-15	8	67
16	8	64
-16	9	451
17	9	448
-17	9	323
18	9	132
-18	9	131
19	10	900
-19	10	899
20	10	644
-20	10	267
21	10	261
-21	10	260
22	11	1 797
-22	11	1 796
23	11	533
-23	11	532
24	12	3 605
-24	12	3 604
25	12	2 582
-25	12	2 581
26	13	7 215
-26	13	7 214
27	13	5 167

Quantization level	Code length	Code
-27	13	5 166
28	13	5 160
-28	14	14 427
29	14	10 323
-29	14	10 322
30	15	28 853
-30	15	28 852
31	15	28 851
-31	15	28 850
32	15	28 849
-32	15	28 848

Table E.65

Quantization level	Code length	Code
0	3	4
1	3	0
-1	4	15
2	4	7
-2	4	6
3	5	29
-3	5	28
4	5	23
-4	5	22
5	5	10
-5	5	9
6	5	6
-6	5	5
7	6	54
-7	6	53
8	6	48
-8	6	43
9	6	40
-9	6	23
10	6	16
-10	6	15
11	6	9
-11	6	8
12	7	105
-12	7	104
13	7	100
-13	7	99
14	7	84
-14	7	83
15	7	45
-15	7	44
16	7	29
-16	7	28
17	8	221
-17	8	220
18	8	206
-18	8	205
19	8	202
-19	8	197
20	8	171
-20	8	170
21	8	164
-21	8	71
22	8	69
-22	8	68
23	9	446
-23	9	445
24	9	415
-24	9	414
25	9	408
-25	9	407

Quantization level	Code length	Code
26	9	393
-26	9	392
27	9	331
-27	9	330
28	9	141
-28	9	140
29	10	895
-29	10	894
30	10	889
-30	10	888
31	10	819
-31	10	818
32	10	813
-32	10	812

Table F.65

Quantization level	Code length	Code
0	3	6
1	3	3
-1	3	2
2	4	15
-2	4	14
3	4	9
-3	4	8
4	4	1
-4	4	0
5	5	21
-5	5	20
6	5	5
-6	5	4
7	6	45
-7	6	44
8	6	13
-8	6	12
9	7	93
-9	7	92
10	7	29
-10	7	28
11	8	189
-11	8	188
12	8	61
-12	8	60
13	9	381
-13	9	380
14	9	125
-14	9	124
15	10	765
-15	10	764
16	10	252
-16	11	1 535
17	11	1 532
-17	11	511
18	11	506
-18	12	3 069
19	12	3 067
-19	12	3 066
20	12	1 015
-20	12	1 014
21	13	6 136
-21	13	2 043
22	13	2 035
-22	13	2 034
23	14	12 275
-23	14	12 274

Quantization level	Code length	Code
24	14	4 085
-24	14	4 084
25	14	4 083
-25	14	4 082
26	14	4 081
-26	14	4 080
27	14	4 079
-27	14	4 078
28	14	4 077
-28	14	4 076
29	14	4 075
-29	14	4 074
30	14	4 073
-30	14	4 072
31	14	4 067
-31	14	4 066
32	14	4 065
-32	14	4 064

Table G.65

Quantization level	Code length	Code
0	4	14
1	4	11
-1	4	10
2	4	8
-2	4	6
3	4	4
-3	4	3
4	4	0
-4	5	31
5	5	26
-5	5	25
6	5	18
-6	5	15
7	5	10
-7	5	5
8	5	2
-8	6	61
9	6	54
-9	6	49
10	6	38
-10	6	29
11	6	22
-11	6	9
12	6	6
-12	7	121
13	7	110
-13	7	97
14	7	78
-14	7	57
15	7	46
-15	7	17
16	7	14
-16	8	241
17	8	223
-17	8	222
18	8	159
-18	8	158
19	8	95
-19	8	94
20	8	31
-20	8	30
21	9	480
-21	9	387
22	9	384

Quantization level	Code length	Code
-22	9	227
23	9	225
-23	9	224
24	9	65
-24	9	64
25	10	962
-25	10	773
26	10	771
-26	10	770
27	10	452
-27	10	135
28	10	133
-28	10	132
29	11	1 927
-29	11	1 926
30	11	1 545
-30	11	1 544
31	11	907
-31	11	906
32	11	269
-32	11	268

D.5.12 129 Levels

Table SA.129

Quantization level	Code length	Code
0	2	1
1	3	6
-1	3	5
2	3	0
-2	4	15
3	4	8
-3	4	3
4	5	28
-4	5	19
5	5	4
-5	6	59
6	6	36
-6	6	11
7	7	75
-7	7	74
8	8	233
-8	8	232
9	8	41
-9	8	40
10	9	87
-10	9	86
11	10	937
-11	10	936
12	11	1 877
-12	11	1 876
13	11	341
-13	11	340
14	12	686
-14	12	685
15	13	1 375
-15	13	1 374
16	13	1 369
-16	13	1 368
17	13	1 359
-17	13	1 358
18	13	1 357
-18	13	1 356
19	13	1 355

Quantization level	Code length	Code
-19	13	1 354
20	13	1 353
-20	13	1 352
21	13	1 351
-21	13	1 350
22	13	1 349
-22	13	1 348
23	13	1 347
-23	13	1 346
24	13	1 345
-24	13	1 344
25	14	15 103
-25	14	15 102
26	14	15 101
-26	14	15 100
27	14	15 099
-27	14	15 098
28	14	15 097
-28	14	15 096
29	14	15 095
-29	14	15 094
30	14	15 093
-30	14	15 092
31	14	15 091
-31	14	15 090
32	14	15 089
-32	14	15 088
33	14	15 087
-33	14	15 086
34	14	15 085
-34	14	15 084
35	14	15 083
-35	14	15 082
36	14	15 081
-36	14	15 080
37	14	15 079
-37	14	15 078
38	14	15 077
-38	14	15 076
39	14	15 075
-39	14	15 074
40	14	15 073
-40	14	15 072
41	14	15 071
-41	14	15 070
42	14	15 069
-42	14	15 068
43	14	15 067
-43	14	15 066
44	14	15 065
-44	14	15 064
45	14	15 063
-45	14	15 062
46	14	15 061
-46	14	15 060
47	14	15 059
-47	14	15 058
48	14	15 057
-48	14	15 056
49	14	15 055
-49	14	15 054
50	14	15 053
-50	14	15 052
51	14	15 051
-51	14	15 050

Quantization level	Code length	Code
52	14	15 049
-52	14	15 048
53	14	15 047
-53	14	15 046
54	14	15 045
-54	14	15 044
55	14	15 043
-55	14	15 042
56	14	15 041
-56	14	15 040
57	14	15 039
-57	14	15 038
58	14	15 037
-58	14	15 036
59	14	15 035
-59	14	15 034
60	14	15 033
-60	14	15 032
61	14	15 031
-61	14	15 030
62	14	15 029
-62	14	15 028
63	14	15 027
-63	14	15 026
64	14	15 025
-64	14	15 024

Table SB.129

Quantization level	Code length	Code
0	3	3
1	3	2
-1	3	1
2	4	15
-2	4	14
3	4	12
-3	4	11
4	4	10
-4	4	9
5	4	0
-5	5	27
6	5	17
-6	5	16
7	6	53
-7	6	52
8	6	5
-8	6	4
9	7	13
-9	7	12
10	8	29
-10	8	28
11	9	60
-11	10	127
12	11	253
-12	11	252
13	12	491
-13	12	490
14	13	979
-14	13	978
15	14	1 955
-15	14	1 954
16	14	1 953
-16	14	1 952
17	15	4 031
-17	15	4 030
18	15	4 029
-18	15	4 028
19	15	4 027
-19	15	4 026
20	15	4 025
-20	15	4 024
21	15	4 023
-21	15	4 022
22	15	4 021
-22	15	4 020
23	15	4 019
-23	15	4 018
24	15	4 017
-24	15	4 016
25	15	4 015
-25	15	4 014
26	15	4 013
-26	15	4 012
27	15	4 011
-27	15	4 010
28	15	4 009
-28	15	4 008
29	15	4 007
-29	15	4 006
30	15	4 005
-30	15	4 004
31	15	4 003
-31	15	4 002
32	15	4 001
-32	15	4 000
33	15	3 999

Quantization level	Code length	Code
-33	15	3 998
34	15	3 997
-34	15	3 996
35	15	3 995
-35	15	3 994
36	15	3 993
-36	15	3 992
37	15	3 991
-37	15	3 990
38	15	3 989
-38	15	3 988
39	15	3 987
-39	15	3 986
40	15	3 985
-40	15	3 984
41	15	3 983
-41	15	3 982
42	15	3 981
-42	15	3 980
43	15	3 979
-43	15	3 978
44	15	3 977
-44	15	3 976
45	15	3 975
-45	15	3 974
46	15	3 973
-46	15	3 972
47	15	3 971
-47	15	3 970
48	15	3 969
-48	15	3 968
49	15	3 967
-49	15	3 966
50	15	3 965
-50	15	3 964
51	15	3 963
-51	15	3 962
52	15	3 961
-52	15	3 960
53	15	3 959
-53	15	3 958
54	15	3 957
-54	15	3 956
55	15	3 955
-55	15	3 954
56	15	3 953
-56	15	3 952
57	15	3 951
-57	15	3 950
58	15	3 949
-58	15	3 948
59	15	3 947
-59	15	3 946
60	15	3 945
-60	15	3 944
61	15	3 943
-61	15	3 942
62	15	3 941
-62	15	3 940
63	15	3 939
-63	15	3 938
64	15	3 937
-64	15	3 936

Table SC.129

Quantization level	Code length	Code
0	3	4
1	3	1
-1	3	0
2	4	13
-2	4	12
3	4	7
-3	4	6
4	5	31
-4	5	30
5	5	23
-5	5	22
6	5	11
-6	5	10
7	6	59
-7	6	58
8	6	43
-8	6	42
9	6	19
-9	6	18
10	7	115
-10	7	114
11	7	83
-11	7	82
12	7	35
-12	7	34
13	8	227
-13	8	226
14	8	162
-14	8	161
15	8	66
-15	8	65
16	9	450
-16	9	449
17	9	321
-17	9	320
18	9	129
-18	9	128
19	10	897
-19	10	896
20	10	652
-20	10	271
21	10	268
-21	11	1 807
22	11	1 308
-22	11	1 307
23	11	540
-23	11	539
24	12	3 612
-24	12	3 611
25	12	2 613
-25	12	2 612
26	12	1 077
-26	12	1 076
27	13	7 226
-27	13	7 221
28	13	2 167
-28	13	2 166
29	13	2 164
-29	14	14 455
30	14	14 441
-30	14	14 440
31	14	4 331
-31	14	4 330
32	15	28 909
-32	15	28 908
33	15	28 879

Quantization level	Code length	Code
-33	15	28 878
34	15	28 877
-34	15	28 876
35	15	28 875
-35	15	28 874
36	15	28 873
-36	15	28 872
37	15	28 871
-37	15	28 870
38	15	28 869
-38	15	28 868
39	15	28 867
-39	15	28 866
40	15	28 865
-40	15	28 864
41	15	20 991
-41	15	20 990
42	15	20 989
-42	15	20 988
43	15	20 987
-43	15	20 986
44	15	20 985
-44	15	20 984
45	15	20 983
-45	15	20 982
46	15	20 981
-46	15	20 980
47	15	20 979
-47	15	20 978
48	15	20 977
-48	15	20 976
49	15	20 975
-49	15	20 974
50	15	20 973
-50	15	20 972
51	15	20 971
-51	15	20 970
52	15	20 969
-52	15	20 968
53	15	20 967
-53	15	20 966
54	15	20 965
-54	15	20 964
55	15	20 963
-55	15	20 962
56	15	20 961
-56	15	20 960
57	15	20 959
-57	15	20 958
58	15	20 957
-58	15	20 956
59	15	20 955
-59	15	20 954
60	15	20 953
-60	15	20 952
61	15	20 951
-61	15	20 950
62	15	20 949
-62	15	20 948
63	15	20 947
-63	15	20 946
64	15	20 945
-64	15	20 944

Table SD.129

Quantization level	Code length	Code
0	2	0
1	3	5
-1	3	4
2	4	15
-2	4	14
3	4	7
-3	4	6
4	5	26
-4	5	25
5	5	10
-5	5	9
6	6	54
-6	6	49
7	6	22
-7	6	17
8	7	110
-8	7	97
9	7	46
-9	7	33
10	8	193
-10	8	192
11	8	65
-11	8	64
12	9	444
-12	9	191
13	9	188
-13	10	895
14	10	890
-14	10	381
15	10	378
-15	11	1 789
16	11	761
-16	11	760
17	12	3 577
-17	12	3 576
18	12	1 519
-18	12	1 518
19	12	1 516
-19	13	7 151
20	13	7 128
-20	13	3 035
21	14	14 301
-21	14	14 300
22	14	6 069
-22	14	6 068
23	15	28 599
-23	15	28 598
24	15	28 597
-24	15	28 596
25	15	28 595
-25	15	28 594
26	15	28 593
-26	15	28 592
27	15	28 591
-27	15	28 590
28	15	28 589
-28	15	28 588
29	15	28 587
-29	15	28 586
30	15	28 585
-30	15	28 584
31	15	28 583
-31	15	28 582
32	15	28 581
-32	15	28 580
33	15	28 579

Quantization level	Code length	Code
-33	15	28 578
34	15	28 577
-34	15	28 576
35	15	28 575
-35	15	28 574
36	15	28 573
-36	15	28 572
37	15	28 571
-37	15	28 570
38	15	28 569
-38	15	28 568
39	15	28 567
-39	15	28 566
40	15	28 565
-40	15	28 564
41	15	28 563
-41	15	28 562
42	15	28 561
-42	15	28 560
43	15	28 559
-43	15	28 558
44	15	28 557
-44	15	28 556
45	15	28 555
-45	15	28 554
46	15	28 553
-46	15	28 552
47	15	28 551
-47	15	28 550
48	15	28 549
-48	15	28 548
49	15	28 547
-49	15	28 546
50	15	28 545
-50	15	28 544
51	15	28 543
-51	15	28 542
52	15	28 541
-52	15	28 540
53	15	28 539
-53	15	28 538
54	15	28 537
-54	15	28 536
55	15	28 535
-55	15	28 534
56	15	28 533
-56	15	28 532
57	15	28 531
-57	15	28 530
58	15	28 529
-58	15	28 528
59	15	28 527
-59	15	28 526
60	15	28 525
-60	15	28 524
61	15	28 523
-61	15	28 522
62	15	28 521
-62	15	28 520
63	15	28 519
-63	15	28 518
64	15	28 517
-64	15	28 516

Table SE.129

Quantization level	Code length	Code
0	4	14
1	4	11
-1	4	10
2	4	7
-2	4	6
3	4	3
-3	4	2
4	5	31
-4	5	30
5	5	25
-5	5	24
6	5	17
-6	5	16
7	5	9
-7	5	8
8	5	1
-8	5	0
9	6	53
-9	6	52
10	6	37
-10	6	36
11	6	21
-11	6	20
12	6	5
-12	6	4
13	7	109
-13	7	108
14	7	77
-14	7	76
15	7	45
-15	7	44
16	7	13
-16	7	12
17	8	221
-17	8	220
18	8	157
-18	8	156
19	8	93
-19	8	92
20	8	29
-20	8	28
21	9	445
-21	9	444
22	9	317
-22	9	316
23	9	189
-23	9	188
24	9	61
-24	9	60
25	10	892
-25	10	639
26	10	637
-26	10	636
27	10	381
-27	10	380
28	10	125
-28	10	124
29	11	1 788
-29	11	1 787
30	11	1 276
-30	11	767
31	11	764
-31	11	255
32	11	252
-32	12	3 583
33	12	3 579

Quantization level	Code length	Code
-33	12	3 578
34	12	2 555
-34	12	2 554
35	12	1 531
-35	12	1 530
36	12	507
-36	12	506
37	13	7 160
-37	13	7 147
38	13	7 144
-38	13	3 067
39	13	3 065
-39	13	3 064
40	13	1 017
-40	13	1 016
41	14	14 330
-41	14	14 329
42	14	14 291
-42	14	14 290
43	14	6 132
-43	14	2 039
44	14	2 038
-44	14	2 037
45	15	28 663
-45	15	28 662
46	15	28 585
-46	15	28 584
47	15	12 267
-47	15	12 266
48	15	4 073
-48	15	4 072
49	16	57 315
-49	16	57 314
50	16	57 313
-50	16	57 312
51	16	57 311
-51	16	57 310
52	16	57 309
-52	16	57 308
53	16	57 307
-53	16	57 306
54	16	57 305
-54	16	57 304
55	16	57 303
-55	16	57 302
56	16	57 301
-56	16	57 300
57	16	57 299
-57	16	57 298
58	16	57 297
-58	16	57 296
59	16	57 295
-59	16	57 294
60	16	57 293
-60	16	57 292
61	16	57 291
-61	16	57 290
62	16	57 289
-62	16	57 288
63	16	57 175
-63	16	57 174
64	16	57 173
-64	16	57 172

Table A.129

Quantization level	Code length	Code
0	4	8
1	4	10
-1	4	9
2	4	0
-2	5	31
3	5	24
-3	5	23
4	5	12
-4	5	11
5	5	5
-5	5	4
6	6	60
-6	6	58
7	6	54
-7	6	53
8	6	45
-8	6	44
9	6	28
-9	6	27
10	6	19
-10	6	18
11	6	14
-11	6	13
12	6	6
-12	6	5
13	7	122
-13	7	119
14	7	113
-14	7	112
15	7	104
-15	7	103
16	7	100
-16	7	63
17	7	60
-17	7	59
18	7	52
-18	7	43
19	7	40
-19	7	35
20	7	32
-20	7	31
21	7	15
-21	7	14
22	8	247
-22	8	246
23	8	231
-23	8	230
24	8	223
-24	8	222
25	8	211
-25	8	210
26	8	203
-26	8	202
27	8	123
-27	8	122
28	8	116
-28	8	107
29	8	84
-29	8	83
30	8	68
-30	8	67
31	8	60
-31	8	51
32	8	49
-32	8	48
33	8	17

Quantization level	Code length	Code
-33	8	16
34	9	474
-34	9	473
35	9	458
-35	9	457
36	9	442
-36	9	441
37	9	411
-37	9	410
38	9	251
-38	9	250
39	9	248
-39	9	235
40	9	213
-40	9	212
41	9	170
-41	9	165
42	9	139
-42	9	138
43	9	132
-43	9	123
44	9	101
-44	9	100
45	9	37
-45	9	36
46	10	950
-46	10	945
47	10	919
-47	10	918
48	10	912
-48	10	887
49	10	881
-49	10	880
50	10	818
-50	10	817
51	10	499
-51	10	498
52	10	469
-52	10	468
53	10	343
-53	10	342
54	10	329
-54	10	328
55	10	267
-55	10	266
56	10	245
-56	10	244
57	10	79
-57	10	78
58	10	77
-58	10	76
59	11	1 903
-59	11	1 902
60	11	1 889
-60	11	1 888
61	11	1 827
-61	11	1 826
62	11	1 773
-62	11	1 772
63	11	1 639
-63	11	1 638
64	11	1 633
-64	11	1 632

Table B.129

Quantization level	Code length	Code
0	5	10
1	5	7
-1	5	6
2	5	4
-2	5	3
3	5	0
-3	6	63
4	6	60
-4	6	59
5	6	57
-5	6	56
6	6	53
-6	6	52
7	6	50
-7	6	49
8	6	46
-8	6	45
9	6	43
-9	6	42
10	6	39
-10	6	38
11	6	35
-11	6	34
12	6	32
-12	6	31
13	6	28
-13	6	27
14	6	25
-14	6	24
15	6	22
-15	6	19
16	6	16
-16	6	11
17	6	5
-17	6	4
18	7	125
-18	7	124
19	7	122
-19	7	117
20	7	110
-20	7	109
21	7	103
-21	7	102
22	7	96
-22	7	95
23	7	89
-23	7	88
24	7	81
-24	7	80
25	7	74
-25	7	73
26	7	66
-26	7	61
27	7	59
-27	7	58
28	7	52
-28	7	47
29	7	37
-29	7	36
30	7	21
-30	7	20
31	7	6
-31	7	5
32	8	247
-32	8	246
33	8	223

Quantization level	Code length	Code
-33	8	222
34	8	217
-34	8	216
35	8	189
-35	8	188
36	8	166
-36	8	165
37	8	151
-37	8	150
38	8	144
-38	8	135
39	8	121
-39	8	120
40	8	106
-40	8	93
41	8	71
-41	8	70
42	8	68
-42	8	15
43	8	9
-43	8	8
44	9	466
-44	9	465
45	9	391
-45	9	390
46	9	388
-46	9	335
47	9	329
-47	9	328
48	9	269
-48	9	268
49	9	215
-49	9	214
50	9	184
-50	9	139
51	9	29
-51	9	28
52	10	934
-52	10	929
53	10	779
-53	10	778
54	10	668
-54	10	583
55	10	582
-55	10	581
56	10	371
-56	10	370
57	10	276
-57	11	1 871
58	11	1 857
-58	11	1 856
59	11	1 338
-59	11	1 161
60	11	1 160
-60	11	555
61	12	3 741
-61	12	3 740
62	12	2 678
-62	12	1 109
63	12	1 108
-63	13	5 359
64	14	10 717
-64	14	10 716

Table C.129

Quantization level	Code length	Code
0	6	58
1	6	55
-1	6	54
2	6	52
-2	6	51
3	6	49
-3	6	48
4	6	46
-4	6	45
5	6	43
-5	6	42
6	6	40
-6	6	39
7	6	37
-7	6	36
8	6	34
-8	6	33
9	6	30
-9	6	29
10	6	27
-10	6	26
11	6	24
-11	6	23
12	6	21
-12	6	20
13	6	18
-13	6	17
14	6	14
-14	6	13
15	6	12
-15	6	11
16	6	8
-16	6	7
17	6	6
-17	6	5
18	6	3
-18	6	2
19	7	127
-19	7	126
20	7	124
-20	7	123
21	7	121
-21	7	120
22	7	118
-22	7	115
23	7	113
-23	7	112
24	7	106
-24	7	101
25	7	95
-25	7	94
26	7	88
-26	7	83
27	7	77
-27	7	76
28	7	70
-28	7	65
29	7	64
-29	7	63
30	7	56
-30	7	51
31	7	45
-31	7	44
32	7	39
-32	7	38
33	7	31

Quantization level	Code length	Code
-33	7	30
34	7	20
-34	7	19
35	7	18
-35	7	9
36	7	3
-36	7	2
37	7	0
-37	8	251
38	8	245
-38	8	244
39	8	238
-39	8	229
40	8	215
-40	8	214
41	8	200
-41	8	179
42	8	165
-42	8	164
43	8	143
-43	8	142
44	8	124
-44	8	115
45	8	101
-45	8	100
46	8	66
-46	8	65
47	8	43
-47	8	42
48	8	17
-48	8	16
49	8	2
-49	9	501
50	9	479
-50	9	478
51	9	456
-51	9	403
52	9	357
-52	9	356
53	9	251
-53	9	250
54	9	228
-54	9	135
55	9	129
-55	9	128
56	9	6
-56	10	1 001
57	10	1 000
-57	10	915
58	10	805
-58	10	804
59	10	458
-59	10	269
60	10	268
-60	10	15
61	11	1 829
-61	11	1 828
62	11	918
-62	11	29
63	11	28
-63	12	1 839
64	13	3 677
-64	13	3 676

Table D.129

Quantization level	Code length	Code
0	4	9
1	4	6
-1	4	5
2	4	2
-2	4	1
3	5	30
-3	5	29
4	5	26
-4	5	25
5	5	22
-5	5	21
6	5	16
-6	5	15
7	5	8
-7	5	7
8	5	0
-8	6	63
9	6	56
-9	6	55
10	6	48
-10	6	47
11	6	40
-11	6	35
12	6	28
-12	6	19
13	6	12
-13	6	3
14	7	124
-14	7	115
15	7	108
-15	7	99
16	7	92
-16	7	83
17	7	68
-17	7	59
18	7	36
-18	7	27
19	7	4
-19	8	251
20	8	228
-20	8	219
21	8	196
-21	8	187
22	8	164
-22	8	139
23	8	116
-23	8	75
24	8	52
-24	8	11
25	9	501
-25	9	500
26	9	437
-26	9	436
27	9	373
-27	9	372
28	9	277
-28	9	276
29	9	149
-29	9	148
30	9	21
-30	9	20
31	10	917
-31	10	916
32	10	789
-32	10	788
33	10	661

Quantization level	Code length	Code
-33	10	660
34	10	469
-34	10	468
35	10	214
-35	10	213
36	11	1 838
-36	11	1 837
37	11	1 582
-37	11	1 581
38	11	1 326
-38	11	1 325
39	11	942
-39	11	941
40	11	431
-40	11	430
41	12	3 679
-41	12	3 678
42	12	3 167
-42	12	3 166
43	12	3 160
-43	12	2 655
44	12	2 648
-44	12	1 887
45	12	1 880
-45	12	851
46	12	849
-46	12	848
47	13	7 346
-47	13	7 345
48	13	6 322
-48	13	5 309
49	13	3 773
-49	13	3 772
50	13	3 762
-50	13	1 701
51	14	14 695
-51	14	14 694
52	14	14 688
-52	14	12 647
53	14	10 617
-53	14	10 616
54	14	10 596
-54	14	7 527
55	14	3 401
-55	14	3 400
56	15	29 378
-56	15	25 293
57	15	21 195
-57	15	21 194
58	15	15 053
-58	15	15 052
59	16	58 759
-59	16	58 758
60	16	50 585
-60	16	50 584
61	16	42 399
-61	16	42 398
62	16	42 397
-62	16	42 396
63	16	42 395
-63	16	42 394
64	16	42 393
-64	16	42 392

Table E.129

Quantization level	Code length	Code
0	5	12
1	5	11
-1	5	10
2	5	9
-2	5	8
3	5	7
-3	5	6
4	5	4
-4	5	3
5	5	2
-5	5	1
6	5	0
-6	6	63
7	6	61
-7	6	60
8	6	59
-8	6	58
9	6	56
-9	6	55
10	6	53
-10	6	52
11	6	51
-11	6	50
12	6	47
-12	6	46
13	6	45
-13	6	44
14	6	42
-14	6	41
15	6	38
-15	6	37
16	6	36
-16	6	35
17	6	32
-17	6	31
18	6	29
-18	6	28
19	6	26
-19	6	11
20	7	125
-20	7	124
21	7	109
-21	7	108
22	7	98
-22	7	97
23	7	87
-23	7	86
24	7	79
-24	7	78
25	7	68
-25	7	67
26	7	60
-26	7	55
27	7	21
-27	7	20
28	8	230
-28	8	229
29	8	198
-29	8	193
30	8	163
-30	8	162
31	8	139
-31	8	138
32	8	123
-32	8	122
33	8	108

Quantization level	Code length	Code
-33	9	463
34	9	457
-34	9	456
35	9	385
-35	9	384
36	9	321
-36	9	320
37	9	266
-37	9	265
38	9	218
-38	10	925
39	10	798
-39	10	797
40	10	646
-40	10	645
41	10	535
-41	10	534
42	10	528
-42	10	439
43	11	1 848
-43	11	1 599
44	11	1 592
-44	11	1 295
45	11	1 288
-45	11	1 059
46	11	877
-46	11	876
47	12	3 197
-47	12	3 196
48	12	2 589
-48	12	2 588
49	12	2 117
-49	12	2 116
50	13	7 398
-50	13	7 397
51	13	6 374
-51	13	6 373
52	13	5 158
-52	13	5 157
53	14	14 799
-53	14	14 798
54	14	12 751
-54	14	12 750
55	14	10 318
-55	14	10 313
56	15	29 587
-56	15	29 586
57	15	29 584
-57	15	25 491
58	15	20 625
-58	15	20 624
59	16	59 171
-59	16	59 170
60	16	50 980
-60	16	41 277
61	16	50 981
-61	16	41 278
62	16	50 978
-62	16	41 279
63	16	50 979
-63	16	50 976
64	16	50 977
-64	16	41 276

Table F.129

Quantization level	Code length	Code
0	6	56
1	6	55
-1	6	54
2	6	52
-2	6	51
3	6	50
-3	6	49
4	6	48
-4	6	47
5	6	46
-5	6	45
6	6	44
-6	6	43
7	6	41
-7	6	40
8	6	39
-8	6	38
9	6	36
-9	6	35
10	6	34
-10	6	33
11	6	31
-11	6	30
12	6	29
-12	6	28
13	6	26
-13	6	25
14	6	23
-14	6	22
15	6	21
-15	6	20
16	6	18
-16	6	17
17	6	15
-17	6	14
18	6	12
-18	6	11
19	6	9
-19	6	8
20	6	7
-20	6	6
21	6	3
-21	6	2
22	6	1
-22	6	0
23	7	125
-23	7	124
24	7	123
-24	7	122
25	7	120
-25	7	119
26	7	116
-26	7	115
27	7	114
-27	7	107
28	7	84
-28	7	75
29	7	65
-29	7	64
30	7	54
-30	7	49
31	7	39
-31	7	38
32	7	27
-32	7	26
33	7	20

Quantization level	Code length	Code
-33	7	11
34	7	10
-34	7	9
35	8	254
-35	8	253
36	8	243
-36	8	242
37	8	235
-37	8	234
38	8	213
-38	8	212
39	8	149
-39	8	148
40	8	110
-40	8	97
41	8	66
-41	8	65
42	8	43
-42	8	42
43	8	16
-43	9	511
44	9	505
-44	9	504
45	9	474
-45	9	473
46	9	343
-46	9	342
47	9	340
-47	9	223
48	9	192
-48	9	135
49	9	129
-49	9	128
50	9	34
-50	10	1 021
51	10	951
-51	10	950
52	10	944
-52	10	683
53	10	445
-53	10	444
54	10	269
-54	10	268
55	10	71
-55	10	70
56	11	2 040
-56	11	1 891
57	11	1 364
-57	11	775
58	11	774
-58	11	773
59	12	4 083
-59	12	4 082
60	12	3 780
-60	12	2 731
61	12	1 545
-61	12	1 544
62	13	7 562
-62	13	5 461
63	13	5 460
-63	14	15 127
64	15	30 253
-64	15	30 252

Table G.129

Quantization level	Code length	Code
0	4	0
1	5	29
-1	5	28
2	5	25
-2	5	24
3	5	21
-3	5	20
4	5	17
-4	5	16
5	5	13
-5	5	12
6	5	9
-6	5	8
7	5	5
-7	5	4
8	6	63
-8	6	62
9	6	55
-9	6	54
10	6	47
-10	6	46
11	6	39
-11	6	38
12	6	31
-12	6	30
13	6	23
-13	6	22
14	6	15
-14	6	14
15	6	7
-15	6	6
16	7	123
-16	7	122
17	7	107
-17	7	106
18	7	91
-18	7	90
19	7	75
-19	7	74
20	7	59
-20	7	58
21	7	43
-21	7	42
22	7	27
-22	7	26
23	7	11
-23	7	10
24	7	8
-24	8	243
25	8	240
-25	8	211
26	8	208
-26	8	179
27	8	176
-27	8	147
28	8	144
-28	8	115
29	8	112
-29	8	83
30	8	80
-30	8	51
31	8	48
-31	8	19
32	9	484
-32	9	483
33	9	421

Quantization level	Code length	Code
-33	9	420
34	9	357
-34	9	356
35	9	293
-35	9	292
36	9	229
-36	9	228
37	9	226
-37	9	165
38	9	162
-38	9	101
39	9	98
-39	9	37
40	10	970
-40	10	965
41	10	839
-41	10	838
42	10	711
-42	10	710
43	10	708
-43	10	583
44	10	580
-44	10	455
45	10	329
-45	10	328
46	10	201
-46	10	200
47	10	198
-47	10	73
48	11	1 942
-48	11	1 929
49	11	1 675
-49	11	1 674
50	11	1 672
-50	11	1 419
51	11	1 165
-51	11	1 164
52	11	1 162
-52	11	909
53	11	655
-53	11	654
54	11	652
-54	11	399
55	11	145
-55	11	144
56	12	3 886
-56	12	3 857
57	12	3 347
-57	12	3 346
58	12	2 837
-58	12	2 836
59	12	2 327
-59	12	2 326
60	12	1 817
-60	12	1 816
61	12	1 307
-61	12	1 306
62	12	797
-62	12	796
63	13	7 775
-63	13	7 774
64	13	7 713
-64	13	7 712

D.6 Block Code Books

D.6.1 3 Levels

Table V.3: 3-level 4-element 7-bit Block Code Book

Level index	Code for 1st element
-1	0
0	1
1	2
Level index	**Code for 2nd element**
-1	0
0	3
1	6
Level index	**Code for 3rd element**
-1	0
0	9
1	18
Level index	**Code for 4th element**
-1	0
0	27
1	54

D.6.2 5 Levels

Table V.5: 5-level 4-element 10-bit Block Code Book

Level index	Code for 1st element
-2	0
-1	1
0	2
1	3
2	4
Level index	**Code for 2nd element**
-2	0
-1	5
0	10
1	15
2	20
Level index	**Code for 3rd element**
-2	0
-1	25
0	50
1	75
2	100
Level index	**Code for 4th element**
-2	0
-1	125
0	250
1	375
2	500

D.6.3 7 Levels

Table V.7: 7-level 4-element 12-bit Block Code Book

Level index	Code for 1st element
-3	0
-2	1
-1	2
0	3
1	4
2	5
3	6
Level index	**Code for 2nd element**
-3	0
-2	7
-1	14
0	21
1	28
2	35
3	42
Level index	**Code for 3rd element**
-3	0
-2	49
-1	98
0	47
1	196
2	245
3	294
Level index	**Code for 4th element**
-3	0
-2	343
-1	686
0	1 029
1	1 372
2	1 715
3	2 058

D.6.4 9 Levels

Table V.9: 9-level 4-element 13-bit Block Code Book

Level index	Code for 1st element
-4	0
-3	1
-2	2
-1	3
0	4
1	5
2	6
3	7
4	8
Level index	**Code for 2nd element**
-4	0
-3	9
-2	18
-1	27
0	36
1	45
2	54
3	63
4	72
Level index	**Code for 3rd element**
-4	0
-3	81
-2	162
-1	243
0	324
1	405
2	486
3	567
4	648
Level index	**Code for 4th element**
-4	0
-3	729
-2	1 458
-1	2 187
0	2 916
1	3 645
2	4 374
3	5 103
4	5 832

D.6.5 13 Levels

Table V.13: 13-level 4-element 15-bit block quantizer

Level index	Code for 1st element
-6	0
-5	1
-4	2
-3	3
-2	4
-1	5
0	6
1	7
2	8
3	9
4	10
5	11
6	12
Level index	**Code for 2nd element**
-6	0
-5	13
-4	26
-3	39
-2	52
-1	65
0	78
1	91
2	104
3	117
4	130
5	143
6	156
Level index	**Code for 3rd element**
-6	0
-5	169
-4	338
-3	507
-2	676
-1	845
0	1 014
1	1 183
2	1 352
3	1 521
4	1 690
5	1 859
6	2 028
Level index	**Code for 4th element**
-6	0
-5	2 197
-4	4 394
-3	6 591
-2	8 788
-1	10 985
0	13 182
1	15 379
2	17 576
3	19 773
4	21 970
5	24 167
6	26 364

D.6.6 17 Levels

Table V.17: 17-level 4-element 17-bit Block Code Book

Level index	Code for 1st element
-8	0
-7	1
-6	2
-5	3
-4	4
-3	5
-2	6
-1	7
0	8
1	9
2	10
3	11
4	12
5	13
6	14
7	15
8	16
Level index	**Code for 2nd element**
-8	0
-7	17
-6	34
-5	51
-4	68
-3	85
-2	102
-1	119
0	136
1	153
2	170
3	187
4	204
5	221
6	238
7	255
8	272
Level index	**Code for 3rd element**
-8	0
-7	289
-6	578
-5	867
-4	1 156
-3	1 445
-2	1 734
-1	2 023
0	2 312
1	2 601
2	2 890
3	3 179
4	3 468
5	3 757
6	4 046
7	4 335
8	4 624

Level index	Code for 4th element
-8	0
-7	4 913
-6	9 826
-5	14 739
-4	19 652
-3	24 565
-2	29 478
-1	34 391
0	39 304
1	44 217
2	49 130
3	54 043
4	58 956
5	63 869
6	68 782
7	73 695
8	78 608

D.6.7 25 Levels

Table V.25: 25-level 4-element 19-bit Block Code Book

Level index	Code for 1st element
-12	0
-11	1
-10	2
-9	3
-8	4
-7	5
-6	6
-5	7
-4	8
-3	9
-2	10
-1	11
0	12
1	13
2	14
3	15
4	16
5	17
6	18
7	19
8	20
9	21
10	22
11	23
12	24
Level index	**Code for 2nd element**
-12	0
-11	25
-10	50
-9	75
-8	100
-7	125
-6	150
-5	175
-4	200
-3	225
-2	250
-1	275
0	300
1	325
2	350
3	375
4	400
5	425
6	450
7	475
8	500
9	525
10	550
11	575
12	600

Level index	Code for 3rd element
-12	0
-11	625
-10	1 250
-9	1 875
-8	2 500
-7	3 125
-6	3 750
-5	4 375
-4	5 000
-3	5 625
-2	6 250
-1	6 875
0	7 500
1	8 125
2	8 750
3	9 375
4	10 000
5	10 625
6	11 250
7	11 875
8	12 500
9	13 125
10	13 750
11	14 375
12	15 000

Level index	Code for 4th element
-12	0
-11	15 625
-10	31 250
-9	46 875
-8	62 500
-7	78 125
-6	93 750
-5	109 375
-4	125 000
-3	140 625
-2	156 250
-1	171 875
0	187 500
1	203 125
2	218 750
3	234 375
4	250 000
5	265 625
6	281 250
7	296 875
8	312 500
9	328 125
10	343 750
11	359 375
12	375 000

D.7 Interpolation FIR

D.7.1 2 x Interpolation

0.00000330524
-0.00000010955
-0.00001133348
-0.00000550946
0.00002381930
0.00002278368
-0.00003684078
-0.00005886791
0.00004053684
0.00011868291
-0.00001809484
-0.00020025449
-0.00005299183
0.00028929862
0.00019636558
-0.00035464740
-0.00042854782
0.00034668882
0.00074765814
-0.00020110645
-0.00112112367
-0.00015036913
0.00147567503
0.00076182623
-0.00169373665
-0.00164926716
0.00162025949
0.00276480708
-0.00108283700
-0.00397485122
-0.00007440893
0.00504923845
0.00194591074
-0.00566803338
-0.00451489678
0.00545062358
0.00760785490
-0.00400814833
-0.01086365897
0.00101561449
0.01372703910
0.00370476092
-0.01547267288
-0.01010103151
0.01526044402
0.01782309450
-0.01221452747
-0.02617896535
0.00550970528
0.03411839902
0.00556340395
-0.04023005813
-0.02160109766
0.04270342737
0.04324966297
-0.03911506757
-0.07177370787
0.02557834797
0.11093838513
0.00677702902
-0.17523027956
-0.09289701283
0.36306288838
0.82348650694
0.82348650694
0.36306288838
-0.09289701283
-0.17523027956
0.00677702902
0.11093838513
0.02557834797
-0.07177370787
-0.03911506757
0.04324966297
0.04270342737
-0.02160109766
-0.04023005813
0.00556340395
0.03411839902
0.00550970528
-0.02617896535
-0.01221452747
0.01782309450
0.01526044402
-0.01010103151
-0.01547267288
0.00370476092
0.01372703910
0.00101561449
-0.01086365897
-0.00400814833
0.00760785490
0.00545062358

-0.00451489678
-0.00566803338
0.00194591074
0.00504923845
-0.00007440893
-0.00397485122
-0.00108283700
0.00276480708
0.00162025949
-0.00164926716
-0.00169373665
0.00076182623
0.00147567503
-0.00015036913
-0.00112112367
-0.00020110645
0.00074765814
0.00034668882
-0.00042854782
-0.00035464740
0.00019636558
0.00028929862
-0.00005299183
-0.00020025449
-0.00001809484
0.00011868291
0.00004053684
-0.00005886791
-0.00003684078
0.00002278368
0.00002381930
-0.00000550946
-0.00001133348
-0.00000010955
0.00000330524

D.7.2 4 x Interpolation

0.00000210763
0.00001094810
0.00002290807
0.00002839700
0.00001428398
-0.00002752976
-0.00008951150
-0.00014279621
-0.00014358315
-0.00005408613
0.00012832218
0.00034889783
0.00049825106
0.00045058018
0.00013060049
-0.00041822359
-0.00100400147
-0.00132885773
-0.00110866863
-0.00023321882
0.00110653217
0.00240567792
0.00299475086
0.00232767221
0.00030306191
-0.00253753108
-0.00507534668
-0.00599124469
-0.00435559964
-0.00019723058
0.00523723615
0.00974622648
0.01096914243
0.00746764848
-0.00035646744
-0.00998689700
-0.01744846255
-0.01880371012
-0.01198318321
0.00182849320
0.01799243502
0.02975338697
0.03083455376
0.01837555505
-0.00504227728
-0.03137993813
-0.04954963177
-0.04967092723
-0.02767589502
0.01166744903
0.05492079630
0.08387579024
0.08227037638
0.04309020936
-0.02637432516
-0.10408806801
-0.15836296976
-0.15739876032
-0.08037899435
0.07367454469
0.28265473247
0.50538766384
0.69214117527

0.79854756594
0.79854756594
0.69214117527
0.50538766384
0.28265473247
0.07367454469
-0.08037899435
-0.15739876032
-0.15836296976
-0.10408806801
-0.02637432516
0.04309020936
0.08227037638
0.08387579024
0.05492079630
0.01166744903
-0.02767589502
-0.04967092723
-0.04954963177
-0.03137993813
-0.00504227728
0.01837555505

0.03083455376
0.02975338697
0.01799243502
0.00182849320
-0.01198318321
-0.01880371012
-0.01744846255
-0.00998689700
-0.00035646744
0.00746764848
0.01096914243
0.00974622648
0.00523723615
-0.00019723058
-0.00435559964
-0.00599124469
-0.00507534668
-0.00253753108
0.00030306191
0.00232767221
0.00299475086
0.00240567792

0.00110653217
-0.00023321882
-0.00110866863
-0.00132885773
-0.00100400147
-0.00041822359
0.00013060049
0.00045058018
0.00049825106
0.00034889783
0.00012832218
-0.00005408613
-0.00014358315
-0.00014279621
-0.00008951150
-0.00002752976
0.00001428398
0.00002839700
0.00002290807
0.00001094810
0.00000210763

D.8 32-Band Interpolation FIR

D.8.1 Perfect Reconstruction

+1.135985195E-010
+7.018770981E-011
-1.608403011E-008
-5.083275667E-008
-1.543309907E-007
-3.961981463E-007
-7.342250683E-007
-3.970030775E-007
-4.741137047E-007

-6.022448247E-007
-6.628192182E-007
-6.982898526E-007
-7.020648809E-007
-6.767839409E-007
-6.262345096E-007
-5.564140224E-007
+7.003467317E-007
+8.419976893E-007

+9.742954035E-007
+1.085227950E-006
+1.162929266E-006
+1.194632091E-006
+1.179182050E-006
+1.033426656E-006
+9.451737242E-007
+1.975324267E-006
+1.190443072E-006

+5.234479659E-007
+2.014677420E-007
+7.834767501E-008

-6.702406963E-010
-1.613285505E-009
-2.682709610E-009
-3.399493131E-009
+1.314406006E-008
+7.506701927E-009
+2.788728892E-008
+1.444918922E-007
+3.132386439E-007
+1.399798180E-006
+2.032118118E-006
+2.715013807E-006
+3.453840463E-006
+4.195037945E-006
+4.896494374E-006
+5.516381407E-006
+6.015239251E-006
+6.361419310E-006
+8.006985809E-006
+8.087732567E-006
+7.941360309E-006
+7.568834008E-006
+6.986399967E-006
+6.225028756E-006
+5.315936960E-006
+4.429412002E-006
+3.332600045E-006
+8.427224429E-007
+4.341498823E-007
+9.458596395E-008
+2.975164826E-008

+6.402664354E-008
-3.246264413E-008
-3.809887872E-008
+8.434094667E-008
+6.437721822E-008
+1.189317118E-006
+2.497214155E-006
+3.617151151E-006
+3.157242645E-006
+2.319611212E-006
+7.869333785E-006
+9.826449968E-006
+1.177108606E-005
+1.379448349E-005
+1.571428584E-005
+1.743183020E-005
+1.884208177E-005
+1.987093310E-005
+2.042970118E-005
-3.144468428E-005
-3.334947178E-005
-3.460439257E-005
-3.515914432E-005
-3.495384954E-005
-3.397853652E-005
-3.225446198E-005
-2.978993689E-005
-2.677291741E-005
-1.806914770E-005
-1.776598037E-005
-1.661818715E-005
-1.207003334E-005
-6.993315310E-006
-5.633860383E-007
-9.984935332E-007

-1.470520488E-006
-1.853591357E-006
+7.198007665E-007
+3.086857760E-006
+6.084746474E-006
+9.561075785E-006
+1.309637537E-005
+2.263354872E-005
+2.847247197E-005
+3.415624451E-005
+3.946387005E-005
+4.425736552E-005
+4.839275425E-005
+5.176846025E-005
+5.429694284E-005
+5.595519906E-005
+4.916387297E-006
+9.299508747E-006
+1.356193479E-005
+1.751866148E-005
+2.093936746E-005
+2.362549276E-005
+2.537086584E-005
+2.618136386E-005
+2.554462844E-005
+3.018750249E-005
+2.570833203E-005
+1.985177369E-005
+1.191342653E-005
+2.525620175E-006
-1.521241393E-005
-1.617751332E-005
+1.992636317E-005
+1.774702469E-005
+4.624524081E-005

+5.610509834E-005

+6.568001118E-005

+7.513730816E-005

+8.413690375E-005

+8.757545584E-005

+9.517164290E-005

+1.020687996E-004

+1.084438481E-004

+1.140582463E-004

+1.187910311E-004

+1.224978914E-004

+1.250260248E-004

+1.262027217E-004

+1.226499153E-004

+1.213575742E-004

+1.180980107E-004

+1.126275165E-004

+1.047207043E-004

+9.417100227E-005

+8.078388782E-005

+6.447290798E-005

+4.491530854E-005

+2.470704203E-005

-1.714242217E-006

-3.193307566E-005

-6.541742187E-005

-1.024175072E-004

-1.312203676E-004

-1.774113771E-004

-2.233728592E-004

-2.682086197E-004

-3.347633174E-004

-3.906481725E-004

-4.490280990E-004

-5.099929986E-004

-5.729619297E-004

-6.358824321E-004

-7.021900383E-004

-7.698345580E-004

-8.385353722E-004

-9.078957955E-004

-9.775133803E-004

-1.046945457E-003

-1.115717343E-003

-1.183370827E-003

-1.252829796E-003

-1.316190348E-003

-1.376571832E-003

-1.433344092E-003

-1.485876855E-003

-1.533520175E-003

-1.575609902E-003

-1.611457788E-003

-1.640390139E-003

-1.661288203E-003

-1.674512983E-003

-1.678415807E-003

-1.672798418E-003

-1.656501088E-003

-1.633993932E-003

-1.593449386E-003

+1.542080659E-003

+1.479332102E-003

+1.395521569E-003

+1.303116791E-003

+1.196175464E-003

+1.073757303E-003

+9.358961834E-004

+7.817269652E-004

+6.114174030E-004

+4.244441516E-004

+2.206075296E-004

-2.719412748E-007

-2.382978710E-004

-4.935106263E-004

-7.658848190E-004

-1.055365428E-003

-1.361547387E-003

-1.684492454E-003

-2.023874084E-003

-2.379294252E-003

-2.750317100E-003

-3.136433195E-003

-3.537061159E-003

-3.951539751E-003

-4.379155114E-003

-4.819062538E-003

-5.270531867E-003

-5.732392892E-003

-6.203945260E-003

-6.683901884E-003

-7.170005701E-003

-7.664063945E-003

-8.162760176E-003

-8.665001951E-003

-9.170533158E-003

-9.676489048E-003

-1.018219907E-002

-1.068630442E-002

-1.118756086E-002

-1.168460958E-002

-1.217562053E-002

-1.265939046E-002

-1.313448418E-002

-1.359948888E-002

-1.405300573E-002

-1.449365262E-002

-1.492007636E-002

-1.533095632E-002

-1.572482102E-002

-1.610082202E-002

-1.645756140E-002

-1.679391414E-002

-1.710879989E-002

-1.740120351E-002

-1.767017506E-002

-1.791484281E-002

-1.813439466E-002

-1.832821220E-002

-1.849545911E-002

-1.863567345E-002

-1.874836907E-002

-1.883326657E-002

-1.889026538E-002

-1.891860925E-002

+1.891860925E-002

+1.889026538E-002

+1.883326657E-002

+1.874836907E-002

+1.863567345E-002

+1.849545911E-002

+1.832821220E-002

+1.813439466E-002

+1.791484281E-002

+1.767017506E-002

+1.740120351E-002

+1.710879989E-002

+1.679391414E-002

+1.645756140E-002

+1.610082202E-002

+1.572482102E-002

+1.533095632E-002

+1.492007636E-002

+1.449365262E-002

+1.405300573E-002

+1.359948888E-002

+1.313448418E-002

+1.265939046E-002

+1.217562053E-002

+1.168460958E-002

+1.118756086E-002

+1.068630442E-002

+1.018219907E-002

+9.676489048E-003

+9.170533158E-003

+8.665001951E-003

+8.162760176E-003

+7.664063945E-003

+7.170005701E-003

+6.683901884E-003

+6.203945260E-003

+5.732392892E-003

+5.270531867E-003

+4.819062538E-003

+4.379155114E-003

+3.951539751E-003

+3.537061159E-003

+3.136433195E-003

+2.750317100E-003

+2.379294252E-003

+2.023874084E-003

+1.684492454E-003

+1.361547387E-003

+1.055365428E-003

+7.658848190E-004

+4.935106263E-004

+2.382978710E-004

+2.719412748E-007

-2.206075296E-004

-4.244441516E-004

-6.114174030E-004

-7.817269652E-004

-9.358961834E-004

-1.073757303E-003

-1.196175464E-003

-1.303116791E-003

-1.395521569E-003

-1.479332102E-003

-1.542080659E-003

+1.593449386E-003

+1.633993932E-003

+1.656501088E-003

+1.672798418E-003

+1.678415807E-003

+1.674512983E-003

+1.661288203E-003

+1.640390139E-003

+1.611457788E-003

+1.575609902E-003

+1.533520175E-003

+1.485876855E-003

+1.433344092E-003

+1.376571832E-003

+1.316190348E-003

+1.252829796E-003

+1.183370827E-003

+1.115717343E-003

+1.046945457E-003

+9.775133803E-004

+9.078957955E-004

+8.385353722E-004
+7.698345580E-004
+7.021900383E-004
+6.358824321E-004
+5.729619297E-004
+5.099929986E-004
+4.490280990E-004
+3.906481725E-004
+3.347633174E-004
+2.682086197E-004
+2.233728592E-004
+1.774113771E-004
+1.312203676E-004
+1.024175072E-004
+6.541742187E-005
+3.193307566E-005
+1.714242217E-006
-2.470704203E-005
-4.491530854E-005
-6.447290798E-005
-8.078388782E-005
-9.417100227E-005
-1.047207043E-004
-1.126275165E-004
-1.180980107E-004
-1.213575742E-004
-1.226499153E-004
-1.262027217E-004
-1.250260248E-004
-1.224978914E-004
-1.187910311E-004
-1.140582463E-004
-1.084438481E-004
-1.020687996E-004
-9.517164290E-005
-8.757545584E-005
-8.413690375E-005
-7.513730816E-005
-6.568001118E-005
-5.610509834E-005
-4.624524081E-005
-1.774702469E-005
-1.992636317E-005
+1.617751332E-005
+1.521241393E-005
-2.525620175E-006
-1.191342653E-005
-1.985177369E-005
-2.570833203E-005
-3.018750249E-005
-2.554462844E-005
-2.618136386E-005
-2.537086584E-005
-2.362549276E-005
-2.093936746E-005
-1.751866148E-005
-1.356193479E-005
-9.299508747E-006
-4.916387297E-006
-5.595519906E-005
-5.429694284E-005
-5.176846025E-005
-4.839275425E-005
-4.425736552E-005
-3.946387005E-005
-3.415624451E-005
-2.847247197E-005
-2.263354872E-005
-1.309637537E-005
-9.561075785E-006
-6.084746474E-006
-3.086857760E-006
-7.198007665E-007
+1.853591357E-006
+1.470520488E-006
+9.984935332E-007
+5.633860383E-007
+6.993315310E-006
+1.207003334E-005
+1.661818715E-005
+1.776598037E-005
+1.806914770E-005
+2.677291741E-005
+2.978993689E-005
+3.225446198E-005
+3.397853652E-005
+3.495384954E-005
+3.515914432E-005
+3.460439257E-005
+3.334947178E-005
+3.144468428E-005
-2.042970118E-005
-1.987093310E-005
-1.884208177E-005
-1.743183020E-005
-1.571428584E-005
-1.379448349E-005
-1.177108606E-005
-9.826449968E-006
-7.869333785E-006
-2.319611212E-006
-3.157242645E-006
-3.617151151E-006
-2.497214155E-006
-1.189317118E-006

-6.437721822E-008
-8.434094667E-008
+3.809887872E-008
+3.246264413E-008
-6.402664354E-008
-2.975164826E-008
-9.458596395E-008
-4.341498823E-007
-8.427224429E-007
-3.332600045E-006
-4.429412002E-006
-5.315936960E-006
-6.225028756E-006
-6.986399967E-006
-7.568834008E-006
-7.941360309E-006
-8.087732567E-006
-8.006985809E-006
-6.361419310E-006
-6.015239251E-006
-5.516381407E-006
-4.896494374E-006
-4.195037945E-006
-3.453840463E-006
-2.715013807E-006
-2.032118118E-006
-1.399798180E-006
-3.132386439E-007
-1.444918922E-007
-2.788728892E-008
-7.506701927E-009
-1.314406006E-008
+3.399493131E-009
+2.682709610E-009
+1.613285505E-009
+6.702406963E-010
-7.834767501E-008
-2.014677420E-007
-5.234479659E-007
-1.190443072E-006
-1.975324267E-006
-9.451737242E-007
-1.033426656E-006
-1.179182050E-006
-1.194632091E-006
-1.162929266E-006
-1.085227950E-006
-9.742954035E-007
-8.419976893E-007
-7.003467317E-007
+5.564140224E-007
+6.262345096E-007
+6.767839409E-007
+7.020648809E-007
+6.982898526E-007
+6.628192182E-007
+6.022448247E-007
+4.741137047E-007
+3.970030775E-007
+7.342250683E-007
+3.961981463E-007
+1.543309907E-007
+5.083275667E-008
+1.608403011E-008
-7.018770981E-011
-1.135985195E-010

D.8.2 Non-Perfect Reconstruction

-1.390191784E-007
-1.693738625E-007
-2.030677564E-007
-2.404238444E-007
-2.818143514E-007
-3.276689142E-007
-3.784752209E-007
-4.347855338E-007
-4.972276315E-007
-5.665120852E-007
-6.434325428E-007
-7.288739425E-007
-8.238164355E-007
-9.293416952E-007
-1.046637067E-006
-1.176999604E-006
-1.321840614E-006
-1.482681114E-006
-1.661159786E-006
-1.859034001E-006
-2.078171747E-006
-2.320550948E-006
-2.588257530E-006
-2.883470643E-006
-3.208459020E-006
-3.565570978E-006
-3.957220997E-006
-4.385879038E-006
-4.854050530E-006
-5.364252502E-006
-5.918994248E-006
-6.520755960E-006
-7.171964626E-006

-7.874960829E-006
-8.631964192E-006
-9.445050637E-006
-1.031611009E-005
-1.124680875E-005
-1.223855270E-005
-1.329243969E-005
-1.440921824E-005
-1.558924305E-005
-1.683242772E-005
-1.813820381E-005
-1.950545993E-005
-2.093250441E-005
-2.241701623E-005
-2.395598858E-005
-2.554569073E-005
-2.718161704E-005
-2.885844333E-005
-3.056998685E-005
-3.230916263E-005
-3.406793985E-005
-3.583733633E-005
-3.760734762E-005
-3.936696885E-005
-4.110412556E-005
-4.280570283E-005
-4.445751256E-005
-4.604430433E-005
-4.754976908E-005
-4.895655002E-005
-5.024627535E-005
+5.139957648E-005
+5.239612074E-005
+5.321469871E-005
+5.383323878E-005
+5.422891263E-005
+5.437819709E-005
+5.425697600E-005
+5.384063843E-005
+5.310418419E-005
+5.202236207E-005
+5.056979353E-005
+4.872112549E-005
+4.645117951E-005
+4.373511547E-005
+4.054862075E-005
+3.686808850E-005
+3.267079956E-005
+2.793515523E-005
+2.264085742E-005
+1.676913780E-005
+1.030297699E-005
+3.227306706E-006
-4.470633485E-006
-1.280130618E-005
-2.177240640E-005
-3.138873581E-005
-4.165195787E-005
-5.256036457E-005
-6.410864444E-005
-7.628766616E-005
-8.908427117E-005
-1.024810626E-004
-1.164562127E-004
-1.309833024E-004
-1.460311323E-004
-1.615635992E-004
-1.775395358E-004
-1.939126523E-004
-2.106313768E-004
-2.276388550E-004
-2.448728774E-004
-2.622658503E-004
-2.797449124E-004
-2.972317743E-004
-3.146430245E-004
-3.318900708E-004
-3.488793736E-004
-3.655125911E-004
-3.816867538E-004
-3.972945851E-004
-4.122247046E-004
-4.263620067E-004
-4.395879805E-004
-4.517810594E-004
-4.628172028E-004
-4.725702747E-004
-4.809123348E-004
-4.877146275E-004
-4.928477574E-004
-4.961824161E-004
-4.975944757E-004
-4.969481961E-004
-4.941228544E-004
-4.889960401E-004
+4.814492422E-004
+4.713678791E-004
+4.586426076E-004
+4.431701091E-004
+4.248536134E-004
+4.036037717E-004
+3.793396754E-004
+3.519894381E-004
+3.214911267E-004
+2.877934603E-004

+2.508567995E-004
+2.106537577E-004
+1.671699720E-004
+1.204049113E-004
+7.037253090E-005
+1.710198012E-005
-3.936182839E-005
-9.895755647E-005
-1.616069785E-004
-2.272142592E-004
-2.956659591E-004
-3.668301215E-004
-4.405563814E-004
-5.166754709E-004
-5.949990009E-004
-6.753197522E-004
-7.574109477E-004
-8.410271257E-004
-9.259034996E-004
-1.011756598E-003
-1.098284614E-003
-1.185167348E-003
-1.272067428E-003
-1.358630019E-003
-1.444484224E-003
-1.529243193E-003
-1.612505526E-003
-1.693855622E-003
-1.772865304E-003
-1.849094522E-003
-1.922092517E-003
-1.991399564E-003
-2.056547208E-003
-2.117061289E-003
-2.172462177E-003
-2.222266514E-003
-2.265989315E-003
-2.303145360E-003
-2.333251061E-003
-2.355825622E-003
-2.370394068E-003
-2.376487479E-003
-2.373647178E-003
-2.361423569E-003
-2.339380793E-003
-2.307097195E-003
-2.264167881E-003
-2.210205887E-003
-2.144844970E-003
-2.067740774E-003
-1.978572691E-003
-1.877046190E-003
-1.762894331E-003
-1.635878929E-003
+1.495792647E-003
+1.342460280E-003
+1.175740734E-003
+9.955273708E-004
+8.017504588E-004
+5.943773431E-004
+3.734139318E-004
+1.389056415E-004
-1.090620208E-004
-3.703625989E-004
-6.448282511E-004
-9.322494152E-004
-1.232374110E-003
-1.544908970E-003
-1.869517611E-003
-2.205822384E-003
-2.553403843E-003
-2.911801683E-003
-3.280514618E-003
-3.659002949E-003
-4.046686925E-003
-4.442950245E-003
-4.847140983E-003
-5.258570891E-003
-5.676518660E-003
-6.100233644E-003
-6.528933067E-003
-6.961807609E-003
-7.398022339E-003
-7.836719044E-003
-8.277016692E-003
-8.718019351E-003
-9.158811532E-003
-9.598465636E-003
-1.003604382E-002
-1.047059800E-002
-1.090117730E-002
-1.132682897E-002
-1.174659748E-002
-1.215953380E-002
-1.256469358E-002
-1.296114177E-002
-1.334795821E-002
-1.372423489E-002
-1.408908330E-002
-1.444163360E-002
-1.478104480E-002
-1.510649733E-002
-1.541720331E-002
-1.571240649E-002
-1.599138230E-002

-1.625344716E-002
-1.649795473E-002
-1.672429405E-002
-1.693190821E-002
-1.712027565E-002
-1.728892699E-002
-1.743743755E-002
-1.756543480E-002
-1.767260395E-002
-1.775865816E-002
-1.782339066E-002
-1.786663756E-002
-1.788828894E-002
+1.788828894E-002
+1.786663756E-002
+1.782339066E-002
+1.775865816E-002
+1.767260395E-002
+1.756543480E-002
+1.743743755E-002
+1.728892699E-002
+1.712027565E-002
+1.693190821E-002
+1.672429405E-002
+1.649795473E-002
+1.625344716E-002
+1.599138230E-002
+1.571240649E-002
+1.541720331E-002
+1.510649733E-002
+1.478104480E-002
+1.444163360E-002
+1.408908330E-002
+1.372423489E-002
+1.334795821E-002
+1.296114177E-002
+1.256469358E-002
+1.215953380E-002
+1.174659748E-002
+1.132682897E-002
+1.090117730E-002
+1.047059800E-002
+1.003604382E-002
+9.598465636E-003
+9.158811532E-003
+8.718019351E-003
+8.277016692E-003
+7.836719044E-003
+7.398022339E-003
+6.961807609E-003
+6.528933067E-003
+6.100233644E-003
+5.676518660E-003
+5.258570891E-003
+4.847140983E-003
+4.442950245E-003
+4.046686925E-003
+3.659002949E-003
+3.280514618E-003
+2.911801683E-003
+2.553403843E-003
+2.205822384E-003
+1.869517611E-003
+1.544908970E-003
+1.232374110E-003
+9.322494152E-004
+6.448282511E-004
+3.703625989E-004
+1.090620208E-004
-1.389056415E-004
-3.734139318E-004
-5.943773431E-004
-8.017504588E-004
-9.955273708E-004
-1.175740734E-003
-1.342460280E-003
-1.495792647E-003
+1.635878929E-003
+1.762894331E-003
+1.877046190E-003
+1.978572691E-003
+2.067740774E-003
+2.144844970E-003
+2.210205887E-003
+2.264167881E-003
+2.307097195E-003
+2.339380793E-003
+2.361423569E-003
+2.373647178E-003
+2.376487479E-003
+2.370394068E-003
+2.355825622E-003
+2.333251061E-003
+2.303145360E-003
+2.265989315E-003
+2.222266514E-003
+2.172462177E-003
+2.117061289E-003
+2.056547208E-003
+1.991399564E-003
+1.922092517E-003
+1.849094522E-003
+1.772865304E-003
+1.693855622E-003
+1.612505526E-003

+1.529243193E-003
+1.444484224E-003
+1.358630019E-003
+1.272067428E-003
+1.185167348E-003
+1.098284614E-003
+1.011756598E-003
+9.259034996E-004
+8.410271257E-004
+7.574109477E-004
+6.753197522E-004
+5.949990009E-004
+5.166754709E-004
+4.405563814E-004
+3.668301215E-004
+2.956659591E-004
+2.272142592E-004
+1.616069785E-004
+9.895755647E-005
+3.936182839E-005
-1.710198012E-005
-7.037253090E-005
-1.204049113E-004
-1.671699720E-004
-2.106537577E-004
-2.508567995E-004
-2.877934603E-004
-3.214911267E-004
-3.519894381E-004
-3.793396754E-004
-4.036037717E-004
-4.248536134E-004
-4.431701091E-004
-4.586426076E-004
-4.713678791E-004
-4.814492422E-004
+4.889960401E-004
+4.941228544E-004
+4.969481961E-004
+4.975944757E-004
+4.961824161E-004
+4.928477574E-004
+4.877146275E-004
+4.809123348E-004
+4.725702747E-004
+4.628172028E-004
+4.517810594E-004
+4.395879805E-004
+4.263620067E-004
+4.122247046E-004
+3.972945851E-004
+3.816867538E-004
+3.655125911E-004
+3.488793736E-004
+3.318900708E-004
+3.146430245E-004
+2.972317743E-004
+2.797449124E-004
+2.622658503E-004
+2.448728774E-004
+2.276388550E-004
+2.106313768E-004
+1.939126523E-004
+1.775395358E-004
+1.615635992E-004
+1.460311323E-004
+1.309833024E-004
+1.164562127E-004
+1.024810626E-004
+8.908427117E-005
+7.628766616E-005
+6.410864444E-005
+5.256036457E-005
+4.165195787E-005
+3.138873581E-005
+2.177240640E-005
+1.280130618E-005
+4.470633485E-006
-3.227306706E-006
-1.030297699E-005
-1.676913780E-005
-2.264085742E-005
-2.793515523E-005
-3.267079956E-005
-3.686808850E-005
-4.054862075E-005
-4.373511547E-005
-4.645117951E-005
-4.872112549E-005
-5.056979353E-005
-5.202236207E-005
-5.310418419E-005
-5.384063843E-005
-5.425697600E-005
-5.437819709E-005
-5.422891263E-005
-5.383323878E-005
-5.321469871E-005
-5.239612074E-005
-5.139957648E-005
+5.024627535E-005
+4.895655002E-005
+4.754976908E-005
+4.604430433E-005
+4.445751256E-005

+4.280570283E-005

+4.110412556E-005

+3.936696885E-005

+3.760734762E-005

+3.583733633E-005

+3.406793985E-005

+3.230916263E-005

+3.056998685E-005

+2.885844333E-005

+2.718161704E-005

+2.554569073E-005

+2.395598858E-005

+2.241701623E-005

+2.093250441E-005

+1.950545993E-005

+1.813820381E-005

+1.683242772E-005

+1.558924305E-005

+1.440921824E-005

+1.329243969E-005

+1.223855270E-005

+1.124680875E-005

+1.031611009E-005

+9.445050637E-006

+8.631964192E-006

+7.874960829E-006

+7.171964626E-006

+6.520755960E-006

+5.918994248E-006

+5.364252502E-006

+4.854050530E-006

+4.385879038E-006

+3.957220997E-006

+3.565570978E-006

+3.208459020E-006

+2.883470643E-006

+2.588257530E-006

+2.320550948E-006

+2.078171747E-006

+1.859034001E-006

+1.661159786E-006

+1.482681114E-006

+1.321840614E-006

+1.176999604E-006

+1.046637067E-006

+9.293416952E-007

+8.238164355E-007

+7.288739425E-007

+6.434325428E-007

+5.665120852E-007

+4.972276315E-007

+4.347855338E-007

+3.784752209E-007

+3.276689142E-007

+2.818143514E-007

+2.404238444E-007

+2.030677564E-007

+1.693738625E-007

+1.390191784E-007

D.9 LFE Interpolation FIR

D.9.1 64 x Interpolation

2.6584343868307770E-004

8.1793652498163280E-005

9.4393239123746760E-005

1.0821702744578940E-004

1.2333714403212070E-004

1.3974857574794440E-004

1.5759580128360540E-004

1.7699223826639360E-004

1.9817386055365200E-004

2.2118473134469240E-004

2.4602311896160240E-004

2.7261159266345200E-004

3.0138631700538100E-004

3.3283955417573450E-004

3.6589911906048660E-004

4.0182814700528980E-004

4.4018754852004350E-004

4.8127761692740020E-004

5.2524596685543660E-004

5.7215924607589840E-004

6.2221300322562460E-004

6.7555153509601950E-004

7.3241489008069040E-004

7.9285167157649990E-004

8.5701106581836940E-004

9.2511920956894760E-004

9.9747709464281800E-004

1.0739302961155770E-003

1.1550235794857140E-003

1.2406768510118130E-003

1.3312589144334200E-003

1.4268938684836030E-003

1.5278297942131760E-003

1.6342115122824910E-003
1.7463274998590350E-003
1.8643775256350640E-003
1.9886041991412640E-003
2.1191518753767010E-003
2.2563596721738580E-003
2.4004334118217230E-003
2.5515670422464610E-003
2.7100932784378530E-003
2.8761904686689380E-003
3.0501529108732940E-003
3.2322725746780640E-003
3.4227769356220960E-003
3.6219672765582800E-003
3.8300913292914630E-003
4.0474990382790560E-003
4.2744171805679800E-003
4.5111598446965220E-003
4.7580120153725150E-003
5.0153112970292570E-003
5.2832840010523800E-003
5.5623454973101620E-003
5.8526843786239620E-003
6.1547122895717620E-003
6.4686913974583150E-003
6.7949919030070300E-003
7.1338820271193980E-003
7.4857366271317010E-003
7.8508658334612850E-003
8.2296309992671010E-003
8.6223213002085690E-003
9.0293306857347480E-003
9.4509534537792200E-003
9.8875602707266800E-003
1.0339494794607160E-002
1.0807084850966930E-002
1.1290682479739190E-002
1.1790650896728040E-002
1.2307321652770040E-002
1.2841059826314450E-002
1.3392185792326930E-002
1.3961089774966240E-002
1.4548087492585180E-002
1.5153550542891020E-002
1.5777811408042910E-002
1.6421230509877200E-002
1.7084129154682160E-002
1.7766902223229410E-002
1.8469827249646190E-002
1.9193304702639580E-002
1.9937623292207720E-002
2.0703161135315900E-002
2.1490212529897690E-002
2.2299138829112050E-002
2.3130238056182860E-002
2.3983856663107870E-002
2.4860285222530360E-002
2.5759860873222350E-002
2.6682861149311060E-002
2.7629608288407320E-002
2.8600392863154410E-002
2.9595496132969860E-002
3.0615204945206640E-002
3.1659796833992000E-002
3.2729536294937140E-002
3.3824689686298370E-002
3.4945506602525710E-002
3.6092240363359450E-002
3.7265110760927200E-002
3.8464374840259550E-002
3.9690230041742320E-002
4.0942888706922530E-002
4.2222552001476290E-002
4.3529424816370010E-002
4.4863656163215640E-002
4.6225443482398990E-002
4.7614917159080510E-002
4.9032241106033330E-002
5.0477534532547000E-002
5.1950931549072270E-002
5.3452525287866590E-002
5.4982420057058330E-002
5.6540694087743760E-002
5.8127421885728840E-002
5.9742655605077740E-002
6.1386436223983760E-002
6.3058786094188690E-002
6.4759708940982820E-002
6.6489234566688540E-002
6.8247318267822270E-002
7.0033922791481020E-002
7.1849010884761810E-002
7.3692522943019870E-002
7.5564362108707430E-002
7.7464438974857330E-002
7.9392634332180020E-002
8.1348828971385960E-002
8.3332858979702000E-002
8.5344567894935610E-002
8.7383769452571870E-002
8.9450262486934660E-002
9.1543838381767280E-002
9.3664251267910000E-002
9.5811240375041960E-002
9.7984537482261660E-002

1.0018386691808700E-001
1.0240890830755230E-001
1.0465932637453080E-001
1.0693479329347610E-001
1.0923493653535840E-001
1.1155936866998670E-001
1.1390769481658940E-001
1.1627949774265290E-001
1.1867434531450270E-001
1.2109176814556120E-001
1.2353130429983140E-001
1.2599244713783260E-001
1.2847468256950380E-001
1.3097748160362240E-001
1.3350030779838560E-001
1.3604259490966800E-001
1.3860376179218290E-001
1.4118319749832150E-001
1.4378026127815250E-001
1.4639437198638920E-001
1.4902481436729430E-001
1.5167096257209780E-001
1.5433208644390100E-001
1.5700751543045040E-001
1.5969651937484740E-001
1.6239835321903230E-001
1.6511227190494540E-001
1.6783750057220460E-001
1.7057323455810550E-001
1.7331869900226590E-001
1.7607308924198150E-001
1.7883554100990300E-001
1.8160524964332580E-001
1.8438133597373960E-001
1.8716295063495640E-001
1.8994916975498200E-001
1.9273911416530610E-001
1.9553191959857940E-001
1.9832661747932440E-001
2.0112232863903040E-001
2.0391805469989780E-001
2.0671287178993220E-001
2.0950584113597870E-001
2.1229594945907590E-001
2.1508227288722990E-001
2.1786379814147950E-001
2.2063951194286350E-001
2.2340846061706540E-001
2.2616961598396300E-001
2.2892196476459500E-001
2.3166447877883910E-001
2.3439615964889520E-001
2.3711597919464110E-001
2.3982289433479310E-001
2.4251587688922880E-001
2.4519388377666480E-001
2.4785590171813960E-001
2.5050088763237000E-001
2.5312781333923340E-001
2.5573557615280150E-001
2.5832322239875800E-001
2.6088967919349670E-001
2.6343390345573420E-001
2.6595494151115420E-001
2.6845166087150580E-001
2.7092313766479490E-001
2.7336826920509340E-001
2.7578607201576240E-001
2.7817553281784060E-001
2.8053569793701170E-001
2.8286558389663700E-001
2.8516408801078800E-001
2.8743034601211550E-001
2.8966337442398070E-001
2.9186218976974480E-001
2.9402589797973640E-001
2.9615348577499390E-001
2.9824411869049070E-001
3.0029675364494320E-001
3.0231067538261420E-001
3.0428490042686460E-001
3.0621853470802300E-001
3.0811080336570740E-001
3.0996081233024600E-001
3.1176769733428960E-001
3.1353080272674560E-001
3.1524917483329780E-001
3.1692212820053100E-001
3.1854888796806340E-001
3.2012873888015740E-001
3.2166096568107600E-001
3.2314485311508180E-001
3.2457971572875980E-001
3.2596495747566220E-001
3.2729989290237420E-001
3.2858389616012580E-001
3.2981643080711360E-001
3.3099696040153500E-001
3.3212485909461980E-001
3.3319962024688720E-001
3.3422079682350160E-001
3.3518791198730470E-001
3.3610042929649360E-001
3.3695802092552180E-001
3.3776029944419860E-001

3.3850681781768800E-001
3.3919724822044380E-001
3.3983129262924200E-001
3.4040865302085880E-001
3.4092903137207030E-001
3.4139221906661980E-001
3.4179797768592840E-001
3.4214612841606140E-001
3.4243649244308470E-001
3.4266895055770880E-001
3.4284341335296630E-001
3.4295973181724550E-001
3.4301793575286860E-001
3.4301793575286860E-001
3.4295973181724550E-001
3.4284341335296630E-001
3.4266895055770880E-001
3.4243649244308470E-001
3.4214612841606140E-001
3.4179797768592840E-001
3.4139221906661980E-001
3.4092903137207030E-001
3.4040865302085880E-001
3.3983129262924200E-001
3.3919724822044380E-001
3.3850681781768800E-001
3.3776029944419860E-001
3.3695802092552180E-001
3.3610042929649360E-001
3.3518791198730470E-001
3.3422079682350160E-001
3.3319962024688720E-001
3.3212485909461980E-001
3.3099696040153500E-001
3.2981643080711360E-001
3.2858389616012580E-001
3.2729989290237420E-001
3.2596495747566220E-001
3.2457971572875980E-001
3.2314485311508180E-001
3.2166096568107600E-001
3.2012873888015740E-001
3.1854888796806340E-001
3.1692212820053100E-001
3.1524917483329780E-001
3.1353080272674560E-001
3.1176769733428960E-001
3.0996081233024600E-001
3.0811080336570740E-001
3.0621853470802300E-001
3.0428490042686460E-001
3.0231067538261420E-001
3.0029675364494320E-001
2.9824411869049070E-001
2.9615348577499390E-001
2.9402589797973640E-001
2.9186218976974480E-001
2.8966337442398070E-001
2.8743034601211550E-001
2.8516408801078800E-001
2.8286558389663700E-001
2.8053569793701170E-001
2.7817553281784060E-001
2.7578607201576240E-001
2.7336826920509340E-001
2.7092313766479490E-001
2.6845166087150580E-001
2.6595494151115420E-001
2.6343390345573420E-001
2.6088967919349670E-001
2.5832322239875800E-001
2.5573557615280150E-001
2.5312781333923340E-001
2.5050088763237000E-001
2.4785590171813960E-001
2.4519388377666480E-001
2.4251587688922880E-001
2.3982289433479310E-001
2.3711597919464110E-001
2.3439615964889520E-001
2.3166447877883910E-001
2.2892196476459500E-001
2.2616961598396300E-001
2.2340846061706540E-001
2.2063951194286350E-001
2.1786379814147950E-001
2.1508227288722990E-001
2.1229594945907590E-001
2.0950584113597870E-001
2.0671287178993220E-001
2.0391805469989780E-001
2.0112232863903040E-001
1.9832661747932440E-001
1.9553191959857940E-001
1.9273911416530610E-001
1.8994916975498200E-001
1.8716295063495640E-001
1.8438133597373960E-001
1.8160524964332580E-001
1.7883554100990300E-001
1.7607308924198150E-001
1.7331869900226590E-001
1.7057323455810550E-001
1.6783750057220460E-001
1.6511227190494540E-001

1.6239835321903230E-001
1.5969651937484740E-001
1.5700751543045040E-001
1.5433208644390100E-001
1.5167096257209780E-001
1.4902481436729430E-001
1.4639437198638920E-001
1.4378026127815250E-001
1.4118319749832150E-001
1.3860376179218290E-001
1.3604259490966800E-001
1.3350030779838560E-001
1.3097748160362240E-001
1.2847468256950380E-001
1.2599244713783260E-001
1.2353130429983140E-001
1.2109176814556120E-001
1.1867434531450270E-001
1.1627949774265290E-001
1.1390769481658940E-001
1.1155936866998670E-001
1.0923493653535840E-001
1.0693479329347610E-001
1.0465932637453080E-001
1.0240890830755230E-001
1.0018386691808700E-001
9.7984537482261660E-002
9.5811240375041960E-002
9.3664251267910000E-002
9.1543838381767280E-002
8.9450262486934660E-002
8.7383769452571870E-002
8.5344567894935610E-002
8.3332858979702000E-002
8.1348828971385960E-002
7.9392634332180020E-002
7.7464438974857330E-002
7.5564362108707430E-002
7.3692522943019870E-002
7.1849010884761810E-002
7.0033922791481020E-002
6.8247318267822270E-002
6.6489234566688540E-002
6.4759708940982820E-002
6.3058786094188690E-002
6.1386436223983760E-002
5.9742655605077740E-002
5.8127421885728840E-002
5.6540694087743760E-002
5.4982420057058330E-002
5.3452525287866590E-002
5.1950931549072270E-002
5.0477534532547000E-002
4.9032241106033330E-002
4.7614917159080510E-002
4.6225443482398990E-002
4.4863656163215640E-002
4.3529424816370010E-002
4.2222552001476290E-002
4.0942888706922530E-002
3.9690230041742320E-002
3.8464374840259550E-002
3.7265110760927200E-002
3.6092240363359450E-002
3.4945506602525710E-002
3.3824689686298370E-002
3.2729536294937140E-002
3.1659796833992000E-002
3.0615204945206640E-002
2.9595496132969860E-002
2.8600392863154410E-002
2.7629608288407320E-002
2.6682861149311060E-002
2.5759860873222350E-002
2.4860285222530360E-002
2.3983856663107870E-002
2.3130238056182860E-002
2.2299138829112050E-002
2.1490212529897690E-002
2.0703161135315900E-002
1.9937623292207720E-002
1.9193304702639580E-002
1.8469827249646190E-002
1.7766902223229410E-002
1.7084129154682160E-002
1.6421230509877200E-002
1.5777811408042910E-002
1.5153550542891020E-002
1.4548087492585180E-002
1.3961089774966240E-002
1.3392185792326930E-002
1.2841059826314450E-002
1.2307321652770040E-002
1.1790650896728040E-002
1.1290682479739190E-002
1.0807084850966930E-002
1.0339494794607160E-002
9.8875602707266800E-003
9.4509534537792200E-003
9.0293306857347480E-003
8.6223213002085690E-003
8.2296309992671010E-003
7.8508658334612850E-003
7.4857366271317010E-003
7.1338820271193980E-003

6.7949919030070300E-003

6.4686913974583150E-003

6.1547122895717620E-003

5.8526843786239620E-003

5.5623454973101620E-003

5.2832840010523800E-003

5.0153112970292570E-003

4.7580120153725150E-003

4.5111598446965220E-003

4.2744171805679800E-003

4.0474990382790560E-003

3.8300913292914630E-003

3.6219672765582800E-003

3.4227769356220960E-003

3.2322725746780640E-003

3.0501529108732940E-003

2.8761904686689380E-003

2.7100932784378530E-003

2.5515670422464610E-003

2.4004334118217230E-003

2.2563596721738580E-003

2.1191518753767010E-003

1.9886041991412640E-003

1.8643775256350640E-003

1.7463274998590350E-003

1.6342115122824910E-003

1.5278297942131760E-003

1.4268938684836030E-003

1.3312589144334200E-003

1.2406768510118130E-003

1.1550235794857140E-003

1.0739302961155770E-003

9.9747709464281800E-004

9.2511920956894760E-004

8.5701106581836940E-004

7.9285167157649990E-004

7.3241489008069040E-004

6.7555153509601950E-004

6.2221300322562460E-004

5.7215924607589840E-004

5.2524596685543660E-004

4.8127761692740020E-004

4.4018754852004350E-004

4.0182814700528980E-004

3.6589911906048660E-004

3.3283955417573450E-004

3.0138631700538100E-004

2.7261159266345200E-004

2.4602311896160240E-004

2.2118473134469240E-004

1.9817386055365200E-004

1.7699223826639360E-004

1.5759580128360540E-004

1.3974857574794440E-004

1.2333714403212070E-004

1.0821702744578940E-004

9.4393239123746760E-005

8.1793652498163280E-005

2.6584343868307770E-004

D.9.2 128 x Interpolation

0.00053168571

0.00016358691

0.00018878609

0.00021643363

0.00024667382

0.00027949660

0.00031519096

0.00035398375

0.00039634691

0.00044236859

0.00049204525

0.00054522208

0.00060277141

0.00066567765

0.00073179678

0.00080365466

0.00088037323

0.00096255314

0.00105048984

0.00114431616

0.00124442333

0.00135110028

0.00146482687

0.00158570008

0.00171401864

0.00185023469

0.00199495023

0.00214785640

0.00231004250

0.00248134881

0.00266251224

0.00285378192

0.00305565330

0.00326841651

0.00349264755

0.00372874714

0.00397720048

0.00423829490

0.00451271003

0.00480085658
0.00510312291
0.00542017492
0.00575236930
0.00610029325
0.00646453211
0.00684553990
0.00724391919
0.00766016589
0.00809498038
0.00854881573
0.00902230106
0.00951600447
0.01003060210
0.01056654565
0.01112466771
0.01170534454
0.01230939943
0.01293735672
0.01358995494
0.01426773332
0.01497144438
0.01570170000
0.01645922661
0.01724460535
0.01805862412
0.01890186779
0.01977507770
0.02067894675
0.02161412500
0.02258131653
0.02358125709
0.02461459488
0.02568206564
0.02678431384
0.02792212367
0.02909611352
0.03030703776
0.03155555204
0.03284239396
0.03416819125
0.03553372994
0.03693958372
0.03838652745
0.03987516090
0.04140623659
0.04298033938
0.04459818453
0.04626038298
0.04796761274
0.04972046614
0.05151961371
0.05336561054
0.05525910854
0.05720067024
0.05919086933
0.06123027951
0.06331945211
0.06545893103
0.06764923781
0.06989086419
0.07218432426
0.07453006506
0.07692859322
0.07938029617
0.08188561350
0.08444493264
0.08705867827
0.08972713351
0.09245070815
0.09522963315
0.09806428105
0.10095486045
0.10390164703
0.10690483451
0.10996460915
0.11308115721
0.11625462025
0.11948505789
0.12277261168
0.12611730397
0.12951917946
0.13297818601
0.13649433851
0.14006754756
0.14369773865
0.14738474786
0.15112841129
0.15492856503
0.15878495574
0.16269733012
0.16666537523
0.17068879306
0.17476719618
0.17890018225
0.18308731914
0.18732811511
0.19162209332
0.19596865773
0.20036731660
0.20481738448
0.20931822062
0.21386915445
0.21846942604
0.22311829031

0.22781492770
0.23255851865
0.23734821379
0.24218304455
0.24706205726
0.25198432803
0.25694879889
0.26195442677
0.26700007915
0.27208462358
0.27720692754
0.28236576915
0.28755992651
0.29278811812
0.29804900289
0.30334126949
0.30866351724
0.31401440501
0.31939238310
0.32479602098
0.33022382855
0.33567428589
0.34114575386
0.34663668275
0.35214546323
0.35767036676
0.36320972443
0.36876192689
0.37432509661
0.37989753485
0.38547745347
0.39106300473
0.39665243030
0.40224379301
0.40783521533
0.41342487931
0.41901078820
0.42459106445
0.43016362190
0.43572667241
0.44127810001
0.44681602716
0.45233830810
0.45784294605
0.46332800388
0.46879136562
0.47423094511
0.47964480519
0.48503074050
0.49038675427
0.49571081996
0.50100076199
0.50625455379
0.51147013903
0.51664537191
0.52177828550
0.52686679363
0.53190881014
0.53690224886
0.54184508324
0.54673534632
0.55157101154
0.55634999275
0.56107026339
0.56572991610
0.57032698393
0.57485944033
0.57932555676
0.58372318745
0.58805054426
0.59230577946
0.59648692608
0.60059231520
0.60462015867
0.60856848955
0.61243581772
0.61622029543
0.61992025375
0.62353414297
0.62706029415
0.63049703836
0.63384294510
0.63709646463
0.64025616646
0.64332056046
0.64628833532
0.64915806055
0.65192854404
0.65459835529
0.65716648102
0.65963155031
0.66199249029
0.66424828768
0.66639786959
0.66844022274
0.67037439346
0.67219948769
0.67391467094
0.67551922798
0.67701220512
0.67839306593
0.67966115475
0.68081587553
0.68185669184
0.68278300762

0.68359452486
0.68429082632
0.68487155437
0.68533653021
0.68568539619
0.68591803312
0.68603444099
0.68603444099
0.68591803312
0.68568539619
0.68533653021
0.68487155437
0.68429082632
0.68359452486
0.68278300762
0.68185669184
0.68081587553
0.67966115475
0.67839306593
0.67701220512
0.67551922798
0.67391467094
0.67219948769
0.67037439346
0.66844022274
0.66639786959
0.66424828768
0.66199249029
0.65963155031
0.65716648102
0.65459835529
0.65192854404
0.64915806055
0.64628833532
0.64332056046
0.64025616646
0.63709646463
0.63384294510
0.63049703836
0.62706029415
0.62353414297
0.61992025375
0.61622029543
0.61243581772
0.60856848955
0.60462015867
0.60059231520
0.59648692608
0.59230577946
0.58805054426
0.58372318745
0.57932555676
0.57485944033
0.57032698393
0.56572991610
0.56107026339
0.55634999275
0.55157101154
0.54673534632
0.54184508324
0.53690224886
0.53190881014
0.52686679363
0.52177828550
0.51664537191
0.51147013903
0.50625455379
0.50100076199
0.49571081996
0.49038675427
0.48503074050
0.47964480519
0.47423094511
0.46879136562
0.46332800388
0.45784294605
0.45233830810
0.44681602716
0.44127810001
0.43572667241
0.43016362190
0.42459106445
0.41901078820
0.41342487931
0.40783521533
0.40224379301
0.39665243030
0.39106300473
0.38547745347
0.37989753485
0.37432509661
0.36876192689
0.36320972443
0.35767036676
0.35214546323
0.34663668275
0.34114575386
0.33567428589
0.33022382855
0.32479602098
0.31939238310
0.31401440501
0.30866351724
0.30334126949
0.29804900289

0.29278811812

0.28755992651

0.28236576915

0.27720692754

0.27208462358

0.26700007915

0.26195442677

0.25694879889

0.25198432803

0.24706205726

0.24218304455

0.23734821379

0.23255851865

0.22781492770

0.22311829031

0.21846942604

0.21386915445

0.20931822062

0.20481738448

0.20036731660

0.19596865773

0.19162209332

0.18732811511

0.18308731914

0.17890018225

0.17476719618

0.17068879306

0.16666537523

0.16269733012

0.15878495574

0.15492856503

0.15112841129

0.14738474786

0.14369773865

0.14006754756

0.13649433851

0.13297818601

0.12951917946

0.12611730397

0.12277261168

0.11948505789

0.11625462025

0.11308115721

0.10996460915

0.10690483451

0.10390164703

0.10095486045

0.09806428105

0.09522963315

0.09245070815

0.08972713351

0.08705867827

0.08444493264

0.08188561350

0.07938029617

0.07692859322

0.07453006506

0.07218432426

0.06989086419

0.06764923781

0.06545893103

0.06331945211

0.06123027951

0.05919086933

0.05720067024

0.05525910854

0.05336561054

0.05151961371

0.04972046614

0.04796761274

0.04626038298

0.04459818453

0.04298033938

0.04140623659

0.03987516090

0.03838652745

0.03693958372

0.03553372994

0.03416819125

0.03284239396

0.03155555204

0.03030703776

0.02909611352

0.02792212367

0.02678431384

0.02568206564

0.02461459488

0.02358125709

0.02258131653

0.02161412500

0.02067894675

0.01977507770

0.01890186779

0.01805862412

0.01724460535

0.01645922661

0.01570170000

0.01497144438

0.01426773332

0.01358995494

0.01293735672

0.01230939943

0.01170534454

0.01112466771

0.01056654565

0.01003060210

0.00951600447

0.00902230106

0.00854881573

0.00809498038

0.00766016589

0.00724391919

0.00684553990

0.00646453211

0.00610029325

0.00575236930

0.00542017492

0.00510312291

0.00480085658

0.00451271003

0.00423829490

0.00397720048

0.00372874714

0.00349264755

0.00326841651

0.00305565330

0.00285378192

0.00266251224

0.00248134881

0.00231004250

0.00214785640

0.00199495023

0.00185023469

0.00171401864

0.00158570008

0.00146482687

0.00135110028

0.00124442333

0.00114431616

0.00105048984

0.00096255314

0.00088037323

0.00080365466

0.00073179678

0.00066567765

0.00060277141

0.00054522208

0.00049204525

0.00044236859

0.00039634691

0.00035398375

0.00031519096

0.00027949660

0.00024667382

0.00021643363

0.00018878609

0.00016358691

0.00053168571

D.10 VQ Tables

D.10.1 ADPCM Coefficients

Each vector consists of 4 elements and the Codebook has 2^{12} = 4 096 vectors. In the following table, each entry represents an element multi-plied by 2^{13}. So the actual value of each element is:

Actual Element Value = **Entry**

2^{13}

For example, the first entry in the table gives:

9928 = 1.2119140625.

2^{13}

List of Tables

Table 4.1: DTS Coherent Acoustics is optimized for 5.1 channel applications, but is extensible to deliver 8 channels with sampling frequency up to 192 kHz 7

Table 5.1: Frame Type 10

Table 5.2: Deficit Sample Count 11

Table 5.3: CRC Present Flag 11

Table 5.4: Audio channel arrangement 12

Table 5.5: Core audio sampling frequencies 12

Table 5.6: Sub-sampled audio decoding for standard sampling rates. 13

Table 5.7: RATE parameter vs. targeted bit-rate 13

Table 5.8: Targeted and actual bit-rate for the CD and DVD-Video applications 14

Table 5.9: Status of embedded down mixing coefficients 14

Table 5.10: Embedded Dynamic Range Flag 14

Table 5.11: Embedded Time Stamp Flag 14

Table 5.12: Auxiliary Data Flag 14

Table 5.13: Extension Audio Descriptor Flag 15

Table 5.14: Extended Coding Flag 15

Table 5.15: Audio Sync Word Insertion Flag 15

Table 5.16: Flag for LFE channel 15

Table 5.17: Multirate interpolation filter bank switch 16

Table 5.18: Encoder software revision 16

Table 5.19: Quantization resolution of source PCM samples 16

Table 5.20: Sum/difference decoding status of front left and right channels 16

Table 5.21: Sum/difference decoding status of left and right surround channels. 17

Table 5.22: Dialog Normalization Parameter 17

Table 7.1: X96k Algorithm Revision Number 22

Table B.1: Joint subband coding status and source channels 28

Table B.2: Selection of Huffman codebook for encoding the transient mode data TMODE 28

Table B.3: Code books and square root tables for scale factors 28

Table B.4: Codebooks for encoding bit allocation index ABITS 29

Table B.5: Selection of quantization levels and codebooks 29

Table B.6: Scale factor adjustment values if Huffman coding is used to encode the subband quantization indexes 30

Table C.1: 3-level 4-element 7-bit Block Code Book 39

Table A.3 54

Table A.4 54

Table B.4 54

Table C.4 54

Table D.4 54

Table A.5 54

Table B.5 55

Table C.5 55

Table A.7 55

Table B.7 55

Table C.7 55

Table A.9 56

Table B.9 56

Table C.9 56

Table A.12 56

Table B.12 57

Table C.12 57

Table D.12 57

Table E.12 57

Table A.13 58

Table B.13 58

Table C.13 58

Table A.17 59

Table B.17 59

Table C.17 59

Table D.17 60

Table E.17 60

Table F.17 60

Table G.17 61

Table A.25 61

Table B.25 62

Table C.25 62

Table D.25 63

Table E.25 63

Table F.25 64

Table G.25 64

Table A.33 ... 65
Table B.33 ... 66
Table C.33 ... 67
Table D.33 ... 68
Table E.33 ... 69
Table F.33 ... 70
Table G.33 ... 71
Table A.65 ... 72
Table B.65 ... 73
Table C.65 ... 74
Table D.65 ... 75
Table E.65 ... 76
Table F.65 ... 77
Table G.65 ... 78
Table SA.129 ... 79
Table SB.129 ... 82
Table SC.129 ... 84
Table SD.129 ... 86
Table SE.129 ... 88
Table A.129 ... 90
Table B.129 ... 92
Table C.129 ... 94
Table D.129 ... 96
Table E.129 ... 98
Table F.129 ... 100
Table G.129 ... 102
Table V.3: 3-level 4-element 7-bit Block Code Book ... 104
Table V.5: 5-level 4-element 10-bit Block Code Book ... 104
Table V.7: 7-level 4-element 12-bit Block Code Book ... 105
Table V.9: 9-level 4-element 13-bit Block Code Book ... 106
Table V.13: 13-level 4-element 15-bit block quantizer ... 107
Table V.17: 17-level 4-element 17-bit Block Code Book ... 108
Table V.25: 25-level 4-element 19-bit Block Code Book ... 110

History

Document history		
V1.1.1	August 2002	Publication
V1.2.1	December 2002	Publication