Davidson, G.A. “Digital Audio Coding: Dolby AC-3”
*Digital Signal Processing Handbook*
Ed. Vijay K. Madisetti and Douglas B. Williams
Boca Raton: CRC Press LLC, 1999

# 41

# Digital Audio Coding: Dolby AC-3



Grant A. Davidson
*Dolby Laboratories, Inc.*

## 41.1 Overview

In order to more efficiently transmit or store high-quality audio signals, it is often desirable to reduce the amount of information required to represent them. In the case of digital audio signals, the amount of binary information needed to accurately reproduce the original pulse code modulation (PCM) samples may be reduced by applying compression algorithm. A primary goal of audio compression algorithms is to maximally reduce the amount of digital information (bit-rate) required for conveyance of an audio signal while rendering differences between the original and decoded signals inaudible.

Digital audio compression is useful wherever there is an economic benefit realized by reducing the bit-rate. Typical applications are in satellite or terrestrial audio broadcasting, delivery of audio over electrical or optical cables, or storage of audio on magnetic, optical, semiconductor, or other storage media. One application which has received considerable attention in the United States is digital television (DTV). Audio and video compression are both necessary in DTV to meet the requirement that one high-definition DTV channel fit within the 6 MHz transmission bandwidth occupied by one preexisting NTSC (analog) channel. In December 1996, the United States Federal Communications Commission adopted the ATSC standard for DTV which is consistent with a consensus agreement developed by a broad cross-section of parties, including the broadcasting and computer industries. The audio technology used in the ATSC digital audio compression standard [1] is Dolby AC-3.

Dolby AC-3 is an audio compression technology capable of encoding a range of audio channel formats into a bit stream ranging from 32 kb/s to 640 kb/s. AC-3 technology is primarily targeted toward delivery of multiple discrete channels intended for simultaneous presentation to consumers. Channel formats range from 1 to 5.1 channels, and may include a number of associated audio services. The 5.1 channel format consists of five full bandwidth (20 kHz) channels plus an optional low frequency effects (lfe or subwoofer) channel.

A typical application of the algorithm is shown in Fig. 41.1. In this example, a 5.1 channel audio program is converted from a PCM representation requiring more than 5 Mbps (6 channels $\times$ 48 kHz $\times$ 18 bits = 5.184 Mbps) into a 384 kbps serial bit stream by the AC-3 encoder. Satellite transmission equipment converts this bit stream to an RF transmission which is directed to a satellite transponder. The amount of bandwidth and power required by the transmission has been reduced by more than a factor of 13 by the AC-3 digital compression. The signal received from the satellite is demodulated back into the 384 kbps serial bit stream, and decoded by the AC-3 decoder. The result is the original 5.1 channel audio program.

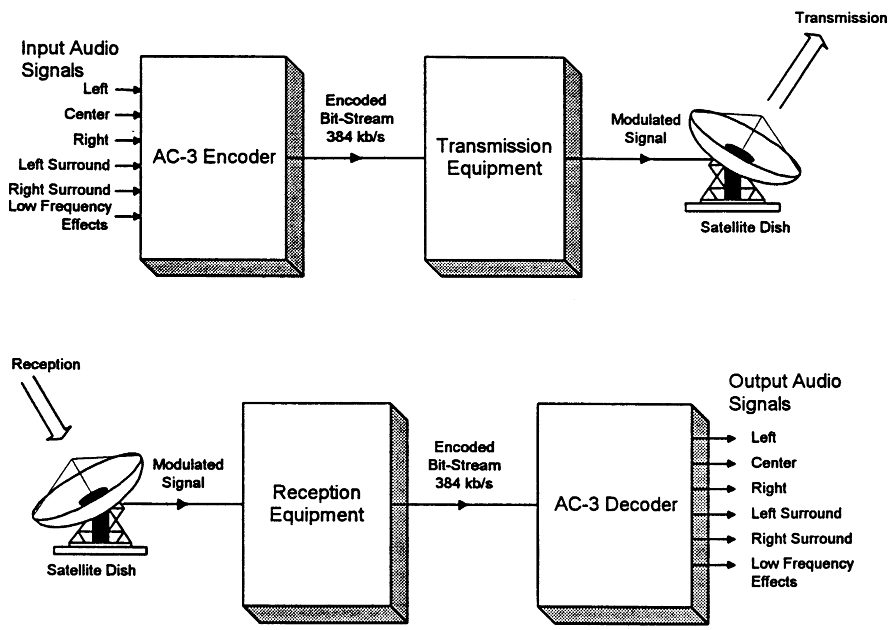


FIGURE 41.1: Example application of satellite transmission using AC-3.

There are a diverse set of requirements for a coder intended for widespread application. While the most critical members of the audience may be anticipated to have complete 6-speaker multi-channel reproduction systems, most of the audience may be listening in mono or stereo, and still others will have three front channels only. Some of the audience may have matrix-based (e.g., Dolby Surround) multi-channel reproduction equipment without discrete channel inputs, thus requiring a dual-channel matrix-encoded output from the AC-3 decoder. Most of the audience welcomes a restricted dynamic range reproduction, while a few in the audience will wish to experience the full dynamic range of the original signal. The visually and hearing impaired wish to be served. All of these and other diverse needs were considered early in the AC-3 design process. Solutions to these requirements have been incorporated from the beginning, leading to a self-contained and efficient system.

As an example, one of the more important listener features built-in to AC-3 is dynamic range compression. This feature allows the program provider to implement subjectively pleasing dynamic range reduction for most of the intended audience, while allowing individual members of the audience the option to experience more (or all) of the original dynamic range. At the discretion of the program originator, the encoder computes dynamic range control values and places them into the AC-3 bit stream. The compression is actually applied in the decoder, so the encoded audio has full dynamic range. It is permissible (under listener control) for the decoder to fully or partially apply the dynamic range control values. In this case, some of the dynamic range will be limited. It is also permissible (again under listener control) for the decoder to ignore the control words, and hence reproduce full-range audio. By default, AC-3 decoders will apply the compression intended by the program provider.

Other user features include decoder downmixing to fewer channels than were present in the bit stream, dialog normalization, and Dolby Surround compatibility. A complete description of these features and the rest of the ATSC Digital Audio Compression Standard is contained in [1].

AC-3 achieves high coding gain (the ratio of the encoder input bit-rate to the encoder output bit-rate) by quantizing a frequency domain representation of the audio signal. A block diagram of this process is shown in Fig. 41.2. The first step in the encoding process is to transform the representation of audio from a sequence of PCM signal sample blocks into a sequence of frequency coefficient blocks. This is done in the analysis filter bank as follows. Signal sample blocks of length 512 are multiplied by a set of window coefficients and then transformed into the frequency domain. Each sample block is overlapped by 256 samples with the two adjoining blocks. Due to the overlap, every PCM input sample is represented in two adjacent transformed blocks. The frequency domain representation includes decimation by an extra factor of two so that each frequency block contains only 256 coefficients. The individual frequency coefficients are then converted into a binary exponential notation as a binary exponent and a mantissa. The set of exponents is encoded into a coarse representation of the signal spectrum which is referred to as the spectral envelope. This spectral envelope is processed by a bit allocation routine to calculate the amplitude resolution required for encoding each individual mantissa. The spectral envelope and the quantized mantissas for 6 audio blocks (1536 audio samples) are formatted into one AC-3 synchronization frame. The AC-3 bit stream is a sequence of consecutive AC-3 frames.

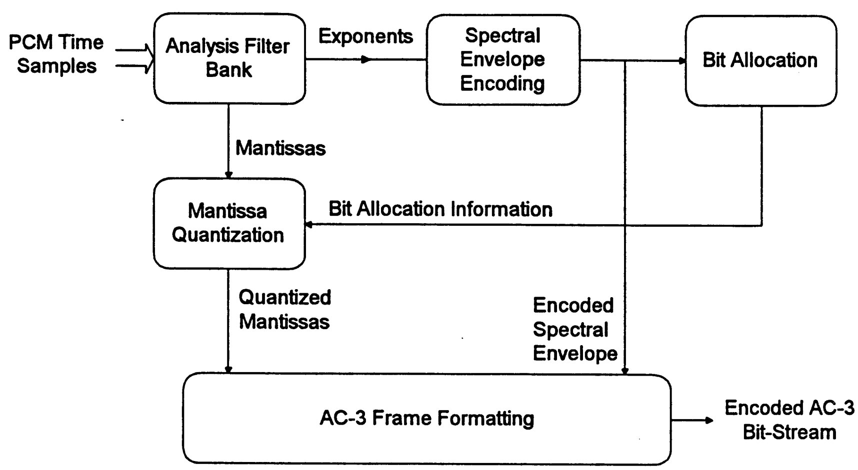


FIGURE 41.2: The AC-3 Encoder.

The decoding process is essentially a mirror-inverse of the encoding process. The decoder, shown in Fig. 41.3, must synchronize to the encoded bit stream, check for errors, and deformat the various types of data such as the encoded spectral envelope and the quantized mantissas. The spectral envelope is decoded to reproduce the exponents. The bit allocation routine is run and the results used to unpack and dequantize the mantissas. The exponents and mantissas are recombined into frequency coefficients, which are then transformed back into the time domain to produce decoded PCM time samples. Figs. 41.2 and 41.3 present a somewhat simplified, high-level view of an AC-3 encoder and decoder.

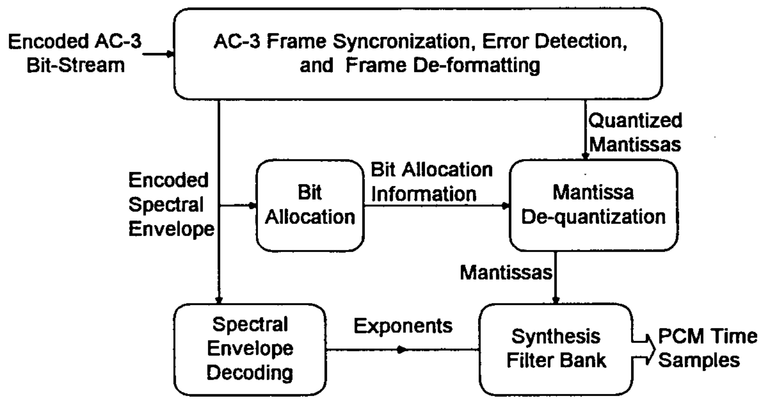


FIGURE 41.3: The AC-3 Decoder.

Table 41.1 presents the different channel formats that are accommodated by AC-3. The three-bit control variable acmod is embedded in the bit stream to convey the encoder channel configuration to the decoder. If acmod is '000', then two completely independent program channels (dual mono) are encoded into the bit stream (referenced as Ch1, Ch2). The traditional mono and stereo formats are denoted when acmod equals '001' and '010', respectively. If acmod is greater than '100', the bit stream format includes one or more surround channels. The optional lfe channel is enabled/disabled by a separate control bit called lfeon.

**TABLE 41.1** AC-3 Audio Coding Modes

| acmod | Audio coding mode | Number of full bandwidth channels | Channel array ordering |
|---|---|---|---|
| '000' | 1 + 1 | 2 | Ch1, Ch2 |
| '001' | 1/0 | 1 | C |
| '010' | 2/0 | 2 | L, R |
| '011' | 3/0 | 3 | L, C, R |
| '100' | 2/1 | 3 | L, R, S |
| '101' | 3/1 | 4 | L, C, R, S |
| '110' | 2/2 | 4 | L, R, SL, SR |
| '111' | 3/2 | 5 | L, C, R, SL, SR |

Table 41.2 presents the different bit-rates that are accommodated by AC-3. The six-bit control variable frmsizecod is embedded in the bit stream to convey the encoder bit-rate to the decoder. In principle, it is possible to use the bit-rates in Table 41.2 with any of the channel formats from Table 41.1. However, in high-quality applications employing the best known encoder, the typical bit-rate for 2 channels is 192 kb/s, and for 5.1 channels is 384 kb/s. As AC-3 encoding technologies mature in the future, these bit-rates can be expected to drop farther.

TABLE 41.2 AC-3 Audio Coding Bit-Rates

| frmsizecod | Nominal bit-rate (kb/sec) | frmsizecod | Nominal bit-rate (kb/sec) | frmsizecod | Nominal bit-rate (kb/sec) |
|---|---|---|---|---|---|
| 0 | 32 | 14 | 112 | 28 | 384 |
| 2 | 40 | 16 | 128 | 30 | 448 |
| 4 | 48 | 18 | 160 | 32 | 512 |
| 6 | 56 | 20 | 192 | 34 | 576 |
| 8 | 64 | 22 | 224 | 36 | 640 |
| 10 | 80 | 24 | 256 | | |
| 12 | 96 | 26 | 320 | | |

## 41.2 Bit Stream Syntax

An AC-3 serial coded audio bit stream is composed of a contiguous sequence of synchronization frames. A synchronization frame is defined as the minimum-length bit stream unit which can be decoded independently of any other bit stream information. Each synchronization frame represents a time interval corresponding to 1536 samples of digital audio (for example, 32 ms at a sampling rate of 48 kHz). All of the synchronization codes, preamble, coded audio, error correction, and auxiliary information associated with this time interval is completely contained within the boundaries of one audio frame.

Figure 41.4 presents the various bit stream elements within each synchronization frame. The five different components are: SI (Synchronization Information), BSI (Bit Stream Information), AB (Audio Block), AUX (Auxiliary Data Field), and CRC (Cyclic Redundancy Code). The SI and CRC fields are of fixed-length, while the length of the other four depends upon programming parameters such as the number of encoded audio channels, the audio coding mode, and the number of optionally-conveyed listener features. The length of the AUX field is adjusted by the encoder such that the CRC element falls on the last 16-bit word of the frame. A summary of the bit stream elements and their purpose is provided in Table 41.3.

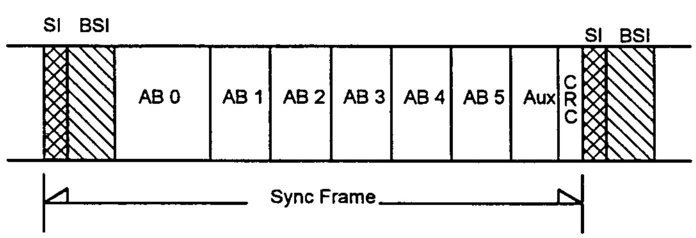


FIGURE 41.4: AC-3 synchronization frame.

The number of bits in a synchronization frame (frame length) is a function of sampling rate and total bit-rate. In a conventional encoding scenario, these two parameters are fixed, resulting in synchronization frames of constant length. However, AC-3 also supports variable-rate audio applications, as will be discussed shortly.

Each Audio Block contains coded information for 256 samples from each input channel. Within one synchronization frame, the AC-3 encoder can change the relative size of the six Audio Blocks depending on audio signal bit demand. This feature is particularly useful when the audio signal is non-stationary over the 1536-sample synchronization frame. Audio Blocks containing signals with a high bit demand can be weighted more heavily than others in the distribution of the available bits (bit pool) for one frame. This feature provides one mechanism for local variation of bit-rate while keeping the overall bit-rate fixed.

**TABLE 41.3** AC-3 Bit Stream Elements

| Bit stream element | Purpose | Length (bits) |
|---|---|---|
| SI | Synchronization information — Header at the beginning of each frame containing information needed to acquire and maintain bit stream synchronization. | 40 |
| BSI | Bit stream information — Preamble following SI containing parameters describing the coded audio service, e.g., number of input channels (acmod), dynamic compression control word (dynrng), and program time codes (timecod1, timecod2). | Variable |
| AB | Audio block — Coded information pertaining to 256 quantized samples of audio from all input channels. There are six audio blocks per AC-3 synchronization frame. | Variable |
| Aux | Auxiliary data field — Block used to convey additional information not already defined in the AC-3 bit stream syntax. | Variable |
| CRC | Frame error detection field — Error check field containing a CRC word for error detection. An additional CRC word is located in the SI header, the use of which is optional. | 17 |

In applications such as digital audio storage, an improvement in audio quality can often be achieved by varying the bit-rate on a long-term basis (more than one synchronization frame). This can also be realized in AC-3 by adjusting the bit-rate of different synchronization frames on a signal-dependent basis. In regions where the audio signal is less bit-demanding (for example, during quiet passages), the frame bit-rate (frmsizecod) is reduced. As the audio signal becomes more demanding, the frame bit-rate is increased so that coding distortion remains inaudible. Frame-to-frame bit-rate changes selected by the encoder are automatically tracked by the decoder.

## 41.3 Analysis/Synthesis Filterbank

The design of an analysis/synthesis filterbank is fundamental to any frequency-domain audio coding system. The frequency and time resolution of the filterbank play critical roles in determining the achievable coding gain. Of significant importance as well are the properties of critical sampling and overlap-add reconstruction. This section discusses these properties in the context of the AC-3 multichannel audio coding system.

Of the many considerations involved in filterbank design, two of the most important for audio coding are the window shape and the impulse response length. The window shape affects the ability to resolve frequency components which are in close proximity, and the impulse response length affects the ability to resolve signal events which are short in time duration. For transform coders, the impulse response length is determined by the transform block length.

A long transform length is most suitable for input signals whose spectrum remains stationary, or varies only slowly with time. A long transform length provides greater frequency resolution, and hence improved coding performance for such signals. On the other hand, a shorter transform length, possessing greater time resolution, is more effective for coding signals that change rapidly in time. The best of both cases can be obtained by dynamically adjusting the frequency/time resolution of the transform depending upon spectral and temporal characteristics of the signal being coded. This behavior is very similar to that known to occur in human hearing, and is embodied in AC-3.

The transform selected for use in AC-3 is based on a 512-point Modified Discrete Cosine Transform (MDCT) [2]. In the encoder, the input PCM block for each successive transform is constructed by taking 256 samples from the last half of the previous audio block and concatenating 256 new samples from the current block. Each PCM block is therefore overlapped by 50% with its two neighbors. In the decoder, each inverse transform produces 512 new PCM samples, which are subsequently windowed, 50% overlapped, and added together with the previous block. This approach has the desirable property of crossfade reconstruction, which reduces waveform discontinuities (and audible distortion) at block boundaries.

### 41.3.1 Window Design

To achieve perfect-reconstruction with a unity-gain MDCT transform filterbank, the shape of the analysis and synthesis windows must satisfy two design constraints. First of all, the analysis/synthesis windows for two overlapping transform blocks must be related by:

$$a_i(n+N/2)s_i(n+N/2)+a_{i+1}(n)s_{i+1}(n)=1, \qquad n=0,\ldots,N/2-1 \tag{41.1}$$

where $a_i(n)$ is the analysis window, $s_i(n)$ is the synthesis window, $n$ is the sample number, $N$ is the transform block length, and $i$ is the transform block index. This is the well-known condition that the analysis/synthesis windows must add so that the result is flat [3]. The second design constraint is:

$$a_i(N/2-n-1)s_i(n)-a_i(n)s_i(N/2-n-1)=0, \qquad n=0,\ldots,N/2-1 \tag{41.2}$$

This constraint must be satisfied so that the time-domain alias distortion introduced by the forward transform is completely canceled during synthesis.

To design the window used in AC-3, a convolution technique was employed which guarantees that the resultant window satisfies Eq. (41.1). Equation (41.2) is then satisfied by choosing the analysis and synthesis windows to be equal. The procedure consists of convolving an appropriately chosen symmetric kernel window with a rectangular window. The window obtained by taking the square root of the result satisfies Eq. (41.1). Tradeoffs between the width of the window main-lobe and the ultimate rejection can be made simply by choosing different kernel windows. This method provides a means for transforming a kernel window having desirable spectral analysis properties (such as in [4]) into one satisfying the MDCT window design constraints.

The window generation technique is based on the following equation:

$$a_i(n)=s_i(n)=\sqrt{\frac{\displaystyle\sum_{j=L}^{M}[w(j)r(n-j)]}{\displaystyle\sum_{j=0}^{K}[w(j)]}} \quad \text{for } n=0,\ldots,N-1, \text{ where} \tag{41.3}$$

$$L=\begin{cases} 0 & 0\le n<N-K \\ n-N+K+1 & N-K\le n<N \end{cases}$$

$$M=\begin{cases} n & 0\le n<K \\ K & K\le n<N \end{cases}$$

In this equation, $w(n)$ is the kernel window of length $K+1$, $r(n)$ is a rectangular window of length $N-K$, $N$ is the transform sample block length, and $K$ is the width of the (non-flat) transition region in the resulting window (note that $K$ must satisfy $0\le K\le N/2$). The rectangular window is defined as:

$$r(n)=\begin{cases} 0 & 0\le n<(N/2-K)/2 \;\text{ and }\; (3N/2-K)/2\le n<N-K \\ 1 & (N/2-K)/2\le n<(3N/2-K)/2 \end{cases} \tag{41.4}$$

The rectangular window is defined to contain $(N/2-K)/2$ zeros, followed by $N/2$ unity samples, followed by another $(N/2-K)/2$ zeros. The AC-3 window uses $K=N/2$, implying the transition region length is one-half the total window length.

The Kaiser-Bessel window is used as the kernel in designing the AC-3 analysis/synthesis windows because of its near-optimal transition band slope and good ultimate rejection characteristic. A scalar parameter $\alpha$ in the Kaiser-Bessel window definition can be adjusted to vary this ratio. The AC-3 window uses $\alpha = 5$.

The selection of the Kaiser-Bessel window function and alpha factor used for the AC-3 algorithm is determined by considering the shape of masking template curves. A useful criterion is to use a filter response which is at or below the worst-case combination of all masking templates [5]. Such a filter response is advantageous in reducing the number of bits required for a given level of audio quality. When the filter response is at or below the worst-case combination of all masking templates, the number of bits assigned to transform coefficients adjacent to each tonal component is reduced.

### 41.3.2 Transform Equations

The transform employed in AC-3 is an extension of the oddly-stacked TDAC (OTDAC) filter bank reported by Princen and Bradley [2]. The extension involves the capability to switch transform block length from $N = 512$ to 256 for audio signals with rapid amplitude changes. As originally formulated by Princen, the filter bank operates with a time-invariant block-length, and therefore has constant time/frequency resolution. An adaptive time/frequency resolution transform can be implemented by changing the time offset of the transform basis functions during short blocks. The time offset is selected to preserve critical sampling and perfect reconstruction before, during, and following transform length changes.

Prior to transforming the audio signal from time to frequency dimension, the encoder performs an analysis of the spectral and/or temporal nature of the input signal and selects the appropriate block length. A one-bit code per channel per Audio Block is embedded in the bit stream which conveys length information: (blksw $=$ 0 or 1 for 512 or 256 samples, respectively). The decoder uses this information to deformat the bit stream, reconstruct the mantissa data, and apply the appropriate inverse transform equations.

Transforming a long block (512 samples) produces 256 unique transform coefficients. Short blocks are constructed starting with 512 windowed audio samples and splitting them into two abutting subblocks of length 256. Each subblock is transformed independently, producing 128 unique non-zero transform coefficients. Hence, the total number of transform coefficients produced in the short-block mode is identical to that produced in long-block mode, but with doubly improved temporal resolution. Transform coefficients from the two subblocks are interleaved together on a coefficient-by-coefficient basis. This block is quantized and transmitted identically to a single long block.

A similar, mirror image procedure is applied in the decoder. Quantized transform coefficients for the two short transforms arrive in the decoder interleaved in frequency. The decoder processes the interleaved sequences identically to long-block sequences, except during the inverse transformation as described below.

A definition of the AC-3 forward transform equation for long and short blocks is:

$$X(k) = 1/N \sum_{n=0}^{N-1} x(n) \cos((2\pi/N)(k+1/2)(n+n_0)), \quad k = 0, 1, \ldots, N-1\,, \tag{41.5}$$

where $n$ is the sample index, $k$ is the frequency index, $x(n)$ is the windowed sequence of $N$ audio samples, and $X(k)$ is the resulting sequence of transform coefficients.

The corresponding inverse transform equation for long and short blocks is:

$$y(n) = \sum_{k=0}^{N-1} X(k) \cos((2\pi/N)(k+1/2)(n+n_0)), \quad n = 0, 1, \ldots, N-1 \tag{41.6}$$

Parameter $n_0$ represents a time offset of the modulator basis vectors used in the transform kernel. For long blocks, and for the second of each short block pair, $n_0 = 257/2$. For the first short block, $n_0 = 1/2$.

When $x(n)$ in Eq. (41.5) is real, $X(k)$ is odd-symmetric for the MDCT. Therefore, only $N/2$ unique non-zero transform coefficients are generated for each new block of $N$ samples. Accordingly, some information is lost during the transform, which ultimately leads to an alias component in $y(n)$. However, with an appropriate choice of $n_0$, and in the absence of transform coefficient quantization, the aliasing is completely canceled during the window/overlap/add procedure following the inverse transform. Hence, the AC-3 filterbank has the properties of critical sampling and perfect reconstruction. A fundamental advantage of this approach is that 50% frame overlap is achieved without increasing the required bit-rate. Any non-zero overlap used with conventional transforms (such as the DFT or standard DCT) precludes critical sampling, generally resulting in a higher bit-rate for the same level of subjective quality.

Several memory and computation-efficient techniques are available for implementing the AC-3 forward and inverse transforms (for example, see [6]). The most efficient ones can be derived by rewriting Eqs. (41.5) and (41.6) in the form of an $N$-point DFT and IDFT, respectively, combined with two complex vector multiplies. The DFT and IDFT can be efficiently computed using an FFT and IFFT, respectively. Two properties further reduce the fast transform length. First, the input signal is real, and second, the $N$-length sequence $y(n)$ contains only $N/2$ unique samples. When these two properties are combined, the result is an $N/4$-point complex FFT or IFFT. The AC-3 decoder filter bank computation rate is about 13 multiply-accumulate operations per sample per channel, including the window/overlap/add. This computation rate remains virtually unchanged during block length changes.

## 41.4 Spectral Envelope

The most basic form of audio information conveyed by an AC-3 bit stream consists of quantized frequency coefficients. The coefficients are delivered in floating-point form, whereby each consists of an exponent and a mantissa. The exponents from one audio block provide an estimate of the overall spectral content as a function of frequency. This representation is often termed a spectral envelope. This section describes spectral envelope coding strategies in AC-3, and explores an important relationship between exponent coding and mantissa bit allocation.

Due to the inherent variety of audio spectra within one frame, the AC-3 spectral envelope coding scheme contains significant degrees of freedom. In essence, the six spectral envelopes contained in one frame represent a two-dimensional signal, varying in time (block index) and frequency. AC-3 spectral envelope coding provides for variable coarseness of representation in both dimensions. In the frequency dimension, either one, two, or four mantissas can be shared by one floating-point exponent. In the time dimension, any two or more consecutive audio blocks from one frame can share common set of exponents.

The concepts of spectral envelope coding and bit allocation are closely linked in AC-3. More specifically, the effectiveness with which mantissa bits are utilized can depend greatly upon the encoder's choice of spectral envelope coding. To see this, note that the dominant contributors to the total bit-rate for a frame are the audio exponents and mantissas. Sharing exponents in either the time or frequency dimension, or both, reduces the total cost of exponent transmission for one frame. More liberal use of exponent sharing therefore frees more bits for mantissa quantization. Conversely, retransmitting exponents increases the total cost of exponent transmission for one frame relative to mantissa quantization. Furthermore, the block positions at which exponents are retransmitted can significantly alter the effectiveness of mantissa bit assignments among the various audio blocks. As will be seen later in Section 41.6, bit assignments are derived in part from the coded spectral envelope.

In summary, the encoder decisions regarding when to use frequency or time exponent sharing, and when to retransmit exponents depend upon signal conditions. Collectively, these decisions are called exponent strategy.

For short-term stationary signals, the signal spectrum remains substantially invariant from block-to-block. In this case, the AC-3 encoder transmits exponents once in audio block 0, and then typically reuses them for blocks 1-5. The resulting bit allocation would generally be identical for all 6 blocks, which is appropriate for these signal conditions.

For short-term non-stationary signals, the signal spectrum changes significantly from block-to-block. In this case, the AC-3 encoder transmits exponents in block 0 and typically in one or more other blocks as well. In this case, exponent retransmission produces a time trajectory of coded spectral envelopes which better matches dynamics of the original signal. Ultimately, this results in a quality improvement if the cost of exponent retransmission is less than the benefit of redistributing mantissa bits among blocks.

Exponent strategy decisions can be based, for example, on a cost-benefit analysis for each frame. The objective of such an analysis would be to minimize a cost-benefit ratio by considering encoding parameters such as total available bit-rate, audibility of quantization noise (noise-to-mask ratio), exponent coding mode for each audio block (reuse, D15, D25, or D45), channel coupling on/off, and reconstructed audio bandwidth.

The block(s) at which bit assignment updates occur is governed by several different parameters, but primarily by the exponent strategy fields. AC-3 bit streams contain coded exponents for up to five independent channels, and for the coupling and low frequency effects channels (when enabled). The respective exponent strategy fields are called chexpstr[ch], cplexpstr, and lfeexpstr. Bit allocation updates are triggered if the state of any one or more strategy flags is D15, D25, or D45; however, updates can be triggered in between shared exponent block boundaries as well.

Exponents are 5-bit values which indicate the number of leading zeros in the binary representation of a frequency coefficient. For the D15 exponent strategy, the unsigned integer exponent $e(i)$ represents a scale factor for the $i$th mantissa, equal to $2^{-e(i)}$. Frequency coefficients are normalized in the encoder by multiplying by $2^{e(i)}$, and denormalized in the decoder by multiplying by $2^{-e(i)}$. Exponent values are allowed to range from 0 (for the largest value coefficients with no leading zeros) to 24. Exponents for coefficients which have more than 24 leading zeros are fixed at 24, and the corresponding mantissas are allowed to have leading zeros. Exponents require 5 bits in order to represent all allowed values.

AC-3 exponent transmission employs differential coding, in which the exponents for a channel are differentially coded across frequency. The first exponent of a full bandwidth or lfe channel is always sent as a 4-bit absolute value, ranging from 0-15. The value indicates the number of leading zeros of the first (DC term) transform coefficient. Successive exponents (ascending in frequency) are sent as differential values which must be added to the prior exponent value in order to form the next absolute value.

The differential exponents are combined into groups in the audio block. The grouping is done by one of three methods, D15, D25, or D45. The number of grouped differential exponents placed in the audio block for a particular channel depends on the exponent strategy and on the frequency bandwidth information for that channel. The number of exponents in each group depends only on the exponent strategy.

Exponent strategy information for every channel is included in every AC-3 audio block. Information is never shared across frames, so block 0 will always contain a strategy indication for each channel.

The three exponent strategies provide a tradeoff between bit-rate required for exponents, and their frequency resolution. The overall exponent bit-rate for a frame depends on the exponent strategy, the number of blocks over which the exponents are shared, and the audio signal bandwidth. Table 41.4 presents the per-coefficient bit-rate required to transmit the spectral envelope for each strategy, and for each block share interval. The D15 mode provides the finest frequency resolution (one exponent per frequency coefficient), while the D45 mode consumes the lowest per-coefficient bit-rate.

**TABLE 41.4** Exponent Bit-Rate for Different Exponent Strategies

| Exponent strategy | Share interval (number of audio blocks) | | | | | |
|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 |
| D15 | 2.33 | 1.17 | 0.78 | 0.58 | 0.47 | 0.39 |
| D25 | 1.17 | 0.58 | 0.39 | 0.29 | 0.23 | 0.19 |
| D45 | 0.58 | 0.29 | 0.19 | 0.15 | 0.12 | 0.10 |

Note: Bits per frequency coefficient

The absolute exponents found in the bit stream at the beginning of the differentially coded exponent sets are sent as 4-bit values which have been limited in either range or resolution in order to save one bit. For full bandwidth and lfe channels, the initial 4-bit absolute exponent represents a value from 0 to 15. Exponent values larger than 15 are limited to a value of 15. For the coupled channel, the 5-bit absolute exponent is limited to even values, and the least significant bit is not transmitted. The resolution has been limited to valid values of 0, 2, 4,..., 24. Each differential exponent can take on one of five values: $-2$, $-1$, 0, $+1$, $+2$. This allows deltas of up to $\pm 2$ ($\pm 12$ $dB$) between exponents. These five values are mapped into the values 0, 1, 2, 3, 4 before being grouped, as shown in Table 41.5.

**TABLE 41.5** Mapping of Differential Exponent Values

| Differential exponent | Mapped value, $M_i$ |
|---|---|
| $+2$ | 4 |
| $+1$ | 3 |
| 0 | 2 |
| $-1$ | 1 |
| $-2$ | 0 |

In D15 mode, the above mapping is applied to each individual differential exponent for coding into the bit stream. In D25 mode, each pair of differential exponents is represented by a single mapped value in the bit stream. In this mode, the differential exponent is used once to compute an exponent that is shared between two consecutive frequency coefficients.

The D45 mode is similar to D25 mode except that quadruplets of differential exponents are represented by a single mapped value. Again, the differential exponent is used once to compute an exponent that is shared between four consecutive frequency coefficients.

For all modes, sets of three adjoining (in frequency) mapped values ($M_1$, $M_2$ and $M_3$) are grouped together and coded as a 7-bit unsigned integer I according to the following relation:

$$I = 25M_1 + 5M_2 + M_3 \; . \tag{41.7}$$

Following the exponent strategy fields in the bit stream is a set of 6-bit channel bandwidth codes, chbwcod[ch]. These are only present for independent channels (not in coupling) that have new exponents in the current block. The channel bandwidth code defines the end mantissa bin number for that channel according to the following:

$$\text{endmant[ch]} = ((\text{chbwcod[ch]} + 12)3) + 37 \tag{41.8}$$

Exponent strategy for each full bandwidth channel and the lfe channel can be updated independently. Lfe channel exponents are restricted only to reuse or D15 mode. If a full bandwidth channel is in coupling, exponents up to the start coupling frequency are transmitted. If the full bandwidth channel is not in coupling, exponents up to the channel bandwidth code are transmitted. If coupling is on for any full bandwidth channel, a separate and independent set of exponents is transmitted for the coupling channel. Coupling start and end frequencies are transmitted as 4-bit indices.

## 41.5 Multichannel Coding

In the context of AC-3, multichannel audio is defined as two or more full bandwidth channels that are intended for simultaneous presentation to a listener. Multichannel audio coding offers new opportunities in bit-rate reduction beyond those commonly employed in monophonic coders. The goal of multichannel coding is to compress an audio program by exploiting redundancy between the channels and irrelevancy in the signal while preserving both sound clarity and spatial characteristics of the original program. AC-3 achieves this goal by preserving listener cues which affect perceived directionality of hearing (localization).

The motivation for multichannel audio coding is provided by an understanding of how the ear extracts directional information from an incident sound wave. Hearing research suggests that the auditory system does not evaluate every detail of the complicated interaural signal differences, but rather derives what information is needed from definite, easily recognizable attributes [7]. For example, localization of signals are generally distinguished by:

1. Interaural time differences (ITD)
2. Interaural level differences (ILD)

The ITD cues are caused by the difference between the time of arrival of a sound at both ears. ILD cues are sound pressure level differences caused by a different acoustic transfer functions from the acoustic source to the two ears. Most authors agree ITD is the most important attribute of the audio signal relating to the formation of lateral displacements [7]. For tones below about 800 Hz, perceived lateral displacement is approximately linear as a function of the difference in the time of arrival for the two ears, up to an ITD of 600 $\mu$sec. Full lateral displacement is obtained with an ITD of approximately 630 $\mu$sec. At any given time, the auditory event corresponding to the shorter time of arrival is dominant.

The ear is able to evaluate spectral components of the ear input signals individually with respect to ITD. Lateral displacement of the auditory event attainable for pure tones is most perceptible below 800 Hz. However, for some non-tonal signals above 800 Hz, such as narrowband noise, the ear is still able to detect ITDs. In this case, the interaural temporal displacement of the energy envelope of the signal is generally regarded as the criterion involved. Experiments have indicated that the ITD of the signal's fine temporal structure contributes negligibly to localization; instead, the ear evaluates only the energy envelopes. The processing occurs individually in each of a multiplicity of spectral bands. The spectrum is dissected to a degree determined by the finite spectral resolution of the inner ear. Then, the envelopes of the separate spectral components are evaluated individually. These experimental results form the basis for the use of channel coupling in AC-3.

### 41.5.1 Channel Coupling

Channel coupling is a method for reducing the bit-rate of multichannel programs by summing two or more correlated channel spectra in the encoder. Frequency coefficients for the single combined (coupled) channel are transmitted in place of the individual channel spectra, together with additional side information. The side information consists of a unique set of coupling coefficients for each channel. In the decoder, frequency coefficients for each output channel are computed by multiplying the coupled channel frequency coefficients by the coupling coefficients for that channel. Coupling coefficients are computed in the encoder in a manner which preserves the short-time energy envelope of the original signals, thereby preserving spatialization cues used by the listener.

Coupling is active only above the coupling start frequency; below this boundary, frequency coefficients are coded independently. The coupling start frequency can be changed from one audio block to the next, and coupling can be disabled if desired. Any combination of two or more full bandwidth channels can be coupled; each channel has an associated channel-in-coupling bit to indicate if it has been included in the coupling channel.

Channel coupling is intended for use only when independent channel coding at the given bit-rate and desired audio bandwidth would result in audible artifacts. As the audio bit-rate is lowered with a fixed bandwidth of 20 kHz, a point is eventually reached where audible coding errors will occur for critical signals. In these circumstances, channel coupling reduces the need for encoders to take more drastic measures to eliminate artifacts, such as lowering the audio bandwidth.

A diagram depicting the encoder coupling procedure for the case of three input channels is depicted in Fig. 41.5. The coupling channel is formed as the vector summation of frequency coefficients from all channels in coupling. An optional signal-dependent phase adjustment is applied to the frequency coefficients prior to summation so that phase cancellation does not occur. For each input channel in coupling, the AC-3 encoder then calculates the power of the original signal and the coupled signal. The power summation is performed individually on a number of bands. For the simplified case of Fig. 41.5, there are two such bands. In a typical application the number of bands is 14, but can vary between 1 and 18. Next, the power ratio between the original signal and the coupled channel is computed for each input channel and each band. Denoted a coupling coordinate, these ratios are quantized and transmitted to the decoder.

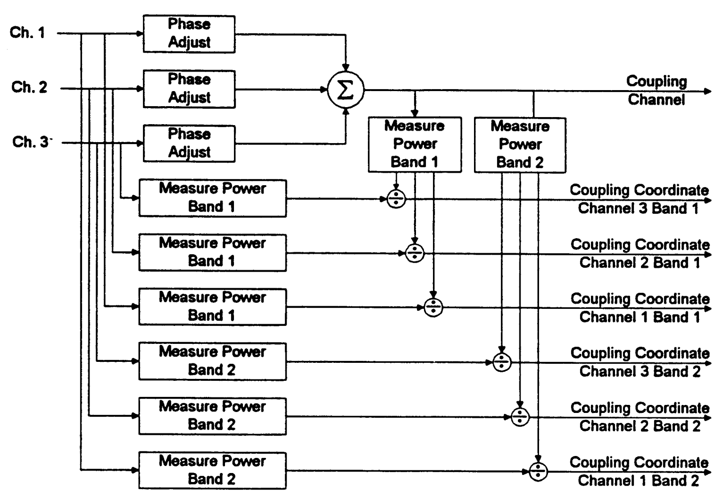


FIGURE 41.5: Block diagram of encoder coupling for three input channels.

To reconstruct the spectral coefficients corresponding to one channel's worth of transform data, quantized spectral coefficients representing the uncoupled portion of the transform block are prepended to a set of scaled coupling channel spectral coefficients. The scaled coupling channel coefficients are generated for each channel by multiplying the coupling coordinates for each band by the received coupling channel coefficients, as shown in Fig. 41.6.

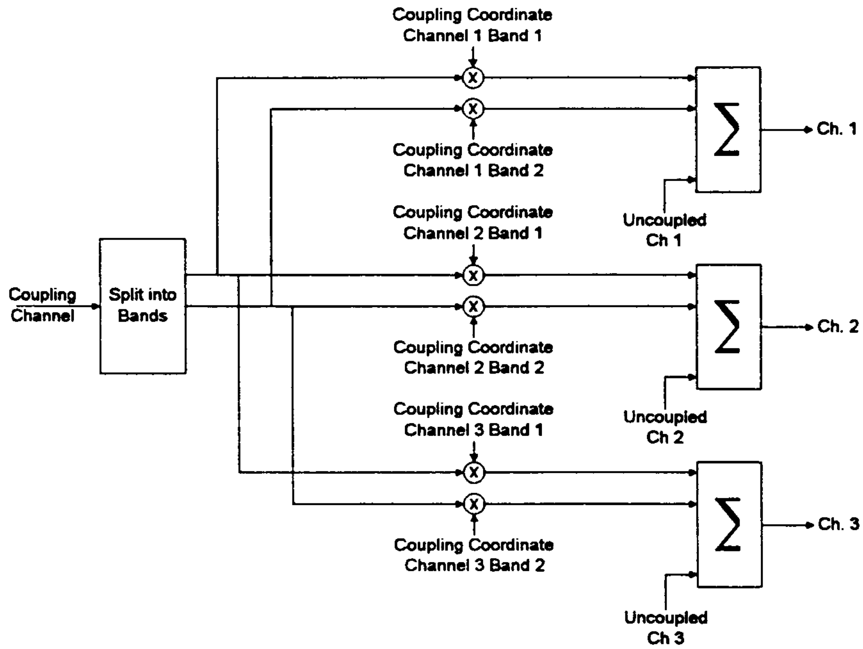


FIGURE 41.6: Block diagram of decoder coupling for three input channels.

Coupling parameters, such as the coupling start/end frequencies and which channels are in coupling, are always transmitted in block 0. They are also optionally transmitted in blocks 1 through 5. Typically only channels with similar spectral shapes are coupled. Level differences between channels are accounted for by the coupling coefficients. It is noteworthy that if in-band spectral differences between coupled input channels are due to level only, the original input channels can still be recovered exactly in the decoder, in the absence of frequency coefficient quantization.

The coupling coefficient dynamic range is $-132$ to $+18$ dB, with step sizes varying between 0.28 and 0.53 dB. The lowest coupling start frequency is 3.42 kHz at a 48 kHz sampling frequency.

### 41.5.2 Rematrixing

Rematrixing in AC-3 is a channel combining technique in which sum and difference signals of highly correlated channels are coded rather than the original channels themselves. That is, rather than code and pack left and right (L and R) in a two channel coder, the encoder constructs:

$$\begin{aligned} L' &= (L+R)/2 \\ R' &= (L-R)/2\,. \end{aligned} \tag{41.9}$$

The usual quantization and data packing operations are then performed on $L'$ and $R'$.

In the decoder, the original $L$ and $R$ signals are reconstructed using the inverse equations:

$$\begin{aligned} L &= L'+R' \\ R &= L'-R'\,. \end{aligned} \tag{41.10}$$

Clearly, if the original stereo signal were identical in both channels (i.e., two-channel mono), $L'$ is identical to $L$ and $R$, and $R'$ is identically zero. Therefore, the $R'$ channel can be coded with very few bits, increasing accuracy in the more important $L'$ channel. Rematrixing is only applicable in the 2/0 encoding mode (acmod = '010').

Rematrixing is particularly important when conveying Dolby Surround encoded programs. Consider again a two channel mono source signal. A Dolby Pro Logic decoder will steer all in-phase information to the center channel, and all out-of-phase information to the surround channel. Without rematrixing, the Pro Logic decoder will receive the signals:

$$\begin{aligned} Q(L) &= L + n_1 \\ Q(R) &= R + n_2 \end{aligned} \tag{41.11}$$

where $n_1$ and $n_2$ are uncorrelated quantization noise sequences added by the process of bit-rate reduction. The Pro Logic decoder will then construct the center and surround channels as:

$$\begin{aligned} C &= ((L + n_1) + (R + n_2))/2 \\ S &= ((L + n_1) - (R + n_2))/2 \end{aligned} \tag{41.12}$$

In the case of the center channel, $n_1$ and $n_2$ add, but remain masked by, the dominant $L + R$ signal. In the surround channel, however, $L - R$ cancels to zero, and the surround speakers reproduce the difference in the quantization noise sequences $(n_1 - n_2)$.

If rematrixing is active, the left and right channels will be reproduced as:

$$\begin{aligned} L &= (L' + n_3) + (R' + n_4) \cong L + n_3 \\ R &= (L' + n_3) - (R' + n_4) \cong R + n_3 \,, \end{aligned} \tag{41.13}$$

where the approximation is made since the quantization noise $n_3 >> n_4$. More importantly, the center and surround channels will be more faithfully reproduced as:

$$\begin{aligned} C &\cong ((L + n_3) + (R + n_3))/2 = (L + R)/2 + n_3 \\ S &\cong ((L + n_3) - (R + n_3))/2 = (L - R)/2 \,. \end{aligned} \tag{41.14}$$

In this case, the quantization noise in the surround channel is much lower in level.

In AC-3, rematrixing is performed independently in separate frequency bands. There are two to four contiguous bands with boundary locations dependent on coupling information. At a sampling rate of 48 kHz, bands 0 and 1 start at 1.17 and 2.30 kHz, respectively. Bands 2 and 3 start at 3.42 and 5.67 kHz. If coupling is not in use, band 3 stops at a frequency given by the channel bandwidth code. The band boundaries scale proportionally for other sampling frequencies.

Rematrixing is never used in the coupling channel. If coupling and rematrixing are simultaneously in use, the highest rematrixing band ends at the coupling start frequency.

## 41.6 Parametric Bit Allocation

The process of distributing a finite number of bits $B$ to a block of $M$ frequency bands so as to minimize a suitable distortion criterion is called bit allocation. The result is a bit assignment $b(k)$, $k =$ 0, 1, ..., $M - 1$, which defines the word length of the frequency coefficient(s) transmitted in the $k$th band. The bit assignment is performed subject to the constraint:

$$\sum_{k=0}^{M-1} b(k) = B \,. \tag{41.15}$$

$B$ is determined from the transmission channel capacity, expressed in bits/sec, the block length, and other parameters as well. Performance gains may be realized by allowing $B$ to vary from block to block depending on signal characteristics.

### 41.6.1 Bit Allocation Strategies

In applications such as digital audio broadcasting and high definition television, one encoder typically distributes programs to many decoders. In these situations, it is advantageous to make the encoder as flexible as possible. If quality improvements are possible even after the decoder design is standardized, the useful life of the coding algorithm can be extended. The bit allocation strategy is a natural candidate for improvement because it plays a crucial role in determining the ultimate quality achievable by a coding algorithm.

One approach to achieving flexibility is to use a forward-adaptive bit allocation strategy, in which the bit assignment $b(k)$ for all bands is explicitly conveyed in the bit stream as side information. A second strategy is termed backward-adaptive allocation, in which $b(k)$ is computed in the encoder and then recomputed in the decoder. Since the computation is based upon quantized information which is transmitted to the decoder anyway, the side information to convey $b(k)$ is not required. The bits that are saved can be used to encode the frequency coefficients themselves.

AC-3 employs an alternative approach [8]. Called parametric bit allocation, the technique combines the advantages of forward and backward-adaptive strategies. It employs a hearing model combining elements from [9] with new features based on more recent psychoacoustic experiments. The term "parametric" refers to the notion that the model is defined by several key variables that influence the masking curve shape and amplitude, and hence the bit assignment. A difference between AC-3 and previous coders is that both the encoder and decoder contain the model, eliminating the need to transmit $b(k)$ explicitly. Only the essential model parameters (psychoacoustic features) are conveyed to the decoder. These parameters can be transmitted with significantly fewer bits than $b(k)$ itself. Furthermore, an improvement path is provided since the specific parameter values are selected by the encoder.

Equally significant, the parametric approach provides latitude for the encoder to adjust the time and frequency resolution of $b(k)$. Bit allocation updates are always present in block 0, and are optionally transmitted in blocks 1 through 5. The frequency resolution of $b(k)$ can be adjusted from 2.7 to 10.7 bands/kHz (one bit assignment every 94 to 375 Hz). The AC-3 encoder typically makes these adjustments in accordance with spectral and temporal changes in the audio signal itself, in a manner similar to the human ear.

Secondary strategy information which also affects block-to-block changes in bit assignment is the bit allocation information, coupling strategy, and SNR offset. This information is always transmitted in block 0, and is optionally transmitted in each subsequent audio block. The presence/absence of the information is controlled by the respective "exists" bits baie, cplstre, and snroffste. The exists bits are set to 1 in blocks 1 to 5 only when a change in strategy results in better audio quality than would be obtained by reusing parameters from the preceding block.

Signal conditions may arise in which the masking curve, and therefore $b(k)$, cannot be sufficiently optimized using the built-in parametric model. Therefore, AC-3 encoders contain a provision for adjusting the masking curve in accordance with an independent psychoacoustic analysis. This is accomplished by transmitting additional bit stream codes, designated as deltas, which convey differences between the two masking curves.

### 41.6.2 Spreading Function Shape

In Schroeder's model for computing a masking threshold [9], one of the key variables influencing the degree of masking of one spectral component by another is the shape of the spreading function. If the spreading function is a unit impulse, the excitation function (and therefore the masking curve) will be identical in shape to the input signal spectrum. This corresponds to the case where no masking whatsoever is assumed, with the result that all frequency coefficients receive the same bit assignment, and the quantization noise spectrum will conform in shape to the input signal spectrum. As the spreading function is broadened, progressively greater degrees of masking are modeled. This yields a noise spectrum which, in general, contains peaks and valleys that are aligned with features of the input signal spectrum, but broadened in character. As the spreading function is flattened further, eventually a limit is reached where the noise spectrum will be white, corresponding to the minimum mean-square error bit assignment. This noise shaping behavior is identical to the concept of [10, 11].

Since the spreading function strongly influences the level and extent of assumed masking, a parametric description is provided in AC-3. The parametric spreading function is one mechanism available to an AC-3 encoder for making compatible adjustments to the masking model.

The range of spreading function parameter variation was obtained by distilling a family of prototype functions from the available masking data [8]. The variation in shape of the four composite masking curves for 0.5, 1, 2, and 4 kHz tones is shown in Fig. 41.7. We have approximated the envelope of upward masking for each composite curve by two linear segments. The spreading function is defined as the point-by-point maximum of the two segments across frequency. In the example shown in Fig. 41.7, the composite curves can be reasonably approximated by choosing an appropriate slope and vertical offset for each linear segment. Hence, four parameters are transmitted in the bit stream to define the spreading function shape.

### 41.6.3 Algorithm Description

The AC-3 bit allocation strategy places the majority of computation in the encoder. For example, the encoder can trade off reconstructed signal bandwidth vs. quantization noise power, change the degree of upward masking of one spectral component by another, modify the bit assignment as a function of acoustic signal level, and adaptively control total harmonic distortion. The bit assignment can also be adjusted according to an arbitrary second masking model, as outlined later in the section describing "Delta Bit Allocation". The encoder iteratively converges on an optimal solution. On the other hand, the decoder makes only one pass through the received parameters and exponent data, and is therefore considerably simpler. The bit assignment is reconstructed in the decoder using only basic two's complement operators: add, compare, arithmetic left/right shift, and table lookup.

#### Frequency Banding

The first step in the masking curve computation is to convert a block of power spectrum samples, taken at equidistant frequency intervals, into a Bark spectrum. This is accomplished by subdividing the power spectrum into multiple frequency bands and then integrating the spectrum samples within each band. The bands are non-uniform in width and derived from the critical-bandwidths defined by Zwicker [12]. The psychoacoustic basis for this procedure is that each critical band corresponds to a fixed distance along the basilar membrane, and therefore to a constant number of auditory nerve fibers.

The summation of the power spectrum samples during linear to Bark frequency conversion requires a linear summation. However, the logarithm of those quantities is most readily available in AC-3. Therefore, a log-adder is employed. The log-addition of two quantities $\log(a)$ and $\log(b)$ is computed using the relation:

$$\log(a+b) = \max(\log(a), \log(b)) + \log(1+e^{d}) \tag{41.16}$$

where $e$ is the logarithm base, and

$$d = |\log(a) - \log(b)| \, . \tag{41.17}$$

The second term on the right side of the equation is implemented as a subtraction $\log(a) - \log(b)$, followed by absolute value and a table lookup. The contents of the table at address $d$ is: $\log(1 + e^{d})$. Therefore, the complete log-addition is performed with only add, compare, and table lookup instructions.

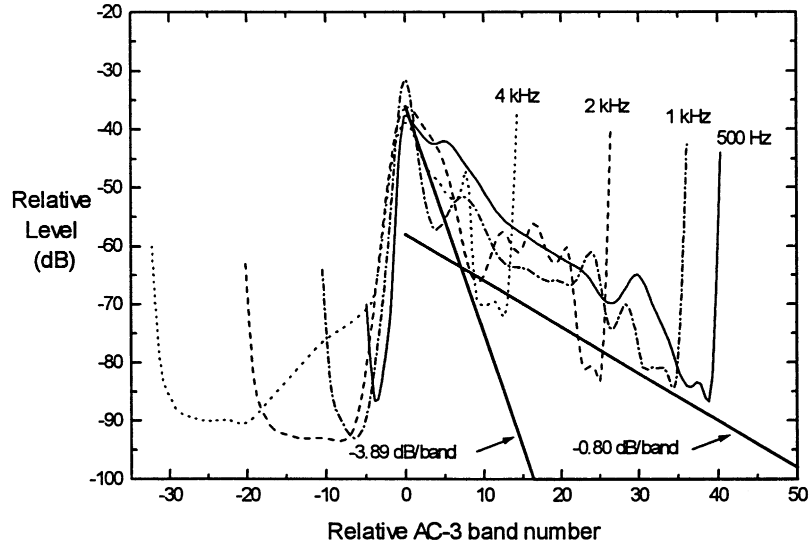


FIGURE 41.7: Comparison between 500 Hz to 4 kHz masking templates and the two slope spreading function.

### Masking Convolution

The technique for modeling masking effects developed by Schroeder specifies convolution. At every frequency point, masking contributions from all other spectral components are weighted and summed. The output of a linear recursive (e.g., IIR) filter may also be viewed as a weighted summation of input samples. Therefore, the convolution of the spreading function with the *linear*-amplitude critical band density in Schroeder's model can be approximated by applying a time-varying linear recursive filter to the spectral components. Upward masking is modeled by filtering the frequency samples from low to high frequency. Downward masking is modeled by filtering input samples in the reverse order. The filter order and coefficients are determined from the desired spreading function. To compute the excitation function using the *logarithmic*-amplitude critical band density used in AC-3, the linear recursive filter is replaced with an equivalent filter which processes logarithmic spectral samples.

The conversion of the linear recursive filter to an equivalent log-domain filter is straightforward. By writing out the difference equation for an IIR filter and taking the logarithm of both sides of the equation, an expression relating the log excitation function with the log power spectrum can be derived. The multiplies and additions of the IIR filter are replaced with additions and log-additions, respectively, in the log domain filter. To implement the two-slope spreading function, two filters are connected in parallel. Each filter implements the characteristic of one of the segments, and the overall excitation value $E(k)$ at band $k$ is computed as the larger of the two filter output samples. The log-domain equations for computing $E(k)$ are:

$$\begin{aligned} x_0 &= (x_0 - d_0) \oplus (P(k) - g_0) \\ x_1 &= (x_1 - d_1) \oplus (P(k) - g_1) \\ E(k) &= \max(x_0, x_1) \end{aligned} \tag{41.18}$$

where $P(k)$ is the log-amplitude power spectrum, $d_0$ and $d_1$ are the dB spreading function decay values for the first and second segment, respectively, $g_0$ and $g_1$ are the dB offsets of the two spreading function segments, and the $\oplus$ symbol denotes log-addition as defined previously. For each of 50 bands, the value of two accumulators $x_0$ and $x_1$ is computed by performing a log-addition of the previous accumulator value, decayed by $d_0$ or $d_1$, and the current power spectrum value scaled by the gain $g_0$ or $g_1$. In AC-3, log-addition is replaced by a maximum operator to reduce computation. This is also more conservative in that additive masking is not assumed.

#### Compensation for Decoder Selectivity

One basis for determining the bit assignments $b(k)$ in a perception-based allocation strategy is to compute the difference between the signal spectrum and the predicted masking curve. An implicit assumption of this technique is that quantization noise in one particular band is independent of bit assignments in neighboring bands. This is not always a reasonable assumption because the finite frequency selectivity and the high degree of overlap between bands in the decoder filter bank cause localized spreading of the error spectrum (leakage from one band into neighboring bands). The effect is predominant at low frequencies where the slope of the masking curve can equal or exceed the slope of the filter bank transition skirts. Hence, under some conditions, a basis other than the difference between the signal spectrum and masking curve is warranted.

As discussed in [8], decoder selectivity compensation has been found to improve subjective coding performance at low frequencies. Accordingly, the AC-3 masking model employs a straightforward, recursive algorithm for applying compensation from 0 to 2.3 kHz. Although the compensation is a filter bank response correction, not a psychoacoustic effect of human hearing, it can be incorporated into the computation of the excitation curve.

#### Parameter Variation

The masking computation primarily represents the backward-adaptive portion of the bit allocation strategy. However a number of parameters defining the masking model are transmitted in the compressed data stream. These represent part of the forward-adaptive portion of the bit allocation strategy. As discussed earlier, the shape of the prototype spreading function is controlled by four parameters, where a pair of parameters correspond to each segment. The first linear segment slope is adjustable between $-2.95$ to $-5.77$ dB per band, with offsets ranging from $-6$ to $-48$ dB. The second segment slope can be adjusted between $-0.70$ to $-0.98$ dB per band, with offsets ranging from $-49$ to $-63$ dB. The syntax of AC-3 allows the first segment to be controlled independently for each channel. Parameters for the second segment are common to all channels. There are 512 unique spreading function shapes available to an AC-3 encoder.

#### Delta Bit Allocation

Another forward-adaptive component of the bit allocation strategy is a parametric adjustment that is optionally made to the masking curve computed by the masking model. This adjustment is conveyed to the decoder with the delta bit allocation. For each channel, the encoder can specify nearly arbitrary adjustments to the computed masking curve at a certain cost in bit-rate that otherwise would be used directly to code audio data. Delta bit allocation is used by the encoder to specify a masking curve, and hence a bit assignment, that cannot be generated by the parametric model alone. This feature is useful, for example, if future research points to masking behavior which cannot be simulated by the existing model. In this case, benefits of any new research can be added to an AC-3 encoder by augmenting the parametric model with the improved one.

Determination of the desired delta bit allocation function is straightforward. In an encoder, both the standard AC-3 masking model and the improved one are run in parallel to determine two masking curves. The desired delta bit allocation function is equal to the difference between the masking curves. It may be advantageous to only approximate the desired difference to reduce the required data expenditure. Note that an encoder should first exhaust the flexibility granted by the non-delta parameters before committing any bits to delta bit allocation. The other parameters are less expensive in terms of bit-rate as they must be transmitted periodically in any case.

The delta function is constrained to have a "stair step" shape. Each tread of the stair step corresponds to the masking level adjustment for an integral number of adjoining one-half Bark bands. Taken together, the stair steps comprise a number of non-overlapping, variable-length segments. The segments are run-length coded for efficient transmission.

## 41.7 Quantization and Coding

All mantissas are quantized to a fixed level of precision indicated by the corresponding bit allocation pointer (bap). Mantissas quantized to 15 or fewer levels use symmetric quantization. Mantissas quantized to more than 15 levels use asymmetric quantization (a conventional two's complement representation).

Some quantized mantissa values are grouped together and encoded into a common codeword. In the case of the 3-level quantizer, 3 quantized values are grouped together and represented by a 5-bit codeword in the data stream. In the case of the 5-level quantizer, 3 quantized values are grouped and represented by a 7-bit codeword. For the 11-level quantizer, 2 quantized values are grouped and represented by a 7-bit codeword. Groups are filled in the order that the mantissas are processed. If the number of mantissas in an exponent set does not fill an integral number of groups, the groups are shared across exponent sets. The next exponent set in the block continues filling the partial groups.

In the encoder, each frequency coefficient is normalized by applying a left-shift equal to its associated exponent (0 to 24). The mantissa is then quantized to a number of levels indicated by the corresponding bap.

Table 41.6 indicates the assignment between bap number and number of quantizer levels. If a bap equals 0, no bits are sent for the mantissa. For more efficient bit utilization, grouping is used for bap values of 1, 2, and 4 (3, 5, and 11 level quantizers).

**TABLE 41.6** Quantizer Levels and Mantissa Bits vs. bap

| bap | Quantizer levels | Mantissa bits (group bits / num in group) |
|---|---|---|
| 0 | 0 | 0 |
| 1 | 3 | 1.67 (5/3) |
| 2 | 5 | 2.33 (7/3) |
| 3 | 7 | 3 |
| 4 | 11 | 3.5 (7/2) |
| 5 | 15 | 4 |
| 6 | 32 | 5 |
| 7 | 64 | 6 |
| 8 | 128 | 7 |
| 9 | 256 | 8 |
| 10 | 512 | 9 |
| 11 | 1024 | 10 |
| 12 | 2048 | 11 |
| 13 | 4096 | 12 |
| 14 | 16,384 | 14 |
| 15 | 65,536 | 16 |

For bit allocation pointer values between 6 and 15, inclusive, asymmetric fractional two's complement quantization is used. No grouping is employed for asymmetrically quantized mantissas.

For bap values of 1 through 5, inclusive, the mantissas are represented by coded values. The coded values are converted to standard 2's complement fractional binary words. The number of bits indicated by a mantissa's bap are extracted from the bit stream and right justified. This coded value is treated as a table index and is used to look up the quantized mantissa value. The resulting mantissa value is right shifted by the corresponding exponent to generate the transform coefficient value.

The AC-3 decoder may use random noise (dither) values instead of quantized values when the number of bits allocated to a mantissa is zero (bap $= 0$). The decoder substitution of random values for the quantized mantissas with bap $= 0$ is conditional on the value of a bit conveyed in the bit stream (dithflag). There is a separate dithflag bit for each transmitted channel. When dithflag $= 1$, the random noise value is used. When dithflag $= 0$, a true zero value is used.

## 41.8 Error Detection

There are several ways in which the AC-3 data may determine that errors are contained within a frame of data. The decoder may be informed of that fact by the transport system which has delivered the data. Data integrity may be checked using the embedded Cyclic Redundancy Check (CRCs) words. Also, some simple consistency checks on the received data can indicate that errors are present. The decoder strategy when errors are detected is user definable. Possible responses include muting, block repeats, frame repeats, or more elaborate schemes based on waveform interpolation to “fill in” missing PCM samples. The amount of error checking performed, and the behavior in the presence of errors are not specified in the AC-3 ATSC standard, but are left to the application and implementation.

Each AC-3 frame contains two 16-bit CRC words. As discussed in Section 41.2, crc1 is the second 16-bit word of the frame, immediately following the synchronization word. crc2 is the last 16-bit word of the frame, immediately preceding the synchronization word of the following frame. crc1 applies to the first 5/8 of the frame, not including the synchronization word. crc2 provides coverage for the last 3/8 of the frame as well as for the entire frame (not including the synchronization word). Decoding of CRC word(s) allows errors to be detected.

The following generator polynomial is used to generate both of the 16-bit CRC words in the encoder:

$$x^{16} + x^{15} + x^2 + 1 \, . \tag{41.19}$$

The CRC calculation may be implemented by one of several standard techniques. A convenient hardware implementation is a linear feedback shift register. Details of this technique are presented in [1].

## References


[1] United States Advanced Television Systems Committee, ATSC Digital Audio Compression Standard (AC-3), Document A/52, December 20, 1995.

[2] Princen, J.P., Johnson, A.W. and Bradley, A.B., Subband/transform coding using filter bank designs based on time domain aliasing cancellation, *IEEE Intl. Conf. on Acoustics, Speech, and Signal Proc.*, 2161–2164, Dallas, 1987.

[3] Crochiere, R.E. and Rabiner, L.R., *Multirate Digital Signal Processing*, Prentice-Hall, Englewood Cliffs, NJ, 1983, 356–358.

[4] Harris, F.J., On the use of windows for harmonic analysis of the discrete fourier transform, *Proc. IEEE*, 66, 51–83, Jan. 1975.

[5] Fielder, L.D., Bosi, M., Davidson, G., Davis, M., Todd, C. and Vernon, S., AC-2 and AC-3: Low-complexity transform-based audio coding, AES Publication *Collected Papers on Digital Audio Bit Rate Reduction*, Neil Gilchrist and Christer Grewin, Eds., 54–72, 1996.

[6] Sevic, D. and Popovic, M., A new efficient implementation of the oddly-stacked Princen-Bradley filter bank, *IEEE Signal Proc. Lett.*, 1(11), Nov. 1994.

[7] Blauert, J., *Spacial Hearing*, The MIT Press, Cambridge, MA, 1974.

[8] Davidson, G., Parametric bit allocation in a perceptual audio coder, presented at the 97th Convention of the Audio Engineering Society, Preprint 3921, November 1994.
[9] Schroeder, M., Atal, B. and Hall, J., Optimizing digital speech coders by exploiting masking properties of the human ear, *J. Acoustical Soc. Am.*, 66(6), Dec. 1979.
[10] Tribolet, J. and Crochiere, R., Frequency domain coding of speech, *IEEE Trans. Acoustics, Speech, and Signal Proc.*, ASSP-27(5), Oct. 1979.
[11] Crochiere, R. and Tribolet, J., Frequency domain techniques for speech coding, *J. Acoustical Soc. Am.*, 66(6), Dec. 1979.
[12] Zwicker, E., Subdivision of the audible frequency range into critical bands (Frequenzgruppen), *J. Acoustical Soc. Am.*, 33, 248, Feb. 1961.