Electgon

A Circuits' Particle

MP3 Format

Theory of the Standard

ELECTGON
www.electgon.com
contact@electgon.com

03.06.2019

Contents

1 Introduction

Smart systems are exploited in many applications recently starting from home applications up to industry and huge manufacturing sectors. These smart systems rely to high extend on information exchange that is carried out using processing systems for analog and digital signal. Audio streams are part of this exchanged information. Many applications are introduced to facilitate dealing with this type of information. MP3 format is serving as best and most common format for these audio application. In this document important introduction about MP3 format is demonstrated to understand how this format works with high efficiency that made it most common format for three decades so far.

2 General Specifications

MPEG-1 was the first international standard for the perceptual coding. This standard was developed by MPEG group in November 1992. It defines coding scheme for the digital audio using three possible techniques (or layers). These three layers are working similarly but the higher layer has more sophisticated configuration than others. MPEG-1 layer III (which is the scope of this project) has more complexity than other layers. It is known also as MP3 and it has successfully found many applications in Video, CD, ISDN, Video games, broadcasting, etc. In order to make this standard applicable in all these applications, MPEG-1 Layer III defined a data representation including a number of options

2.1 Operating mode

MP3 Standard is designed to work with mono and dual channel signals. For Dual channel signals, another extension is also added to cover efficient combined coding of the left and right channels of a stereophonic audio signal. Namely there is stereo joint stereo coding so MP3 allows both mid/side stereo coding and intensity stereo coding. Thus The operating modes are:

- single channel
- dual channel (two independent channels, for example containing different language versions of the audio)
- stereo (no joint stereo coding)
- joint stereo.

2.2 Sampling frequency

MPEG audio compression works on a number of different sampling frequencies. In general MPEG-1 defines audio compression at sampling frequencies 32 kHz, 44.1 kHz and 48 kHz. This means that MP3 encoder is able to handle audio signals that are sampled at these frequencies. MP3 standard is defined to receive 1152 audio samples. If these samples were sampled at 32

KHz, this means that the 1152 audio samples covers 36 ms of the audio source. 44.1 KHz sampling covers each 26 ms. 48 KHz covers 24 ms.

2.3 Bit-rate

Bit rate expresses capacity or number of bits used to represent audio samples. The implementer of the MP3 encoder can choose which bitrate he can use to compress the samples in condition that the selected bitrate within a range of bit-rates from 8 kbit/s up to 320 kbit/s as this the defined range for the standard. MP3 standard gives the option to the implementer to switch bitrate from audio frame to another frame.

2.4 Encoding Process

In very general terms, MP3 is encoding audio signal that are sampled into intervals, each intervals lasts for 24 ms (in case of 48 KHz sampling frequency is used) and results in 1152 audio samples. These 1152 samples pass through filter bank to be analyzed further in frequency domain. This filter bank will result in frequency coefficients that are quantized according to psychoacoustic model. After that it undergoes to further coding using Huffman coding algorithm. These coded frequency coefficients are packed into frames prior to transmission. This process can be depicted in more details in figure 1.

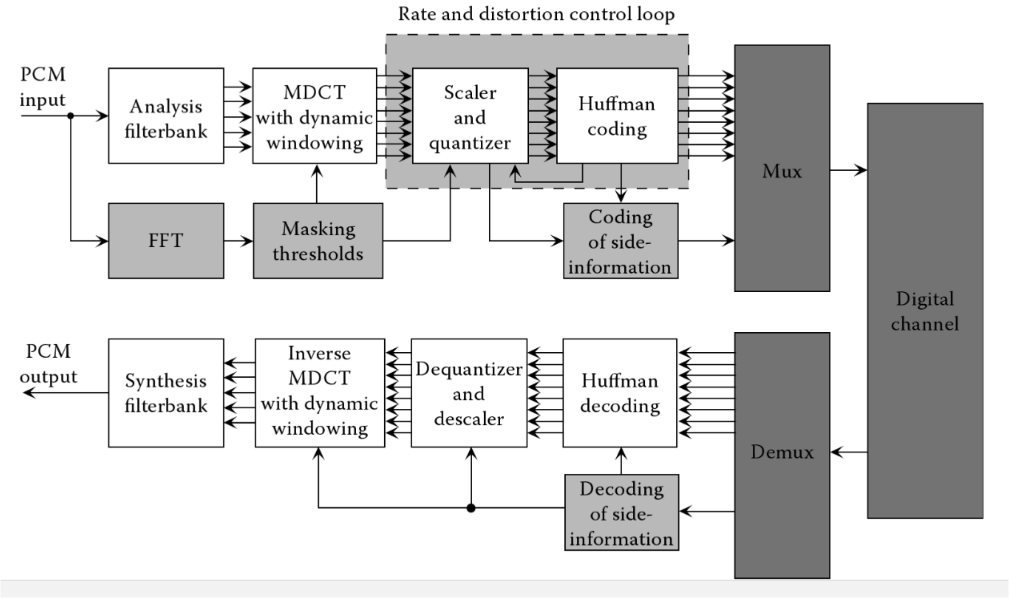

Figure 1: MP3 Encoder Main Architecture[5]

3 Analysis Filter Bank

When we call this stage by "Analysis", this means that this stage is trying to understand input audio signal and analyze its nature for improving coding performance. For Example the input signal is divided into 32 subbands, in parallel another analysis is done based on FFT to calculate energy stored in each subband to know masker components and masking threshold.

This analysis result in what is called Signal to Mask Ratio (SMR). This analysis together with frequency coefficients are provided to next stage for quantization and encoding. At decoder side, the opposite process is done and it is called then "Synthesis" Filters.

Filter Bank that is used in MP3 encoding is Hybrid Filter that is composed of polyphase filter and MDCT filter. Both filters are producing frequency components at the output of its transformation process. The frequency components are called subband samples if the filterbank has low frequency resolution, otherwise they are called frequency lines or frequency coefficients. The polyphase has low frequency resolution so it is said that we get subbands as a result of the polyphase filtering. Polyphase filter is acting as band pass filter. For MP3 it has 32 sub filters. This means that its output is 32 frequency bands. MDCT filter has more resolution so it is able to analyze these 32 subbands into determined frequency coefficients. Typically it produces 18 or 6 frequency coefficients for each subband. Figure 2 shows building blocks for this Hybrid filter.

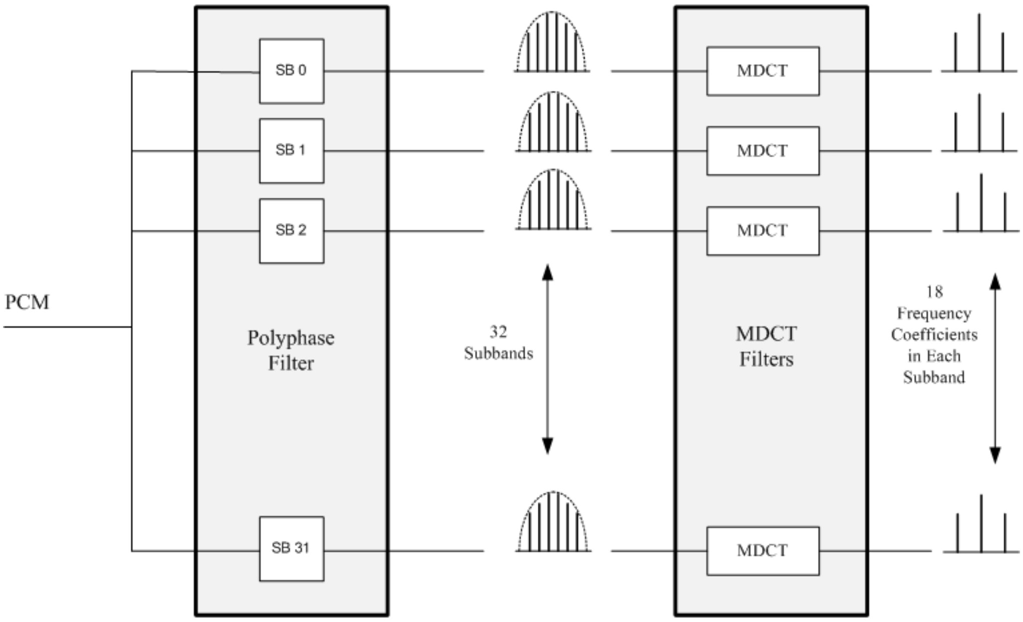

Figure 2: Filter Bank Basic Blocks

4 MDCT Functions

4.0.1 Antialiasing

When reconstructing original signal at decoder side again, some aliasing effects can appear because of missed frequency coefficient that were filtered out during encoding process. MDCT is used to overcome this point by applying overlapping between current filtered frame and the following frame. To make it clear, the Polyphase filter is filtering frame by frame (each frame is 1152 audio samples) but for the MDCT it processes two consecutive frames to make overlapping between them. MDCT is performing 50% overlapping so eventually MDCT will release frame by frame is can be demonstrated from figure 3 where M can be 1152 in our case.

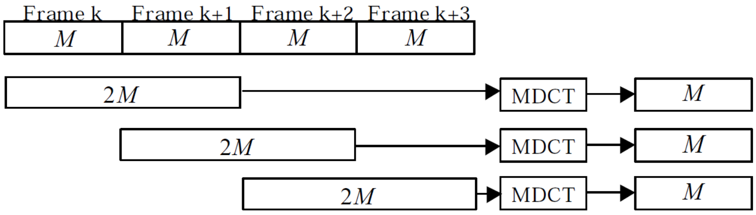

Figure 3: Overlapping in MDCT[6]

4.0.2 Window Switching

Another important role of the MDCT filter comes from its window function. In MP3 standard, MDCT is designed to apply four different window functions to the input spectrum as shown in figure 4.

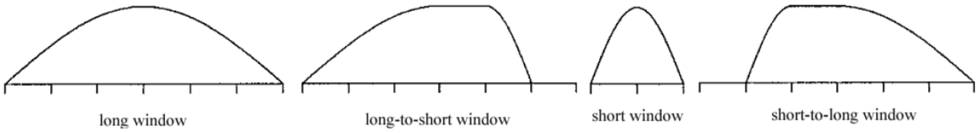

Figure 4: Window Functions of MDCT[7]

long, long-to-short, short-to-long windows are 36 points window functions, this means it generate 36 frequency coefficients if these are applied. Short window is a 12 point window function, this means it generate 12 frequency coefficients if it is applied. These generated 36 or 12 frequency coefficients are interpreted later into 18 or 6 frequency coefficients because of overlapping property of the MDCT.

As explanation for the windowing operation, this can be described as declared in [5] "The reason behind using window switching technique in MP3 can be explained as follows. A crucial issue in frequency domain coding is the potential for preechoes. Consider the case in which a silent period is followed by a percussive sound within the same coding block. Such an onset (attack) will cause comparably large instantaneous quantization errors. This preechoes can become distinctively audible. Preechoes can be masked by the time domain effect of of premasking if the time spread is of short length (in the order of few milliseconds). Therefore they can be reduced or avoided by using blocks of short length. However, a large percentage of the total bit rate is typically required for the transmission of side information if the blocks are shorter. A solution to this problem is to switch between block sizes of different lengths. This means to use short blocks only to control preecho artifacts during nonstationary periods of the signal, otherwise the coder uses long blocks. Figure 5 demonstrates the effect of preecho when it is dealt with short and long blocks."

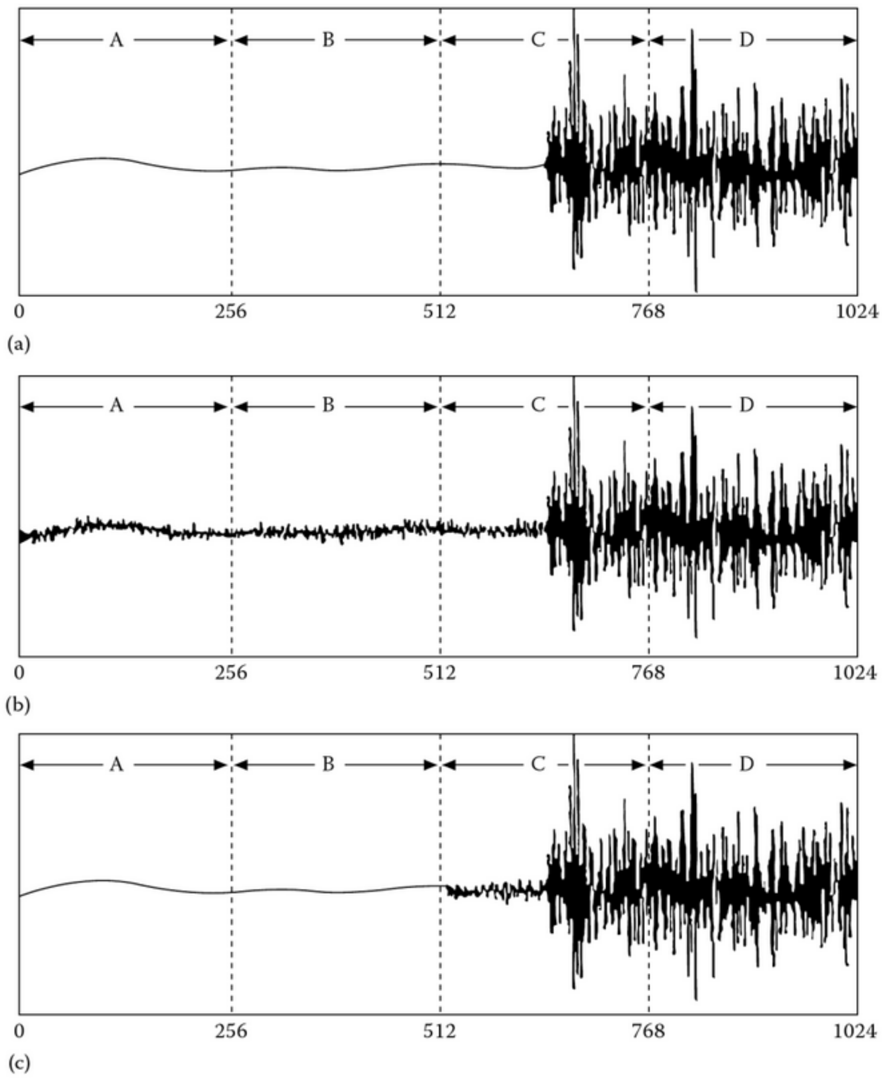

Figure 5: Preecho Handling with Short and Long Blocks.[5]
(a) Source signal, (b) reconstructed signal with block size N=1024, and (c) reconstructed signal with block size, N=256.

MP3 format is applying long window in MDCT for sample that has little change in its sound level. Short window for samples that have considerable change in its sound level. Transition from long window to short window is not happening immediately but as can be shown in figure 6.

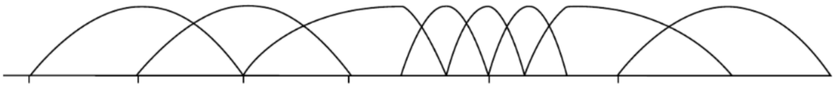

Figure 6: Sequence of Window Transition[5]

This windowing operation will result in three types of blocks at the output of MDCT. Long Block which has 18 frequency coefficients. Short Block which has 6 frequency coefficients. Mixed Block which has 18 frequency coefficients. It is important to point out here that MP3 is using always three consecutive short blocks so that to keep order of number of coded sample for each frame. So eventually one long block will have 32x18=576 frequency coefficients (and also for Mixed Block). One short block is having 32x6=192 frequency coefficients, however there is always three consecutive short blocks and they are grouped together to make 576 frequency coefficients.

5 Scalefactors Bands

At this stage we can summarize previous processes as: Filter Bank is filtering input spectrum of each frame into 32 uniformly distributed subbands, each subband has 18 frequency coefficients which leads to total of 576 frequency coefficients. Actually these 32 subbands are not emulating critical bands of human ear. So Perceptual encoding shouldn't be based on these bands as it will not be able encode audible maskers within each critical band. Therefore frequency coefficients are grouped into scalefactors bands that emulates critical bands of human ear. These scalefactors bands depend on which sampling frequency is used for the audio. Also these scalefactors bands depend on type of the block (long, mixed, short). For long blocks, the 576 frequency coefficients are grouped into 22 scalefactors bands. For short block it is 13 scalefactors bands. For mixed block it is 18 scalefactors bands.

Tables 1 and 2 shows distribution of these frequencies on scalefactors bands.

	32 KHz		44.1 KHz		48 KHz	
Scalefactor Band	Width	Freq.	Width	Freq.	Width	Freq.
0	4	0-3	4	0-3	4	0-3
1	4	4-7	4	4-7	4	4-7
2	4	8-11	4	8-11	4	8-11
3	4	12-15	4	12-15	4	12-15
4	4	16-19	4	16-19	4	16-19
5	4	20-23	4	20-23	4	20-23
6	6	24-29	6	24-29	6	24-29
7	6	30-35	6	30-35	6	30-35
8	8	36-43	8	36-43	6	36-41
9	10	44-53	8	44-51	8	42-49
10	12	54-65	10	52-61	10	50-59
11	16	66-81	12	62-73	12	60-71
12	20	82-101	16	74-89	16	72-87
13	24	102-125	20	90-109	18	88-105
14	30	126-155	24	110-133	22	106-127
15	38	156-193	28	134-161	28	128-155
16	46	194-239	34	162-195	34	156-189
17	56	240-295	42	196-237	40	190-229
18	68	296-363	50	238-287	46	230-275
19	84	364-447	54	288-341	54	276-329
20	102	448-549	76	342-417	54	330-383
21	26	550-575	158	418-575	192	384-575

Table 1: Scalefactors Bands Defined in MP3 - Long Blocks Type

	32 KHz		44.1 KHz		48 KHz	
Scalefactor Band	Width	Freq.	Width	Freq.	Width	Freq.
0	4	0-3	4	0-3	4	0-3
1	4	4-7	4	4-7	4	4-7
2	4	8-11	4	8-11	4	8-11
3	4	12-15	4	12-15	4	12-15
4	6	16-21	6	16-21	6	16-21
5	8	22-29	8	22-29	6	22-27
6	12	30-41	10	30-39	10	28-37
7	16	42-57	12	40-51	12	38-49
8	20	58-77	14	52-65	14	50-63
9	26	78-103	18	66-83	16	64-79
10	34	104-137	22	84-105	20	80-99
11	42	138-179	30	106-135	26	100-125
12	12	180-191	57	135-191	66	126-191

Table 2: Scalefactors Bands Defined in MP3 - Short Blocks Type

Grouping 576 frequency coefficients from subbands into scalefactors bands means nothing is changing the frequency coefficients itself. It means only that the new grouping will undergo to special treatment afterward. This treatment is applying scalefactor values. It means that values of frequency coefficients will be adjusted so that it can be represented correctly with suitable number of bits. In other words, frequency coefficients undergo to scaling by these scalefactor values. MP3 standard is using limited and predefined scalefactor values; within the range 0 to 4. Table 3 shows scalefactor values that are assigned to each scalefactor band in case of long block type and sampling frequency is 44.1 KHz.

SF Value	scfsi Band	SF Band	Freq.
slen1	0	0	0-3
		1	4-7
		2	8-11
		3	12-15
		4	16-19
		5	20-23
	1	6	24-29
		7	30-35
		8	36-43
		9	44-51
		10	52-61
slen2	2	11	62-73
		12	74-89
		13	90-190
		14	110-133
		15	134-161
	3	16	162-195
		17	196-237
		18	238-287
		19	288-341
		20	342-417
		21	418-575

(a)

SF Value	SF Band	Freq.
slen1	0	0-3
		4-7
		8-11
	1	12-15
		16-19
		20-23
	2	24-27
		28-31
		32-35
	3	36-39
		40-43
		44-47
	4	48-53
		54-59
		60-65
	5	66-73
		74-81
		82-89
slen2	6	90-99
		100-109
		110-119
	7	120-131
		132-143
		144-155
	8	156-169
		170-183
		184-197
	9	198-215
		216-233
		234-251
	10	252-273
		274-295
		296-317
	11	318-347
		348-377
		378-407
	12	408-575

(b)

SF Value	SF Band	Freq.
slen1	0	0-3
	1	4-7
	2	8-11
	3	12-15
	4	16-19
	5	20-23
	6	24-29
	7	30-35
	8	36-39
		40-43
		44-47
	9	48-53
		54-59
		60-65
	10	66-73
		74-81
		82-89
slen2	11	90-99
		100-109
		110-119
	12	120-131
		132-143
		144-155
	13	156-169
		170-183
		184-197
	14	198-215
		216-233
		234-251
	15	252-273
		274-295
		296-317
	16	318-347
		348-377
		378-407
	17	408-575

(c)

Table 3: Scalefactors Distribution at 44.1 KHz
(a) Long Block Type, (b) Short Block Type (c) Mixed Block Type

slen1 and slen2 are parameters that have values between 0 and 4. Selection of specific values for slen1 and slen2 is done during quantization process that will be explained shortly. Note that these tables are at 44.1 KHz.Other operating frequencies (32 and 48 KHz) have similar pattern but distribution of frequency coefficients is according to its defined table in ISO-11172-3. Note also that distribution of short block type in table 3 is shown after consolidation of three successive short windows. this is applied also for the mixed block table.

In a nutshell there are critical bands of human ear, subbands which are output of Polyphase filter, scalefactors bands which are used by Quantization and Encoding process to encode suitable frequency coefficients, scfsi bands which are used for further compression.

6 Reordering

Since short block types are consolidated together when it is used in MDCT to compensate for frequency number of each frame, the order of this consolidation has to be known. It is not mentioned explicitly in ISO-11172-3 what the order of these three blocks is. However it is mentioned in section 2.4.3.4.8 in MP3 standard in Decoding section as follows "If short blocks are used (block_type==2), the rescaled data xr[scf_band][window][freq_lines] (as described in huffmancodebits() in 2.4.1.7) shall be reordered in subband order, xr[subband][window][freq_line], prior to the IMDCT operation."

Then in Huffman declaration section it is mentioned that "The Huffman encoded data are given for successive scalefactor bands, beginning with scalefactor band 0 and ending with scalefactor band 11. Within each scalefactor band, the data is given for successive time windows, beginning with window 0 and ending with window 2. The data values within each window are arranged in order of increasing frequency."

What can be understood then is; each scalefactor band will expand to consolidate the same scalefactor band of the three windows. The clause "The data values within each window are arranged in order of increasing frequency." may make confusion understanding what is meant by increasing frequency; Taking for example scalefactor band 9. This scalefactor band has width of 18 for each window, so after consolidation it will be 54. Does it mean then to arrange ascending frequencies of each band by first 18 of first window then first 18 of second window then first 18 of the third window. Or does it mean to arrange the content of each window so that first window will have first frequency of first window then first frequency of second window then first frequency of third window then second frequency of first window, second frequency of second window, second frequency of third window, etc. This confusion may be removed if the clause is meant to be "The data values ***of*** each window are arranged in order of increasing frequency." In this case, it means that frequencies of first, second and third windows are arranged in increasing order together. So the second understanding will be followed. Moreover, a lot of previous context and researches are based on this conception. Figure 7 demonstrates this reorder operation at encoder side. Note that at decoder side, the opposite operation is needed before IMDCT filtering.

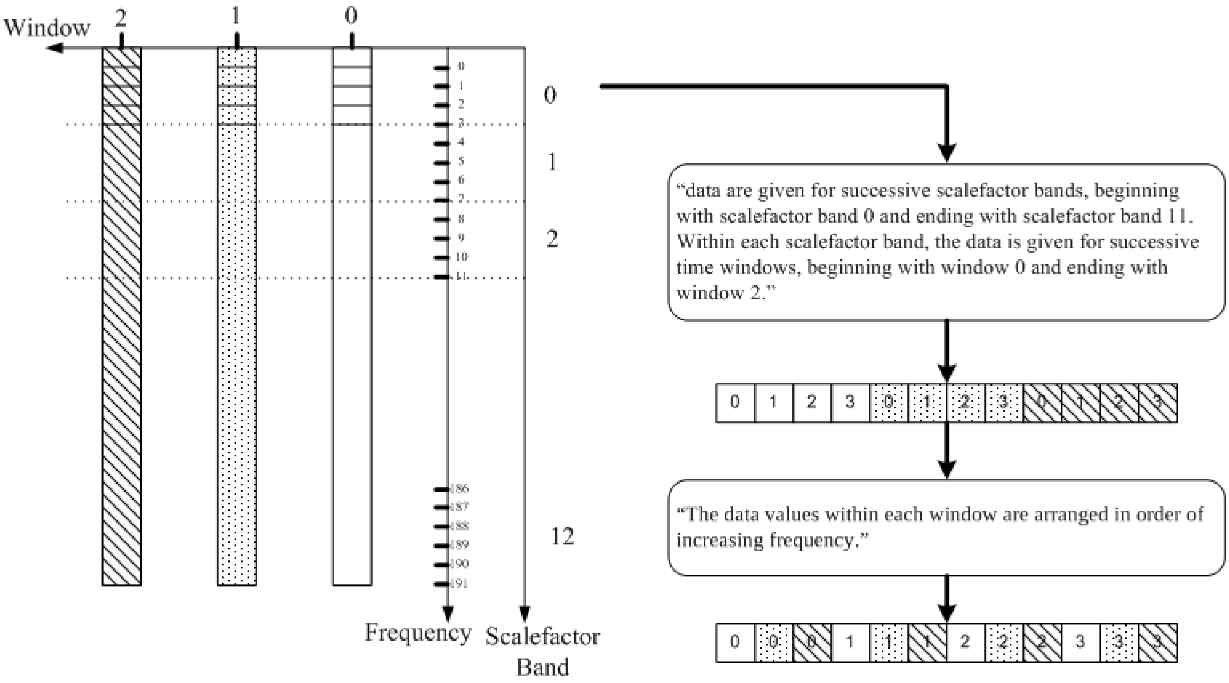
Window
2
1
0
0
1
2
12
Frequency
Scalefactor Band
"data are given for successive scalefactor bands, beginning with scalefactor band 0 and ending with scalefactor band 11. Within each scalefactor band, the data is given for successive time windows, beginning with window 0 and ending with window 2."
0 1 2 3 0 1 2 3 0 1 2 3
"The data values within each window are arranged in order of increasing frequency."
0 0 0 1 1 1 2 2 2 3 3 3

Figure 7: Reorder Process at Encoder

7 Bit Allocation

Quantization process is doing mainly two tasks; first task is to allocate (quantize) input frequency coefficients. Second Task is to scale some frequency coefficients if it found that number of bits assigned to it is too much so that after scaling these frequency coefficients, number of assigned bits can be reduced. These two task are executed with a constraint to keep the quantization noise below the masking threshold. The allocation control algorithm suggested for the layer III encoder uses non uniform and dynamic quantization. Non uniform means step size between quantization levels is not equal. Dynamic means that this non-linearity differs from one scalefactor band to another (i.e. same non-linearity is not applied for each scalefactor band). Quantization is applied to 576 spectral values at a time. This is done iteratively in two nested loops, a distortion control loop (outer loop) and a rate control loop (inner loop). The following explanation of these loops is declared by [8].

rate control loop: The rate loop does the quantization of the frequency domain samples and thus also determines the required quantization step size. Furthermore the subdivision of the big values (see Huffman part below) into regions, the Huffman table selection decision for each region and the calculation of the boundaries between the regions take place here. To begin with the samples are quantized with an increasing step size until the quantized values can be coded using one of the available Huffman code tables. A larger step size leads to smaller quantized values. Then the overall Huffman coded bit sum is calculated and compared with the number of bits available. If the calculated bit sum exceeds the number of bits available the quantization step size is further increased and the entire procedure is repeated until the available bits are sufficient. The non-linearity is achieved raising each sample to the power of 3/4.

distortion control loop: This loop controls the quantization noise which is produced by the quantization of the frequency domain lines within the rate control loop. The aim is to keep the quantization noise below the masking threshold (allowed noise given by the psychoacoustic model) for each scalefactor band. To shape the quantization noise scalefactors are applied to the frequency lines within each scalefactor band. The scalefactors of all scalefactor bands and the quantization step size are then saved before the rate control loop in called. After the inner loop the quantization noise is calculated. This is repeated until there is no more scalefactor band with more noise than allowed by the threshold. The values of the scalefactors belonging to bands that are too noisy are increased for each iteration loop. Finally the noise caused by the quantization will not be audible by a human and the loop will exit. There are still situations where both loops can go on forever depending on the calculated threshold. To avoid this there are several conditions in the distortion control loop that can be checked to stop the iterations more early.

8 Huffman Encoding

Huffman coding algorithm is considered a lossless encoding method which means it is able to represent input samples in fewer bits without losing any information. What is important to point out here is the input to Huffman encoder is 576 quantized frequency coefficients. These 576 lines are classified into three regions; big values, count1, zero regions. Big values region includes frequencies that needs higher gain as it is audible by human ear so it is better to represent it clearly and it is further divided into region0, region1, region2. Count1 region includes frequencies with values between -1 and 1. Zero region is not coded and approximated to 0 as it in not audible by human ear. Figure 8 shows these different regions.

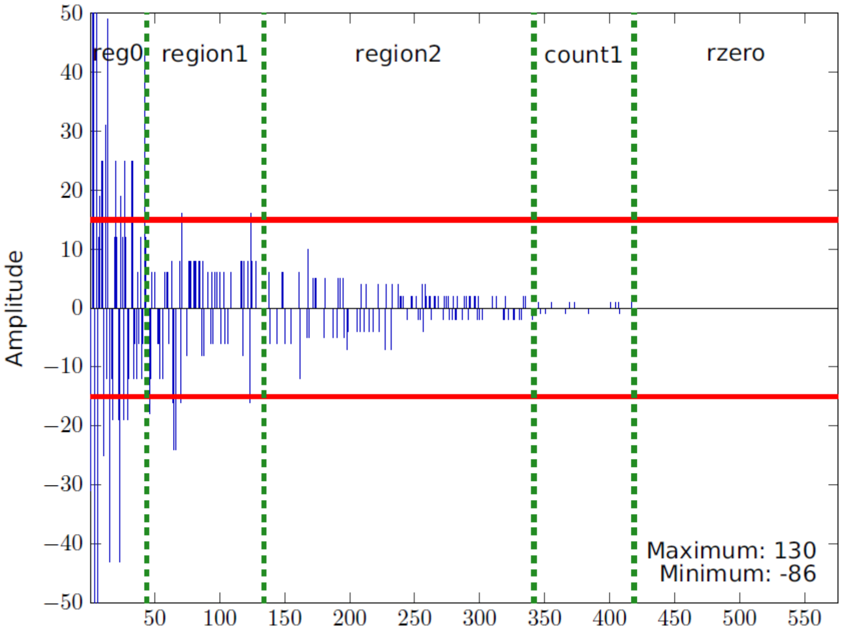

Figure 8: Huffman Regions[9]

9 Frame Packing

Encoding process resulted in frequency coefficients represented with Huffman bits that were scaled with scalefactors values. These are mainly what is needed to be sent to a receiver in order to be able to decode original samples back. Some associated information has to be sent also to clarify to the receiver what are number of bits representing the whole frequencies (part2_3_length), limits of Huffman regions (big_values, count1, etc), sampling frequency, etc. Figure 9 shows anatomy of the MP3 frame.

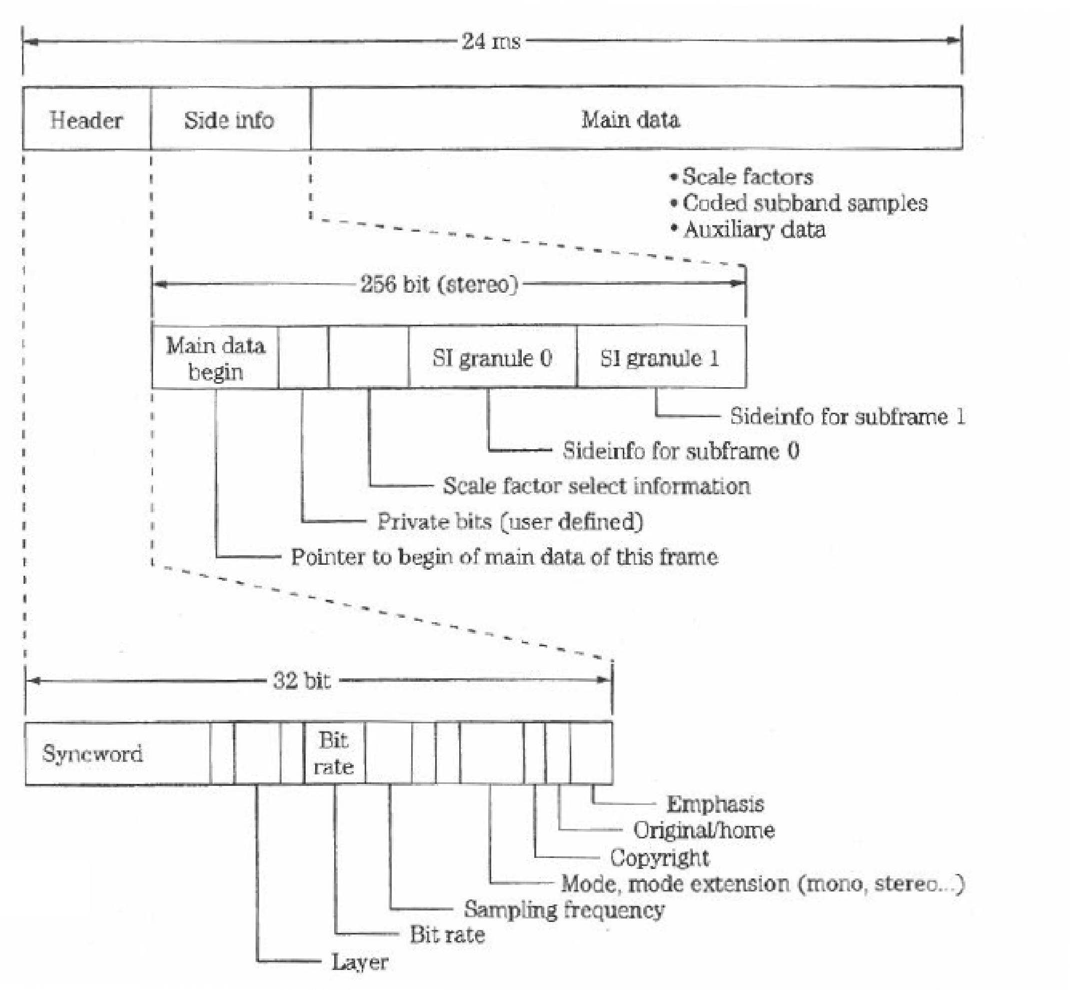

Figure 9: MP3 Frame Structure[10]

It is important to point out here that one MP3 frame carries 1152 frequency samples. i.e. two 576 frequencies. First 576 frequencies are labeled as granule 0, the second is labeled as granule 1. What can be concluded then from this frame anatomy that it starts with Header which is needed by the receiver to know synchronization of bits sequence and to know additional tags like which layer is this frame (layer I, II, III), bit rate and sampling frequency, etc. Then Side Information is sent to inform the receiver about no of Main Data bits, big values region length, count1 length, which table is used in Huffman algorithm, scalefactors bands length, etc. All these details can be reviewed in ISO 11172-3 standard section 2.4.1.7.

Bibliography

[1] S. Committe, Health Risk from Exposure to Noise from Personal Music Players, sep 2008.

[2] http://www.intropsych.com, June 2015.

[3] https://www.wikipedia.org/, June 2015.

[4] V. G. Oklobdzija, Ed., Digital Systems and Applications. CRC Press, 2008.

[5] V. Madisetti, Ed., Digital Signal Processing Handbook. CRC Press, 2009.

[6] A. S. Ted Painter, “Perceptual coding of digital audio,” Telecommunications Research Center, Arizona State University, Tech. Rep.

[7] C.-J. Tsai, Audio Codecs, National Chiao Tung University, dec 2012.

[8] R. Raissi, “The theory behind mp3,” Tech. Rep., December 2002.

[9] M. Schmidt, “Hardware modelle für die mp3-huffman dekodierung,”, Hochschule Bremen, feb 2014.

[10] K. C. Pohlmann, Principles of Digital Audio, 5th ed. McGraw-Hill, 2005.