

Chapter 2 Time-Frequency Mapping Tools in MP3 and MPEG 2/4

Although the proposed patch method is focused on MP3 and AAC, it can apply to any other frequency-based audio decoders. There are many time-frequency mapping tools adopted in a variety of codec, however the concepts of them are similar. Hence, this chapter gives an overview of two main kinds of time-frequency mapping tool used widely in most perceptual audio coders, especially for MP3 and AAC. The first one is cosine modulated PQMF (Pseudo Quadrature Mirror Filter bank) with nearly PR (Perfect Reconstruction) property. Second is MDCT (Modified Discrete Cosine Transform) that is a TDAC (Time Domain Aliasing Concealing) transform with PR property.

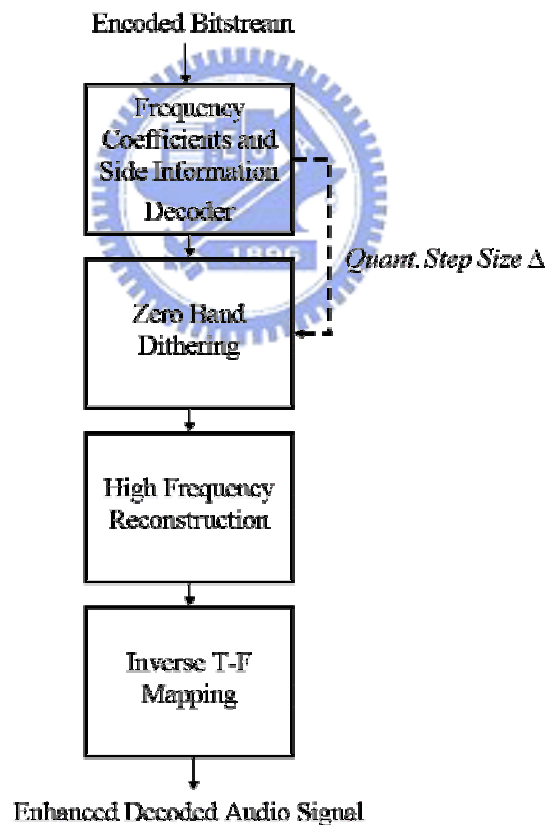


Figure 5: Block diagram of audio patch method incorporated into frequency-based audio decoder.

Roughly speaking, in MP3 encoder, an audio signal is first separated into 32 subband signals by an analysis filter bank that is a cosine modulated PQMF, and subsequently a MDCT with 36 point is applied to every subband signal to raise the frequency resolution further. At the part of decoder, an IMDCT and an overlap-add operation inversely transform the frequency coefficients to 32 subband signals and then a synthesis filterbank regenerates the decoded audio signal by synthesizing the 32 subband signals. The combinative filter bank of the two different mapping tools in MP3 encoder is commonly referred to as the hybrid filter bank that is illustrated in **Figure 6**. For AAC, only a MDCT is used to transform every 2048 time-domain samples of the input signal to 1024 frequency coefficients directly in encoder, and in decoder a IMDCT and an overlap-add operation inversely transform the 1024 frequency coefficients to 2048 decoded time domain samples.

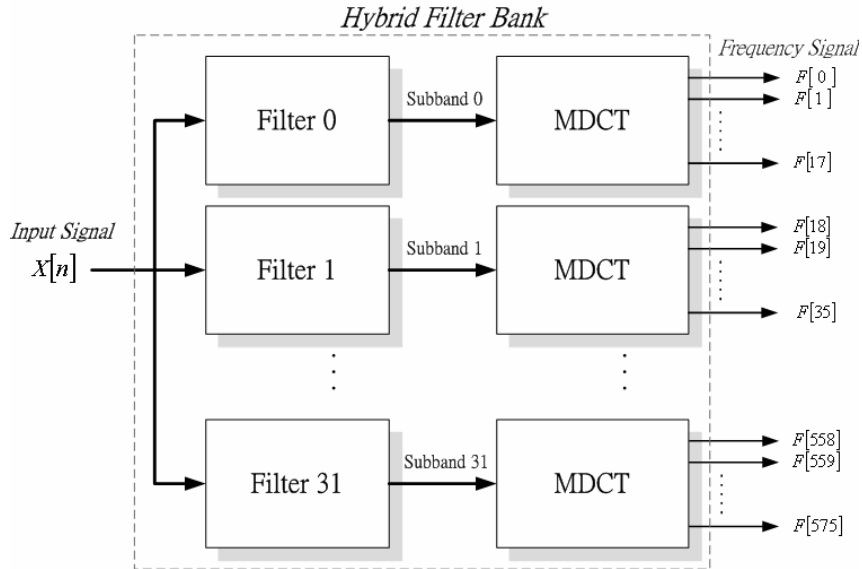


Figure 6: Block diagram of the hybrid filter bank consisting of an analysis polyphase filter bank and MDCT in MP3 encoder.

2.1 Polyphase Filter bank in MP3

Figure 7 illustrates the uniform M -band maximally decimated analysis-synthesis filter bank. The analysis filter bank separates the input signal to M subband signals with uniform bandwidth π/M illustrated in **Figure 8**. Every subband signal, subsequently, is downsampled by a decimator of M factor for critical sampling. At the backward part, an upsampler with M factor is applied to the decimated subband signals firstly. To eliminate the imaging distortions introduced by the upsampling operation, the upsampled subband signals are processed by the parallel bank of the synthesis filters, and then the filter output are combined to form the overall output [22].

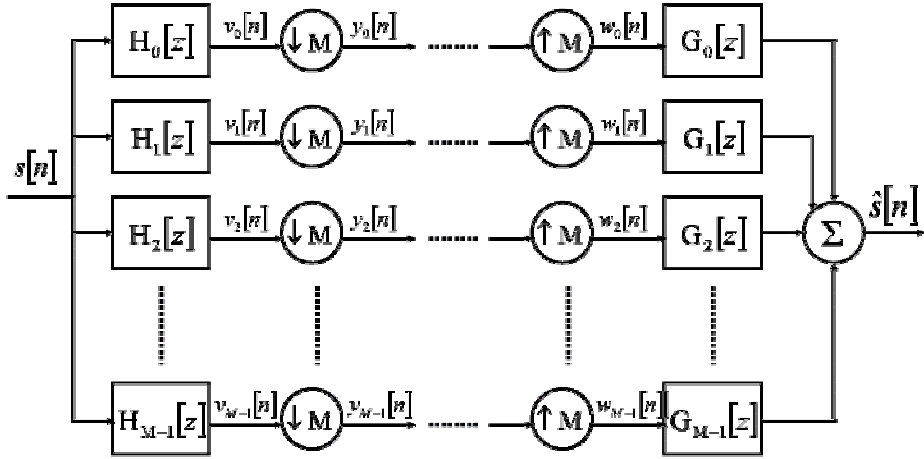


Figure 7: Uniform M-band maximally decimated analysis-synthesis filter bank.

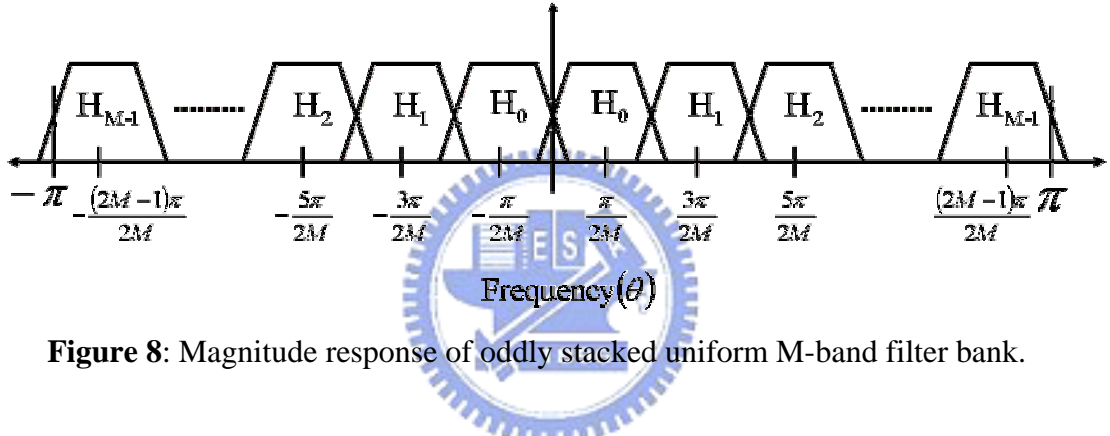


Figure 8: Magnitude response of oddly stacked uniform M-band filter bank.

The framework is also used in MP3 codec, where M is 32. Furthermore, the cosine modulation of lowpass prototype is used in MP3 to realize the parallel M -channel filter banks with nearly perfect reconstruction [22]. The impulse response of the analysis filter bank is given as (1), where $p[n]$ is the impulse response of the lowpass prototype filter and L is 512.

$$h_k[n] = \begin{cases} p[n] \cdot \cos\left[\frac{\pi}{2M} \cdot (2k+1) \cdot \left(n - \frac{M}{2}\right)\right], & \text{for } n = 0 \sim L-1 \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

On the other hand, the impulse response of the synthesis filter bank is as (2).

$$g_k[n] = \begin{cases} p[n] \cdot \cos\left[\frac{\pi}{2M} \cdot (2k+1) \cdot \left(n + \frac{M}{2}\right)\right], & \text{for } n = 0 \sim L-1 \\ 0, & \text{otherwise.} \end{cases} \quad (2)$$

By polyphase decomposition, there are fast implementation methods introduced in standards of MP3 and AAC for the filtering processing of analysis and synthesis filter banks.

2.1.1 Fast Filtering Algorithm of Analysis Filter Bank by Polyphase Decomposition

Let $X[k]$ be the input time-domain signal, and $S_k[n]$ be the output subband signal from the k th filter of the analysis filter bank. By the convolution of the input signal $X[k]$ and the impulse response $h_k[n]$, $S_k[n]$ is given as (3).

$$S_k[n] = X[k] * h_k[n] = \sum_{j=-\infty}^{\infty} X[n-j] \cdot h_k[j] \quad (3)$$

Since each filter of the analysis filter bank is a FIR filter of 512 length, it reduces (3)

$$S_k[n] = \sum_{j=0}^{511} X[n-j] \cdot h_k[j] \quad (4)$$

After the decimator of 32 factor applies, the samples required to recode are

$$S_k[32n] = \sum_{j=0}^{511} X[32n-j] \cdot h_k[j] \quad (5)$$

Using the polyphase decomposition of 64 factor on the impulse response $h_k[n]$, it gives another method of computing $S_k[32n]$.

$$S_k[32n] = \sum_{j=0}^{63} \sum_{m=0}^7 X[32n - (64m + j)] \cdot h_k[64m + j] \quad (6)$$

On the other hand,

$$\begin{aligned} h_k[64m + j] &= p[64m + j] \cdot \cos\left[\frac{\pi}{64} \cdot (2k + 1) \cdot ((64m + j) - 16)\right] \\ &= p[64m + j] \cdot (-1)^m \cdot \cos\left[\frac{\pi}{64} \cdot (2k + 1) \cdot (j - 16)\right] \end{aligned} \quad (7)$$

Hence, by substituting (7) to (6), it gives

$$\begin{aligned} S_k[32n] &= \sum_{j=0}^{63} \left(\sum_{m=0}^7 X[32n - (64m + j)] \cdot p[64m + j] \cdot (-1)^m \right) \cdot \cos\left[\frac{\pi}{64} \cdot (2k + 1) \cdot (j - 16)\right] \\ &= \sum_{j=0}^{63} \left(\sum_{m=0}^7 X[32n - (64m + j)] \cdot C[64m + j] \right) \cdot M_k[j] \end{aligned} \quad (8)$$

Where $C[64m + j]$ represents $p[64m + j] \cdot (-1)^m$, and $M_k[j]$ represents

$\cos\left[\frac{\pi}{64} \cdot (2k + 1) \cdot (j - 16)\right]$. The numbers of multiplications for computing the 32

samples from the 32 filters by the formulation (4) is totally 16384 that is 512×32 .

However, by the method of polyphase decomposition, it only needs 2560 times of

multiplication, that is $512 + 64 \times 32$. This is because the summations $\sum_{m=0}^7 X[32n - (64m + j)] \cdot C[64m + j]$ can be used repeatedly for every subband filtering processing to reduce the computing complexity.

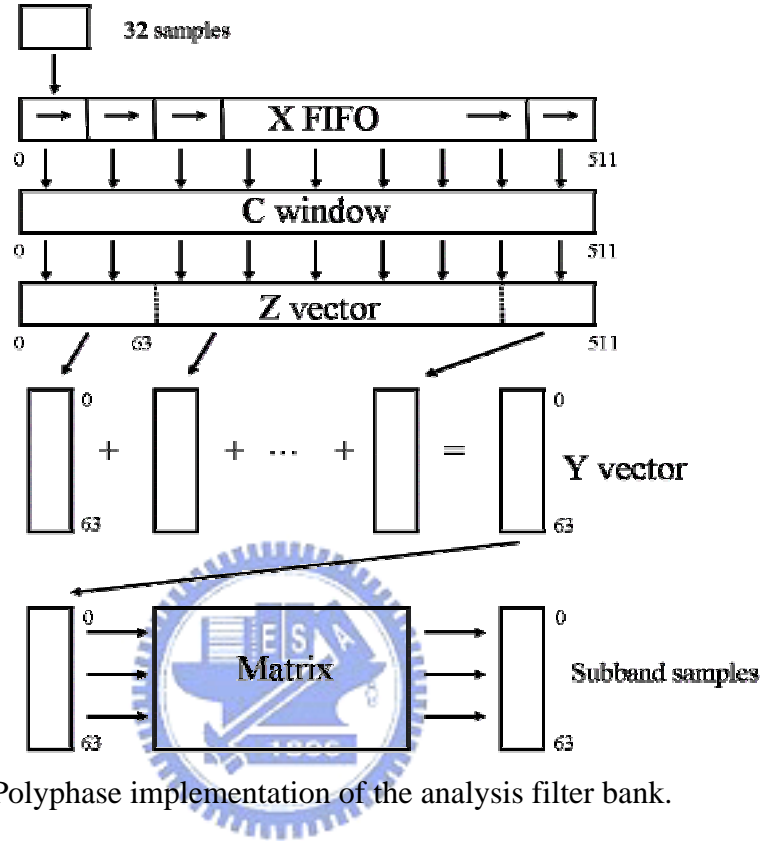


Figure 9: Polyphase implementation of the analysis filter bank.

Figure 9 illustrates the procedures of the method of polyphase decomposition, where $Z[64m + j]$ recodes the value of $X[32n - (64m + j)] \cdot C[64m + j]$ for m is from 0 to 7 and j is from 0 to 63 and $Y[j]$ recodes the summation $\sum_{m=0}^7 Z[64m + j]$ for j is from 0 to 63. For every time to compute the new 32 subband samples, 32 new signal samples are putted into the FIFO buffer and the oldest 32 signal samples are scarified for data updating.

2.1.2 Fast Filtering Algorithm of Synthesis Filter Bank by Polyphase Decomposition

Let $\hat{S}_k[n]$ be the interpolated subband signal after the processing of interpolator with 32 factor. That is,

$$\hat{S}_k[n] = \begin{cases} S_k[n], & \text{if } 32 \mid n. \\ 0, & \text{otherwise.} \end{cases} \quad (9)$$

The filtering output from the k th synthesis filter is given as (10).

$$\tilde{X}_k[n] = \hat{S}_k[n] * g_k[n] = \sum_{j=-\infty}^{\infty} \hat{S}_k[j] \cdot g_k[n-j] \quad (10)$$

That is

$$\tilde{X}_k[n] = \sum_{j=-\infty}^{\infty} S_k[j] \cdot g_k[n-32j] \quad (11)$$

The reconstructed signal $\tilde{X}[n]$ is obtained by the combination of the 32 filtering output subband signals.

$$\tilde{X}[n] = \sum_{k=0}^{31} \sum_{j=-\infty}^{\infty} S_k[j] \cdot g_k[n-32j] \quad (12)$$

That is

$$\begin{aligned} \tilde{X}[n] &= \sum_{j=-\infty}^{\infty} \sum_{k=0}^{31} S_k[j] \cdot p[n-32j] \cdot \cos\left[\frac{\pi}{64} \cdot (2k+1) \cdot (n-32j+16)\right] \\ &= \sum_{j=-\infty}^{\infty} p[n-32j] \cdot \sum_{k=0}^{31} S_k[j] \cdot \cos\left[\frac{\pi}{64} \cdot (2k+1) \cdot (n-32j+16)\right] \end{aligned} \quad (13)$$

Let $n = 32l + s$ for polyphase decomposition,

$$\tilde{X}[32l+s] = \sum_{j=-\infty}^{\infty} p[32(l-j)+s] \cdot \sum_{k=0}^{31} S_k[j] \cdot \cos\left[\frac{\pi}{64} \cdot (2k+1) \cdot (32(l-j)+s+16)\right] \quad (14)$$

Let q denote $l-j$, (14) is reduced to (15).

$$\tilde{X}[32l+s] = \sum_{q=-\infty}^{\infty} p[32q+s] \cdot \sum_{k=0}^{31} S_k[l-q] \cdot \cos\left[\frac{\pi}{64} \cdot (2k+1) \cdot (32q+s+16)\right] \quad (15)$$

Since the lowpass prototype filter is a FIR filter of 512 length, it gives

$$\tilde{X}[32l+s] = \sum_{q=0}^{15} p[32q+s] \cdot \sum_{k=0}^{31} S_k[l-q] \cdot \cos\left[\frac{\pi}{64} \cdot (2k+1) \cdot (32q+s+16)\right], \quad (16)$$

for s is from 0 to 31. On the other hand, we define $U[l, 32q+s]$ as follow

$$U[l, 32q+s] = \sum_{k=0}^{31} S_k[l-q] \cdot \cos\left[\frac{\pi}{64} \cdot (2k+1) \cdot (32q+s+16)\right] \quad (17)$$

Hence,

$$\tilde{X}[32l+s] = \sum_{q=0}^{15} p[32q+s] \cdot U[l, 32q+s]. \quad (18)$$

Furthermore, if q is an even integer and denoted as $2t$, it implies

$$\begin{aligned}
U[l,32q+s] &= \sum_{k=0}^{31} S_k[l-2t] \cdot \cos\left[\frac{\pi}{64} \cdot (2k+1) \cdot (64t+s+16)\right] \\
&= \sum_{k=0}^{31} S_k[l-2t] \cdot (-1)^t \cdot \cos\left[\frac{\pi}{64} \cdot (2k+1) \cdot (s+16)\right]
\end{aligned} \tag{19}$$

For q is an even integer and denoted as $2t+1$, it implies

$$\begin{aligned}
U[l,32q+s] &= \sum_{k=0}^{31} S_k[l-(2t+1)] \cdot \cos\left[\frac{\pi}{64} \cdot (2k+1) \cdot (64t+32+s+16)\right] \\
&= \sum_{k=0}^{31} S_k[l-(2t+1)] \cdot (-1)^t \cdot \cos\left[\frac{\pi}{64} \cdot (2k+1) \cdot (32+s+16)\right]
\end{aligned} \tag{20}$$

For convenience, define a vector V as follow

$$V[l,64i+j] = \sum_{k=0}^{31} S_k[l-i] \cdot \cos\left[\frac{\pi}{64} \cdot (2k+1) \cdot (j+16)\right] \tag{21}$$

Therefore,

$$U[l,32q+s] = \begin{cases} (-1)^t \cdot V[l,64 \cdot (2t) + s], & \text{for } q = 2t. \\ (-1)^t \cdot V[l,64 \cdot (2t+1) + (32+s)], & \text{for } q = 2t+1. \end{cases} \tag{22}$$

That is

$$U[l,32q+s] = \begin{cases} (-1)^t \cdot V[l,128 \cdot t + s], & \text{if } q = 2t. \\ (-1)^t \cdot V[l,128 \cdot t + 96 + s], & \text{if } q = 2t+1. \end{cases} \tag{23}$$

for q is from 0 to 15, and s is from 0 to 31. From (21), it gives

$$V[l+1,64 \cdot (i+1) + j] = V[l,64i + j] \tag{24}$$

In the other word, the data in the vector V can be updated by inputting a new vector of 64 length and shifting the old data rightly 64 elements to scarify the oldest 64 elements. This gives an efficient method to compute the vector U , and decrease the complexity of reconstruct the synthesis output signal from 544 multiplications required (i.e. $512+32$) to 80 multiplications (i.e. $(32 \cdot 64 + 512)/32$) for every one reconstructed signal sample. **Figure 10** illustrates the procedures of the method of polyphase decomposition for the synthesis filter bank.

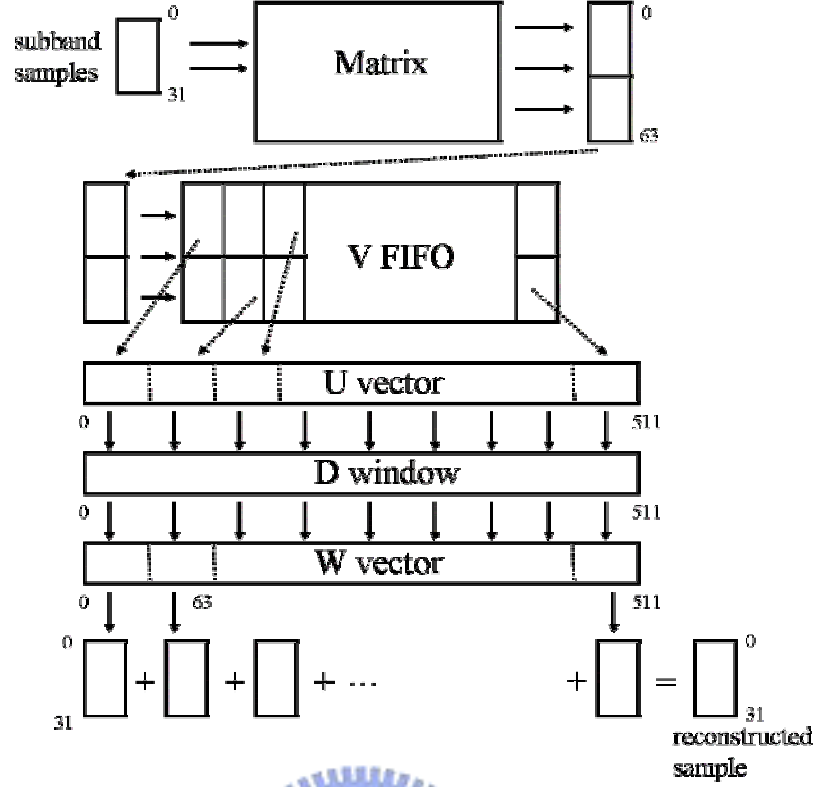


Figure 10: Polyphase implementation of the synthesis filter bank.

2.2 Modified Discrete Cosine Transform

The human hearing is very sensitive to the block effect of quantization noise between the consecutive frames in time. Hence, the concealing or smoothing of the block effect is an important issue of perceptual audio coding. On the other hand, the frequency coefficients encoded must be real number for coding efficiency. Therefore, a transform that can map R^n to R^m , not C^m , is required. The MDCT is exactly a best tool to satisfy the two issues on audio coding. Given a time frame consisting of $2N$ samples a_k , formally, MDCT is given as (25), where h_k is a window function.

$$\alpha_r \equiv \frac{2}{\sqrt{2N}} \cdot \sum_{k=0}^{2N-1} h_k \cdot a_k \cdot \cos \left(\frac{2\pi \cdot \left(k + \frac{N+1}{2} \right) \cdot \left(r + \frac{1}{2} \right)}{2N} \right), \text{ for } r = 0 \sim 2N-1. \quad (25)$$

And IMDCT is given as (26).

$$\hat{a}_k \equiv h_k \cdot \frac{2}{\sqrt{2N}} \cdot \sum_{r=0}^{N-1} \alpha_r \cdot \cos \left(\frac{2\pi \cdot \left(k + \frac{N+1}{2} \right) \cdot \left(r + \frac{1}{2} \right)}{2N} \right), \text{ for } k = 0 \sim 2N-1 \quad (26)$$

Furthermore, by the overlap-add operation with the last frame, the original samples in the front half of the current frame can be reconstructed perfectly if there is no quantization noise to harm. That is

$$a_k = \hat{a}_k + \hat{a}'_{N+k}, \text{ for } k = 0 \sim N-1, \quad (27)$$

where \hat{a}'_k represents the output of IMDCT of the last frame. The behind half of the current frame will be reconstructed perfectly by the overlap-add operation between the next frame and the current frame.

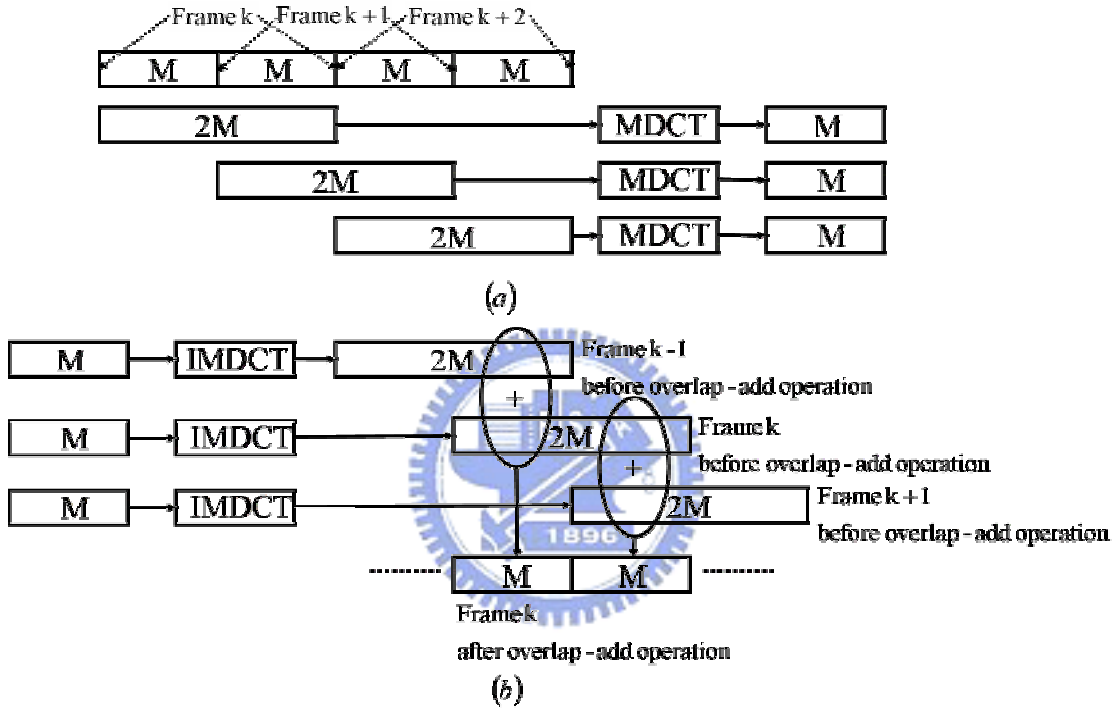


Figure 11: MDCT: (a) Lapped forward transform (analysis). (b) Inverse transform (synthesis).

The property of perfect reconstruction by time domain aliasing concealing makes MDCT able to ease the block effect. This is because the time domain aliasing concealing is achieved by an overlap-add operation over the two consecutive time frames and then smooth the quantization noise shape across the two frames simultaneously. **Figure 11** (b) illustrates the overlap-add operation after IMDCT. For the following, the design conception of MDCT is explained in detail.

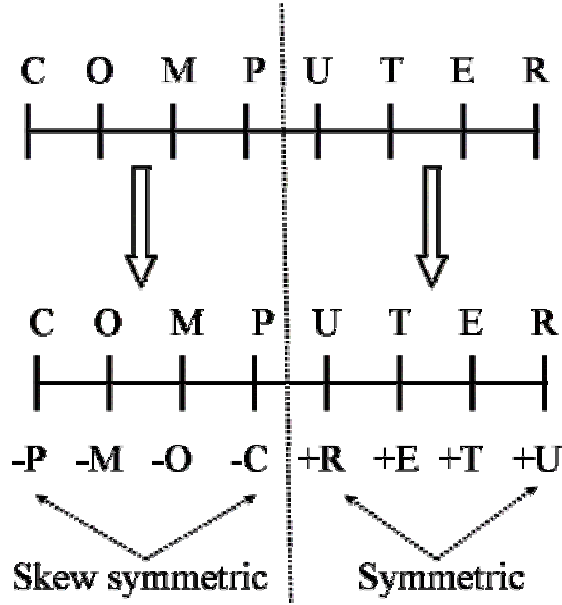


Figure 12: The artificial aliasing of MDCT.

Simply to speaking, without considering of the action of the window function, the processing conception of the transform system consisting of MDCT and IMDCT is equivalent to firstly alias the time-domain signal in a frame artificially as illustrated in Figure 12 and transform the aliased signal to a frequency signal by some invertible special mapping in the forward part, and in backward part, inversely transform the frequency signal and conceal the aliasing artifact by the overlap-add operation. The action of aliasing is to apply “subtraction by crossing” at the first half of the time frame and “addition by crossing” at the behind half. Specifically speaking, for a frame consisting of $2N$ samples a_k , the aliased signal samples is \hat{a}_k

$$\hat{a}_k = \begin{cases} a_k - a_{N-1-k} & \text{for } k = 0 \sim N-1 \\ a_k + a_{3N-1-k} & \text{for } k = N \sim 2N-1 \end{cases} \quad (28)$$

Hence, the “difference” information of the artificial aliasing can be recoded by the front half part of the current frame and the “summation” information can be recoded by the tail half part of the last frame. By the overlap-add operation between the current frame and the last frame, the front half part of the current frame will be reconstruction perfectly and the aliasing is concealed if the composed transform of the transform and the inverse transform adapted in middle is an identity transform.

DFT (Discrete Fourier transform) is a standard time-frequency mapping. However, in general, DFT maps a real vector to a complex vector, and hence is unsuitable to be applied as the mapping tool we desire. However, due to the skew symmetric property of the front half of the aliased signal and the symmetric property of the tail half, it is easy to solve the inconsistent problem of real domain and complex

domain by modifying the basis function of DFT by time-domain shift and frequency-domain shift to conceal the image part of the transform coefficients. The new transform with the modified basis function is called SDFT (Shifted DFT) that is a generalization of DFT [23]. The following Theorems explain how the system of MDCT and IMDCT is constructed.

Theorem 1 For any given $u, v \in R$, the ordered vector set $\{b_0, b_1, \dots, b_{2N-1}\}$ is an orthonormal basis of C^{2N} over field C , where

$$b_r = \begin{bmatrix} \frac{1}{\sqrt{2N}} \cdot \exp\left(\frac{i \cdot 2\pi \cdot u \cdot (r+v)}{2N}\right) \\ \frac{1}{\sqrt{2N}} \cdot \exp\left(\frac{i \cdot 2\pi \cdot (1+u) \cdot (r+v)}{2N}\right) \\ \vdots \\ \frac{1}{\sqrt{2N}} \cdot \exp\left(\frac{i \cdot 2\pi \cdot (2N-1+u) \cdot (r+v)}{2N}\right) \end{bmatrix}, \text{ for } r = 0 \sim 2N-1. \quad (29)$$

<Proof> For $m \neq n$,

$$\begin{aligned} \langle b_m, b_n \rangle &= \sum_{k=0}^{2N-1} \frac{1}{2N} \cdot \exp\left(\frac{i \cdot 2\pi \cdot (k+u) \cdot (m-n)}{2N}\right) \\ &= \frac{1}{2N} \cdot \frac{\exp\left(\frac{i \cdot 2\pi \cdot u \cdot (m-n)}{2N}\right) \cdot \left(1 - \exp\left(\frac{i \cdot 2\pi \cdot (m-n) \cdot 2N}{2N}\right)\right)}{1 - \exp\left(\frac{i \cdot 2\pi \cdot (m-n)}{2N}\right)} = 0 \end{aligned} \quad (30)$$

For $m = n$,

$$\langle b_m, b_n \rangle = \sum_{k=0}^{2N-1} \frac{1}{2N} \cdot \exp\left(\frac{i \cdot 2\pi \cdot (k+u) \cdot (m-n)}{2N}\right) = \sum_{k=0}^{2N-1} \frac{1}{2N} = 1 \quad (31)$$

■

From Theorem 1, it tells that no matter which (u, v) is chosen, the modified vectors from the basis functions of DFT is still a basis of the vector space C^{2N} over field C and construct a new invertible transform SDFT. Hence, if we can adapt a suitable tuple (u, v) to make the two products $\hat{a}_k \cdot \frac{1}{\sqrt{2N}} \exp^*\left(\frac{i \cdot 2\pi \cdot (k+u) \cdot (r+v)}{2N}\right)$ and

$\hat{a}_{N-1-k} \cdot \frac{1}{\sqrt{2N}} \exp^*\left(\frac{i \cdot 2\pi \cdot (N-1-k+u) \cdot (r+v)}{2N}\right)$ for $k = 0 \sim N-1$ and the two products

$\hat{a}_k \cdot \frac{1}{\sqrt{2N}} \exp^*\left(\frac{i \cdot 2\pi \cdot (k+u) \cdot (r+v)}{2N}\right)$ and $\hat{a}_{3N-1-k} \cdot \frac{1}{\sqrt{2N}} \exp^*\left(\frac{i \cdot 2\pi \cdot (3N-1-k+u) \cdot (r+v)}{2N}\right)$ for $k =$

$N \sim 2N-1$ are conjugate, respectively, the image parts of the products in the inner product of the aliased time frame vector and the r th basis vector modified are concealed mutually to get a real value transform coefficient. Because the symmetric

and skew symmetric properties of the aliased signal, the problem mentioned above is equivalent to find a tuple (u, v) that makes the number $\exp\left(\frac{i \cdot 2\pi \cdot (k+u) \cdot (r+v)}{2N}\right)$ is skew conjugate to $\exp\left(\frac{i \cdot 2\pi \cdot (N-1-k+u) \cdot (r+v)}{2N}\right)$ for $k=0 \sim N-1$, and the number $\exp\left(\frac{i \cdot 2\pi \cdot (k+u) \cdot (r+v)}{2N}\right)$ is conjugate to $\exp\left(\frac{i \cdot 2\pi \cdot (3N-1-k+u) \cdot (r+v)}{2N}\right)$ for $k = N \sim 2N-1$. To analysis further, we know

$$\exp\left(\frac{i \cdot 2\pi \cdot (N-1-k+u) \cdot (r+v)}{2N}\right) = \exp\left(\frac{i \cdot 2\pi \cdot (N-1+2u-(k+u)) \cdot (r+v)}{2N}\right) \quad (32)$$

If let $N-1+2u$ be $2N$ to be substituted into (32), then

$$\begin{aligned} \exp\left(\frac{i \cdot 2\pi \cdot (N-1-k+u) \cdot (r+v)}{2N}\right) &= \exp\left(\frac{i \cdot 2\pi \cdot -(k+u) \cdot (r+v)}{2N} + \frac{i \cdot 2\pi \cdot 2N \cdot (r+v)}{2N}\right) \\ &= \exp\left(\frac{i \cdot 2\pi \cdot -(k+u) \cdot (r+v)}{2N}\right) \cdot \exp(i \cdot 2\pi \cdot (r+v)) \\ &= \exp^*\left(\frac{i \cdot 2\pi \cdot (k+u) \cdot (r+v)}{2N}\right) \cdot \exp(i \cdot 2\pi \cdot v) \end{aligned} \quad (33)$$

Therefore, if v is set as $\frac{1}{2}$, then the number $\exp\left(\frac{i \cdot 2\pi \cdot (k+u) \cdot (r+v)}{2N}\right)$ can be skew conjugate to $\exp\left(\frac{i \cdot 2\pi \cdot (N-1-k+u) \cdot (r+v)}{2N}\right)$ for $k = 0 \sim N-1$. Under the candidate tuple (u, v) that is $(\frac{N+1}{2}, \frac{1}{2})$, a fine result we want is confirmed.

$$\begin{aligned} \exp\left(\frac{i \cdot 2\pi \cdot (3N-1-k+u) \cdot (r+v)}{2N}\right) &= \exp\left(\frac{i \cdot 2\pi \cdot (3N-1-(k+u)+N+1) \cdot \left(r+\frac{1}{2}\right)}{2N}\right) \\ &= \exp\left(\frac{i \cdot 2\pi \cdot (4N-(k+u)) \cdot \left(r+\frac{1}{2}\right)}{2N} + i \cdot 2\pi \cdot 2 \cdot \left(r+\frac{1}{2}\right)\right) \\ &= \exp^*\left(\frac{i \cdot 2\pi \cdot (k+u) \cdot (r+v)}{2N}\right) \end{aligned} \quad (34)$$

Hence, if the special tuple $(\frac{N+1}{2}, \frac{1}{2})$ is adapted to shift the original basis function of DFT, the resulting SDFT is exactly the transform we desire. The following theorem gives an equivalent method on real domain to get the transform coefficients of SDFT of the aliased signal without any complex value calculation. This conducts simultaneously that the action of SDFT on the aliased signal is equivalent to the action of MDCT on the original signal without window function.

Theorem 2 For $r = 0 \sim 2N-1$,

$$\sum_{k=0}^{2N-1} \hat{a}_k \cdot \frac{1}{\sqrt{2N}} \exp^* \left(\frac{i \cdot 2\pi \cdot \left(k + \frac{N+1}{2}\right) \cdot \left(r + \frac{1}{2}\right)}{2N} \right) = \frac{2}{\sqrt{2N}} \cdot \sum_{k=0}^{2N-1} a_k \cdot \cos \left(\frac{2\pi \cdot \left(k + \frac{N+1}{2}\right) \cdot \left(r + \frac{1}{2}\right)}{2N} \right) \quad (35)$$

<Proof> Since the special tuple $\left(\frac{N+1}{2}, \frac{1}{2}\right)$ is adapted, we have

$$\begin{aligned} \sum_{k=0}^{2N-1} \hat{a}_k \cdot \exp^* \left(\frac{i \cdot 2\pi \cdot \left(k + \frac{N+1}{2}\right) \cdot \left(r + \frac{1}{2}\right)}{2N} \right) &= \sum_{k=0}^{2N-1} \hat{a}_k \cdot \cos \left(\frac{2\pi \cdot \left(k + \frac{N+1}{2}\right) \cdot \left(r + \frac{1}{2}\right)}{2N} \right) \\ &= \sum_{k=0}^{N-1} (a_k - a_{N-1-k}) \cdot \cos \left(\frac{2\pi \cdot \left(k + \frac{N+1}{2}\right) \cdot \left(r + \frac{1}{2}\right)}{2N} \right) \\ &\quad + \sum_{k=N}^{2N-1} (a_k + a_{3N-1-k}) \cdot \cos \left(\frac{2\pi \cdot \left(k + \frac{N+1}{2}\right) \cdot \left(r + \frac{1}{2}\right)}{2N} \right) \end{aligned} \quad (36)$$

By the conjugate and skew conjugate property of the exponential vectors from (33) and (34), the proof can be completed.

$$\begin{aligned} \sum_{k=0}^{2N-1} \hat{a}_k \cdot \exp \left(\frac{i \cdot 2\pi \cdot \left(k + \frac{N+1}{2}\right) \cdot \left(r + \frac{1}{2}\right)}{2N} \right) &= \sum_{k=0}^{N-1} a_k \cdot \cos \left(\frac{2\pi \cdot \left(k + \frac{N+1}{2}\right) \cdot \left(r + \frac{1}{2}\right)}{2N} \right) \\ &\quad - \sum_{k=0}^{N-1} a_{N-1-k} \cdot (-1) \cdot \cos \left(\frac{2\pi \cdot \left(N-1-k + \frac{N+1}{2}\right) \cdot \left(r + \frac{1}{2}\right)}{2N} \right) \\ &\quad + \sum_{k=N}^{2N-1} a_k \cdot \cos \left(\frac{2\pi \cdot \left(k + \frac{N+1}{2}\right) \cdot \left(r + \frac{1}{2}\right)}{2N} \right) \\ &\quad + \sum_{k=N}^{2N-1} a_{3N-1-k} \cdot \cos \left(\frac{2\pi \cdot \left(3N-1-k + \frac{N+1}{2}\right) \cdot \left(r + \frac{1}{2}\right)}{2N} \right) \\ &= 2 \cdot \sum_{k=0}^{2N-1} a_k \cdot \cos \left(\frac{2\pi \cdot \left(k + \frac{N+1}{2}\right) \cdot \left(r + \frac{1}{2}\right)}{2N} \right) \end{aligned} \quad (37)$$

■

On the other hand, for the backward part, a relative ISDFT can be used to inversely transform the frequency coefficients to the aliased signal. However, in according to the fact that the aliased signal is real values, it suggests there should be also symmetric properties on the frequency coefficients.

Theorem 3 For $r = 0 \sim 2N-1$,

$$\alpha_{2N-1-r} = (-1)^{N+1} \cdot \alpha_r \quad (38)$$

<Proof> Consider the relations of the elements of the r th basis vector and the $(2N-1-r)$ th basis vector,

$$\begin{aligned} \exp\left(\frac{i \cdot 2\pi \cdot \left(k + \frac{N+1}{2}\right) \cdot \left(2N-1-r + \frac{1}{2}\right)}{2N}\right) &= \exp\left(\frac{i \cdot 2\pi \cdot \left(k + \frac{N+1}{2}\right) \cdot \left(2N - \left(r + \frac{1}{2}\right)\right)}{2N}\right) \\ &= \exp\left(\frac{i \cdot 2\pi \cdot \left(k + \frac{N+1}{2}\right) \cdot \left(r + \frac{1}{2}\right)}{2N} + i \cdot \pi \cdot (N+1)\right) \\ &= (-1)^{N+1} \cdot \exp^*\left(\frac{i \cdot 2\pi \cdot \left(k + \frac{N+1}{2}\right) \cdot \left(r + \frac{1}{2}\right)}{2N}\right) \end{aligned} \quad (39)$$

From (39), we have

$$\begin{aligned} \alpha_{2N-1-r} &= \sum_{k=0}^{2N-1} \hat{a}_k \cdot \frac{1}{\sqrt{2N}} \cdot (-1)^{N+1} \cdot \exp^*\left(\frac{i \cdot 2\pi \cdot \left(k + \frac{N+1}{2}\right) \cdot \left(r + \frac{1}{2}\right)}{2N}\right) \\ &= (-1)^{N+1} \cdot \left(\sum_{k=0}^{2N-1} \hat{a}_k \cdot \frac{1}{\sqrt{2N}} \cdot \exp\left(\frac{i \cdot 2\pi \cdot \left(k + \frac{N+1}{2}\right) \cdot \left(r + \frac{1}{2}\right)}{2N}\right) \right)^* \\ &= (-1)^{N+1} \cdot \alpha_r^* \end{aligned} \quad (40)$$

Since α_r is real value, it implies that $\alpha_{2N-1-r} = (-1)^{N+1} \cdot \alpha_r$. ■

The next theorem gives an equivalent method on real domain to get the aliased signal by ISDFT without any complex value calculation. That is exactly IMDCT.

Theorem 4 For $k=0 \sim 2N-1$,

$$\hat{a}_k = \frac{2}{\sqrt{2N}} \cdot \sum_{r=0}^{N-1} \alpha_r \cdot \cos\left(\frac{2\pi \cdot \left(k + \frac{N+1}{2}\right) \cdot \left(r + \frac{1}{2}\right)}{2N}\right) \quad (41)$$

<Proof> Apply ISDFT, we have

$$\hat{a}_k = \sum_{r=0}^{2N-1} \alpha_r \cdot \frac{1}{\sqrt{2N}} \exp \left(\frac{i \cdot 2\pi \cdot \left(k + \frac{N+1}{2} \right) \cdot \left(r + \frac{1}{2} \right)}{2N} \right) \quad (42)$$

Furthermore, since the fact that \hat{a}_k and α_r are both real value, it shows the image part of the summation in (42) can be vanished and then (42) can be simplified to (43).

$$\hat{a}_k = \frac{1}{\sqrt{2N}} \cdot \sum_{r=0}^{2N-1} \alpha_r \cdot \cos \left(\frac{2\pi \cdot \left(k + \frac{N+1}{2} \right) \cdot \left(r + \frac{1}{2} \right)}{2N} \right) \quad (43)$$

By the theorem3 and (39),

$$\begin{aligned} & \sum_{r=0}^{2N-1} \alpha_r \cdot \cos \left(\frac{2\pi \cdot \left(k + \frac{N+1}{2} \right) \cdot \left(r + \frac{1}{2} \right)}{2N} \right) \\ &= \sum_{r=0}^{N-1} \alpha_r \cdot \cos \left(\frac{2\pi \cdot \left(k + \frac{N+1}{2} \right) \cdot \left(r + \frac{1}{2} \right)}{2N} \right) \\ & \quad + \sum_{r=N}^{2N-1} (-1)^{N+1} \cdot \alpha_{2N+1-r} \cdot (-1)^{N+1} \cdot \cos \left(\frac{2\pi \cdot \left(k + \frac{N+1}{2} \right) \cdot \left(2N+1-r + \frac{1}{2} \right)}{2N} \right) \\ &= 2 \cdot \sum_{r=0}^{N-1} \alpha_r \cdot \cos \left(\frac{2\pi \cdot \left(k + \frac{N+1}{2} \right) \cdot \left(r + \frac{1}{2} \right)}{2N} \right) \end{aligned} \quad (44)$$

■

To summary the discussion above, (45) and (46) give a transform system consisting of MDCT and IMDCT to map the frame vector processed and inversely reconstruct an aliased frame vector.

$$\alpha_r = \frac{2}{\sqrt{2N}} \cdot \sum_{k=0}^{2N-1} a_k \cdot \cos \left(\frac{2\pi \cdot \left(k + \frac{N+1}{2} \right) \cdot \left(r + \frac{1}{2} \right)}{2N} \right), \text{ for } r = 0 \sim 2N-1 \quad (45)$$

$$\hat{a}_k = \frac{2}{\sqrt{2N}} \cdot \sum_{r=0}^{N-1} \alpha_r \cdot \cos \left(\frac{2\pi \cdot \left(k + \frac{N+1}{2} \right) \cdot \left(r + \frac{1}{2} \right)}{2N} \right), \text{ for } k = 0 \sim 2N-1 \quad (46)$$

Furthermore, the behind half of the $2N$ transform coefficients α_r can be discarded without any information lost.

However, a window function h_k always applies to the frames processed in audio coding [22]. This will cause the fault of reconstruction by overlap-add operation in general. To conceal the affect of the window function h_k , the window function g_k is also required at the backward part. The conditions on the window functions for PR are discussed as follow. Once the window function applies, the aliased signal reconstructed is

$$\hat{a}_k = \begin{cases} h_k \cdot a_k - h_{N-1-k} \cdot a_{N-1-k}, & \text{for } k = 0 \sim N-1 \\ h_k \cdot a_k + h_{3N-1-k} \cdot a_{3N-1-k}, & \text{for } k = N \sim 2N-1 \end{cases} \quad (47)$$

At the backward part, after the window function g_k applies, the windowed aliased signal is

$$g_k \cdot \hat{a}_k = \begin{cases} g_k \cdot h_k \cdot a_k - g_k \cdot h_{N-1-k} \cdot a_{N-1-k} & \text{for } k = 0 \sim N-1 \\ g_k \cdot h_k \cdot a_k + g_k \cdot h_{3N-1-k} \cdot a_{3N-1-k} & \text{for } k = N \sim 2N-1 \end{cases} \quad (48)$$

Furthermore, after the overlap-add operation between the front half of the current frame and the tail half of the last frame, the reconstructed signal sample of the front half of the current frame is

$$\tilde{a}_k = g_k \cdot h_k \cdot a_k - g_k \cdot h_{N-1-k} \cdot a_{N-1-k} + g_{N+k} \cdot h_{N+k} \cdot a'_{N+k} + g_{N+k} \cdot h_{2N-1-k} \cdot a'_{2N-1-k} \quad (49)$$

for $k = 0 \sim N-1$, where a'_k represent the samples of the last frame. Since $a_k = a'_{N+k}$, (49) is simplified as (50).

$$\tilde{a}_k = (g_k \cdot h_k + g_{N+k} \cdot h_{N+k}) \cdot a_k - (g_k \cdot h_{N-1-k} - g_{N+k} \cdot h_{2N-1-k}) \cdot a_{N-1-k} \quad (50)$$

Assume the window function g_k is the same as the window function h_k , that is

$$\tilde{a}_k = (h_k^2 + h_{N+k}^2) \cdot a_k - (h_k \cdot h_{N-1-k} - h_{N+k} \cdot h_{2N-1-k}) \cdot a_{N-1-k} \quad (51)$$

Hence, the sufficient conditions on the window functions for PR are, for $k = 0 \sim 2N-1$,

$$g_k = h_k \quad (52)$$

$$h_k^2 + h_{N+k}^2 = 1 \quad (53)$$

$$h_k \cdot h_{N-1-k} - h_{N+k} \cdot h_{2N-1-k} = 0 \quad (54)$$

The assumption of the symmetry of the window function that $h_k = h_{2N-1-k}$ can guarantee (54) holds. On the assumptions, the sin function is used widely in audio coding.

$$h_k = \sin\left(\frac{\pi \cdot \frac{k + \frac{1}{2}}{2}}{2N}\right), \text{ for } k = 0 \sim 2N - 1. \quad (55)$$

In AAC, the Kaiser window is another example of the window function satisfying the assumptions [23]. Especially, there are also several other window types offered in MP3 and AAC to adapt to different signal situations to avoid the annoying pre-echo effect [22].

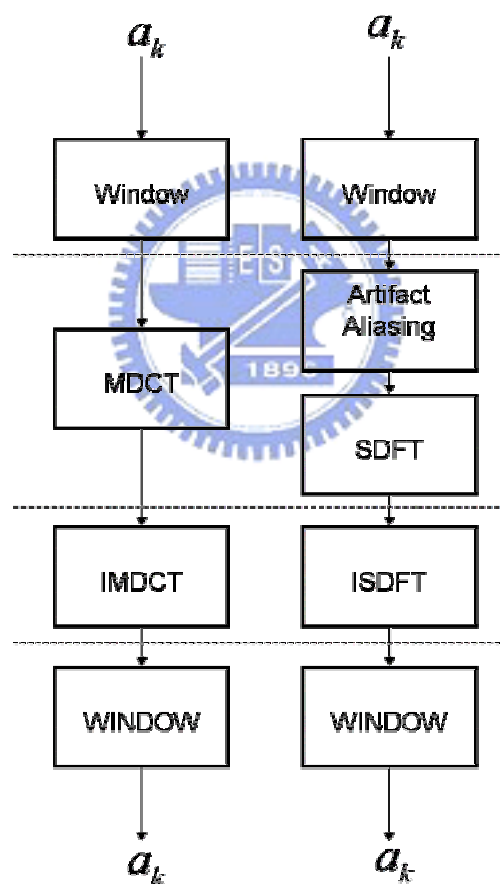


Figure 13: The MDCT-based transform system and the relative equivalent SDFT-based transform system.