ISSN No: - 2456 – 2165

# A Novel Approach for Transformation Technique using Clenshaws Recurrence Formula

S.mounika#[1], P.sai chaitanya[#2]

[1]Student of MTECH (DSCE), Dept. of ECE, KITS(S), email: suddalamounika12@gmail.com,

[2] Asst Professor in ECE dept. KITS(S), email:chaithu456@gmail.com.

***Abstract-*** **The MPEG sound coding standard utilizes the progressively windowed adjusted discrete cosine change (MDCT) to accomplish an excellent execution. Coordinate calculation of the MDCT in MPEG coding and of the converse MDCT (IMDCT) in MPEG deciphering are computationally concentrated errands. Accordingly, proficient calculations for the MDCT and IMDCT are of prime significance inside the sound coding and interpreting process.**

**The forward and converse altered discrete cosine change (MDCT) are two of the most computational escalated operations in the MPEG sound coding standard. In this venture, we utilized Clenshaw's repeat equation to change parts of the MDCT and IMDCT of the general length. Clenshaw's repeat recipe is an effective approach to assess the entirety of results of ordered coefficients that obey recursive relations. Proficient usage of MDCT and IMDCT are gotten. The proposed consistent structures are especially reasonable for parallel VLSI realization.**

**Simulation results are carried out using MATLAB for signal reconstruction by considering different signals. Further, the proposed approach is applied for signal compression and the compression results are obtained. It is observed from the simulation results that the proposed approach is more efficient that straightforward method.**

***Index terms:*** *Audio coding modified discrete cosine transform (MDCT), MPEG.*

## I. INTRODUCTION

The MPEG audio coding standard employs the dynamically windowed modified discrete cosine transform (MDCT) to achieve a high quality performance direct computation of the inverse MDCT(IMDCT)in MPEG decoding are computationally intensive tasks .therefore ,efficient algorithms for the MDCT and IMDCT are of prime importance within the audio coding and decoding process .Recently recursive algorithms for the forward and inverse MDCT,while can be the implemented by parallel vlsi filters in addition to it we have proposed recursive algorithms for the general length discrete cosine transform (DCT).based on the concept of the previously reported algorithms ,we used clenshaw's recurrence formula to transform kernels of the MDCT and IMDCT, and we propose new recursive structures for the general length MDCT and IMDCT.the proposed regular structures are less particularly suitable for parallel VLSI realization and require less hard ware and number of operations than the one proposed in direct method .

### *A. Clenshaw's Recurrence Formula*

Clenshaw's recurrence formula (with an associated sum) is an efficient way to evaluate a sum of coefficients multiplied by functions that obey a recurrence formula. It has been used extensively in physical geodesy in the evaluation of sums of high degree and order spherical harmonic series approximating the earth's gravitational potential. It is also used in other applications like map projections where Clenshaw's method is used to develop compact formula for meridian distance and for computation of coordinates, grid convergence and point scale factor on the Transverse Mercator (TM) projection using the Karney-Krueger equations

The forward and inverse modified discrete cosine transform (MDCT) are two of the most computational intensive operations in the MPEG audio coding standard. In this paper, we used Clenshaw's recurrence formula to transform kernels of the MDCT and IMDCT of the general length. Efficient implementations of MDCT and IMDCT are obtained. The proposed regular structures are particularly suitable for parallel VLSI realization. Our solution requires less hardware, and we achieved significant savings for a number of operations compared with existing related systems.

Clenshaw's recurrence formula is an elegant and efficient way to evaluate a sum of coefficients time's functions that obey a recurrence formula. In this paper, it is used to obtain recursive algorithms for the forward and inverse MDCT. Let us first see the formulation of Clenshaw's recurrence formula.-
Suppose that the desired sum is

$$f(x) = \sum_{k=0}^{N-1} C_K\, F_K(x)$$

ISSN No: - 2456 – 2165

and that $F_K(x)$ obeys the recurrence relation

$F_{n+1}(x) = \alpha(n,x)F_n(x) + \beta(n,x)F_{n-1}(x)$ **……(a)**

For some functions $\alpha(n,x)$ and $\beta(n,x)$.

**1 *Downward order*:** We define the quantities $y_k$,

$k = N-1, \ldots\ldots, 0$ by the following recurrence: $y_{N+1} = y_N = 0$

$y_k = \alpha(k,x)y_{k+1} + \beta(k+1,x)y_{k+2} + c_k$. Clenshaw's recurrence formula states that the sum $f(x)$ can be Computed by $f(x) = \beta(1,x)F_0(x)y_2 + F_1(x)y_1 + F_0(x)c_0$

**2 *Upward order*:** We define the quantities $y_k, 0, \ldots\ldots, N-1$ by the following recurrence:

$$y_{-2} = y_{-1} = 0$$

$y_k = 1/\beta(k+1,x)[y_{k-2} - \alpha(k,x)y_{k-1} - c_k]$ **………. (C)**

Clenshaw's recurrence formula states that the sum $f(x)$ can be computed by

$f(x) = c_{N-1}F_{N-1}(x) - \beta(N-1,x)F_{N-2}(x)y_{N-2} - F_{N-1}(x)y_{N-3} \ldots\ldots\ldots(\boldsymbol{D})$

## II. DERIVATION OF THE MDCT ALGORITHM

Let x(n),n=0,1,2….N-1.the MDCT of x(n) is given by

$$X(k) = \sum_{n=0}^{N-1} x(n)\cos[(n + (M+1)/2)\,(K+1/2)\,\pi/M]$$

K=0,1,…..M-1

Where N=2M is the window length, and M is the number of transform coefficients. To obtain the recursive transfer function of X(K),

$$\text{let } \theta_K = (K+1/2)\,\pi/M.$$

we define $F_n(\theta_k) = cos[(n + \frac{M+1}{2})\,(K+1/2)\,\pi/M] = \cos[(n + (M+1)/2)\theta_k]$.

Further we have

$F_{n+1}(\theta_K) = \cos[(n+1+(M+1)/2)\theta_K]$

$= 2cos\theta_k \cos\left[\left(n + \frac{M+1}{2}\right)\theta_K\right] - COS[(n-1+\frac{M+1}{2})\theta_k]$

$= 2cos\theta_k F_n(\theta_k) - F_{n-1}(\theta_k)$

And comparing with equation **(a),** we will get

$$\alpha(n,k) = \alpha(k) = 2cos\theta_k$$

And $\beta(n,k) = \beta = -1.$

Now, according to equation (C) for upward order, we define

$$a_{-2} = a_{-1} = 0$$

$a_n = x(n) + 2cos\theta_k a_{n-1} - a_{n-2}$

$n = 0,1,\ldots\ldots,N-1$ **………………..(F)**

and from (D), it follows that

$X(k) = x(N-1)F_{N-1}(\theta_K) + F_{N-2}(\theta_K)a_{N-2} - F_{N-1}(\theta_K)a_{N-3}$. From the definition of $F_n(\theta_K)$, we have $F_{N-1}(\theta_K) = COS[\left(N-1+\frac{M+1}{2}\right)\theta_K]$

$= -COS\left[\frac{M-1}{2}\,\theta_K\right]$ $F_{N-2}(\theta_K) = COS[\left(N-2+\frac{M+1}{2}\right)\theta_K]$

$= -COS[\frac{M-3}{2}\,\theta_K]$

And if we put this in the previous equation, we obtain

$$X(k) = -\cos\left[\frac{(M-1)\,\theta_K}{2}\right]x(N-1) - \cos\left[\frac{(M-3)\,\theta_K}{2}\right]a_{N-2} + \cos\left[\frac{(M-1)\,\theta_K}{2}\right]a_{N-3}$$

**………. (E)**

Formula (E) can be written as

$\boldsymbol{X(k) = [a_{n-3} - x(N-1)]COS\left[\frac{(M-1)\theta_K}{2}\right] - \cos\left[\frac{(M-3)\,\theta_K}{2}\right]a_{N-2}}$ **….(G)**

From (F)

for n=N-1, we have $a_{N-3} = x(N-1) = -a_{N-1} + 2COS\theta_K a_{N-2}$

if we and if we put this into the previous equation, after mathematical transforming, we will obtain $\boldsymbol{X(k)} = -\boldsymbol{COS}\left[\frac{(M-1)\theta_K}{2}\right]\boldsymbol{a_{N-1}} + \mathbf{cos}\left[\frac{(M+1)\,\theta_K}{2}\right]\boldsymbol{a_{N-2}}$**..........(H)**

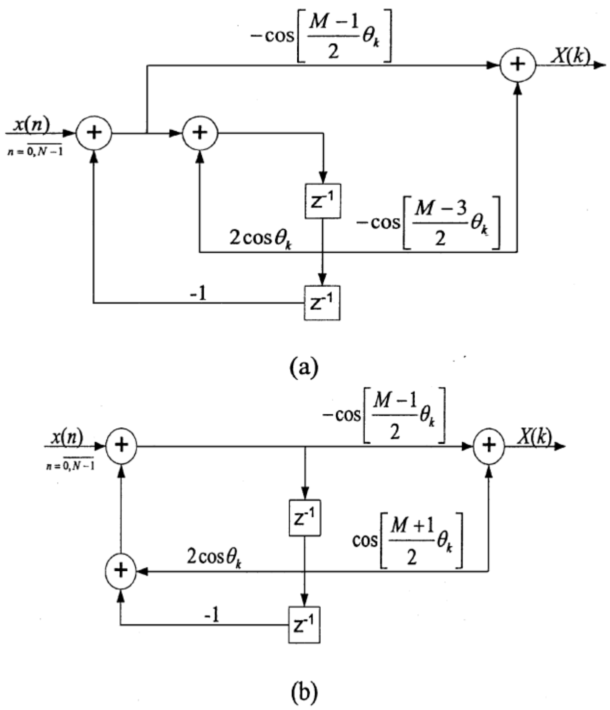



Fig 1.a and fig 1.b(recursive implementation of mdct derived by using upward order clenshaw's recurrence formula.

Therefore, we recursively generate the $a_n$ from the input sequence x(n) according to upward order. At the Nth step, the Kth MDCT coefficient X(k) is evaluated from either (G)or (H).

Fig. 1(a) and (b) shows the digital implementation for the computation of the kth transform element according to (G) and (H), respectively. All elements of the transformation can be computed in parallel and can be implemented in VLSI.
These implementations require one fewer delay element than the method presented in [I]; see Table I. They require the same number of multipliers and adders. To compute M points of output, one needs (N+2) multiplications and (2N+1) additions per output sample, whereas the approach requires (2N + 1) multiplications and 3N additions.

According to down ward order, we can also define

$$a_{N+2} = a_{N+1} = 0$$

$$a_n = 2cos\theta_k a_{n+1} - a_{n+2} + x(n)$$

n=N-1 …1, 0,**…………….**(M)

Similar to upward order if we derive the equation for down ward order final equation that we will get

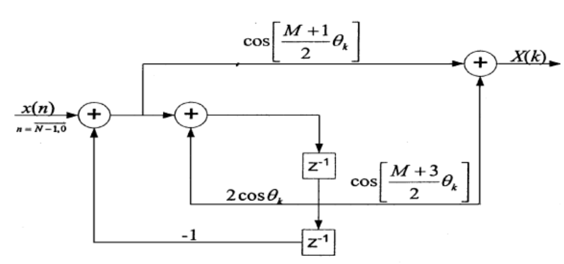
$$X(k) = cos\left[\frac{(M+1)\theta_k}{2}\right]a_0 cos\left[\frac{(M-1)\theta_k}{2}\right]a_1 \text{............(N)}$$

Table1

| | N-point MDCT | | | N/2-point IMDCT | | |
|---|---|---|---|---|---|---|
| | Proposed upward | Proposed downward | [1] | Proposed upward | Proposed downward | [1] |
| Latches | 2 | 2 | 3 | 2 | 2 | 3 |
| Multipliers | 3 | 3 | 3 | 2 | 2 | 2 |
| Adders | 3 | 3 | 3 | 3 | 3 | 3 |
| Multiplications | N+2 | N+2 | 2N+1 | N/2+1 | N/2+1 | N/2+1 |
| Additions | 2N+1 | 2N+1 | 3N | N+1 | N+1 | 3N/2 |
| Parity | + | + | + | - | + | - |
| Order of input | + | - | + | + | - | + |

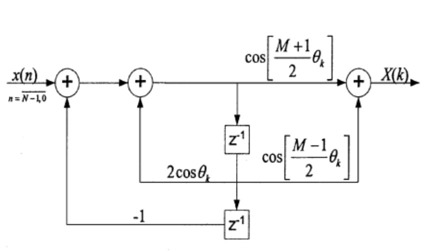
Again, we recursively generate the $a_n$ from the input sequence x (n), only now, it is according to (M). At the Nth step, the Kth MDCT coefficient is evaluated from equation (N). Fig. 2(a) and (b) shows the digital implementation for the computation of the kth transform element according to (18), respectively. All elements of the transformation can be computed in parallel and can be implemented in VLSI. These implementations require one fewer delay element than the method presented in, but the input sequence is in reverse order; see Table I. They require the same number of multipliers and adders.

To compute M points of output, one needs (N+1) multiplications and (2N+1) additions per output sample, whereas the approach requires (2N+1) multiplications and 3N additions. If we summarize the results of proposed implementations for MDCT from Table I, it can be seen that we achieved significant savings for number of operations: approximately a 50% saving for the number of multiplications and a 30% saving for the number of additions. We also decreased the number of delay elements for one. The first solution with upward order is rather superior to the second one with downward order because of natural order of the input sequence.




(a)

ISSN No: - 2456 – 2165




(b)

Fig2.(a) and 2(b) (recursive implementation of mdct derived by using down ward order clenshaw's recurrence formula.

## III. DERIVATION OF THE IMDCT ALGORITHM

Let X(K),K=0,1,…..M-1.the IMDCT of X(k) is given by $x(n) = \sum_{k=0}^{M-1} X(k)\cos[n + (M+1)/2)(K+\frac{1}{2})\pi M]$

n=0, 1,…..N-1.

To obtain the recursive transfer function of x (n) let

$$\theta_n = [n + \frac{M+1}{2}]\,\pi/M$$

$$F_K(\theta_n) = cos\left[\left(n + \frac{M+1}{2}\right)\left(k+\frac{1}{2}\right)\pi/M\right]$$

$$=\cos\left[\left(K+\frac{1}{2}\right)\theta_n\right].$$

Further, we have

$$F_{K+1}(\theta_n) = \mathrm{Cos}\left[\left(K+1+\frac{1}{2}\right)\theta_n\right].$$

$$=2\cos\theta_n\cos\left[\left(K+\frac{1}{2}\right)\theta_n\right] - \cos\left[\left(K-1+\frac{1}{2}\right)\theta_n\right]$$

$$=2\cos\theta_n\, F_K(\theta_n) - F_{K-1}(\theta_n)$$

And comparing with equation …..(a)

We get $\alpha(n,k)= \alpha(n)$ =2cos$\theta n$ and $\beta(n,k)$=β=-1

according to upward order we define $x(n) = X(M-1)F_{M-1}(\theta_n) + F_{M-2}(\theta_n)\,b_{M-2} - F_{M-1}(\theta_n)\,b_{M-3}$

From the definition of $F_K(\theta_n)$,we have

$$F_{M-1}(\theta_n) = \cos[(M-1+\frac{1}{2})\theta_n]$$

$$\cos[(n + \left(M+\frac{1}{2}\right)\pi - \frac{\theta_n}{2})]$$

$$(-1)^{n+\left(\frac{M+1}{2}\right)} cos3(\frac{\theta_n}{2}),\quad M$$

*ODD*

$$(-1)^{n+\left(\frac{M}{2}\right)} sin3(\frac{\theta_n}{2}),\quad M\ EVEN$$

if we put this in to the previous equation ,we obtain

$$x(n) = (-1)^{n+\left(\frac{M+1}{2}\right)} cos\left(\frac{\theta_n}{2}\right) X(M-1)$$

$+(-1)^{n+\left(\frac{M+1}{2}\right)} cos\left(\frac{3\theta_n}{2}\right)\ b_{M-2}$ - $(-1)^{n+\left(\frac{M+1}{2}\right)}.\ cos\left(\frac{\theta_n}{2}\right)\ b_{M-3}$..m odd

$x(n) = (-1)^{n+\left(\frac{M}{2}\right)} sin\left(\frac{\theta_n}{2}\right) X(M-1)$

$+(-1)^{n+\left(\frac{M+1}{2}\right)} sin\left(\frac{3\theta_n}{2}\right)\ b_{M-2}$ - $(-1)^{n+\left(\frac{M+1}{2}\right)}.\ sin\left(\frac{\theta_n}{2}\right)\ b_{M-3}$even

From the concept of upward order which was discussed earlier

$x(n) = (-1)^{n+\left(\frac{M+1}{2}\right)} cos\left(\frac{\theta_n}{2}\right) [\,b_{M-1} - b_{M-2}]$ M odd and **…… (x.a)**

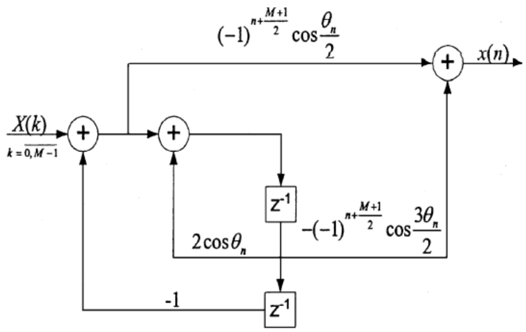
$x(n) = (-1)^{n+\left(\frac{M}{2}\right)} sin\left(\frac{\theta_n}{2}\right) [\,b_{M-1} + b_{M-2}]\ M..EVEN$**……… (x.b)**

$b_k$ is recursively generated from the input sequence x(k) according to upward order . At the Mth step, the Nth IMDCT coefficient is evaluated from either (X.a) or (X.b). Figs. 3 and 4 show the digital implementation for the computation of the nth transform element according to (x.a) and (x.b), respectively. All elements of the transformation can be computed in parallel and can be implemented in VLSI. The implementation in Fig. 3 requires one delay element less than the method presented in one multiplier more. The number of adders is the same. To compute N points of output, one needs (M+2 ) multiplications and (2M+1 ) additions per output sample, whereas the approach requires(M+1) multiplications and additions

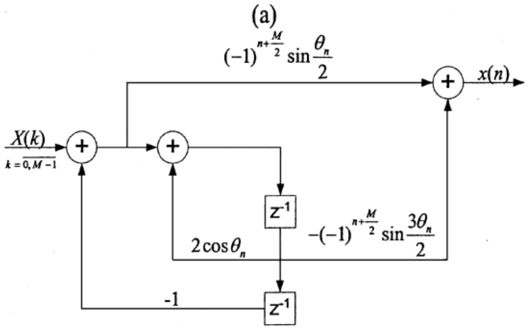
According to down ward order we can also define

$$[\,b_{M+2} = b_{M+1}] = 0$$

$$b_k = 2cos\theta_n\, b_{k+1} - b_{k+2} + X(k), \ldots.k = M-1, \ldots.1,$$




(a)




(b)

Above Fig 3(a) and fig 3(b) explains Recursive implementation of IMDCT according to (x.a) derived by using upward order Clenshaw's recurrence formula. (a)M odd. (b) M even.

According to down ward order which was discussed earlier we can also define

$$[\, b_{M+2} = b_{M+1}] = 0$$

$$b_k = 2cos\theta_n\, b_{k+1} - b_{k+2} + X(k), \ldots. k = M-1, \ldots.1,0$$

And from clenshaw's recurrence formula, it follows that

$$x(n) = -F_0(\theta_n)\, b_2 + F_1(\theta_n)\, b_1 + F_0(\theta_n)X(0).$$

From the definition of

$$F_n(\theta_k), we\ get$$

$$F_0(\theta_n) = cos\frac{\theta_n}{2}$$

$$F_1(\theta_n) = cos\frac{3\theta_n}{2}$$

And if we put this in to previous equation, we will obtain

$$x(n) = X(0)cos\frac{\theta_n}{2} + cos\frac{3\theta_n}{2}b_1 - cos\frac{\theta_n}{2}b_2$$

It can be also rewritten as

$$x(n) = \left[X(0) - b_2\right] cos\frac{\theta_n}{2} + cos\frac{3\theta_n}{2}b_1$$

From down ward order equation for the value of k=0, we have

$$X(0) - b_2 = b_0 - 2cos\theta_n b_1$$

And if we put this in to the previous equation, after mathematical transforming, we will obtain

$$\boldsymbol{x(n) = cos\frac{\theta_n}{2}[b_0 - b_1]}$$

If we summarize the results of proposed implementations for inverse MDCT from Table I, it can be seen that we achieved approximately 30% saving for number of additions. We also decreased the number of delay elements for one. In addition, in proposed solutions for inverse MDCT with downward order, it does not need to be considered the parity of M, but in some implementations, it might be necessary to have additional memory. From the other side in proposed solutions with upward order, it is not necessary to have additional memory, and in most implementations M is even; therefore, only one realization for inverse MDCT is required. Again, we can say that the first solution with upward order is rather superior to the second one with downward order because of natural order of the input sequence.

## II. EXPERIMENTAL RESULTS

In order to verify the proposed faster approach, we have considered different standard signals. The proposed approach is used to reconstruct the signal. The original and reconstructed signals are shown in Figs. 5.1 and 5.2. Further the proposed approach is applied for signal compression. The original and compressed signal after reconstruction are shown in Figs. 5.3 and 5.4.

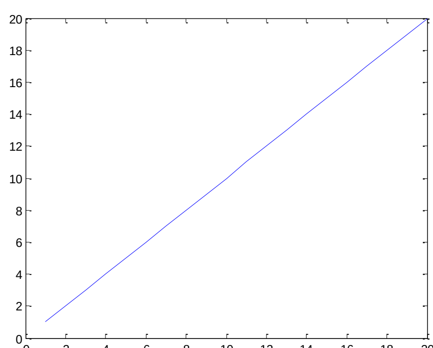



Fig. 5.1 Original signal of 20 points

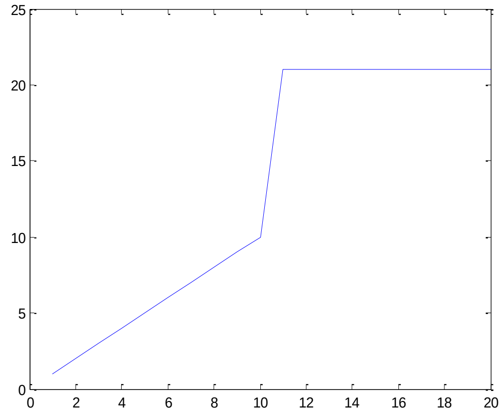



Fig. 5.2 Reconstructed signal using the proposed approach

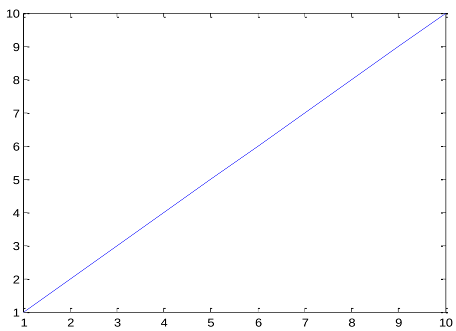



Fig.5.3. Original signal of 10 points

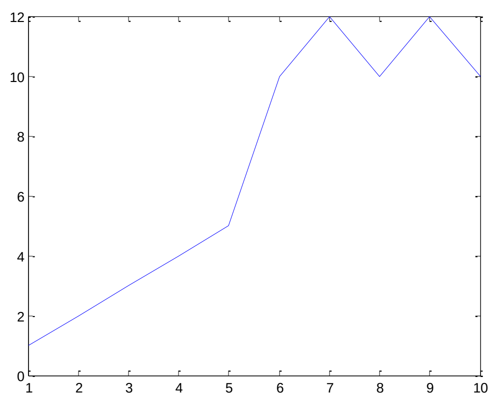



Fig.5.4. Compressed signal using proposed approach

## III. CONCLUSIONS AND SCOPE FOR FUTURE WORK

The MPEG audio coding standard employs the dynamically windowed modified discrete cosine transform (MDCT) to achieve a high-quality performance. Direct computation of the MDCT in MPEG coding and of the inverse MDCT (IMDCT) in MPEG decoding are computationally intensive tasks. Therefore, efficient algorithms for the MDCT and IMDCT are of prime importance within the audio coding and decoding process.

The forward and inverse modified discrete cosine transform (MDCT) are two of the most computational intensive operations in the MPEG audio coding standard. In this project work, we proposed new recursive algorithms for the forward and inverseMDCTof general length, which are suitable for parallel VLSI implementation. The Clenshaw's recurrence formula is used for derivation. We used Clenshaw's recurrence formula to transform kernels of the MDCT and IMDCT of the general length. Clenshaw's recurrence formula is an efficient way to evaluate the sum of products of indexed coefficients that obey recursive relations. Efficient implementations of MDCT and IMDCT are obtained. The proposed regular structures are particularly suitable for parallel VLSI realization.

Simulation results are carried out for signal reconstruction by considering different signals. Further, the proposed approach is applied for signal compression and the compression results are obtained. It is observed from the simulation results that the proposed approach is more efficient that straightforward method. Future work would be to use the method presented for a number of image processing applications like image watermarking, face recognition etc. Further work would also be to extend this faster approach to other transforms.

## REFERENCES

[1] "Regressive implementations for the forward and inverse MDCT in MPEG audio coding," IEEE Signal Processing vol. 3, pp. 116–118, Apr. 1996

[2], "Recursive algorithms for the forward and inverse discrete cosine transform with arbitrary length,"IEEE Signal Processing vol. 1, pp. 101–102, July 1994.

[3] "Recursive algorithm for the discrete cosine transform with general lengths," vol. 30, pp. 197–198, Feb. 1994.

[4] "Efficient recursive algorithm for the inverse discrete cosine transform,"IEEE Signal Processing vol. 7, pp. 276–277, Oct. 2000.

[5] "Computation of DCT using Clenshaw's recurrence formula," IEEE Signal Processing vol. 2, pp. 155–156, Aug. 1995.

[6] W. H. Press, S. A. Teukolsky, W. T. Vetter ling, and B. P. Flannery, Numerical Recipes in C. Cambridge, U.K.: Cambridge Univ. Press, 1992.

[7] V. Nikolayevich and G. Fettweis, “New recursive algorithms for the forward and inverse MDCT,” in Proc. IEEE Workshop Signal Process .Syst., Antwerp, Belgium, Sept. 2001.

[8] “Information technology—Coding of moving pictures and associated audio for digital storage media at about1.5Mbit/s, Part 3: Audio,” SO/IEC JTC1/SC29/WG11 MPEG, IS11172–3, 1992.

[9] “Information technology generic coding of moving pictures and associated audio, Part 3: Audio”. SO/IEC JTC1/SC29/WG11 MPEG, IS13818–3, 1994.

[10] “Computation of forward and inverse MDCT using clenshaw’s recurrence formula”, IEEE transactions on signal processing, vol51, no.5, May 20.