computing@tanet.edu.te.ua
www.tanet.edu.te.ua/computing

International Scientific Journal of Computing

ISSN 1727-6209

NEW RECURSIVE – DCT IMPLEMENTATIONS WITH GOERTZEL FILTERS

Sьleyman Sərrə Demirsoy, Robert Beck, Эzzet Kale, Andrew G. Dempster

Advanced DSP and VLSI Systems
Department of Electronic Systems, University of Westminster
115 New Cavendish St., London, W1M 8JS, United Kingdom
demirss@cmsa.wmin.ac.uk

Abstract: *In this paper we report on a new recursive DCT architecture that is more efficient in terms of area and power in comparison to recently published recursive DCT architectures. Our approach here employs Type A, B and C Goertzel filters. These three different realizations of Goertzel filters together with multiplier-less implementation of loop multiplications are used so as to reduce the area, the multiplier delay, and undesirable transitions hence the power consumption. The newly proposed DCT structure has been compared with conventional recursive implementations at different transform lengths to observe that there are potential savings both in area and power.*

Keywords: *DCT, Goertzel, Low Power, multimedia, multiplier-block*

1. INTRODUCTION

Discrete Cosine Transforms (DCT's) have been widely used in many applications related to spectral decomposition of images and video signals [5]. Different architectural considerations have been applied to result in computationally less complex designs. The general aim of research in recent years has been to make the designs less power hungry, which is of paramount importance for mobile applications [6],[7].

Time-recursive architectures are known for their modularity and reduced computational complexity. They are also free of global interconnects making them attractive for VLSI implementations [4],[5],[7]. Goertzel filter implementations of these time-recursive architectures are commonly employed for all sorts of trigonometric transforms [3],[6],[7].

However, the main recursive block, a 2nd order Infinite Impulse Response (IIR) filter, used in these implementations is always the same as seen in Figure 1 in [5]. The coefficient values used in this recursive filter has the characteristic of being minimal in terms of absolute value for the half Nyquist frequency н=0.25 and getting bigger as DC or Nyquist is approached.

Two alternative structures called Type A and Type C together with Type B, the commonly known form, in the parallel implementation of these transforms were proposed in [3]. These three structures

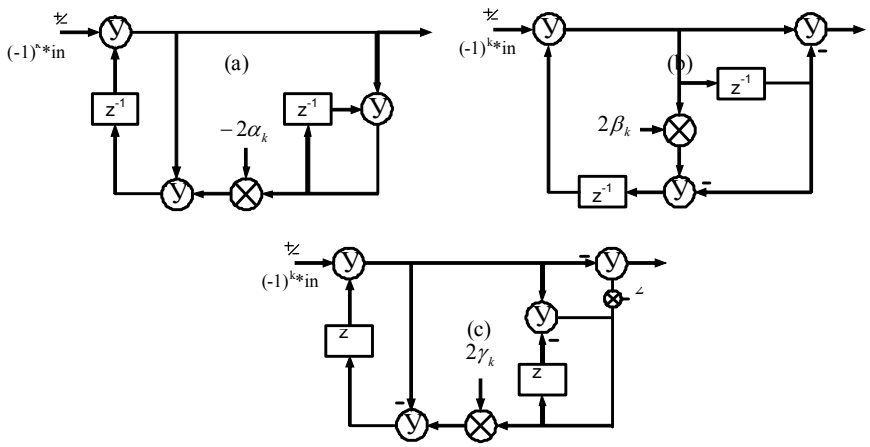

Figure 1. *(a) Type A (b) Type B (c) Type C Goertzel filters given in [3].*

are depicted in Figure 1 (a),(b) and (c). Please note that the multiplication by 2 in Type C structure does not require a physical multiplier but is implemented using hard-wire shifting. Furthermore the conditional scaling by –1 at the inputs of each block is effected by swapping the adders they feed by subtractors. Their coefficient realizations for a given frequency bin are given as follows [3]:

$$2\alpha_k = 2(1-\mathrm{Cos}(kp/N)) \quad (1a)$$

$$2\beta_k = 2\mathrm{Cos}(kp/N) \quad (1b)$$

$$2\gamma_k = 2(1+\mathrm{Cos}(kp/N)) \quad (1c)$$

where k is number of frequency bin and N is the transform length.

Figure 2 shows the numerical characteristics of these equations. It is observed that the coefficient value to be represented in fixed-point form is smallest if Type A is used near DC, Type B is used near н=0.25 and Type C is used near the Nyquist frequency.

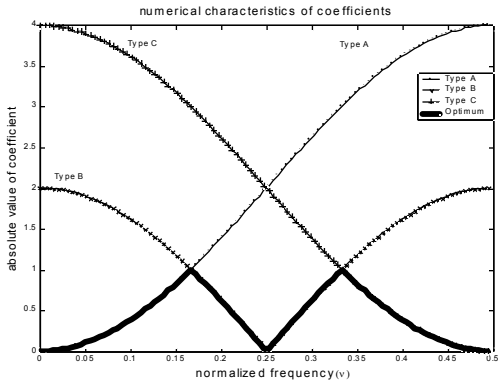

Figure 2. ***Numerical properties of the coefficient values for Type A, B and C Goertzel filters.***

2. IMPLEMENTATION

The behavioural implementation of the recursive modules was undertaken in MATLAB both in floating-point precision and fixed-point precision. The module was carefully optimised for a specified mean square error (MSE) noise level starting from the coefficient value, then fixing the register lengths and then the adder outputs.

For an N point DCT processor, there are N-1 frequencies each having a recursive DCT module. For k=0, the output of the DCT module is the sum of all N input data. Therefore a recursive module is not required. The first N/3 bins starting from k=1 are Type A structures. When N is not divisible by 3, N/3 is truncated to the nearest integer. In the same way, the last N/3 bins are designated to be Type C. The remaining bins in the middle are Type B.

In this study, we have concentrated on the implementation of a 1D-DCT processors to effect a better comparison for Type A and C blocks. Figure 3 gives a symbolic diagram of the DCT processor for the case of N=8. There is a physical loop multiplier inside every bin, which is hardwired for one of its inputs. The benefit of the hard-wired multiplier is that only the set bits of the coefficient are to be implemented. This kind of approach makes a lot of savings in the area of the multiplier.

Another method to reduce the area is to use multiplier-blocks [1]. By implementing the coefficients as the sums of shifted forms of input and cascading them, multiplication can be realized in a more efficient way in terms of area and delay [2]. This implementation also reduces the spurious glitches occurring at the outputs of the adders throughout the multiplication, making the implementation less power consuming. Our coefficient realizations for N=8 and noise-level requirement of $5*10^{-7}$ are:

$$\alpha_1 = \gamma_7 = 2^{-4}*((1-2^{-5})*(1-2^{-10}) + 2^{-2} + 2^{-13}) \quad (2a)$$

$$\alpha_2 = \gamma_6 = 2^{-2}*((1+2^{-2})*(1-2^{-4}-2^{-12}) + 2^{-18}) \quad (2b)$$

$$\beta_3 = -\beta_5 = 2^{-2}*(1+(2^{-1}+2^{-5})*(1-2^{-10})+2^{-19}) \quad (2c)$$

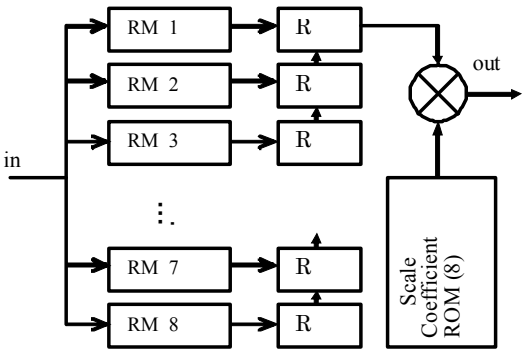

Figure 3. ***Flow diagram of 1D-DCT structure. RMs are Recursive modules, Rs are registers.***

If the same block is to be a multifunctional forward and Inverse DCT (IDCT), one should use a general-purpose multiplier in the loop too [5]. There is another multiplication outside the recursive module. So only one general-purpose multiplier as in Figure 3 with a scale coefficient ROM bank as an input is enough.

As seen from Figure 3, the input is directly taken into the recursive modules with the processing starting without latency. The clock is the same as the input data clock. This kind of approaches minimizes the need for storage elements used in the

pre-processing of some DCT algorithms [4]. However there is a drawback that the loops run for N clock cycles to calculate one output, resulting in the accumulation of errors. Pre-processing the input column data at the expense of storage elements and latency can reduce the accumulated errors [5]. To compare the area and word-lengths, a traditional all-Type-B design has also been realised and simulated.

3. SIMULATIONS

To demonstrate that using Type A and Type C recursive structures for near-DC and near-Nyquist frequency bins would be more area and power efficient, several transform lengths were considered. For each transform length, namely N=8, 16, 32 and 64, the all-Type-B architecture, and Type ABC architecture have been implemented with optimal (MSE sense) word-lengths. These word-lengths were constrained for three different MSE noise levels, namely $5*10^{-7}$, $5*10^{-5}$ and $5*10^{-3}$. When a 2D-DCT design is considered, the MSE noise level of $5*10^{-7}$ for 1D-DCT would result to be close to the MPEG standard [8].

The optimisation simulations were excited with uniformly distributed random numbers in the interval (-300,300). Figure 4 shows the required precision for the loop-multiplier coefficients at different transform lengths to achieve a MSE level of $5*10^{-7}$. It is seen in Fig. 2 that Type A and C coefficients are smaller at near DC and Nyquist, requiring less integer bits to represent. Thus, the loop-multiplier coefficients inside the recursive modules are shorter in length when Type A and Type C structures are used.

The effect of the coefficient word-length on the area of the loop-multiplier depends on the number of set bits (1's) inside each individual coefficient. Therefore, in order to estimate the area savings, it is required to calculate the number of ones inside the coefficient of each frequency bin. Figure 5 depicts the number of set bits for an error level of $5*10^{-7}$ for different transform lengths. It can be observed that as one approaches DC, the number of sets bits for Type A modules are less than those for Type B, since both coefficients are positive and the Type A coefficients are near zero. As one approaches Nyquist, Type C coefficients usually have one less set bit in comparison to Type B. This is due to the sign difference of Type C and Type B coefficients. Since they are complements of each other, the only difference between their binary representations is the MSB coming from the sign change.

The two 1D-DCT implementations are the same for the Type B bins, hence there is no difference in the coefficient lengths as seen in Figures 4 and 5.

A summary of comparative results in terms of number of adders used in the Type ABC structures is given in Table I. There are quite significant savings for N=16 and higher transform lengths, furthermore for N=8, the overhead of the extra adder stage in the Type C recursive module dominates the area savings in the loop-multiplier. However, the figure of area savings for N=8 and MSE level of $5*10^{-7}$ changes dramatically when the coefficient realizations are undertaken as shown in Equation 2. The multiplier area is reduced by 1/3 and the total area is 15.9% smaller than the design done

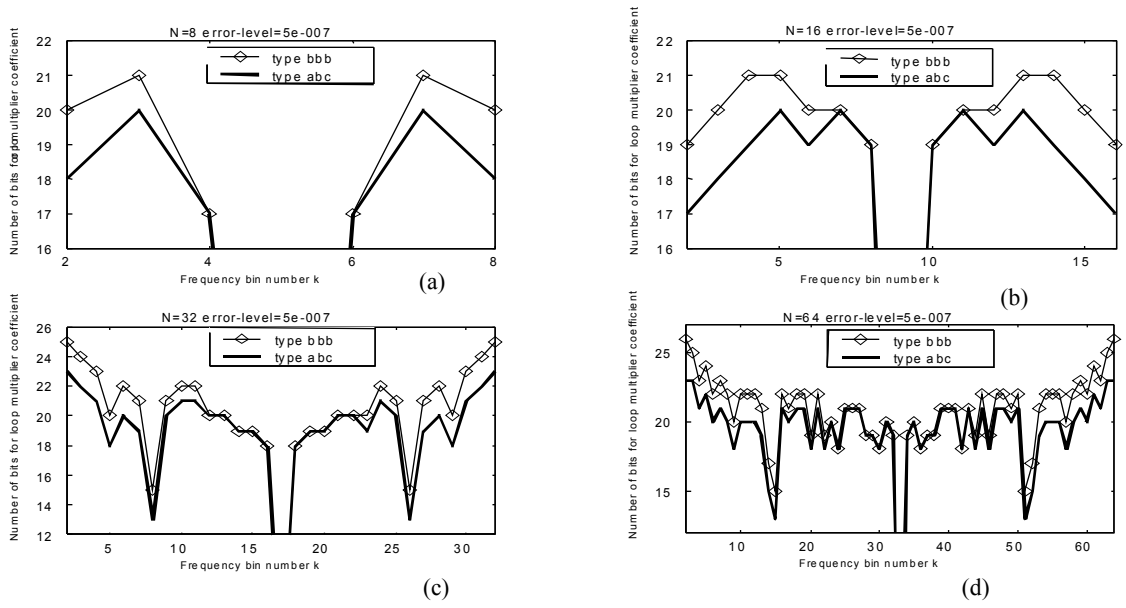

Figure 4. ***Comparison of coefficient word-length for MSE level of $5*10^{-7}$ for a) N=8 b) N=16 c) N=32 d) N=64***

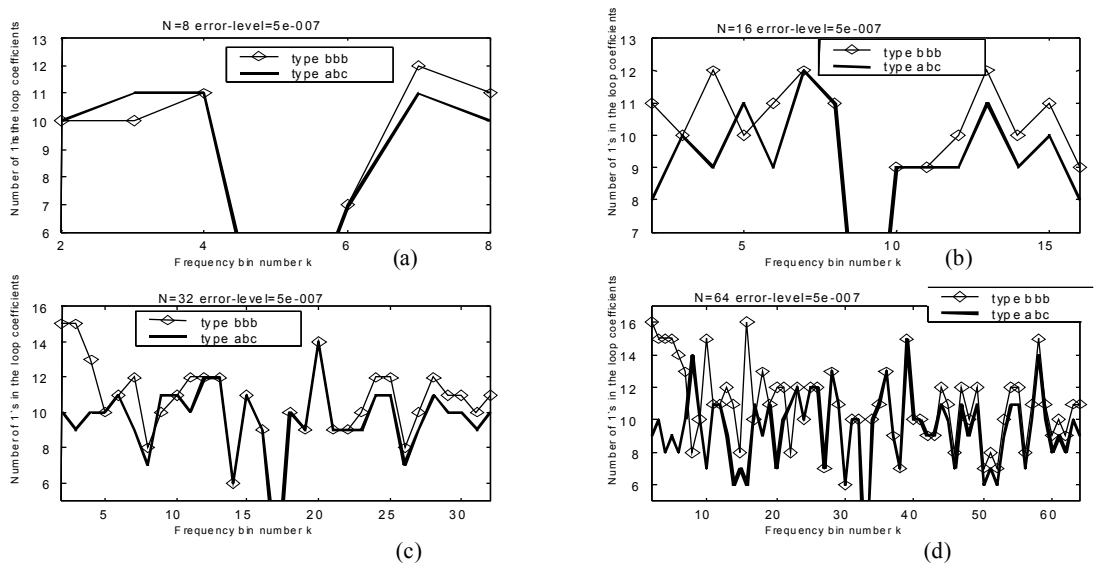

Figure 5. ***Comparison of set bits in the coefficients for MSE level of 5*10^{-7} for different N values.***

by the hard-wired multiplier. This multiplier-block technique together with Type A and Type C structures would give much better results for different N values. It would also result in a significant reduction of transitions occurring throughout the multiplier, thus enhancing the power consumption of the whole circuit. A more general study on power savings will be performed by effecting low-level implementation and power simulations.

The area savings in terms of adder bits if general-purpose multipliers were to be used inside the recursive loop is depicted in Table II. In all cases, the area is reduced by up to 8% with the Type ABC design.

Table 1. Area savings in Type ABC design for three different noise levels, when loop multiplier is hard-wired

N	5*10^{-7}		5*10^{-5}	5*10^{-3}
8	-0.21%	**15.9%**	-0.45%	1.65%
16	5.55%		3.70%	4.41%
32	6.84%		6.45%	5.81%
64	9.13%		8.59%	7.62%

Table II. Area savings in Type ABC design for three different noise levels, where loop multiplier is general purpose multiplier

N	5*10^{-7}	5*10^{-5}	5*10^{-3}
8	3.40%	5.48%	8.03%
16	5.40%	5.68%	6.98%
32	5.92%	6.70%	6.91%
64	6.12%	6.31%	6.17%

4. CONCLUSION

A novel DCT architecture has been proposed where the benefit of Type A and C Goertzel filters were exploited. It has been shown through behavioural simulations that there are area savings of up to 9% when these recursive modules are used instead of Type B near the DC and Nyquist frequencies. The coefficient word-length characterization for a given noise level has been performed for different transform length DCT processors. Combining the techniques of multiplier-blocks these savings have been increased further to 16% for an N=8 1D-DCT design. Our future work will focus on the silicon-level implementation of recursive 2D-DCT using the mentioned techniques as well as low-power implementation styles.

5. REFERENCES

[1] Demspter A.G. and Mecleod M.D., "Use of minimum-adder multiplier-blocks in FIR digital filters", IEEE Trans. CAS-II, vol. 42, no. 9, pp. 569-577, November 1995

[2] Demirsoy S. S., A. Dempster and I. Kale, "Transition analysis in multiplier-block based FIR filter structures", IEEE Int. Conf. on Elect., Circuit & Systems (ICECS), Lebanon, December 2000.

[3] Beck R., A.G. Dempster and I. Kale, "Finite-precision Goertzel filters used for signal tone detection", provisionally accepted to be published in IEEE Trans. CAS-II

[4] Srinivasan V.and K.J. Liu "VLSI Design

of High-Speed Time-Recursive 2-D DCT/IDCT Processor for Video Applications", IEEE Trans. CAS-Video Technology, vol 6,no. 1, pp.87-96, February 1997

[5] Yang J.F. and C.P. Fan, "Compact Recursive Structures for Discrete Cosine Transform", IEEE Trans. CAS-II, vol 47, no.4, April 2000

[6] Kuhlmann M and K. K. Parhi, "Power Comparison of flow-graph & distributed arithmetic based DCT architectures", 32. Asimolar Conf. Signals, Systems & Computers, pp 1214-19, 1998

[7] Wang J.L et al, "Implementation of the DCT and its Inverse by recursive structures", IEEE Workshop on Signal Processing Systems, pp 120-130, Oct 1999

[8] ISO/IEC, "Information technology-Coding of audio-visual object: Visual ISO/IEC 14496-2 Final Proposed Draft", 14496-2, July 1999

Suleyman S. Demirsoy was born in Ankara, Turkey in 1977.

He received the BSc degree in Electrical & Electronics Engineering from Middle East Technical University, Ankara in 1998, the MSc degree in Electronic Systems from University of Westminster in 1999. He is currently working towards the PhD degree at the University of Westminster on low-power filter design techniques, multiplier blocks, novel DCT architectures.

His reseach interest is in the areas of low-power DSP, video compression and ASIC/FPGA implementation.