ISSN 2229-5518

Arithmetic Cosine Transform Architecture for Computing the Discrete Cosine Transform for Signal Processing Applications

Sangeetha Mohan, Praveen Konda

Abstract— This paper introduces the Arithmetic Cosine Transform (ACT), speedy algorithm for the evaluation of Discrete Cosine Transform (DCT) in digital signal processing. The common algorithms which used to calculate accurate value of DCT includes floating point operations and are mainly concentrated on multiplication which in turn causes the round-off error to occur. The Arithmetic Cosine Transform is proposed for the rapid calculation of DCT where the computation is focused on addition and constant multiplications which reduces the internal errors like round off and truncation. ACT results in less area consumption and low power consuming operations in case of zero-mean input signals. Calculation of ACT can be done with reasonable accuracy, reduction in area and less power using the novel architecture described for non-null mean signals as well. For the computation of eight-point DCT, ten non-uniform sampling instances are required when ACT is introduced. The implementation are done with Xilinx ISE Design suite 10.1 and coded in Verilog HDL. The two architectures are simulated and synthesized using Cadence encounter and the physical design is obtained.

Index Terms— Arithmetic Cosine Transform, Discrete Cosine Transform, Digital Signal Processing, Mean value, Mertens function, Non-null mean signal, Null mean signal

1 Introduction

THE Signal processing is a method to analyze the characteristics of a signal like storage, amplification, compression, reconstruction etc. The Discrete Cosine Transform (DCT) is a signal processing technique which converts a signal from its spatial domain in to frequency components. The existing DCT calculations include floating point operations which lead to computational errors caused by rounding off the values. The Arithmetic Cosine Transform (ACT) algorithm consists of only addition and constant multiplication which in turn reduces the computation error. The ACT can be used to calculate the exact and approximate value of the DCT for null mean and non null mean input sequences respectively. The algorithm is with less area complexity and consumes low power. The main application of DCT is in data compression. Its property states that the DCT coefficient contains most of the relevant information about the image so that it can be used in image compression applications. The DCT is mainly used in JPEG applications which are the algorithms for lossy image compression. Some applications include automatic surveillance, geospatial remote sensing, traffic cameras, satellite based imaging, automotives [1].

2 Discrete Cosine Transform

The Discrete Cosine Transform is a conventional signal processing technique used in number of applications. It's property states that the DCT coefficients contains most of the relevant information about the image so that it can be used in image compression applications.The Arithmetic Cosine Transform algorithms (ACT) are used for quick computation of DCT. The ACT consists of only additions and multiplying with constant value. This overcomes the errors associated with rounding off the values when floating point operations are come in to picture. The exact evaluation of ACT is possible if the input data are non-uniformly sampled and has zero mean. This paper unfolds two main issues (i) calculation of mean value of input signal in case of non-uniformly sampled data and (ii) proposition of efficient architectures of ACT for calculating the 8 point DCT when the input data are considered as only non-uniform samples [1].
.

2.1 Arithmetic Cosine Transform

The Arithmetic Cosine Transforms (ACT) is a speedy algorithm for evaluating the DCT of non-uniformly sampled input data. The incoming signal to the DCT are generally treated as continuous signal u(t) , that are uniformly sampled. This produces the column vector $u = \{u_n\}_{n=0}^{N-1}$ with dimension N. Its DCT is represented by the vector $U = \{U_k\}_{k}^{N}$. The corresponding non-uniform samples of the input sequence u (t) are required to compute the vector U. The sampling instants are givenby

$$s = \frac{2rN}{k} - \frac{1}{2} \tag{1}$$

where k = 1,2,….,N-1 and r = 0,1,…k – 1
Substituting for r = 0, k = 1 gives s = – 1 / 2
For r = 1, k = 2 implies s = 15 / 2
Substituting in the same way, all other values of set S is obtained.
A set S with sampling points as the elements is defined as

S = {All values of S} (2)

For an 8-point DCT,

$$s \in S = \left\{-\frac{1}{2}, \frac{25}{14}, \frac{13}{6}, \frac{27}{10}, \frac{7}{2}, \frac{57}{14}, \frac{29}{6}, \frac{59}{10}, \frac{89}{14}, \frac{15}{2}\right\} \quad (3)$$

Here the ACT algorithm is represented in two ways to compute the DCT for zero mean sequence and non-zero mean sequence. When considering zero-mean sequence, the ACT averages the A_k as

$$A_k \triangleq \frac{1}{k}\sum_{r=0}^{k-1} u_{2r\frac{N}{k}-\frac{1}{2}}, \quad k = 1,2,\ldots,N-1 \quad (4)$$

The above ACT averages can be used in the evaluation of DCT of non-uniform input samples by using the expression

$$U_k = \sqrt{\frac{N}{2}} \sum_{j=1}^{\lfloor\frac{N-1}{k}\rfloor} \mu(j).A_{kj} \quad (5)$$

Where k = 1,2……,N-1 and μ(.) is called the Mobius function. The ACT is derived by using the Mobius inversion formula.

In case of non-null mean input signal, a correction term is subtracted from the equation of U_k to calculate the DCT coefficients and it follows as

$$U_k = \sqrt{\frac{N}{2}} \sum_{j=1}^{\lfloor\frac{N-1}{k}\rfloor} \mu(j).S_{kj} - \sqrt{\frac{N}{2}}\, \bar{u}\,.\,M(\lfloor\tfrac{N-1}{k}\rfloor), \quad (6)$$

where $\bar{u}$ is considered as the arithmetic average of the input uniform samples given by

$$\bar{u} = \frac{1}{8}\, w.v_r \quad (7)$$

with w as the interpolation weight

$$M(n) = \sum_{r=1}^{n} \mu(r), \quad (8)$$

where M(n) is the Mertens function.

2.2 ACT Architectures

This paper introduces architectures for the ACT which accepts only non-uniform samples as inputs and calculate the DCT with reduced area complexity and low power consumption. All the above explained methodologies are used for the design of these architectures. There are registers are introduced at different nodes for the temporary storage which gives a fully pipelined structure to the design. This pipelined structure reduces the critical path delay with a slight increse in the latency. Architecture I corresponds to the ACT architecture for computing the DCT of null mean input signals. This architecture can be realized using (4) and (5). This architecture is done with only additions and constant multiplication with integers which reduces the truncation error and complexity. The Architecture I shown in Fig.1 corresponds to N=8 which takes 10 non-uniform samples as inputs according to the values of the set S given by (3). The applications dealing with zero mean input signal uses this architecture with the advantages of less complex computation and area. The simulation result for the Architecture I is shown in Fig.5. The second architecture is used for the calculation of DCT which has non-null mean input signals. It is desired to calculate the mean value of the incoming non-uniform samples. The Mertens correction function is included as per (6). Architecture II consists of Architecture I, mean calculation block and Mertens correction block as shown in Fig.2.

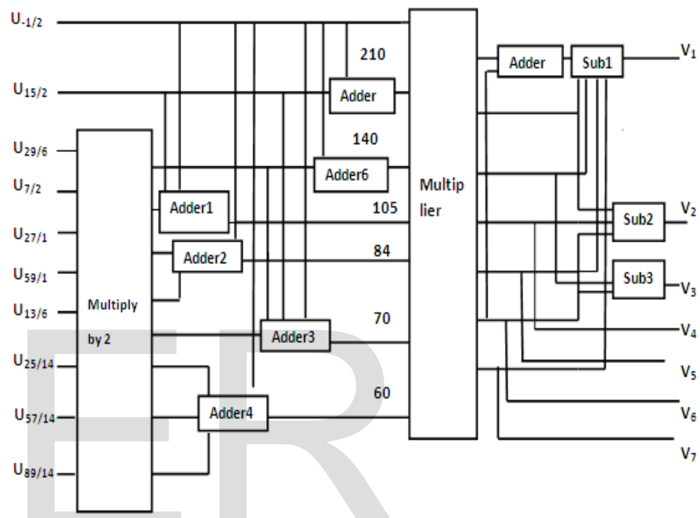

Fig. 1 Architecture I for Null mean input signals

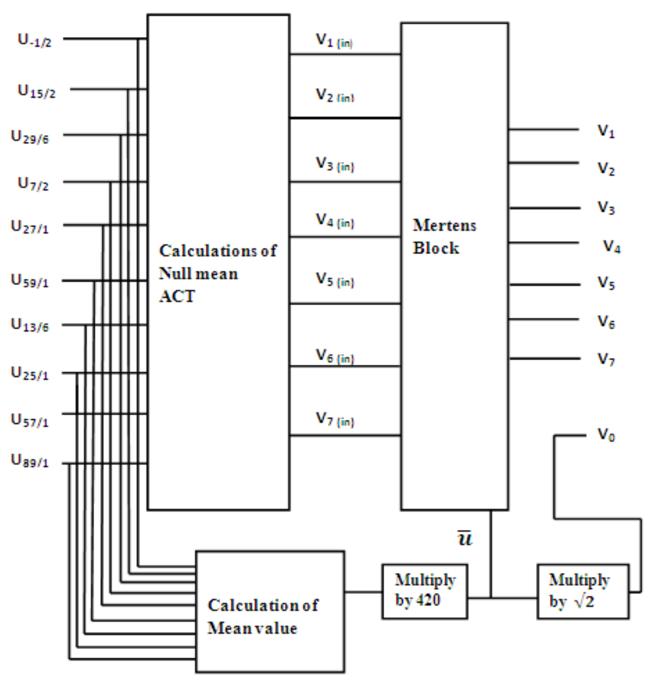

Fig. 2 Architecture II for Non-null mean input signals

2.3 Computing Arithmetic Mean

The calculation of mean value is required if the incoming signals are of non null mean type. The architecture in Fig.(3) is realized using (7). Here the input signals are scaled by the sampling instants and are given as input to the mean value calculation block. In the next step, each inputs are multiplied by the corresponding interpolation weight. The final step of mean value computation is to add all these values which will be the mean value of the incoming non null mean sequence.

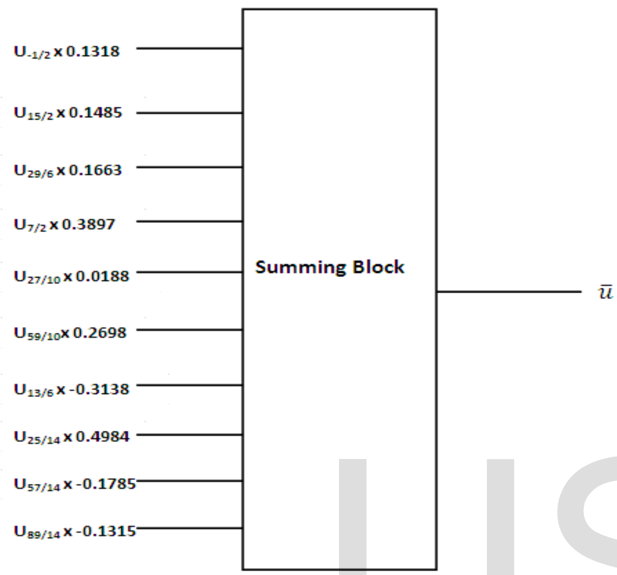

Fig. 3 Mean value calculation

2.4 Mertens Correction Factor

For non null mean input sequence, it is required to subtract a modification term in order to get the DCT coefficients. This is called the Mertens correction term, M(n). This term is the sum of the Mobius function.

- *Sangeetha Mohan is currently pursuing Master of Technology in VLSI design and Embedded System in SCEM , Mangalore India, PH+919538928185. E-mail: sangsangi@gmail.com*
- *Praveen Konda is currently working as Associate Professor in Electronics and Communication Engineering Department, SCEM, Mangalore,India, PH +919591306311. E-mail: pkonda57@gmail.com*

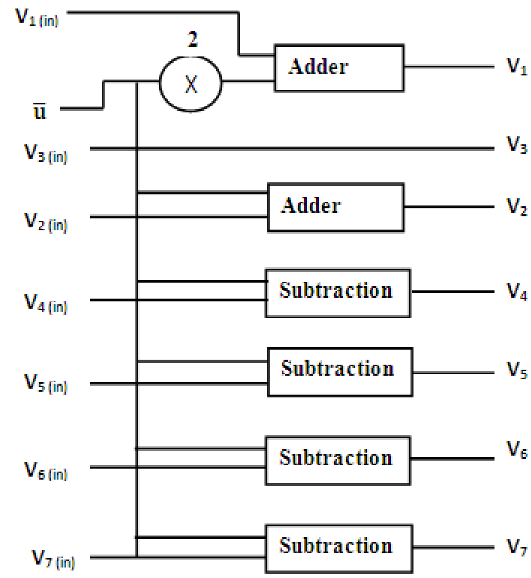

Fig. 4 Mertens correction block

3 Results

The two architectures corresponds to computing DCT for null mean and non null mean inputs are implemented in Verilog HDL using Xilinx 10.1 design suit. The simulation result for the same is shown in Fig. 5 and Fig.6 respectively. The simulation, synthesis and physical design of the two architectures are performed using Cadence Encounter. The synthesis process generates RTL schematic, time, area and power reports. The Cadence result is shown in Fig.7 and Fig.8. The physical design obtained is shown in Fig.9 and Fig.10

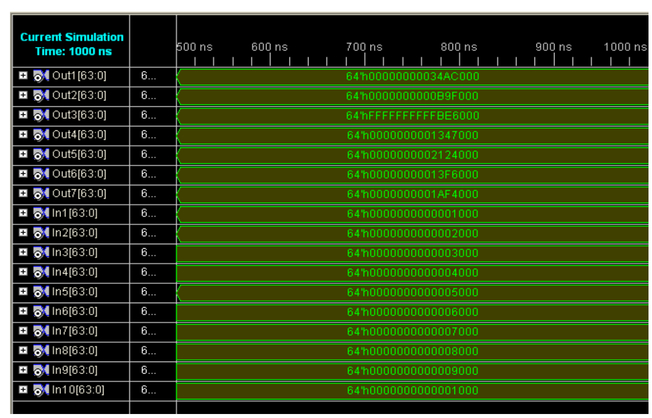

Fig. 5 Simulation result of Architecture I in Xilinx

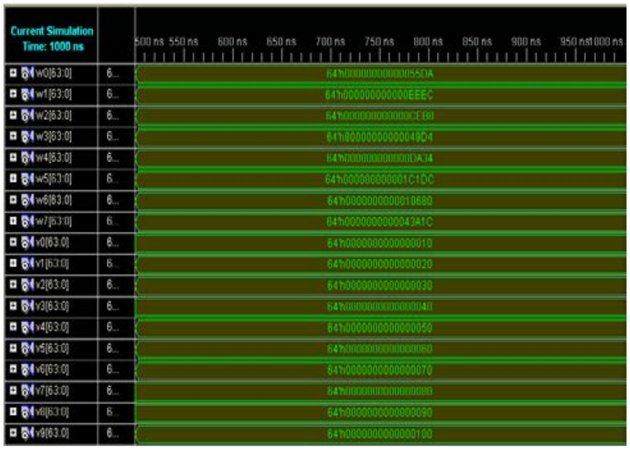

Fig. 6 Simulation result of Architecture II in Xilinx

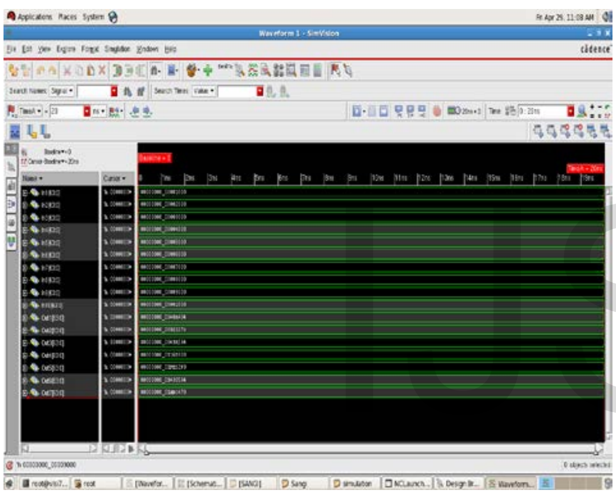

Fig. 7 Simulation result of Architecture I in Cadence

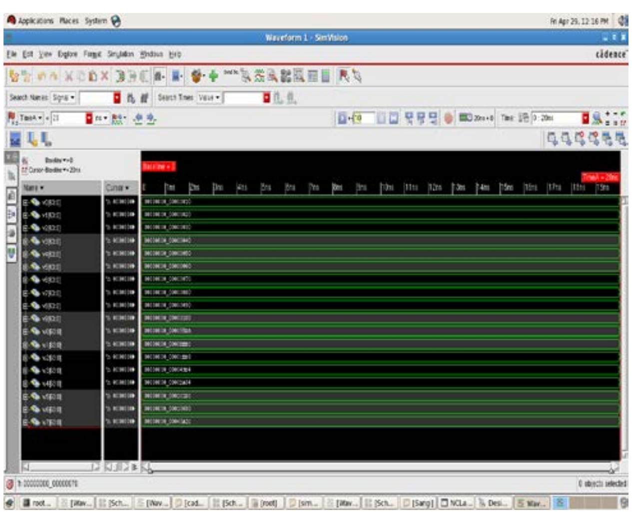

Fig. 8 Simulation result of Architecture II in Cadence

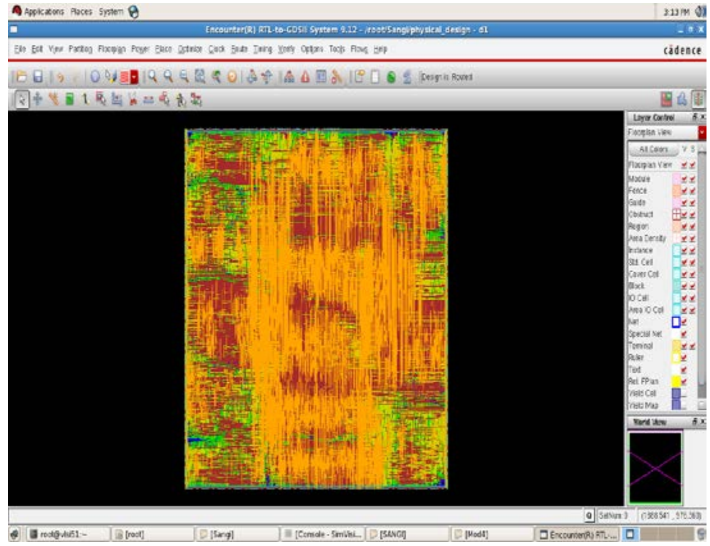

Fig. 9 Physical design of Architecture I using Cadence Encounter

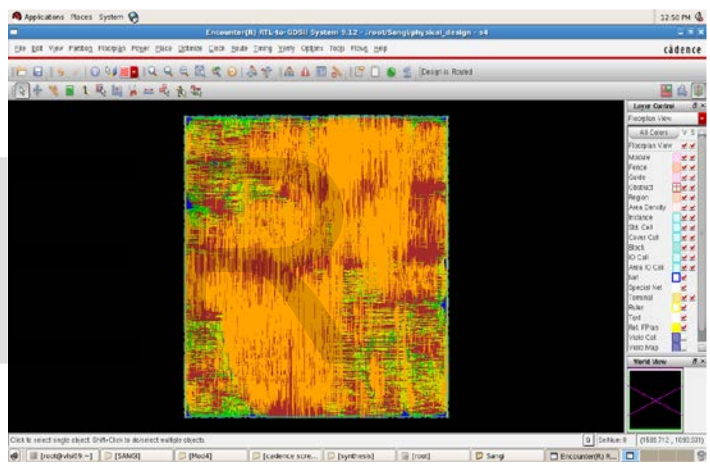

Fig. 10 Physical design of Architecture II using Cadence Encounter

4 Conclusion

The various algorithms for the evaluation of Discrete Cosine Transform are analyzed by considering their number of addition operations, number of multiplications needed, computational difficulties, complexity of area, and probability of occurrence of error and power consumption. The Arithmetic Cosine Transform is found to be a fast one for the computation of DCT. It has got reduced architectural complexity with only adders and constant integer multipliers which make the structure to be free from the truncation errors associated with the floating point operations. The two architectures are designed for null mean and non-null mean input signals which are only non-uniformly sampled.

References

[1] Nilanka Rajapaksha, Arjuna Madanayake, Renato J.Cintra, Jithra

Adikari, “VLSI computational architectures for the Arithmetic cosine transform,” *IEEE Trans. Comput.*, vol. 64, no.9, pp. 2708-2715, Sep.2015.

[2] N. Ahmed, T. Natarajan, and K. R. Rao, “Discrete cosine transform,” *IEEE Trans. Comput.*, vol. 23, no. 1, pp. 90–93, Jan. 1974.

[3] F. A. Kamangar and K. R. Rao, “Fast algorithms for the 2-D discrete cosine transform,” *IEEE Trans. Comput.*, vol. 31, no. 9, pp. 899–906, Sep. 1982.

[4] C. Chakrabarti and J. J_aJ_a, “Systolic architectures for the computation of the discrete Hartley and the discrete cosine transform based on prime factor decomposition,” *IEEE Trans. Comput.,* vol. 39, no. 11, pp. 1359–1368, Nov. 1990.

[5] R. J. Cintra and V. S. Dimitrov, “The arithmetic cosine transform: Exact and approximate algorithms,” *IEEE Trans. Signal Process.*, vol. 58, no. 6, pp. 3076–3085, Jun. 2010.