A Low-power Cordic and CSD based DCT Architecture

Yi Tao[1, a] , Wu Yang[2,b] and Yujing Li[3,c]

[1] No.24 Research Institute of CETC , China

[2] No.24 Research Institute of CETC , China

[3] No.24 Research Institute of CETC , China

[a]ty_sisc@163.com,[b] yw_sisc@163.com, [c] lyj_sisc@163.com

Keywords: Loeffler DCT Cordic CSD Algorithm

Abstract. A low power discrete cosine transform (DCT) architecture is presented. It is implemented through Loffler DCT based on the coordinate rotation digital computer (Cordic) algorithm and the canonical sign-digital (CSD) multiplier-less architecture. The synthesis results show that the proposed 8-point one-dimensional DCT architecture consumes lower power and smaller area.

Introduction

The DCT is a loss-less and reversible mathematical transformation that converts a spatial amplitude representation of data into a spatial frequency representation. One of the advantages of the DCT is its energy compaction property, that is, the signal energy is concentrated on a few components while most other components are zero or are negligibly small.

The DCT was first introduced in 1974 and since then it has been used in many applications such as filtering, speech coding, image coding (still frame, video and image storage), pattern recognition, image enhancement. The DCT is widely used in image compression applications, especially in lossy image compression. For example, the 2-D DCT is used for JPEG still image compression, MPEG moving image compression, and the H.261 and H.263 video-telephony coding schemes [1].

Loeffler based DCT algorithm

DCT is highly computational intensive, which creates prerequisites for performance bottlenecks in systems utilizing it. To overcome this problem, a number of algorithms have been proposed for more efficient computations of these transforms.

In my design I use an 8-point 1-D DCT/IDCT algorithm, proposed by Christoph Loeffler [2].This algorithm is a huge modification of the original DCT transform algorithm, which provides one of the most computationally efficient 1-D DCT /IDCT calculations. The Loeffler algorithm for calculating 8-point 1-D DCT is illustrated in Fig.1.

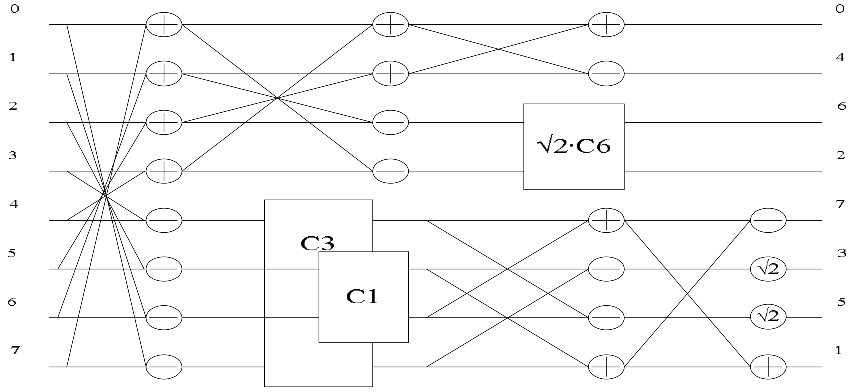
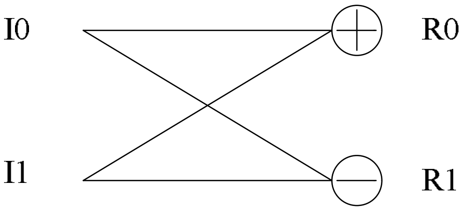


Fig. 1 The 8-point DCT Loeffler algorithm

The round block in Fig.1 signifies a multiplication by √2. The butterfly block and the equations associated to it are presented in Fig.2.

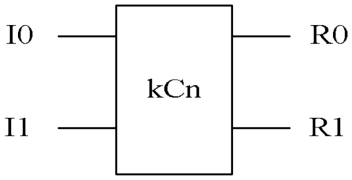


R0=I0+I1

R1=I0-I1

Fig.2 The butterfly

The rectangular block depicts a rotation, which transforms a pair of inputs [I0, I1] into outputs [O0, O1]. The symbol and associated equations are depicted in Fig.3



$$R0 = I0\times k\times\cos(\frac{n\times p}{16}) + I1\times k\times\sin(\frac{n\times p}{16})$$

$$R1 = -I0\times k\times\sin(\frac{n\times p}{16}) + I1\times k\times\cos(\frac{n\times p}{16})$$

Fig.3 The rotator and its associated equations

The implementation of the rotator depicted in Fig.4 utilizes four multipliers and two adders to shorten critical path and improve numerical accuracy. This direct implementation has been proven to be ideal for fixed point arithmetic calculation. Indeed, some other implementations of the rotator are possible, such as implementing with three multipliers and three adders. These alternative designs, however, have longer critical paths and involve initial additions, which may lead to overflows and may affect the accuracy of the calculations.

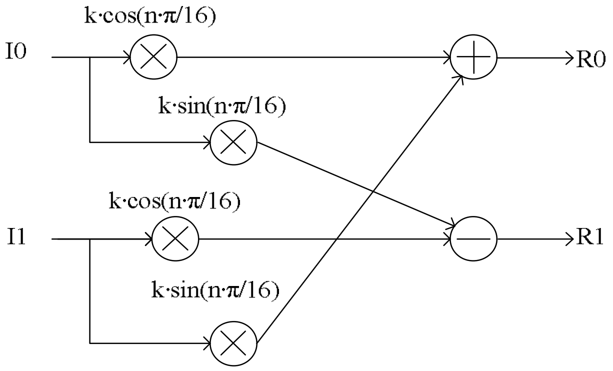


Fig.4 Implementation of the rotator for DCT

Proposed Cordic and CSD based DCT Architecture

Although the Loeffler DCT is very useful for many image/video compressions, it still needs multiplications which are slow in both software and hardware implementations. On the basis of previous work about Cordic-based signal transforms [3], C.-C.Sun propose an optimised Cordic based Loeffler DCT[4]. This implementation only requires 38 add and 16 shift operations to perform a multiplierless DCT transformation, but adding 6 multiplier in the final stage. He took the original Loeffler DCT as the starting point for optimization, because the theoretical lower bound of the number of multiplications required for the 1-D 8-point DCT had been proved to be 11.

Firstly, for the angle θ=3π/8, Sun reduce the number of rotation iterations to three and also shift all compensation steps to the final stage. Although the optimized 3π/8 rotation will decrease the quality of the results, the influences are not noticeable in video sequence streams or image compression. He have implemented a three-stage unfolded Cordic for the angle θ=3π/8 as shown in Fig.5. As illustrated, it needs six add and six shift operations to approximate the 3π/8 rotation.

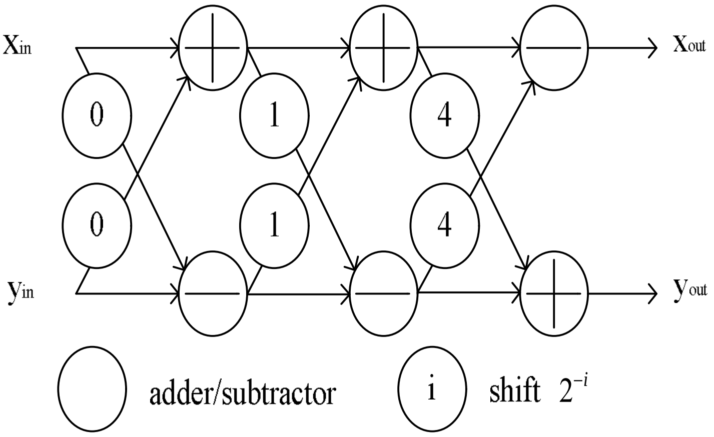


Fig.5 Unfolded flow graph of the 3π/8 angle

Secondly, for the angle θ=π/16, it can be easily observed that the needed compensation of the π/16 rotation is very close to one. Thus, Sun can ignore the compensation iterations of the π/16 rotation. Therefore it only needs two iterations in the rotation calculation as shown in Fig.6.

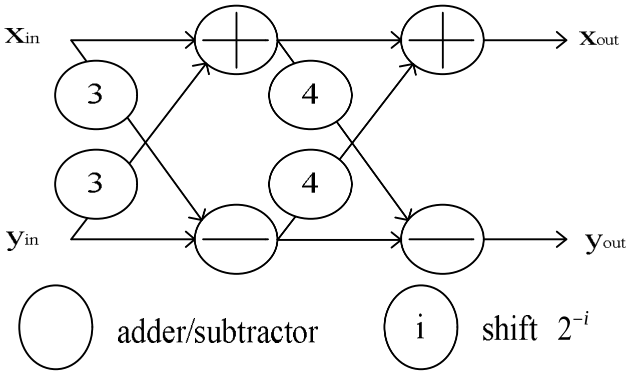


Fig.6 Unfolded flow graph of the π/16 angle

Power reduction can be achieved on the fixed multiplicand by not using 2's complement representation, but using canonical sign–digit (CSD) representation. By definition, the canonical sign–digit representation is a redundant number system that represents numbers with no adjacent non-zero digits. Every number has a unique CSD representation. It represents numbers with fewer or equal non-zero digits as the algebraic sum/subtraction of several power-of-two.

A procedure to transform a conventional binary number to CSD representation is described as below. As an example, the CSD representations of the constant operands used in DCT calculation with 12-bit precision after binary point, is shown in Eq.1

Original Format: 1010 01110 0111 CSD Format: 1010 100-10 100-1 (1)

The CSD representation can reduce the number of non-zero compare with traditional representation. Because 'canonical' means no adjacent nonzero digits, the n-bit number can be represented with less non-zero digits, which in turn reduces the carry–save adder stages compared to a general purpose array multiplier. Since fewer non-zero bits imply less computation, less switching activity and less power consumption, the CSD multiplier–less architecture is a good choice for low-power design.

I have found if I replace the 3π/8 by using the CSD multiplier-less architecture and follow the Loeffler DCT transform architecture, it can lead more lower power and smaller area than the C.-C Sun proposed [4]. This proposed Cordic and CSD based Loeffler DCT architecture is shown in Fig.7.

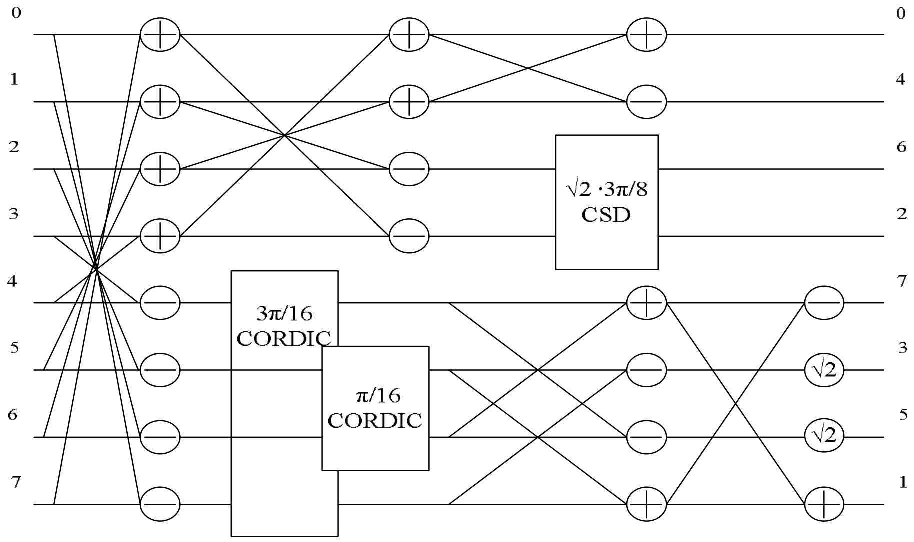


Fig.7 Proposed Cordic and CSD based Loeffler DCT

Synthesis Results

After Synthesis using Chartered 0.18μm CMOS Design Library by DesignComplier under 100Mhz, Table 1 shows the comparison result between the Cordic and CSD based Loeffler DCT with the Loeffer based DCT proposed by the C.-C. Sun[4]. The power is evaluated through PrimeTime PX.

	Cell area	Power(mW)	Leakage power(nV)
Cordic based Loeffler DCT	210188.56	21.40	379.000
Cordic and CSD based Loeffler DCT	203243.04	20.00	374.0000

Table 1 Area and power comparison result

Summary

Discrete Cosine Transform (DCT) are most widely used image compression techniques. It is difficult to make a real-time implementation of it by software method because it takes too many CPU cycles. For the hardware, it still required much more power. In this paper I propose a low-power Cordic and CSD based Loeffler DCT architecture to achieve lower power and smaller area.

References

[1] Conzalez, R.C.,Woods , R.E.: ‘Digital image processing ’(Prentice-Hall, Inc., NJ, 2001)

[2] Loeffler, C., Lightenberg, A., and Moschytz, G.S.: ‘Practical fast 1-D DCT algorithms with 11-multiplications’. Proc. ICASSP, Glasgow,UK, 1989, vol. 2, pp. 988–991

[3] Volder, J.E.: ‘The CORDIC trigonometric computing technique’, IRE Trans. Electron Comput., 1959, 8, pp. 330–334

[4] C.-C. Sun, “Low-power and high-quality Cordic-based Loeffler DCT for signal processing”, IET Circuits Devices Syst., 2007, 1, (6), pp. 453-461