CORDIC-BASED VLSI ARCHITECTURE FOR FLAT TOP WINDOWING IN REAL TIME SPECTRAL ANALYSIS

Ekta Yadav[1] ektayadav.1992.a@gmail.com[1]

Prof. Nilesh Mohota[2] nileshmohota@gmail.com[2]

[1]PG Scholar, Dept of ECE, JD College of Engineering and Management, Valni, Nagpur.

[2]Professor, Dept of ECE, JD College of Engineering and Management, Valni, Nagpur.

Abstract— one of the most important steps in spectral analysis is filtering, where window functions are generally used to design filters. In this paper, we modify the existing architecture for realizing the window functions using CORDIC processor. Firstly, we modify the conventional CORDIC algorithm to reduce its latency and area. The proposed CORDIC algorithm is completely scale-free for the range of convergence that spans the entire coordinate space. Secondly, we realize the window functions using a single CORDIC processor as against two serially connected CORDIC processors inexisting technique, thus optimizing it for area and latency. The linear CORDIC processor is replaced by a shift-add network which drastically reduces the number of pipelining stages required in the existing design. The proposed design on an average requires approximately 64% less pipeline stages and saves up to 44.2% area. Currently, the processor is designed to implement Blackman windowing architecture, which with slight modifications can be extended to other window functions as well. The details of the proposed architecture are discussed in the paper.

Keywords **—** Blackman windowing, CORDIC, VLSI Architecture

I.Introduction

In many digital signal processing applications, fast Fourier transform(FFT) is widely used for real time spectral analysis and filtering. For spectral analysis applications, well-liked windowing functions such as Hamming, Hamming and Blackman windowing methods have been used for preprocessing input signals before FFT to minimize the spectral leakage and picket fence effect [1]. In this paradigm, various algorithms and architectures for FFT have been proposed for high speed implementations with variable transform length. So it requires a windowing computation with high throughput to meet the speed of the FFT processor. But to the knowledge of the authors, there is no new hardware architecture for windowing other than existing ROM based implementation which is having the constraints of speed and flexibility. To overcome these constraints we have proposed CORDIC based architectures for windowing to meet the specifications of recently developed FFT architecture in terms of high throughput and flexibility for real time applications. In this work, we have enhanced the throughput of the proposed architecture for Blackman windowing using a fast adder known as Han-Carlson (HC) adder. This work at the beginning presents expression for Blackman windowing functions, and then, mapping that expression to the architecture for parallelism and pipelining hardware implementation. Here CORDIC units in different modes such as circular and linear mode are used primarily as basic block for trigonometric and linear multiplication functions. The critical path for this architecture is simply the adder/subtractor module in the pipelined CORDIC unit, so we have used a fast adder known as Han-Carlson adder to make it suitable for real time applications that require very high throughput rate.

Window filtering techniques are commonly employed in signal processing paradigm to limit time and frequency resolution. Various window functions are developed to suit different requirements for side-lobe minimization, dynamic range, and so forth. Commonly, many hardware efficient architectures are available for realizing FFT, but the same is not true for windowing–architectures. The conventional hardware implementation of window functions uses lookup tables which give rise to various area and time complexities with increase in word lengths. Moreover, they do not allow user-defined variations in the window length. An efficient implementation of flexible and reconfigurable window functions using CORDIC algorithm is suggested. Though they allow user-defined variations in window length, latency is a major problem. The CORDIC algorithm inherently suffers from latency issues and using two CORDIC processors in series, as is done. The overall latency of the system is hampered.

II. Literature Survey

During spectral analysis, the input signals are to be truncated to fit a finite observation window according to the length of FFT processor. This direct truncation using conventional windowing, known as rectangular window function leads to undesirable effects known as spectral leakage and picket fence effect in frequency domain. To minimize these effects during spectral analysis, researchers have proposed different kinds of windowing functions such as Hanning, Hamming and Blackman windowing functions. These windowing functions are widely adopted because of their good spectral characteristics like central peak width, 6-dB point, Highest side lobe and rate of side lobe fall off and equivalent noise bandwidth (ENBW). Among these, Blackman windowing leads to better side lobe attenuation. It is needless to present all these characteristics in detail here, however readers may refer for the same. Here only Blackman windowing has been discussed for implementation. Though ROM based implementation is already existing, which restricts flexible implementation and also restricts fitting with the advanced FFT processors in terms of variable length and speed. Basic idea of this work is to propose a flexible and fast architecture for Blackman windowing function to fit with the advanced FFT processor. Before presenting the proposed architecture in the next section, Blackman windowing function has been highlighted here briefly. A typical block diagram for real time FFT based spectral analysis system is shown in Fig.1.

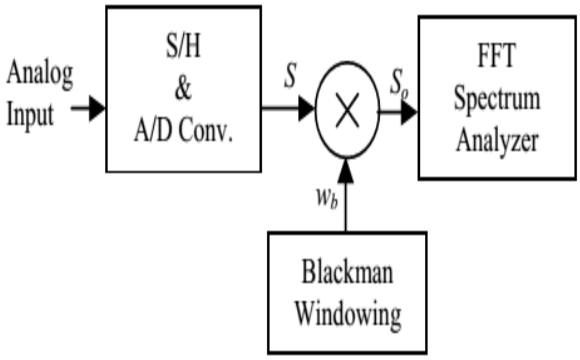

Fig.1. Spectral analysis system

The Blackman window, with the above approximation coefficients, provide attenuation of at least 60dB of side lobes[1] with only a modest increase in computation over that required by the Hanning and Hamming window due to another cosine term as in equation (2). This windowing function demands the attention for designing hardware efficient, flexible window length setting and high throughput VLSI architecture using CORDIC whose implementation is quite economic in terms of hardware. Now from equation (2), we shall have a parallel and pipelined architecture for aforesaid

windowing function, where the selection of window length (N) is user defined as per requirement for the application. Since the equation needs trigonometric computation, so the implementation using CORDIC algorithm is better choice in terms of computation and to change the value of N dynamically. But look up table or ROM method fails to achieve the same. In case of fixed N also, though existing implementation is based on look up table, it consumes more time to access the ROM and to compute multiplication and addition. Whereas CORDIC based proposed architecture gives same result with high throughput and lesser hardware compared to ROM based computation. Here multiplication and trigonometric computations are realized using linear and circular CORDIC algorithm respectively

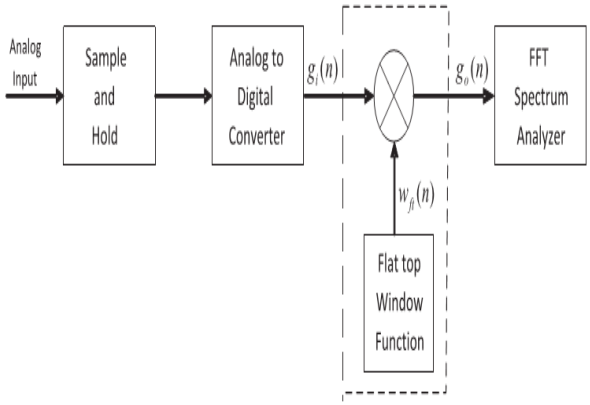

Fig.2. Block representation for spectral analysis

III.Proposed System

Flat top window architecture:

Proposed architecture of flat top window along with each individual blocks have been presented. Further, the computation errors and the implementation results of this proposed architecture are discussed in subsequent sections. In the proposed architecture of flat top window, four circular CORDIC modules and one linear CORDIC module along with angle generator block are being used with reference to Eq.(1)to generate window data. As shown in Fig. 3angle generator unit is being used to produce cosine argumentsh1; h2; h3 and h4to corresponding circular CORDIC.

Angle generator architecture

Angle sequences of h1; h2; h3 andh4 which are generated at each discrete point as follows:

$$\theta_1(n+1) = \theta_1(n) + \frac{2\pi n}{N}$$

$$\theta_2(n+1) = \theta_2(n) + \frac{4\pi n}{N}$$

$$\theta_3(n+1) = \theta_3(n) + \frac{6\pi n}{N}$$

$$\theta_4(n+1) = \theta_4(n) + \frac{8\pi n}{N}$$

Here, the value of $\theta_1(n), \theta_2(n), \theta_3(n), \theta_4(n)$are zero form n=0 and varies from 0 to 2Π in the steps of 2pn N ; 4pnN ; 6pnN ; 8pnN respectively, where n is the discrete time index and varies from 0 to N-1, where N is window length. Here a new hard wired shifter which requires no hardware as shown in Fig. 5is designed and used to generate angle increment required for every next discrete point of angle sequences. Angle generator unit is capable of producing all the discrete angle sequence corresponding to window length, N, selected by user.

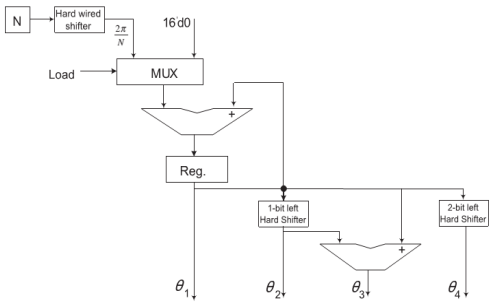

Fig.3 .Angle generator architecture.

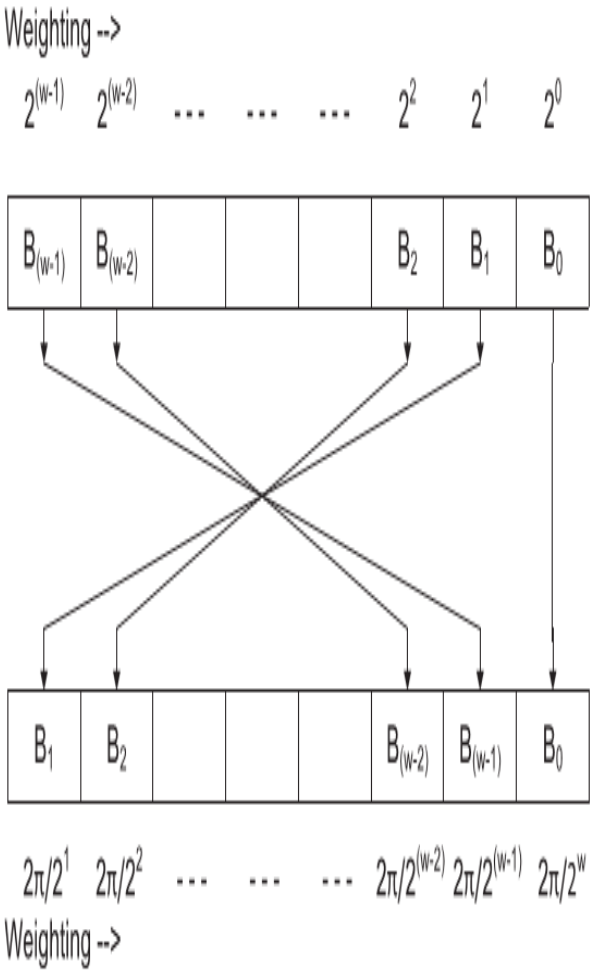

Fig. 4.Hard wired shifter

Scale factor compensated CORDIC architecture

From the previous section, it has been assumed that the scale factor (k) is ignored which lead to an amplified output by a factor equal toA=1/k . But in real time analysis and synthesis of signals, the amplified value needs to be compensated, by the pre or post-multiplication of this cale factor with input samples which introduces more latency in the data path. To achieve the scale factor compensation without additional latency, a parallel scale factor compensated CORDIC architecture[24]is adopted here. In this architecture, two CORDIC modules are used in parallel and their initial input angles are chosen

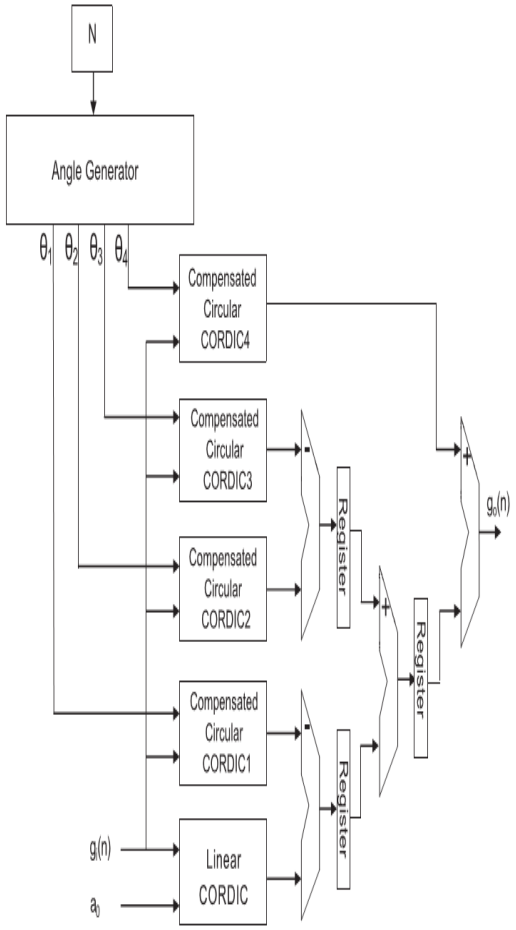

Fig 5. .Flat top window function architecture.

Window functions are being used in spectrum analysis to reduce the undesirable effects known to be spectral leakage and scalloping loss [4,9]present in the spectrum of a signal computed using FFT. These effects arise due to discontinuities present at the boundaries of signal which is truncated to match the length (points) of FFT. Here window data using window functions are smoothly brought to zero at the boundaries of observation length such that periodic extension of the data is continuous in time domain. Window functions are characterized by different parameters like main lobe width,

maximum side lobe level, side lobe roll off rate, coherent gain and equivalent noise bandwidth. Many popular windows like Hanning, Hamming, Blackman, Blackman Harris and Flat top windows have been proposed over the period of time with advantages to each other based on aforesaid parameters. Advantages of flat top window are highlighted and its application has been discussed as follows. During spectral analysis, input signals are preprocessed before FFT by multiplying window function as depicted in Fig. 1and presented mathematically as follows:

$$g_o(n) = w_{ft}(n) \cdot g_i(n)$$

Where n is the discrete time index, gi ðnÞ is the input signal sequence, wftðnÞis flat top window function and g0 ðnÞis the windowed output data sequence. HerewftðnÞis defined as

$$w_{ft}(n) = a_0 - a_1 \cos\left(\frac{2\pi n}{N}\right) + a_2 \cos$$

$$+ a_4 \cos\left(\frac{8\pi n}{N}\right)$$

CORDIC (Co-ordinate Rotation Digital Computer) algorithm is being used for real time computation of linear, trigonometrically and transcendental functions which are essential blocks in the field of signal processing, image processing and communication system. The CORDIC algorithm is a vector rotation algorithm which is presented as follows:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \cos\theta \begin{bmatrix} 1 & -\tan\theta \\ \tan\theta & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

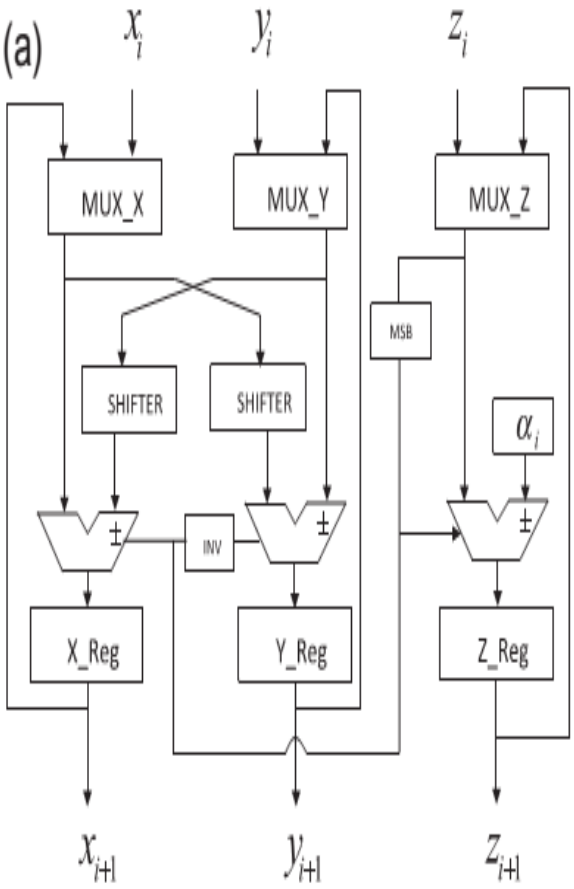

Fig 6.CORDIC architectures (a) circular CORDIC

These iterative equations are used for different CORDIC, i.e. circular, linear and transcendental CORDIC respectively. It is depicted from Eq. (1) that the flat top windowing requires trigonometrically and linear multiplications which can be achieved using circular and linear CORDIC. From Eq.(8), it is obvious that these iterative equations can be implemented using shifter and adder/subtracter blocks and taking n equal to one or zero to realize circular CORDIC or linear CORDIC respectively. The iterative CORDIC architectures for circular and linear CORDIC have been presented in below Fig.

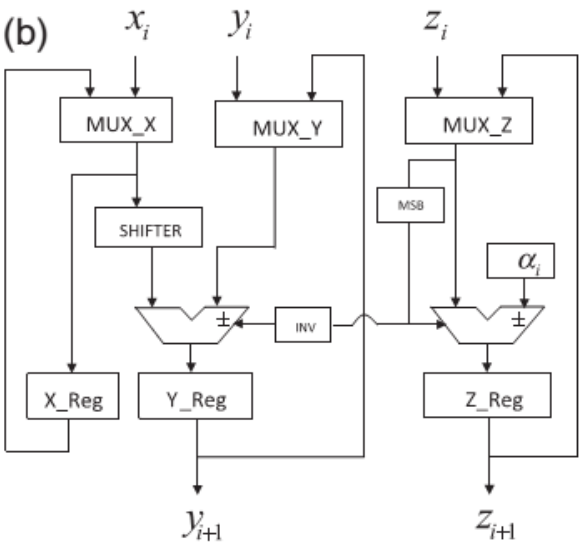

Fig.6.CORDIC architectures (b) linear CORDIC.

IV.Extension Work

Blackman window architecture

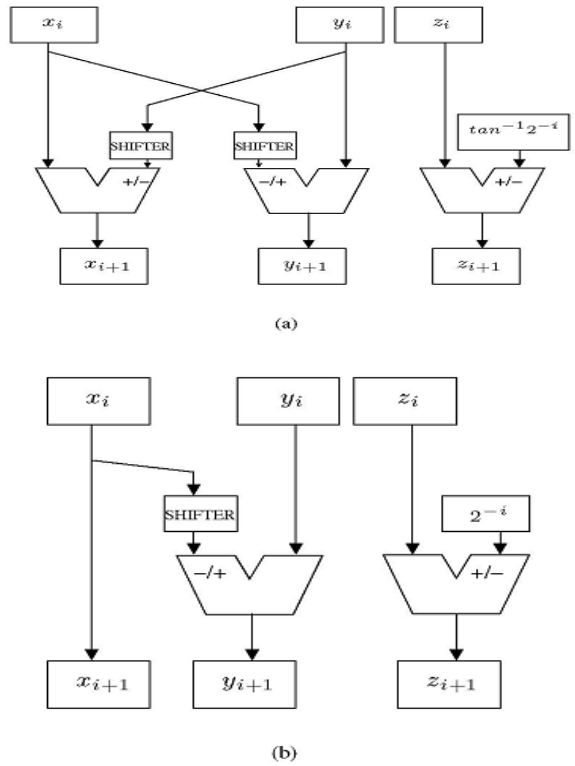

Fig. 7. Pipelined CORDIC Architecture (a) Circular (b) linear

A high throughput VLSI architecture for Blackman windowing. Since most of the implementation of windowing functions for real time applications, are based on either ROM or DSP processor. Here the proposed architecture is designed using major blocks like CORDIC (CO-ordinate Rotation DIgital Computer) and Han-Carlson adder. This architecture is flexible in terms of window length. So that a single chip can be used for those applications, where variable length is required.

The architecture is shown in Fig.7 for VLSI implementation of Blackman windowing function. Major blocks of proposed architecture are described subsequently. Two linear CORDIC blocks are used for multiplying input samples with constant coefficients (b0and b2), however the multiplication of constant coefficient (b1=0.5) with input samples is done with only hard shifter (1-bit right) and passes through FIFO_1 for synchronization with other parallel paths and similarly FIFO_2 is also used for synchronization. Circular CORDIC has been used to compute cosine functions given in equation (2) and multiplication of intermediate values (i.e. values from lower linear CORDIC and FIFO_1 as shown in Fig.7). CORDIC blocks used in our proposed architecture are purely pipelined, where add/sub circuit is the critical path. Here length of FIFOs is equal to the number of stages of pipelined CORDIC minus one, e.g, for 16-bit precision CORDIC, the number of stages are sixteen and thus FIFO length to be fifteen.

V. CONCLUSION

This proposed architecture is a parallel and pipelined structure for Blackman windowing, providing a high throughput, area efficient and having flexibility on changing windowing

(observation) length for user specific applications, as well modifying the register N online with either software protocol or hardware control signal. This architecture can be embedded with advanced FFT processor architectures for real time spectral analysis. This architecture canbe used in real time filtering applications, where Blackman windowing filter is required. The technology independent netlist of this proposed architecture can be implemented in advanced FPGA chips.

REFERENCES

[1] Richard J. Higgins, "Digital Signal Processing in VLSI", Prentice Hall Englewood Cliffs, 1990.

[2] A.Banerjee, A. S. Dhar and S. Banerjee, "FPGA realization of a CORDIC Based FFT Processor for Biomedical Signal Processing", Microprocessors and Micro Systems, vol 25/3, pp131-142, May 2001.

[3] Kai Zhong, Guangxi Zhu, and Hui He, "single-chip, ultra high speed FFT architecture", Proceedings .5th ASIC, 2003, International Conference, Vol 2, pp752-756 Oct.2003.

[4] A. Samad, A. Ragoub, M. Othman and Z.A.M.Shariff, "Implementation of high speed Fast Fourier Transform VLSI chip", Microelectronics Journal, Vol 29,Issue 11,pp881-887 Nov 1998.

[5] Chung-Ping Hung, Sau-Gee Chen,and Kun-Lung Chen, "Design of an efecientvariable length FFT processor", Proc, Circuits and Systems, ISCAS '04. Vol 2, pp 833-836, 23-26 May 2004.

[6] Shousheng HE, Torkelson M., "Design and implementation of a 1024-point pipeline FFT processor", Proc, IEEE- Custom Integrated Circuits Conference, pp131-134, May 1998.

[7] K. C. Ray and A. S. Dhar, "ASIC Architecture for implementing Blackman windowing for real time spectral analysis", Proc, of Intl, Conf, on Signal Processing, Communication and Networking(IEEE-ICSCN 07), pp.388- 391, 22-24th February-2007.

[8] K. C. Ray and A. S. Dhar, "Unified CORDIC based VLSI architecture for implementing windowing functions for real time spectral analysis", IEE Proc.-Circuits Devices Syst., Vol. 53, No. 6, pp539-544, December 2006.

[9] D. Harris, "A Taxonomy of Parallel prefix Networks", Proc, IEEE ACSSC-2003, Vol.2, pp.2213-2217, Nov.2003.

[10] S.Y.Kung, H.J. Whitehouse and T. Kailath, "VLSI and Modern Signal Processing", Prentice Hall Englewood Cliffs, 1985.

[11] S. K. Padala and K M M Prabhu, "Pipelined CORDIC Processor for generating Gaussian Random Numbers", Signal Processing, Vol 72, No.3, pp177-181, Feb 1999.

[12] J. E. Volder, "The CORDIC Trigonometric Computing Technique", IRE Computers, Vol.8, pp 330-334 Sep 1959.

[13] J. S. Walther, "A unified algorithm for elementary functions", Proc, Spring joint comput, conf, pp 379- 385.1971.

[14] G.L. Haviland and A.A. Tuszynski, " A CORDIC Arithmetic Processor Chip", IEEE Trans. On Comput, Vol. C-29, No.2, Feb1980.

[15] Behrooz Parhami, "Computer Arithmetic algorithms and hardware designs", Oxford University Press,New York, 2000.

[16] A. Gyyot, B Hochet and J. M. Muller, "A way to build efficient carry skip Adders", IEEE Trans.Comput.,Vol.C- 36, no.10, pp1144-1151,Oct 1987.

[17] E. Antelo, J. Villalba, J. D. Bruguera, and E. Zapata, "High performance rotation Architectures based on the Radix-4 CORDIC Algorithm", IEEE Trans Comput, Vol.46, No.8, pp855-870, Aug 1997.

[18] J. Duprat and J.M. Muller, "The CORDIC algorithm: new results for fast VLSI implementation", IEEE Trans. Comput. Vol.42, No.2, pp.168-178, Feb 1993.