Grenze International Journal of Engineering and Technology, July 2018

Interveled Double Rotation Cordic architecture for improvement of Performance

Suganda Pendem[1] and Rajshekar B.Shettar[2]
[1]Visvesvaraya Technological University, Belagavi, Karnataka, India, [2] KLE Technological University, Hubballi, Karnataka, India
Email: suganda_p@yahoo.co.in, Raj@bvb.edu

***Abstract*—Very large scale integration made a rapid advance in the research area of DSP in terms of high speed VLSI architecture for real time applications. Numerically intensive calculations are required for real scientific computations. In these calculations round off errors and catastrophic cancellation errors occur, that leads to inaccurate results. In order to control the round off errors and catastrophic cancellation errors interval arithmetic techniques are used which do not use fixed values but works over an interval of values to ensure that the exact solution is within an interval. Thus the efficient tool for monitoring errors is the interval arithmetic method. We propose interval arithmetic CORDIC(Coordinate Rotation Digital Computer) architectures to efficiently support and specify the precision to perform interval sine and cosine functions and to guarantee the correct choice of the bounds of the result with a minimal error for the use of DSP (Digital signal processing) applications. This paper presents interval arithmetic Cordic algorithms for computational accuracy of how it works and its implementation using MATLAB toolbox INTLAB. The results show that the proposed algorithm provides higher computational accuracy and performance.**

***Index Terms*—** DSP, Interval arithmetic, Cordic, MATLAB and INTLAB.

I. INTRODUCTION

The most reliable alternative to the conventional floating point arithmetic is the interval arithmetic. The approach to bound rounding errors in mathematical computation is interval arithmetic. It is a very powerful technique with numerous applications in mathematics, computer science and engineering [16][17]. It corresponds to two values which are represented by lower and upper endpoints of an interval to guarantee that the true value lies on this interval [6]. The accuracy of the result will be given by the width of the interval. Inaccurate results cannot be efficiently solved by floating point arithmetic computations [4] but these can be solved by interval arithmetic which provides a lower and upper bound for each result [5]. It also allows the interval hardware to take advantage of floating-point hardware and VLSI technology. The development of the architecture for interval arithmetic double Cordic algorithm [7] gives an improvement in computational accuracy and an improvement in performance. Numerically intensive calculations are required for real scientific computations, round off errors and catastrophic cancellation occur leads to inaccurate results [13][14][15]. In order to control, interval arithmetic techniques are used which do not use fixed values but works over an interval of values to ensure that the exact solution is within an interval. Thus the efficient

Grenze ID: 01.GIJET.4.2.512

tool for monitoring errors such as input data uncertainties, rounding errors and approximation errors is the interval arithmetic [8]. It is used in digital signal processing, digital image processing, communication, robotics and graphics [3] applications. The Cordic is an approach which is widely used for the evaluation of elementary functions when the primary constraint is silicon area [1][2]. The implementation offers an opportunity to calculate all the functions in a simple way with less hardware. It is easy to implement [10] because it performs only shift and add operations and it is resource friendly. The iterations can be implemented in firmware or even in software using ALU and general purpose registers in microprocessors. It is very well suited for VLSI implementation because of its simplicity in its operations. In double Cordic [11] method the computations are accelerated to converge by duplicating the micro rotation angles to be 2θ. The elementary functions are performed using rotation and circular mode [12]. The increasing performance [9] and reducing latency is a challenge for engineers. Interval arithmetic Double rotation cordic is a suitable method for improving the performance and reducing latency

II. CORDIC ALGORITHM

Real time calculation of trigonometric functions by the use of iterative vector rotations is carried out by the special purpose computer Cordic [1]. The algorithm can be derived from the rotation transform.

$$x^{|} = x.cos\phi - y.sin\phi$$
$$y^{|} = y.cos\phi + x.sin\phi$$

Rearrangement the terms, this can be given as:

$$x^{|} = cos\phi\ [x - y.tan\phi]$$
$$y^{|} = cos\phi\ [y + x.tan\phi]$$

It is complex to implement the above equations due to the presence of the trigonometric functions. However, the values such that $tan\phi = \pm 2^{-i}$, are restricted to the rotation angles, the multiplication by the tangent can be simplified as it uses simple shift operations. Thus performing a series of rotations iteratively arbitrary angles can be obtained. Mathematically, it is given as shown [15].

$$x_{i+1}=k_i[x_i + d_i.2^{-i}] \qquad (1)$$

$$y_{i+1}=k_i[y_i+d_i2^{-i}] \qquad (2)$$

Where $k_i = Cos(tan^{-1} 2^{-i})=1/\sqrt{(1+2^{-2i})}$

$d_i = \pm 1$

If the angle to be obtained is greater than the current iterative angle the value of d_{i} is +1 and is -1 if the current iterative angle is greater. The value K_i =0.607253 is taken to be a constant [12] when the number of iterations are more. During computing the initial value of x-coordinate is chosen to be at 1 and an initial value of y-coordinate as 0. At the beginning of the first iteration, by an angle of 45° the vector rotates which gives the first iteration result. If the angle obtained is greater than the angle (β), the next rotation takes place in the reverse direction, else in the same direction. Finally, after the number of specified rotations, the value of cos(β) is given by the x-coordinate while the value of sin(β) is given by the y-coordinate.

III. INTERVAL ARITHMETIC

The lower bound and upper bound interval end points of an interval X are denoted as xl and xu. A closed interval X= [xl , xu] consists of set of real numbers between and including the two endpoints xl and xu (i.e., X={x: xl<x< xu}. Then outward rounding is used for computing the interval endpoints. The lower endpoint is rounded towards negative infinity (∇) for outward rounding, and positive infinity (∆) for rounding upper endpoint. The resulting interval encloses the true result from outward rounding

IV. INTERVAL ARITHMETIC CORDIC ALGORITHM

A rectangle is pointed by a set of vectors which corresponds to initial interval vector. The vectors point to a new rectangle, if a rotation is performed which is also rotated with respect to the axis. As the real rectangle cannot be represented, so the new set of vectors is enclosed using bigger rectangle. To rotate the initial set of vectors, the vectors that point to the corners of the rectangle have to be rotated, since the rotation operation is a linear. On the final solution, to guarantee correct bounds of the errors involved in the computation have to be taken into account. Figure 1 shows the proposed Interval arithmetic Cordic algorithm. The equations carried out by the architecture are.

$$x_{i+1} = x_i - \sigma i.2^{-i}.y_i \quad (3)$$

$$y_{i+1} = y_i + \sigma i.2^{-i}.x_i \quad (4)$$

$$z_{i+1} = z_i - \sigma i.\tan^{-1}2^{-i} \quad (5)$$

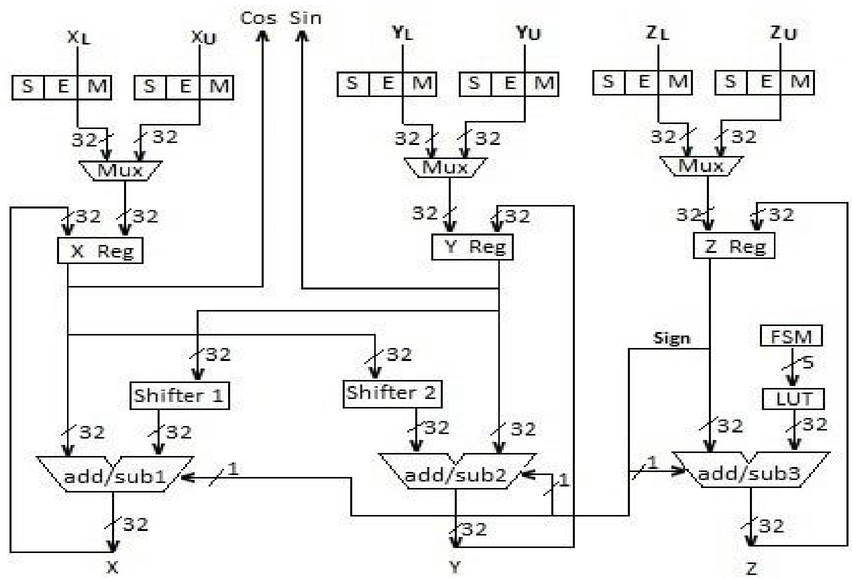

Figure 1. Interval Arithmetic CORDIC Algorithm

V. INTERVAL ARITHMETIC DOUBLE CORDIC ALGORITHM

The challenge for engineers and scientists is the increasing computational accuracy of Cordic. It is based on the rotation of vector on the plane. In rotation mode the double rotation Cordic method is accelerated by duplicating rotation angle θ to be 2θ, the degree of parameter is equal to 2.

$$\begin{pmatrix} x_{out} \\ y_{out} \end{pmatrix} = \cos^2\phi \begin{pmatrix} 1 & -\tan\phi \\ \tan\phi & 1 \end{pmatrix}^2 \begin{pmatrix} x_{in} \\ y_{in} \end{pmatrix}$$

The decomposition of the production of n elementary rotation is shown with $\phi i = \tan^{-1(2-i-1).}$

$$\prod_{i=1}^{n} R[\partial_i] = \prod_{i=1}^{n} 1/(1+2^{-2i-2}) \begin{bmatrix} 1 & -\partial_i.2^{-i-1} \\ \partial_i.2^{-i-1} & 1 \end{bmatrix}^2$$

The main characteristic feature of double rotation method is the high computational accuracy with the small number of iterations. The constant scaling factor is formulated which is approximately equal to 0.9219. The maximum and minimum rotation angle is in range of [-0.9885, +0.9885]. The micro-extension rotation for the double Cordic rotation is given by

$$x_{i+1} = x_i - \partial_i 2^{-i}.y_i - 2^{-2i-2}.x_i$$

$$y_{i+1} = y_i + \partial i.2^{-i}.x_i - 2^{-2i-2}.y_i$$

$$z_{i+1} = z_i - \partial i\ .2.\ \partial i$$

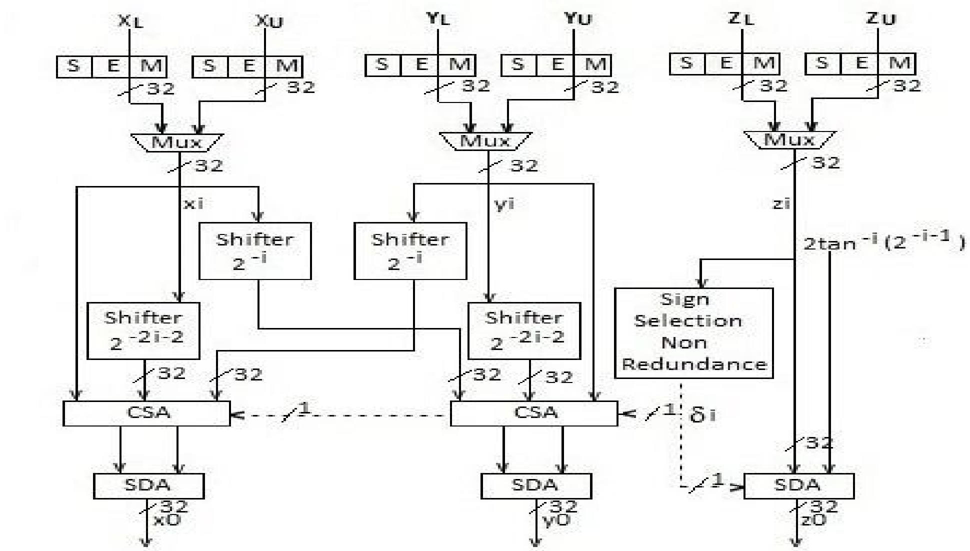

Figure 2: Interval Arithmetic double CORDIC Algorithm

VI. RESULTS

Performance analysis :The Interval arithmetic CORDIC architecture and its double rotation architecture is implemented in the working platform of MATLAB (versionR2017a) Simulink toolbox INTLAB. INTLAB is a MATLAB Toolbox for validating the algorithms. Relative error (RE) is used as a measure of precision. It is the ratio of the absolute error of a measurement to the measurement being taken. RE is expressed as a percentage and has no units. The Relative Error analysis is found using the expression.

Relative Error=Actual value- Simulation / Simulation*100 %

TABLE I. RELATIVE ERROR FOR INTERVAL ARITHMETIC CORDIC

Function	Degrees	Simulation Result	Actual result	Relative error
Sin	45	0.70722605732570	0.70710678118647	0.000168
Cos	45	0.70698830471054	0.70710678118647	0.000167

TABLE II. RELATIVE ERROR FOR INTERVAL ARITHMETIC DOUBLE ROTATION CORDIC

Function	Degrees	Simulation Result	Actual result	Relative error
Sin	45	0.707102883360547	0.70710678118647	0.000005
cos	45	0.707096759932052	0.70710678118647	0.000014

From table 1 and table 2 it can be seen that the error reduction is more than 90% for sin and cosine functions for the Interval Arithmetic Double rotation CORDIC than the Interval Arithmetic CORDIC. The relative error achieved is very small, this indicates that the smallest relative error gives the better performance. And can be concluded that the Interval Arithmetic Double rotation CORDIC has better performance.

The Time and Iteration parameters :

TABLE III. THE TIME AND ITERATIONS PARAMETERS.

Parameters	Interval Arithmetic Cordic	Interval Arithmetic double Cordic
Max iterations	12	07
Execution time	3.236 Sec	2.595 Sec
RE Sin 45	0.000168	0.000005
RE Cos 45	0.000167	0.000014

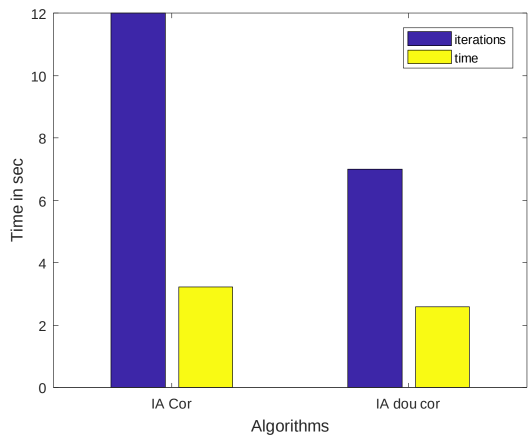

Figure 3: Bar-Graph: Time-iterations of individual algorithm

Table 3 shows the iterations, time and the RE parameters. The maximum iterations taken by IA double Cordic is only 07 where as IA Cordic took 12 iterations and the execution time taken by IA double Cordic is lesser ie 2.595 sec that of 3.236 sec for IA Cordic .The relative error (RE) is very much lesser for the IA double Cordic which indicates better performance. Figure 3 shows the bar graph of time-iterations of individual algorithms, better performance is achieved with the smaller number of iterations and thus avoiding longer computation time with increase in precision.

VII. CONCLUSION

In this paper, interval arithmetic Cordic algorithms with reduced time and error are proposed. The CORDIC algorithm is a powerful and widely used tool for digital signal processing applications. An approach based on interval arithmetic double cordic algorithm has been presented to improve the performance and the accuracy of the result with the reduction in the number of iterations. The design shows that better performance is achieved with the smaller number of iterations and thus avoiding longer computation time.

REFERENCES

[1] J.E Volder,"The CORDIC trigonometric computing technique,"IRE Trans Electronic Computing Volume 8, pp 330-334, 1959

[2] J.Walther,"A unifed algorithm for elementary functions,Spring Joint Computer Con pp 379

[3] P. Meher, J. Valls, J. Tso-Bing, K. Sridharan,and K.Maharatna, "50 Years of CORDIC-Alg orithms,architectures,and Applications,"IEEE Transactions,Circuits and Systems,vol56 pp. 93-97,2009.

[4] D.Goldberg,"What Every Computer Scientist should know about Floating-point arithmetic" ACM Computing Surveys, vol 23, pp. 5-48, and 1991.

[5] R. E. Moore ,"Interval Analysis". Prentice Hall, 1966.

[6] M. J. Schulte and E. E. Swartzlander, Jr, "Hardware design and arithmetic algorithms for a variable-precision,interval arithmetic coprocessor," In Proc. 12th Symposium on Computer Arithmetic, pages 222-229, 1995.

[7] D. M. Chiarulli, W. G. Rudd, and D. A. Buell, "A dynamically recongurable processor For integer arithmetic," In Proc. th Symposium on Computer Arithmetic, pages 309-318, 1985.

[8] M.J. Schulte A Variable Precision, Interval Arithmetic Processor,is, University of Texas at Austin, 1996

[9] E. Antelo, J.Villalaba, D. Bruguera, and E. Zapata, "High performance rotation architecture based on the radix CORDIC algorithm,".IEEE Trans. Comput., vol. 46, pp. 855-870, 1997.

[10] D. Cochrab, “Algorithms and accuracy in the HP-35," HewlettPackard Journal, pp. 1{11,- June 1972.

[11] W. Han, Z.Yousi, and L.Xiaokang,“A Parallel Double-Step CORDIC Algorithm for Digit-al Digital Down Converter," in Communication Networks and Services Research Conference 2009.

[12] J. Muller, Elementary Functions: Algorithms and Implementation. Cambridge, MA: Birkhauser, 1997.

[13] Bohlender G, What Do We Need Beyond IEEE Arithmetic?", Computer Arithmetic and Self-Validating Numerical Methods, ed., pp. 1-32, Boston Academic Press, 1990.

[14] Schulte M. J, Akkas A, Zelov V, and Burley J.C, “The Interval Enhanced GNU Fortran Compiler”, Reliable Computing, vol.5, no. 3, pp. 311-322, Aug. 1999.

[15] M. D. Ercegovac and T. Lang, Digital Arithmetic. Morgan Kaufmann, 2003.

[16] Rump, S., M.: INTLAB - INTerval LABoratory, in Csendes, T. (Ed.), Developments in Reliable Computing, KluwerAcademic Publisher, Dordrecht, pp. 77-104,1998.

[17] Rump, S., M.: Fast and Parallel Interval Arithmetic, Bit, Vol. 39, No. 3, pp.534-554, Sept. 1999.