Review paper on Design and Implementation of Fast Hybrid Reconfigurable CORDIC

[1]Shashank Pramod Zade, [2]Prof. Tushar Upalanchiwar

[1]M.Tech. Final Year, [2]Assistant Professor
Electronics & Comunication Engineering Department.
TGPCET Nagpur India.

***Abstract*—This short introduces the key idea, outline methodology, furthermore, execution of reconfigurable facilitate revolution computerized PC (CORDIC) structures that can be designed to work either for round or for hyperbolic directions in revolution also as vectoring-modes. It can, subsequently, be utilized to play out all the elements of both round and hyperbolic CORDIC. We propose three reconfigurable CORDIC outlines: 1) a reconfigurable pivot mode CORDIC that works either for round or for hyperbolic direction; 2) a reconfigurable vectoring-mode CORDIC for round and hyperbolic directions; and 3) a summed up reconfigurable CORDIC that can work in any of the modes for both round and hyperbolic directions. 4) This paper presents Verilog Hardware Description Language (HDL) execution of CORDIC mixture design and contrasts it and parallel design utilizing Xilinx ISE Webpack and Synopsys Plan Vision. 5) Relative investigation has shown that mixture engineering is quick when contrasted with parallel design yet at the cost of energy utilization and exactness. The reconfigurable CORDIC can play out the calculation of different trigonometric and exponential capacities, logarithms, square-root, and so on of roundabout and hyperbolic CORDIC utilizing either revolution mode or vectoring-mode CORDIC in one single circuit. It can be utilized as a part of advanced synchronizers, illustrations processors, logical mini-computers, et cetera. It offers significant sparing of range many-sided quality over the customary plan for reconfigurable applications.** (Abstract)

***Indexterms*— Circular trigonometry, coordinate rotation digital computer (CORDIC), hyperbolic trigonometry, reconfigurable CORDIC.**

I. Introduction

CORDIC remains for COordinate Rotation DIgital Computer. It was purposed by J.E. Volder in 1959 with the point of acknowledging distinctive processing assignments, for example, the figuring of trigonometric, hyperbolic and logarithmic capacities, genuine and complex duplications, division, square-root, arrangement of straight frameworks, eigenvalue estimation, solitary esteem decay, QR factorization and numerous others utilizing fundamental move and include iterative operation which disposes of the need of multipliers and makes equipment usage of CORDIC calculation straightforward and simple. Because of effortlessness of its design, CORDIC calculation has been a territory of broad review in different range of utilization like immediate and backwards kinematics calculation for robot control, planar and three-dimensional vector revolution for representation and activity, satellite correspondence and some more. Because of inalienable successive calculation of CORDIC calculation, the calculation procedure is moderate, and subsequently, limits its execution and territory of use. Diverse designs have been acquainted with serve this issue in. Cross breed CORDIC design is one of the engineering that was proposed in. It depends on part of the underlying point of pivot to give two rudimentary edge set to quicken the calculation procedure and serve the inactivity issue yet with included drawback of low exactness and high power dispersal .

The CORDIC calculation includes a basic move add iterative methodology to play out a few figuring assignments by working in either turn mode or vectoring-mode taking after any one among straight, hyperbolic, and round directions . Applications, for example, solitary esteem deterioration, eigenvalue estimations, QR disintegration, stage and recurrence estimations, synchronization in advanced beneficiaries, 3-D representation processor, and interpolators require the CORDIC to work in both pivot and vectoring-modes. The 3-D structures, for example, hyperboloids, paraboloids, and ellipsoids require the CORDIC to be worked in both roundabout and hyperbolic directions. The equipment usage of these applications requires more than one CORDIC processor working in various modes and distinctive directions. A reconfigurable CORDIC, which can work in pivot and vectoring-modes, for both roundabout and hyperbolic directions can supplant numerous CORDIC processors, and would be very valuable for such applications. A reconfigurable CORDIC can be used for an assortment of uses in correspondence frameworks, flag handling, 3-D design, mechanical autonomy separated from general logical counts, and waveform eras.

Over the most recent five decades, a few calculations have been proposed for territory delay-effective and control productive execution of CORDIC calculations, either for round direction or for hyperbolic direction . Yet, we don't locate any efficient review on plan and usage of reconfigurable CORDIC in the current writing. An essential plan of reconfigurable CORDIC in view of a brought together CORDIC calculation has been proposed as of late. The reconfigurable outline of is found to include high reconfiguration overhead and results in low equipment usage effectiveness. Along these lines, in this short, we introduce an approach for the plan of reconfigurable CORDIC to be utilized for revolution mode and vectoring-mode in round and hyperbolic directions.

II. LITERTRATUER SURVEY

To plan a reconfigurable CORDIC engineering with least reconfiguration overhead, we have to expand the sharing of regular equipment circuit in various designs. In this manner, to investigate the likelihood of reconfigurable CORDIC, we look at, here, the shared traits in three principle issues of CORDIC execution, to be specific: 1) the arrange turn lattice; 2) choice of basic points; and 3) course of micro rotations.

A. Reference Reconfigurable CORDIC

An essential outline for reconfigurable CORDIC in light of bound together CORDIC calculation was proposed. The significant worry with the plan of regular reconfigurable design is the incongruence in RoC of round and hyperbolic directions. The RoC of round CORDIC is [−99°, 99°], while that of hyperbolic CORDIC is given by |θ| ≤ 1.1182 radians. This constrains the most extreme edge of revolution of the reconfigurable plan to 64°. The inconsistent RoC of round and hyperbolic CORDICs makes it hard to execute them in a similar circuit to perform revolution through [−180°, 180°].

Another real issue with the customary reconfigurable CORDIC is scaling. We need two distinctive scaling circuits for roundabout and hyperbolic CORDIC, and select the yield from on of the scaling circuits relying upon the choice of direction of operation.

B. Plan Strategy for Proposed Reconfigurable CORDIC

As talked about in Section III-A, the roundabout and hyperbolic CORDICs require two distinctive scaling circuits, which is very exorbitant. Subsequently, it is important to utilize a without scale execution in the reconfigurable CORDIC. Here, we examine the without scaling CORDIC furthermore, its impediments, trailed by the dialogs on our plan procedure for a reconfigurable CORDIC.

1) Scaling-Free CORDIC Algorithm and Its Limitations: The sans scaling CORDIC [2] utilizes second-arrange Taylor arrangement estimation, where the pivot lattice is given by

$$\mathbf{R_i} = \begin{bmatrix} 1 - 2^{-(2i+1)} & -2^{-i} \\ 2^{-i} & 1 - 2^{-(2i+1)} \end{bmatrix}.$$

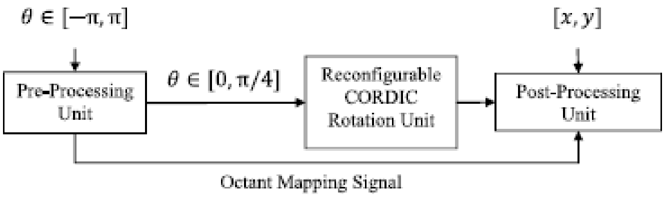


Fig 1. Proposed reconfigurable rotation-mode CORDIC processor.

In CORDIC calculation, the underlying point is disintegrated into set of smaller scale revolutions or rudimentary edges. These basic points are spoken to by arctangent constants, called Arc Digression Radix (ATR) as appeared. Half breed CORDIC calculation can be comprehended utilizing two structures : Blended cross breed and Partitioned Hybrid CORDIC engineering. In both these structures the info point, θ, is separated to give most and minimum huge part.

$$\theta = \theta_H + \theta_L$$

In Mixed cross breed, the most huge piece of the info edge, θ was parceled into two sections to build the computational exactness : the n most-critical bits and the rest of the N - n bits. Each of these gatherings of bits is the double portrayal of an edge given in by.

$$\theta_H = \sum_{i=0}^{n-1} \theta_i 2^{-i}, \theta_L = \sum_{i=n}^{N-1} \theta_i 2^{-i}$$

θH is the most applicable set, while θL changes the position of the pivoted vector to accomplish the last expected (or close-by) position. Blended half and half engineering have two CORDIC processor, appeared in Fig. 2. The principal CORDIC processor utilizes point, θ, and introductory vector X0 and Y0 to create Xm, Ym and Zn toward the finish of n cycle. The turn headings are created successively, same as that of traditional CORDIC calculation. After all revolution, the most huge piece of Zn is zero. The second CORDIC processor begin working with Xm, Ym and Zn to give final result Xn and Yn toward the finish of N-n emphasess.

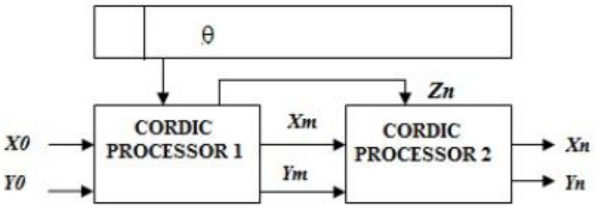


Fig. 2. Architecture of Mixed Hybrid CORDIC algorithm.

A. Rotations in Digital Systems.

This area audits key ideas identified with revolutions in computerized frameworks. Additional data can be found.
In a computerized framework, a revolution by an edge α can be depicted as an increase by a perplexing coefficient P = C + jS,

$$\begin{bmatrix} X_D \\ Y_D \end{bmatrix} = \begin{bmatrix} C & -S \\ S & C \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix},$$

where x + jy is the information and XD + jYD is the aftereffect of the turn. C and S are b-bit whole number numbers in 2's supplement in the range [−2b−1, 2b−1 − 1]. They are gotten from the turn edge as

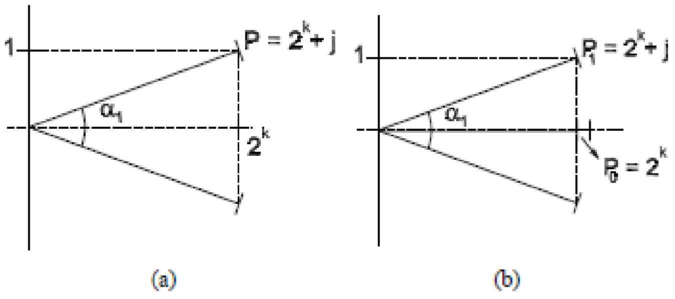
$$\begin{aligned} C &= R \cdot (\cos\alpha + \epsilon_c) \\ S &= R \cdot (\sin\alpha + \epsilon_s), \end{aligned}$$

where c and s are the quantization mistakes of the cosine and sine segments, individually, and R is the scaling component. The yield XD + jYD is likewise scaled by R.
The revolution mistake is the separation between the correct revolution and the genuine turn because of quantization. In the event that the rotator has different turn points αi, i = 1, . . .,M, with their relating coefficients Pi = Ci + jSi, the turn mistake is computed as.

$$\epsilon = \max_i(\epsilon(i)) = \max_i\left(\sqrt{\epsilon_c^2(i) + \epsilon_s^2(i)}\right).$$

Finally, the effective word length is the number of bits of the output that are guaranteed to be accurate and is calculated from the rotation error as

$$WL_E = -\log_2 \frac{\epsilon}{2\sqrt{2}} = -\log_2 \epsilon + \frac{3}{2}.$$



(a) (b)

Fig.3. CORDIC micro-rotation angles. (a) Conventional CORDIC. (b) Redundant CORDIC.

B. The CORDIC Algorithm

The CORDIC calculation considers the coefficients P = C+ jS = 2k + jδk, where δk ∈ {−1, 1} and k = 0, . . .,M is the smaller scale turn arrange. The relating points are αk = tan−1(S/C) = δk tan−1(2−k). This is appeared in Fig.3 a.

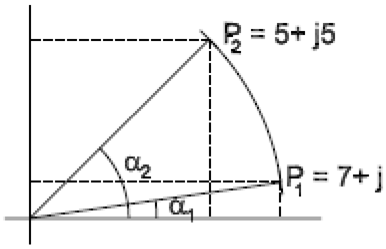


Fig. 4. Example of friend angles for P1 = 7+j and P2 = 5+j5.
The CORDIC calculation separates the revolution point θ into an entirety of miniaturized scale pivots by the edges αk, i.e.,

$$\theta = \sum_{k=0}^{M} \alpha_k + \epsilon_\phi$$

where _φ is the remaining phase error.

Each micro-rotation stage calculates

$$\begin{bmatrix} X_D \\ Y_D \end{bmatrix} = \begin{bmatrix} 2^k & -\delta_k \\ \delta_k & 2^k \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix},$$

where δk determines the direction of the rotation, and the scaling factor of the stage is R $R(k) = \sqrt{2^{2k}+1}$. The revolution blunder at each miniaturized scale pivot stage is ———————————— = 0 and The word length is WLE = ∞. This implies the coefficient Pk turns precisely αk degrees and the scaling component for both points in each smaller scale turn is the same. The last is dependably valid, as the coefficients are conjugated.

C. Redundant CORDIC Algorithm

The excess CORDIC [15] calculation utilizes a similar set of coefficients P1 = C1 + jS1 = 2k + jδk as the CORDIC calculation, with the distinction that δk ∈ {−1, 0, 1}. This includes a turn P0 = 2k by 0◦ in the part, as appeared in Fig. 3(b). This builds the arrangement of option points per organize, which suggests a quicker meeting to the turn edge. In any case, it has the disadvantage that the scaling of the points of the bit is distinctive. Along these lines, repetitive CORDIC calculations require extraordinary stages to repay the scaling and, in this manner, lessen the turn mistake.

III. CONCLUSION

In this paper, a rearranged CORDIC Architecture is displayed which gives improvement regarding multifaceted nature, time and space. It requires greatly less memory i.e. just two 32 registers. Fine stage and recurrence determination is additionally kept up with lessening the gadget usage as the engineering uses just two adders, two multiplexers, two multipliers and two registers. This paper likewise shows a productive SDR utilizing the Optimized CORDIC which can quickly bounce between the frequencies. The proposed structure expends lesser silicon zone and speedier responsive and is feasible on FPGA as the memory required is less and the design of CORDIC is basic. The future extent of the work is plan the SDR to bolster more extensive recurrence extend, complex tweak strategies and to work at higher clock rate.

References

[1] Supriya Aggarwal, Pramod K. Meher, and Kavita Khare "Concept, Design, and Implementation of Reconfigurable CORDIC". Second International Conference on Computational Intelligence & Communication Technology. IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS 2016.

[2] Bhawna tiwari, Nidhi Goel , "Implementation of a Fast Hybrid CORDIC Architecture**"**, IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS 1063-8210 © 2016 IEEE.

[3] Mario Garrido, Petter Källström, Martin Kumm and Oscar Gustafsson, " CORDIC II: A New Improve CORDIC Algorithm", This article has been accepted for publication in a future issue of this journal 2016.

[4] Pramod K. Meher, Javier Valls, Tso-Bing Juang, K. Sridharan, Koushik Maharatna, "50 Years of CORDIC: Algorithms, Architectures and Applications", IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS-I: REGULAR PAPERS 2010.

[5] Leena Vachhani, K. Sridharan, Pramod K. Meher. "Efficient CORDIC Algorithms and Architectures for Low Area and High Throughput Implementation." 2008 IEEE. Vol. 5, No. 2, Indian Institute of Technology Madras, Chennai -600036, India.

[6] Shaoyun Wang, Vincenzo Piuri and Earl E. Swartzlander , "Hybrid CORDIC Algorithms." IEEEE TRANSACTIONS ON COMPUTERS, VOL. 46, NO. 11, NOVEMBER 1997.

[7] F. Jaime, M. Sánchez, J. Hormigo, J. Villalba, and E. Zapata, "Enhanced scaling-free CORDIC," IEEE Trans. Circuits Syst. I, vol. 57, no. 7, pp. 1654–1662, July 2010.

[8] Y. Liu, L. Fan, and T. Ma, "A modified CORDIC FPGA implementation for wave generation," Circuits Syst. Signal Process., vol. 33, no. 1, pp.321–329, 2014.

[9] C.-Y. Yu, S.-G. Chen, and J.-C. Chih, "Efficient CORDIC designs for multi-mode OFDM FFT," in Proc. IEEE Int. Conf. Acoust. Speech Signal Process., vol. 3, May 2006, pp. 1036–1039.

[10] C.-Y. Chen and C.-Y. Lin, "High-resolution architecture for CORDIC algorithm realization," in Proc. Int. Conf. Comm. Circuits Syst., vol. 1,Jun. 2006, pp. 579–582.

[11] M. Garrido and J. Grajal, "Efficient memory less CORDIC for FFT computation," in Proc. IEEE Int. Conf. Acoust. Speech Signal Process.,vol. 2, Apr. 2007, pp. 113–116.

[12] R. Andraka, "A survey of CORDIC algorithms for FPGA based computers," in Proc. ACM/SIGDA Int. Symp. FPGAs, Feb. 1998, pp. 191–200.

[13] P. K. Meher, J. Valls, T.-B. Juang, K. Sridharan, and K. Maharatna, "50years of CORDIC: Algorithms, architectures, and applications," IEEE Trans. Circuits Syst. I, vol. 56, no. 9, pp. 1893–1907, Sep. 2009.

[14] M. Garrido, O. Gustafsson, and J. Grajal, "Accurate rotations based on coefficient scaling," IEEE Trans. Circuits Syst. II, vol. 58, no. 10, pp. 662–666, Oct. 2011.

[15] N. Takagi, T. Asada, and S. Yajima, "Redundant CORDIC methods with a constant scale factor for sine and cosine computation," IEEE Trans. Comput., vol. 40, no. 9, pp. 989–995, Sep. 1991.