Design and Implementation of Reconfigurable CORDIC in Rotation and vectoring-modes

S.RAMAKRISHNA[1] P.PARVATHI[2]
[1]PG Scholar, Dept of ECE, G. Pulla Reddy Engineering College (Autonomous), Kurnool, Andhra Pradesh, India
[2]Assistant Professor, Dept of ECE, G. Pulla Reddy Engineering College (Autonomous), Kurnool, Andhra Pradesh, India

Abstract
CORDIC or CO-ordinate Rotation Digital Computer is a quick, straightforward, productive and capable calculation utilized for various Digital Signal Processing applications. Essentially produced for continuous airborne calculations, it utilizes a special processing strategy which is particularly reasonable for understanding the trigonometric connections engaged with plane coordinate turn and change from rectangular to polar shape. It includes an extraordinary serial number-crunching unit having three move registers, three adders/sub tractors, a Look-Up table and unique interconnections. In this venture, A CORDIC-based processor for sine/cosine count was composed utilizing VHDL programming in Xilinx. The CORDIC module was tried for its usefulness Furthermore, accuracy by the test-seat investigation. Consequently, FPGA usage of the CORDIC center took after by Chip Scope Pro investigation of the yield rationale waveforms was performed.

Keywords: *Circular trigonometry, Coordinate rotation digital computer (CORDIC), hyperbolic trigonometry, reconfigurable CORDIC.*

I. INTRODUCTION

For quite a while the field of Digital Signal Processing has been ruled by Microchips. This is for the most part since they give architects the upsides of single cycle increase gathers guideline and in addition unique tending to modes. Despite the fact that these processors are modest and adaptable they are generally easing back with regards to performing certain requesting signal handling errands e.g. Picture Compression, Digital Communication, what's more, Video Processing Recently, fast headways have been made in the field of VLSI and IC plan. Subsequently uncommon reason processors with custom-models have come up. Higher velocities can be accomplished by these tweaked equipment arrangements at aggressive expenses.

To add to this, different straightforward and equipment productive calculations exist which delineate onto these chips and can be utilized to improve speed and adaptability while playing out the coveted flag handling undertakings. One such straightforward and equipment effective calculation is CORDIC, an acronym for Coordinate Rotation Digital Computer, proposed by Jack E Volder . CORDIC utilizes just Shift-and Add number juggling with table Look-Up to execute changed capacities. By making slight changes in accordance with the underlying conditions and the LUT esteems, it can be utilized to effectively actualize Trigonometric, Hyperbolic, Exponential capacities, Coordinate changes and so on utilizing the similar equipment. Since it utilizes just move include number-crunching, VLSI execution of such a calculation is effectively achievable. DCT calculation has different applications and is generally utilized for Image pressure. This reduces the overall power consumption.

FPGA gives the equipment condition in which devoted processors can be tried for their usefulness. They perform different rapid operations that can't be acknowledged by a straightforward chip. Consequently, it shapes the perfect stage to actualize and test the usefulness of a devoted processor outlined utilizing CORDIC calculation.

II. LITERATURE SURVEY

In [1] a brought together calculation for rudimentary capacities three reconfigurable CORDIC plans: a reconfigurable turn mode CORDIC that works either for roundabout or hyperbolic direction, a reconfigurable vectoring-mode CORDIC for roundabout and hyperbolic directions, and a summed up reconfigurable CORDIC that

can work in any of the modes for both roundabout and also hyperbolic directions.

The reconfigurable CORDIC can play out the calculation of different trigonometric and exponential capacities, logarithms, square-root, and so on of roundabout and hyperbolic CORDICs utilizing either revolution mode or vectoring method of operation in one single circuit. It can be utilized as a part of advanced synchronizers, illustrations processors, logical number crunchers and numerous different applications, with noteworthy range sparing over that of utilizing two CORDICs for various directions.

In [2] CORDIC circuits essentially work in two modes: in particular, the revolution mode and the vectoring-mode. Revolution mode CORDIC decides the segments of a vector that outcome because of turn of a given vector by a given edge. Utilizing vectoring mode CORDIC, the size and in addition the stage point of a planar vector are evaluated from its part esteems. It additionally has an extensive variety of uses. CORDIC calculations are additionally classified as straight, round or hyperbolic, as per the directions of the vectors created by progressive CORDIC cycles. Out of these three classifications, roundabout and hyperbolic CORDICs are prominently utilized. Round CORDIC is basically utilized for the calculation of sine/cosine capacities, waveform era execution of computerized channels change calculation, lattice figuring's and so on.

III. PROPOSED SYSTEM

RECONFIGURABLE CORDIC

To design a reconfigurable CORDIC architecture with minimum reconfiguration overhead, we need to maximize the sharing of common hardware circuit in different configurations. Therefore, to explore the possibility of reconfigurable CORDIC, we examine, here, the commonalities in three main issues of CORDIC implementation, namely: 1) the coordinate-rotation matrix; 2) selection of elementary angles; and 3) direction of micro rotations.

A. Reference Reconfigurable CORDIC

A fundamental outline for reconfigurable CORDIC in light of brought together CORDIC calculation was proposed. The RoC of round CORDIC is [−99°, 99°], while that of hyperbolic CORDIC is given by $|\theta| \leq 1.1182$ radians. The inconsistent RoC of round and hyperbolic CORDIC makes it hard to execute them in a similar circuit to perform rotation through [−180°, 180°]. Another significant issue with the regular reconfigurable CORDIC is scaling. We need two diverse scaling circuits for round and hyperbolic CORDIC and select the yield from one of the scaling circuits relying upon the determination of the direction of operation.

B. Design Strategy for Proposed Reconfigurable CORDIC

The circular and hyperbolic CORDIC require two diverse scaling circuits, which is very expensive. Hence, it is important to utilize a without scale execution in the reconfigurable CORDIC. Here, we talk about the scaling free CORDIC and its constraints, trailed by the dialogs on our outline technique for a reconfigurable CORDIC.

1) Re-configurability of Rotation-Mode CORDIC: Without scaling, calculations for round and hyperbolic directions are proposed. Besides, in both the sans scaling calculations, third request of estimation of Taylor arrangement is utilized to determine the CORDIC pivot frameworks, as

$$\mathbf{R_{ci}} = \begin{bmatrix} 1 - 2^{-(2s_i+1)} & -(2^{-s_i} - 2^{-(3s_i+3)}) \\ 2^{-s_i} - 2^{-(3s_i+3)} & 1 - 2^{-(2s_i+1)} \end{bmatrix}$$

$$\mathbf{R_{hi}} = \begin{bmatrix} 1 + 2^{-(2s_i+1)} & 2^{-s_i} + 2^{-(3s_i+2)} \\ 2^{-s_i} + 2^{-(3s_i+2)} & 1 + 2^{-(2s_i+1)} \end{bmatrix}.$$

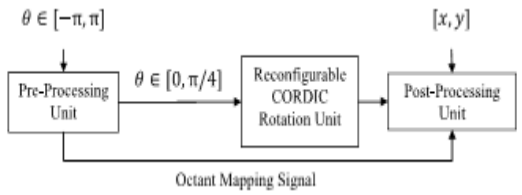


Fig.3. Proposed reconfigurable rotation-mode CORDIC processor.

Note that a similar arrangement of basic edges is utilized for both circular and hyperbolic rotation modes. This is a major favorable position to infer the reconfigurable CORDIC since no separation is required to distinguish the small scale rotation as indicated by the directions. For circular and hyperbolic directions, the basic edges are reclassified as

$$\alpha_i = 2^{-s_i}$$

While i is the number of shifts for i[th] iteration. The RoC for both the directions is good and reaches out to the whole organize space.

2)Re-configurability of Vectoring-Mode CORDIC: To understand a vectoring-mode CORDIC, all the small scale rotation will be performed the clockwise way for both the circular and hyperbolic directions. The revolution grids are given by

$$\mathbf{R_{ci}} = \begin{bmatrix} 1 - 2^{-(2s_i+1)} & 2^{-s_i} - 2^{-(3s_i+3)} \\ -(2^{-s_i} - 2^{-(3s_i+3)}) & 1 - 2^{-(2s_i+1)} \end{bmatrix}$$

$$\mathbf{R_{hi}} = \begin{bmatrix} 1 + 2^{-(2s_i+1)} & -(2^{-s_i} + 2^{-(3s_i+2)}) \\ -(2^{-s_i} + 2^{-(3s_i+2)}) & 1 + 2^{-(2s_i+1)} \end{bmatrix}$$

While i is the shift index i[th] iteration. The sign-piece of the y-coordinate over progressive emphases decides the edge of turn θ. For vectoring-mode, the greatest point of turn that can be processed lies in the range [0,π/4]. Be that as it may, this range can be stretched out to the whole arrange space utilizing octant wave symmetry of sine and cosine capacities for round direction.

C) Proposed Reconfigurable CORDIC:

The organize count frameworks for round and hyperbolic CORDIC contrast by the indication of operands, and to understand that increments are to be supplanted by subtractions and the other way around. This can be effortlessly acknowledged by a reconfigurable add/subtract circuit. In the two cases, the fundamental move could be either 2 or 3, yet the quantity of smaller scale revolutions differs with the method of operation.

1) Rotation mode reconfigurable CORDIC;

2) Vectoring-mode reconfigurable CORDIC;

3) Generalized reconfigurable CORDIC.

1). Rotation-Mode Reconfigurable CORDIC

The proposed plan for reconfigurable rotation mode CORDIC (appeared in Fig. 3) comprises of three sections: 1) preprocessing unit; 2) reconfigurable CORDIC rotation unit, and 3) post processing unit. The preprocessing unit guarantees that the information turn point to the CORDIC preparing structure dependably lies in the range [0,π/4], as the greatest pivot edge that can be taken care of by smaller scale revolution grouping generator is π/4. The client can control the direction of the reconfigurable CORDIC by changing a 1-bit flag T. The rotation network for reconfigurable turn mode CORDIC is gotten in the wake of bringing together the revolution grids of round and hyperbolic case given by (4a) and (4b), individually, as

$$\mathbf{R_i} = \begin{bmatrix} 1 \pm 2^{-(2s_i+1)} & \pm(2^{-s_i} \pm 2^{-(3s_i+2+T)}) \\ 2^{-s_i} \pm 2^{-(3s_i+2+T)} & 1 \pm 2^{-(2s_i+1)} \end{bmatrix}$$

Where T= 0 hyperbolic
T=1 circular

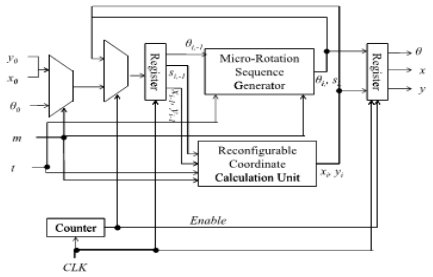


Fig.4. The structure of the proposed reconfigurable recursive CORDIC architectures.

Proposed Recursive Architecture:

The recursive design (appeared in Fig. 4) utilizes a solitary CORDIC small scale rotator to play out all the CORDIC cycles. The circular CORDIC of requires one cycle, not exactly the hyperbolic CORDIC, yet here we understand the engineering for a similar number of emphasis (eight for s basic =2 and eleven for s basic=3) for both circular and hyperbolic directions. The reconfigurable coordinate calculation unit (RCCU) shown in below fig .(5).

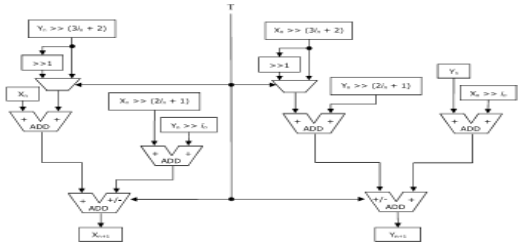


Fig.5. RCCU for recursive design

Proposed Pipelined Architecture:

Fig. 6 demonstrates the reconfigurable CORDIC rotation unit for fundamental move 2. The move record s_i is fixed in each RCCU, and consequently, the shifters are hardwired and don't include high multifaceted nature

barrel-shifters. The execution of RCCUs differs as per the fundamental move s_i. With slight changes, the pipeline can be reached out for essential move 3.

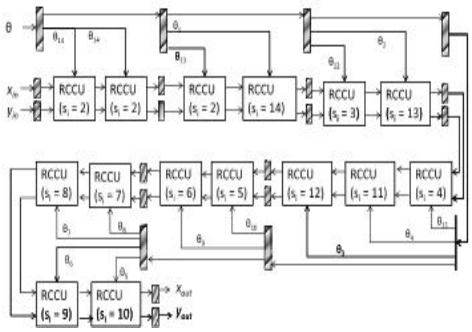


Fig.6. Reconfigurable rotation-mode CORDIC unit for basic-shift 2

2). Vectoring-Mode Reconfigurable CORDIC:

The reconfigurable rotation matrix for vectoring mode is obtained by unifying (6a) and (6b), as

$$\mathbf{R_i} = \begin{bmatrix} 1 \pm 2^{-(2s_i+1)} & \mp(2^{-s_i} \pm 2^{-(3s_i+2+T)}) \\ -(2^{-s_i} \pm 2^{-(3s_i+2+T)}) & 1 \pm 2^{-(2s_i+1)} \end{bmatrix}$$

Where T=0 hyperbolic T=1 circular

By changing the usage of the RCCU to actualize, the recursive design of Fig. 2 can be utilized to acknowledge CORDIC cycles for vectoring-mode. The rollover counter esteem is 15 for s basic=2 and 17 for s basic=3. The pipelined design of vectoring-mode reconfigurable CORDIC comprises of eight phases for basis = 2, as appeared in Fig. 7. Like reconfigurable rotation mode CORDIC, for expanding shift-records, the usage of RCC has streamlined for reconfigurable vectoring-mode CORDIC also. The information facilitates [Xin', Yin'] are first preprocessed to acquire coordinates [xin, yin] and octant mapping signals. The directions [xin, yin] are a contribution to the vectoring-mode CORDIC pipeline to create an edge $\theta \in [0, \pi/4]$. The revolution point θ produced by the vectoring-mode CORDIC pipeline is mapped to the coveted octant utilizing the octant mapping signals created by the preprocessing unit.

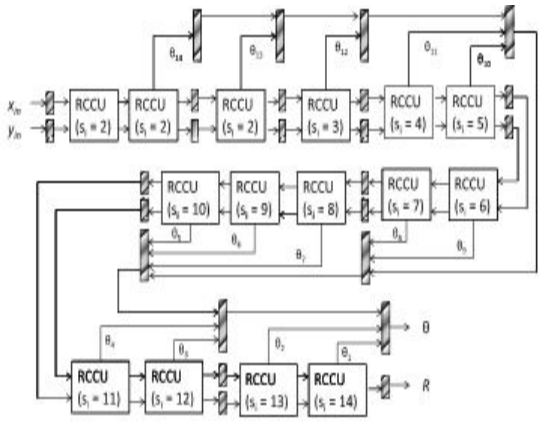


Fig 7 Proposed pipeline reconfigurable vectoring-mode CORDIC unit for sbasic=2

In this manner, the RoC upheld by the proposed vectoring-mode reconfigurable CORDIC is $[-\pi, \pi]$. C. Proposed Generalized Reconfigurable CORDIC The summed up reconfigurable CORDIC can work either in vectoring-mode or in revolution mode for both round and hyperbolic directions. The client can choose the direction of operation utilizing a solitary piece flag T (T=1 for roundabout and T=0 for hyperbolic). Another single piece flag M is utilized to control the method of operation (M=0 for revolution mode and M=1 for vectoring-mode). The recursive engineering of the proposed summed up reconfigurable CORDIC is executed by joining the CORDIC miniaturized scale rotators for both rotation mode and vectoring-mode CORDICs, as appeared in Fig. 8. The piece graph for pipelined summed up reconfigurable CORDIC utilizing essential movements basic=2 has appeared in Fig. 9. It can be effectively reached out to essential movements basic=3 as is accomplished for reconfigurable rotation mode and vectoring-mode CORDICs.

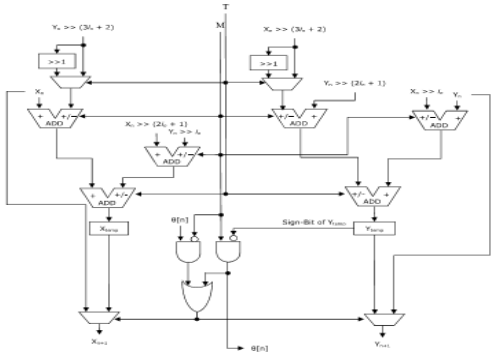


Fig.8. The structure micro rotator for the proposed CORDIC

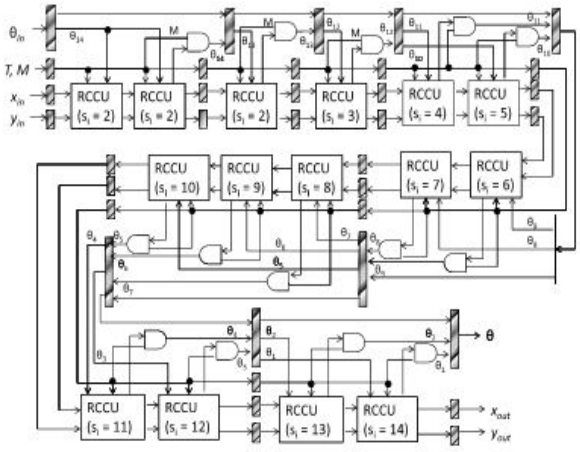


Fig.9.Proposed pipeline generalized reconfigurable CORDIC unit for sbasic=2.

IV. Extension Work

PASTA (Parallel self timed adder) is based on recursive formulation for performing multi bit binary addition. The operation is parallel for those do not need any carry change propagation. Thus the design attains logarithmic performance over random operand conditions.

Architecture of PASTA

Let a_{n-1}, a_{n-2}. . . a_0 and b_{n-1},b_{n-2}. . . b_0 be two n-bit paired numbers with aggregate and convey indicated by S_{n-1}, S_{n-2}. . . S_o and c_n, c_{n-2}. . . c_0 where 0th piece speaks to the minimum huge bit. The two information multiplexer has determination input that compares to the demand hand shake flag and will be a solitary zero to one progress meant by SEL. It will initially choose the real operand amid SEL = 0 and will change to input ways for rehashed emphases utilizing SEL =1. The HAs input way empowers the consistent a few emphases until the point when all the convey signs will be accepted to have zero esteems.

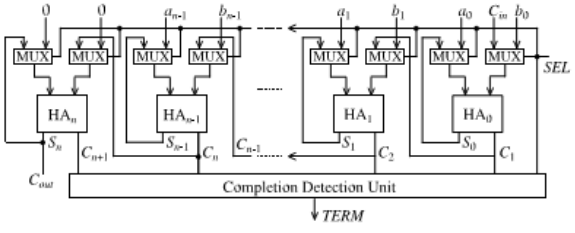


Fig.10. General block diagram of PASTA

V. RESULTS

The written Verilog HDL Modules have successfully simulated and verified using Modelsim III 6.4b and synthesized using Xilinx ISE 14.7.

Results of Generalized Reconfigurable CORDIC

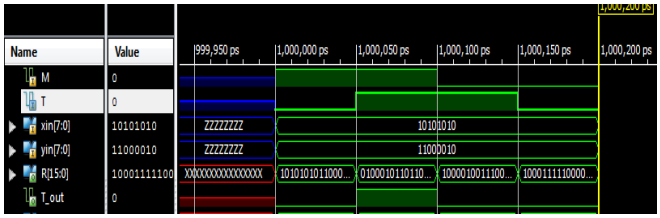
Design Summary:

Logic utilization	Used	Available	utilization
Number of slices	1650	4656	35%
Number of 4 input LUTs	2876	9312	30%
Number of Bonded IOBs	35	232	15%

Synthesis Report:

```
LUT4:I2->O            1   0.612   0.000  m15/y1<3>1091 (m15/y1<3>109)
MUXF5:I1->O           1   0.278   0.357  m15/y1<3>109_f5 (R_3_OBUF)
OBUF:I->O             3.169          R_3_OBUF (R<3>)
----------------------------------------
Total                 183.460ns (96.403ns logic, 87.057ns route)
                                (52.5% logic, 47.5% route)
```

Simulation Results:



Inputs	M	T	Output
X=10101010	1	1	0100010110110001
Y=11000010	1	0	1010101011000010
	0	0	1000111110000011
	0	1	1000010011100000

Results of Generalized Reconfigurable CRDIC with PASTA

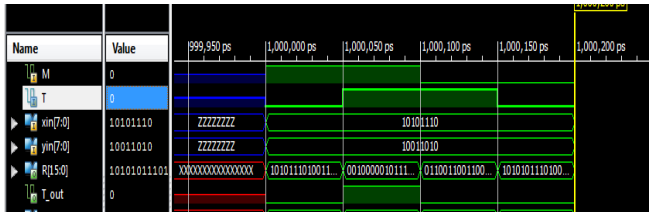
Design Summary

Logic utilization	Used	Available	Utilization
Number of slices	1300	4656	27%
Number of 4 input LUTs	2438	9312	26%
Number of bonded IOBs	35	232	15%

Synthesis Report:

```
MUXF5:I0->O           1   0.278   0.357  m15/y1<6>_f5 (R_6_OBUF)
OBUF:I->O                 3.169          R_6_OBUF (R<6>)
----------------------------------------
Total                   161.350ns (98.763ns logic, 62.586ns route)
                                 (61.2% logic, 38.8% route)
```

Simulation Results:



inputs	M	T	output
X=10101110 Y=10011010	1	1	0010000010111110
	1	0	1010111010011010
	0	0	1010101110100001
	0	1	0010000010111110

VI. CONCLUSION

CORDIC is a powerful algorithm, and a popular algorithm of choice when it comes to various Digital Signal Processing applications. Implementation of a CORDIC-based the processor on FPGA gives us a powerful mechanism of implementing complex computations on a platform that provides a lot of resources and flexibility at a relatively lesser cost. Further, since the algorithm is simple and efficient the design and VLSI implementation of a CORDIC based processor is easily achievable. In this project, a CORDIC module is designed and simulated using Xilinx ISE using VHDL as a synthesis tool. The output of the CORDIC core is analyzed and verified on the test-bench.

REFERENCES

[1] Ray Andraka, FPGA '98. Proceedings of the 1998 ACM/SIGDA sixth international 9999 symposium on Field programmable gate arrays, Feb. 22-24, 1998, Monterey, CA. pp19200

[2] Vikas Sharma, "FPGA Implementation of EEAS CORDIC based Sine and Cosine Generator", M.Tech Thesis, Dept. Electron. Comm. Eng., Thapar Uni., Patiala, 2009.

[3] Satyasen Panda,"Performance Analysis and Design of a Discrete Cosine Transform Processor using CORDIC Algorithm", M.Tech Thesis, Dept. Electron.Comm.Eng., NIT Rourkela, Rourkela, Orissa, 2010.

[4] Kris Raj, Gaurav Doshi, Hiren Shah [online] Available:http://teal.gmu.edu/courses/ECE645/projects_S06/talks/CORDIC.pdf(URL)

[5] S.K.Pattanaik and K.K.Mahapatra,'DHT Based JPEG Image Compression Using a Novel Energy Quantization Method", IEEE International Conference on Industrial Technology,pp.2827-2832, Dec 2006.

[6] Volder, J.,"Binary Computation Algorithms for coordinate rotation and function generation,"Convair Report IAR-1 148 Aeroelectrics Group, June 1956.

[7] Volder, J.,"The CORDIC Trigonometric Computing Technique,"IRE Trans. Electronic Computing, Vol EC-8, pp330-334 Sept 1959.

[8] Walther J.S.,"A unified algorithm for elementary functions," Spring Joint Computer Conf., 1971, proc., pp.379-385.

Author Biography:

S. Ramakrishna is pursuing M. Tech in VLSI & Embedded Systems at G. Pulla Reddy Engineering College, Kurnool, A.P, INDIA. His area of interest is in the field of VLSI & Analog Electronics.

P. Parvathi received her B.Tech degree in ECE in 2007 from VTU, Belgaum,INDIA. M.Tech degree in VLSI & Embedded Systems in 2009 from VTU, Bangalore, INDIA. Currently she is working as assistant professor in ECE department in G.Pulla Reddy Engineering College, Kurnool, A.P, INDIA. Her area of interest is in Digital VLSI Design & Embedded Systems. She has published two IEEE international journals and worked as a Project Engineer in BEL (Bangalore). She has 7 years of teaching experience.