International Journal of Research
Available at https://edupediapublications.org/journals
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 05 Issue 15
May 2018

Generation of Sine and Cosine signals for Fixed Rotation Angle

Jetti Rangamma & K. Thriveni

[1] *M.Tech student, Department of ECE, BIT Institute of Technology, Hindupur*
[2] *Assistant Professor, Department of ECE, BIT Institute of Technology, Hindupur*

ABSTRACT

CORDIC is the more efficient algorithm to perform trigonometric operations for generating sine & Cosine waveforms. By using rotation and vectoring modes in the coordinate system we can implement in hardware. Different scaling factors and its compensated techniques can also be calculated further. Using this algorithm we can generate Sine and cosine signals in signal processing applications. Also, we can perform Hyperbolic and Exponential calculations. It can be further enhanced by extending its version for Extended Hyperbolic and Linear Coordinates for the trigonometric functions and inverse trigonometric functions. This project is implemented in Xilinx ISE 13.2 and the output waveforms are simulated in Questasim 10.0b. The design implementation of CORIDC algorithm thus reduces the hardware implementation by using shift registers and adders. Thereby increases its speed.

Keywords:
CORDIC algorithm, Coordinate Systems, Vector modes

INTRODUCTION

In the field of electronics, the need for high speed increases day by day. Although, the speed increases but still, optimizing the size and power would be a challenging task. The need for multipliers has increased drastically in the applications of Communication Systems and Signal Processing. Optimization of these multipliers can be done using various techniques in allocating different Look-up tables. Also, generating the analogous signal using multipliers would consume more power and area. In this project, the algorithm is simulated that occupies the minimum area that is implemented in SPARTAN 3E Fpga family that consumes very few Look – up tables and slices.

Coordinate Rotational Digital Computer (CORDIC) is a well known algorithm used for performing trigonometric, hyperbolic and Exponential operations. These functions are computed more often in Scientific and Engineering applications.

CORDIC Algorithm:

Two vectors are defined as [X0,Y0] and [Xn,Yn].This algorithm is designed for n number of iterations which access both Vectoring modes and Rotation modes. In rotation mode, the coordinate components of a vector and an angle of rotation are given to the original vectors of the coordinate components for which the rotations are allowed to pass through the given angle for computation. Where as in vectoring mode, the coordinate components of a vector are given and the magnitude of angular movement of the original vectors are computed.

The conventional method of implementation of 2D vector rotation is done using coordinate system is as:

$$\begin{bmatrix} cos\Theta & -sin\Theta \\ sin\Theta & cos\Theta \end{bmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} x' \\ y' \end{pmatrix}$$

Where (x', y') are the coordinates of (x, y) for a unit circle rotated at an angle of Θ.

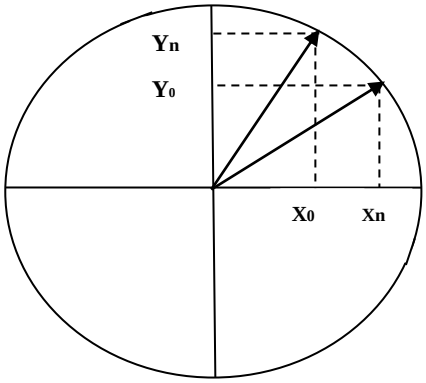

Fig1: Point on the unit circle rotated by an angle Θ

The matrix operation can be expressed below as:

$$X' = X \cos\Theta - Y \sin\Theta$$
$$Y' = Y \cos\Theta + X \sin\Theta$$

Re arranging the above equations as:

$$X' = \cos\Theta[x - y\tan\Theta]$$
$$Y' = \cos\Theta[y + x\tan\Theta]$$

Assume that the angle of rotation is fixed to $\tan(\Theta) = +2^{-i}$

Table1: Different Iteration values from 0 to 9

i	$\tan(\Theta) = +2^{-i}$	$\Theta = \arctan(+2^{-i})$
0	1	4.5
1	0.5	26.565
2	0.25	14.036
3	0.125	7.125

International Journal of Research
Available at https://edupediapublications.org/journals
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 05 Issue 15
May 2018

4	0.0625	3.576
5	0.03125	1.79
6	0.015625	0.895
7	0.0078125	0.448
8	0.00390625	0.224
9	0.001953125	0.112

This is performed by shifting the x & y variables to Right.

This expression further can be reduced as:

$$COS\theta = \frac{1}{\sqrt{1+\tan(\theta)2}} = \frac{1}{\sqrt{1+(\pm 2^{i})^{2}}} = Ki$$

$$Xn = Ki\,[X0 - Y0\,di\,2^{-i}]$$

$$Yn = Ki\,[Y0 + X0\,di\,2^{-i}] \text{ where } di = \pm 1$$

Where, the positive unit refers to the positive iterations and the negative unit refers to the negative iterations.

Assume, the rotational angle is 45 degrees as input to the CORDIC algorithm.

The desired angle of rotation is obtained by performing a series of successive smaller elementary rotations.
Where i = 0, 1, 2 ….n-1.

For tan $\theta = 2^{-i}$ the angle of rotations are

Table2: Desired angle of rotation for successively smaller elementary rotations

i	tan $\theta = 2^{-i}$	θ=arc tan(2^{-i})	Zi	Rotation θ	Final Angle
0	1	45	20	-45	-25
1	0.5	26.565	-25	26.565	1.565
2	0.25	14.036	1.565	-14.036	-12.47
3	0.125	7.125	-12.471	7.125	-5.346
4	0.0625	3.576	-5.346	3.576	-1.77
5	0.03125	1.79	-1.77	1.79	0.020
6	.....	.....	.....	.....	.....
7	.....	.....	.....	.....	.....
8	.....	.....	.....	.....	.....
..	.....	.....	.....	.....	.....
..	.....	.....	.....	.....	.....
..	.....	.....	.....	.....	.....
19	.....	.....	-0.000195	.....	.....

From this calculation we can confirm that, the CORDIC algorithm has the advancement of high precision that is capable of generating a continuous analogous signals.

$$Xn = Ki\,[X0 - Y0\,di\,2^{-i}]$$

$$Yn = Ki\,[Y0 + X0\,di\,2^{-i}]$$

$$Ki = \frac{1}{1 + (\pm 2^{i})^{2}}$$

$$Xi+1 = Ki\,[Xi - Yi\,di\,2^{-i}]$$

$$Yi+1 = Ki\,[Yi + Xi\,di\,2^{-i}]$$

$$Zi+1 = Zi - di\,tan^{-1}(2^{-i})$$

Where di = $\pm i$

The decision at each iteration i, is the direction to rotate rather than whether or not to rotate because eventually to remove the Ki term from the expressions.

In the hardware Xi is fixed to 16 bit wide and if the limit exceeds then the term Xi di 2^{-i} term moves to 0.

For example, if the design of CORDIC is 4 bit then, iterations beyond 4 bit would leads to a garbage memory. In order to restrict to the limit of its iterations we move for avoiding arc tangent calculations.

Removing the scale constant from the iterative equations would yield a shift and add type of algorithm for vector rotations. The product of the Ki can be applied elsewhere in the system or treated as part of a system processing gain.

The exact gain, without the Ki terms, depend on the number of iterations as follows:

International Journal of Research
Available at https://edupediapublications.org/journals
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 05 Issue 15
May 2018

$$An = \sqrt{1 + (2)^{-2i}}$$

For which the limits of π ranges from i=0 to n-1

The design of sine & Cosine outputs can be implemented in digital hardware by setting the variables at

$Y_0 = 0$ and $X_0 = 1/An$

$Xn \sim Cos(Z_0)$ and $Yn \sim Sin(Z_0)$

This algorithm is limited to the rotational angle between

$$\frac{-\pi}{2} \leq 0 \leq \frac{\pi}{2}$$

In order to increase the convergence range of all rotation angle $|Zi|<2\pi$.

Also, angles outside the range must be pre rotated as below:

$$X_0' = -dy_0$$

$$Y_0' = dx_0$$

$$Z_0' = Z_0 - d\frac{\pi}{2}$$

Where d = +1 if Y0 > 0, -1 otherwise

Evaluation of Algorithm for generating sine – cosine waveforms:

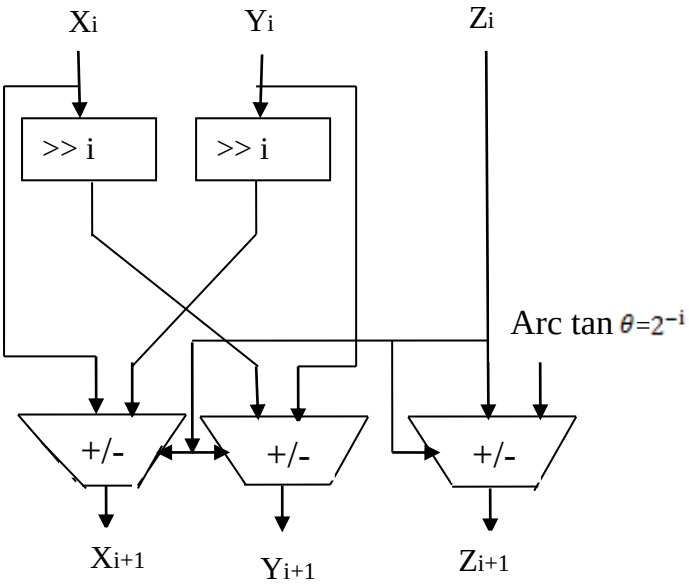

Fig2: Hardware implementation of a CORDIC iteration

Multiple iterations are used in a single angle operation. The direction of each iteration is determined according to the results of the last iteration.

Simulation and result analysis:

The design is simulated in Questasim 10.0b version and the output is verified using Xilinx ISE 13.2.

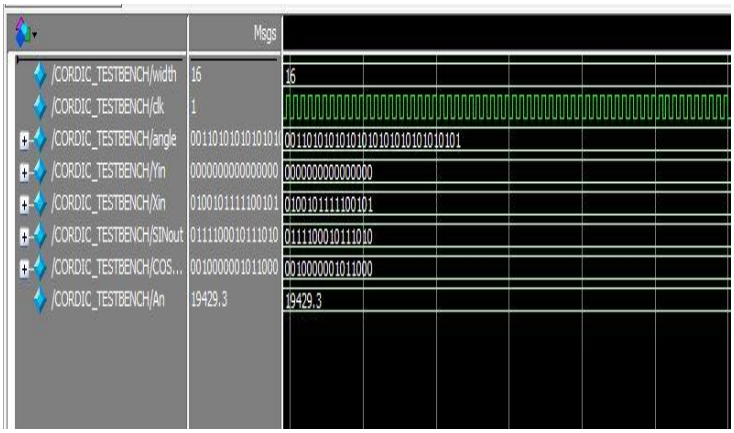

Fig3: CORDIC algorithm output generating Sine & Cosine waveforms

Device Utilization Summary				
Logic Utilization	Used	Available	Utilization	Note(s)
Number of Slice Flip Flops	955	17,344	5%	
Number of 4 input LUTs	1,420	17,344	8%	
Number of occupied Slices	935	8,672	10%	
Number of Slices containing only related logic	935	935	100%	
Number of Slices containing unrelated logic	0	935	0%	
Total Number of 4 input LUTs	1,828	17,344	10%	
Number used as logic	1,389			
Number used as a route-thru	408			
Number used as Shift registers	31			
Number of bonded IOBs	97	304	31%	
Number of BUFGMUXs	1	24	4%	
Average Fanout of Non-Clock Nets	1.83			

Fig4: Internal block of IOBs

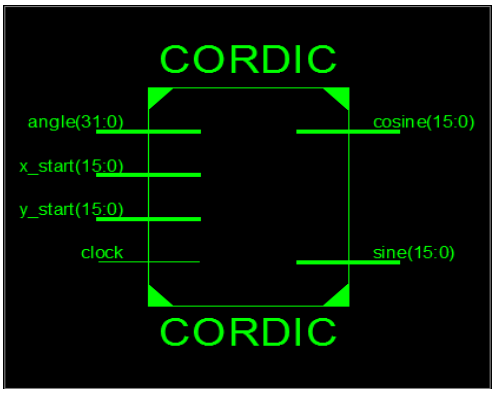

Fig5: RTL Schematic view of CORDIC algorithm.

A total of 4.283ns of delay is being observed in the Synthesis report generated by SPARTAN 3E (XA3S1200E) FPGA.

CONCLUSION:

The implementation of CORDIC algorithm for a fixed rotation angle that calculates using number of iterations to generate sine and cosine waveforms using Questasim 10.0b simulator using Verilog through repeated shift & add operations. There by reducing the hardware implementation. This design module is much more efficient and consumes less area than the conventional approach of designing hardware. The CORDIC algorithm can be further implemented for different variable rotational angles.

REFERENCES:

[1] Junwei Li, Jiandong Fang "Study of CORDIC Algorithm based on FPGA" IEEE Xplore – Control and Decision Conference(CCDC), 2016 Chinese

[2] "CORDIC II: A New Improved CORDIC Algorithm " by Mario Garrido, Petter Kallstom 2015, IEEE Transactions on Circuits and systems II, Vol:63, Issue:2

[3] YAC – Yet Another CORDIC Core Christian Haettich, March 30, 2014

[4] "Decimal CORDIC Rotation based on Selection by Rounding: Algorithm and Architecture" by Amir Kaivani, Ghassem Jaberipur. Published in: The Computer Journal , Vol: 54, Issue: 11, Nov 2011

[5] Fayez Elguibaly, "α-CORDIC: An adaptive CORDIC algorithm, Canadian Journal of Electrical and Computer Engineering, Vol: 23, Issue:3, Publilsher: IEEE, 1998