

A VERILOG BASED IMPLEMENTATION OF TRANSCENDENTAL FUNCTION CALCULATOR USING CORDIC ALGORITHM FOR SDR

Sharanagouda N Patil¹, P.V.Hunagund², R.M.Vani³

¹Dept. of Applied Electronics, Gulbarga University, India, snpaatil@gmail.com

²Dept. of Applied Electronics, Gulbarga University, India, pvhunagund@gmail.com

³Dept. of Applied Electronics, Gulbarga University, India, rmvani123@rediffmail.com

Abstract

The CORDIC (COordinate Rotation Digital Computer) algorithm is an iterative algorithm developed by Volder [1] in 1959. It rotates the vector, iteratively and in finite steps to calculate Sine and Cosine of a given angle. Additional work has been done by Walther [2] in 1971. The main principle of CORDIC are calculations based on shift-registers and adders instead of multiplications, which makes use of limited reconfigurable CLB's in FPGA efficiently saving hardware resources.

All trigonometric functions can be computed using vector rotation. CORDIC is also used for polar to rectangular and rectangular to polar conversions, calculation of vector magnitude, and also for transforms like discrete Fourier transform (DFT)/FFT on reconfigurable platform.

This paper presents the CORDIC Algorithm for calculation of elementary functions Sine and Cosine in IEEE-754 Format and Q-Format. The paper analyses the feasibility of CORDIC algorithm for implementing the elementary angle computation in FPGA. The CORDIC algorithm is implemented using Verilog language and results are obtained from Xilinx ISE simulation.

Keywords: CORDIC, Elementary angle, FPGA, SDR, VHDL

1. INTRODUCTION

CORDIC is an acronym for COordinate Rotation Digital Computer. CORDIC implements the calculations using only adders, and shift registers circumventing the need for multiplier circuits. By using this it saves and overcomes the need of complex multiplier based hardware resources. CORDIC is used for calculations of trigonometric functions. It is used for polar to rectangular & rectangular to polar conversion, also in some transformations like DFT, FFT. CORDIC Algorithm was mainly developed to find a solution for real time navigation problems. CORDIC Algorithm found application in 8087 math coprocessor [3], radar signal processor [4], Hp-35 calculator and robotics, and Wireless LAN at receiver side.

2. CORDIC ALGORITHM

CORDIC is a method of calculating values of elementary functions using much simpler math operations in a loop called a Binary Search. Most commonly CORDIC is used to calculate ArcTan-Angle, and Hypotenuse -Distance of a point. CORDIC can also be used to calculate other math functions like SIN and COS.

Binary search: The CORDIC Algorithm extensively uses binary search mechanism as its main unit for convergence computation. A binary search works as follows. Suppose you are asked to think and guess a number between 0 and 100. You make a guess and the number can be guessed in finite iterations using a binary search.

CORDIC is based on the common rotation equations:

Consider the following equations

$$\begin{aligned}x' &= x \cdot \cos \phi - y \cdot \sin \phi = \cos \phi \cdot [x - y \cdot \tan \phi] \\y' &= y \cdot \cos \phi + x \cdot \sin \phi = \cos \phi \cdot [y + x \cdot \tan \phi]\end{aligned}$$

Eqn -- (1)

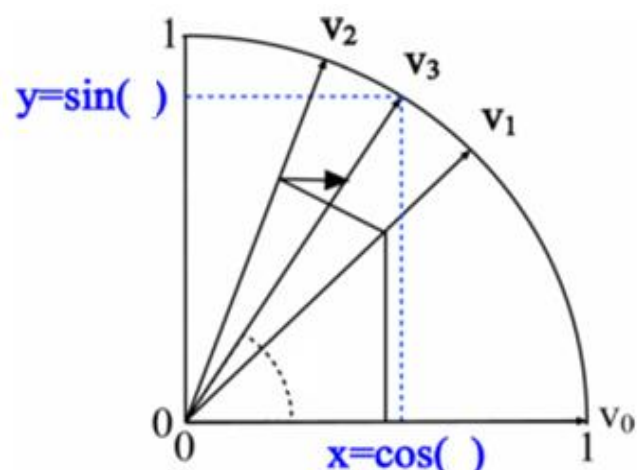


Figure 1: A Unit vector Rotation in quadrant - I.

If $\tan \phi = \pm 2^{-i}$, multiplication in (1) can be performed by a simple shift operation. This allows the vector to be iteratively rotated by desired angle in a sequence of smaller rotations by angle $\phi_i = \pm \tan^{-1}(2^{-i})$

Then in general, we have

$$x_{i+1} = k_i [x_i - y_i \cdot d_i \cdot 2^{-i}]$$

$$y_{i+1} = k_i [y_i + x_i \cdot d_i \cdot 2^{-i}] \quad \text{Eqn -- (2)}$$

where,

$$k_i = \cos(\tan^{-1}(2^{-i})) = 1 / \sqrt{1+2^{-2i}}, \quad d_i = \pm 1 \quad \text{Eqn -- (3)}$$

CORDIC rotator works in two modes: **rotation and vectoring** (Figure 1) [5]

a). **Rotation mode** : The CORDIC can be used to calculate a number of different elementary functions. The CORDIC in a Rotation mode is used to calculate the sine and cosine of an angle given in radians/degrees. The angle may be represented in various binary formats such as fixed point, floating point. In our paper the Q-Format is used to represent the angles. In the first mode it rotates the input vector by specified angle. The angle accumulator is initialized with the desired rotation angle.

CORDIC equations for this mode are:

$$x_{i+1} = x_i - y_i * d_i * 2^{-i}$$

$$y_{i+1} = y_i + x_i * d_i * 2^{-i}$$

$$z_{i+1} = z_i - d_i * \tan^{-1}(2^{-i}), \quad \text{Eqn -- (4)}$$

Where

$$d_i = \begin{cases} -1, & z_i < 0 \\ 1, & \text{otherwise} \end{cases}$$

Which evaluates to

$$x_n = A_n \cdot [x_0 \cos z_0 - y_0 \sin z_0]$$

$$y_n = A_n \cdot [y_0 \cos z_0 + x_0 \sin z_0]$$

$$z_n = 0,$$

$$\text{Eqn -- (5)}$$

Table 1: Look Up Table used for rotational angles, $i = 0 - 15$

Iteration number, i	Rotation angle, $\phi = \tan^{-1}(2^{-i})$	$\tanh \phi = 2^{-i}$
1	0.5493061443	1/2
2	0.2554128119	1/4
3	0.1256572141	1/8
4	0.0625815715	1/16
5	0.0312601785	1/32
6	0.0156262718	1/64
7	0.0078126580	1/128
8	0.0039062690	1/256
9	0.0019531270	1/512
10	0.0009765620	1/1024
11	0.0004882810	1/2048
12	0.0002441400	1/4096
13	0.0001220700	1/8192
14	0.0000610350	1/16384
15	0.0000305170	1/32768

Vectoring Mode: the vectoring mode input vector is being rotated to the x axis while recording the angle required to make that rotation. The result of the vectoring operation is a rotation angle and the scaled magnitude of original vector. The CORDIC algorithm in every mode is limited between $-\pi/2$ and $+\pi/2$. This limitation is caused by the first rotation angle $\phi^0 = \tan^{-1}(2^0)$.

3. IMPLEMENTATION:

The following figure shows the architecture diagram for CORDIC implementation in Rotation mode. It consists of a shifter and adder. The elements are generated based on the number of stages of computation.

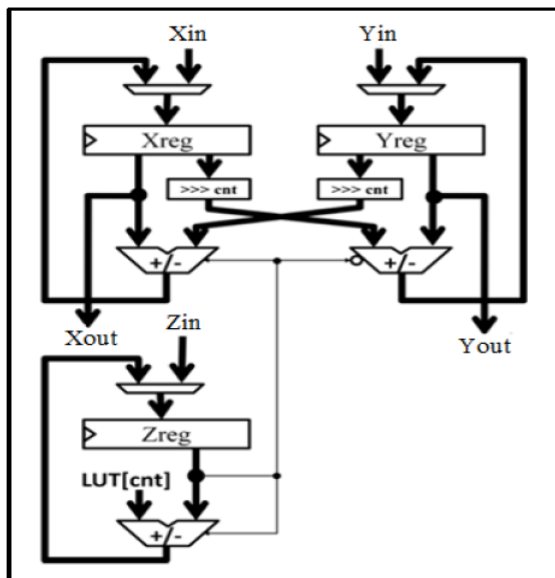


Figure 2: An architecture for CORDIC implementation

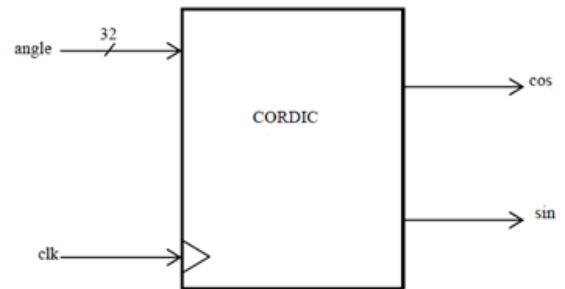


Figure 3: Verilog based CORDIC in rotation mode.

4. RESULTS

The input angle is given in fixed point Q-format. The results were obtained in IEEE-754 floating point format. The following results were obtained for CORDIC using rotation method. The design has been coded by Verilog HDL. All the results are simulated using Xilinx ISE simulator.

The simulation results for the CORDIC algorithm from the ISE simulator are shown below

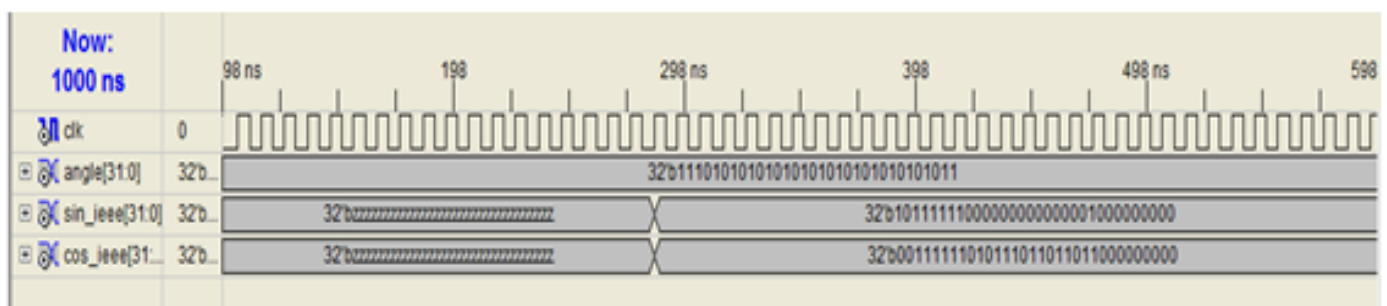


Figure 4: Simulation waveforms on XILINX - ISE simulator

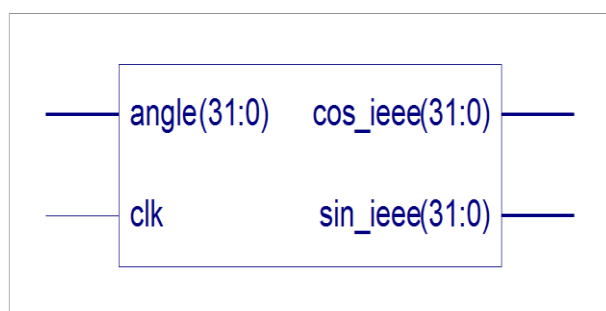


Figure 5 : Synthesized Cordic Block

5. CONCLUSIONS

CORDIC Algorithm is an indispensable tool which is used in Digital Signal Processing. The main application of the CORDIC algorithm in this paper is to design a CORDIC based Software Defined Receivers. This paper shows that this computation tool is available for use in FPGA based computing machines. This paper attempts to present a method to calculate angle in Software Defined Radio/Receiver (SDR) block by using CORDIC Algorithm.

REFERENCES

- [1]. Nee R., Prasad R.: "OFDM for Wireless Multimedia Communications", Boston: Artech House, 2000.
- [2]. Terry J., Heiskala J.: "OFDM Wireless LANs: A Theoretical and Practical Guide", Indianapolis, Ind.: Sams, 2002.
- [3] Duprat J., and Mullar J. M. "The CORDIC Algorithm: New results for VLSI implementation", IEEE Transactions on Computers, Vol. 42 pp. 168-178, 1993.
- [4]. Andraka R. J. "Building a high performance Bit serial processor in an FPGA", proceeding of design supercon 96, Jan 1996, pp 5.1-5.21.
- [5]. Valls, J.; Kuhlmann, M.; Parhi, K.K.: "Efficient Mapping of CORDIC Algorithms on FPGA", Signal Processing Systems, 2000. IEEE Workshop, pp 336-343 , 11-13 Oct. 2000.