FPGA Implementation of Pipelined CORDIC Sine Cosine Digital Wave Generator

Navdeep Prashar[1] and Balwinder Singh[2]

[1-2]Acadmic and Consultancy Services Division,
Centre for Development of Advanced Computing(C-DAC), Mohali, India
nav.prashar@gmail.com , balwinder_cdacmohali@yahoo.com

ABSTRACT

The coordinate rotation digital computer (CORDIC) algorithm is well known iterative algorithm for performing rotations in digital signal processing applications. Hardware implementation of CORDIC results increase in Critical path delay. Pipelined architecture is used in CORDIC to increase the clock speed and to reduce the Critical path delay. In this paper a hardware efficient Digital sine and cosine wave generator is designed and implemented using Pipelined CORDIC architecture. FPGA based architecture is presented and design has been implemented using Xilinx 12.3 device.

KEYWORDS

CORDIC, FPGA, Pipelined architecture, Micro -rotations.

1. INTRODUCTION

CORDIC stands for Coordinate Rotation Digital Computer is a shift and add algorithm used to compute trigonometric, hyperbolic, linear and logarithmic functions. The CORDIC algorithm is first introduced by Jack.E Volder in year 1959[1] and further extended by Walther [2]. The CORDIC algorithm has found its various applications such as pocket calculator, numerical co-processor, and image processing applications, direct digital synthesis and analog digital modulation.

CORDIC operates mainly in two modes for computation of different functions. These modes are known as rotation mode and vector mode. In rotation mode, the co-ordinate components of a vector and an angle of rotation is given and the co-ordinate component of original vector, after rotation through given angle are computed. In vector mode, the coordinate component of a given vector is given and the magnitude and angular argument of original vector are computed.

The CORDIC technique uses a one bit at a time approach to make computation to an arbitrary precision [3]. Typically, these tables only one to two entries per bit of precision. CORDIC algorithms also use only right shifts and additions, minimizing the computation time. It is hardware efficient algorithm because no multipliers are presenting in CORDIC, to save gate required implementing on FPGA. If multiplier is present, then cost and number of gates increases. The CORDIC algorithm has become a widely used approach to elementary function evaluation where the silicon area is a primary constraint. Pipelined CORDIC architecture is implemented in order to reduce iterative cycle and to increase the clock speed.

David C. Wyld, et al. (Eds): CCSEA, SEA, CLOUD, DKMP, CS & IT 05, pp. 435–440, 2012.
 DOI : 10.5121/csit.2012.2243

This paper is organized as follows. Section 2: Introduces CORDIC Algorithm. Section 3: Describes the Pipelined architecture. Section 4: Discuss the simulation and result Section 5: Conclusion.

2. CORDIC ALGORITHM

The basic idea of CORDIC is to rotate the vector over given angle. Each basic rotation is realized by using shift and add operations. A vector is rotated through fixed number of steps called as iterations. If a vector v having co-ordinates (x and y) is rotated through an angle ϕ then obtaining a new vector with co-ordinates where x' and y' can be obtained using following method.

$$X = r\cos\theta\ ,\ Y = r\sin\theta \tag{1}$$

$$V' = \begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} x\cos\phi - y\sin\phi \\ y\cos\phi + x\sin\phi \end{bmatrix} \tag{2}$$

Micro rotation ϕi is performed by vector at each iteration i, so new vector is given by

$$x_{i+1} = x_i.\cos\phi_i - y_i.\sin\phi_i \tag{3}$$

$$y_{i+1} = y_i.\cos\phi_i + x_i.\sin\phi_i \tag{4}$$

Factorizing cos terms vector components given as

$$x_{i+1} = \cos\phi_i(x_i - y_i.\tan\phi_i) \tag{5}$$

$$y_{i+1} = \cos\phi_i(y_i + x_i.\tan\phi_i) \tag{6}$$

As cosine is an even function, so cos (α) = cos (-α).then equation (5) and (6) becomes

$$x_{i+1} = k_i(x_i - y_i d_i 2^{-i}) \tag{7}$$

$$y_{i+1} = k_i(y_i + x_i d_i 2^{-i}) \tag{8}$$

Where i is the number of iteration required by vector to reach the required angle, k factor is given as

$$k = \prod_{i=0}^{n-1} k_i \tag{9}$$

Where k_i is CORDIC gain.

Reducing original given rotation to add shift algorithm given as

$$x_{i+1} = x_i - d_i y_i 2^{-i} \tag{10}$$

$$y_{i+1} = y_i + d_i x_i 2^{-i} \tag{11}$$

A new variable known as accumulator is given as

$$z_{i+1} = z_i - d_i\phi_i \tag{12}$$

$d_i = \pm 1$ (d_i is the direction of angle of rotation)

Where $\phi_i = \tan^{-1} 2^{-i}$ is pre-computed and stored in table for different value of i.

3. PIPELINED ARCHITECTURE

Depending upon the application, CORDIC Processor is implemented in number of ways. The simple architecture is serial architecture consist of three adder/subtractor and two shifter with a rom containing lookup table. Serial architecture perform one micro rotation for every clock cycle. Output is obtained after n clock cycle. Since serial architecture uses n clock cycle for every rotation hence it is very slow. Figure 1 shows the serial architecture.

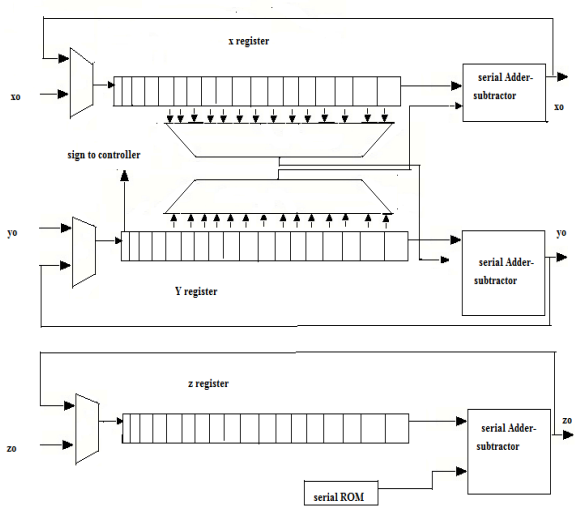

Figure.1. Shows Serial architecture

Pipelined architecture converts iterations in to pipeline phrases. It consists of n cascaded blocks. The first output of n stage CORDIC is obtained after n clock cycle. Thereafter output is obtained after every clock cycle. Pipelined architecture having shift register that perform fixed number of shifts every time. Registers are used to store the angle for a particular micro rotation. Figure 2 shows the pipelined architecture.

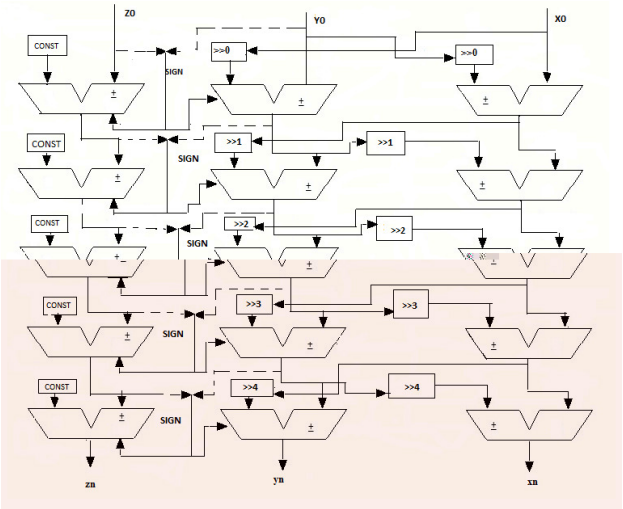

Figure 2. Shows Pipelined architecture

Pipelined architecture is much faster than serial architecture. Sign 'z' gives the direction of iteration at each stage.

In this paper a six stage pipeline sine cosine digital wave generator is developed performing specific micro-rotations. This architecture is fast than serial architecture since it doesn't require any lookup table. It operates in circular rotation mode .Sine and Cosine terms are given by

$$Xn = \cos\theta \quad (13)$$

$$Yn = \sin\theta \quad (14)$$

The RTL view of Pipelined CORDIC is shown in Figure 3.

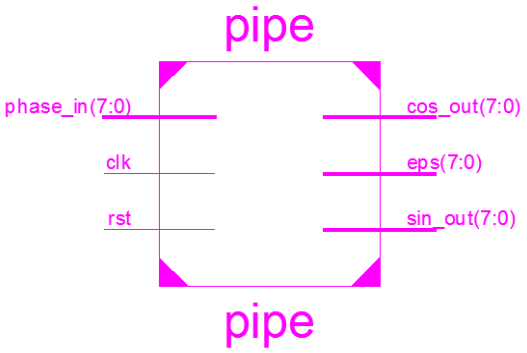

Figure 3. RTL view of Pipelined CORDIC

For any angle as an input three outputs are generated sine function, cosine function and eps.eps gives the error proximity to required angle.

4. SIMULATION AND RESULTS

The code of digital wave generator is written in VHDL and simulated using ModelSim SE 6.3 f. The digital wave generator implemented on XILINX SPARTAN 3 xc3s200-5ft256 FPGA device, using XILINX 12.3. Area and timing reports are given for particular target device. Table 1 shows the device utilization summary of digital wave generator using CORDIC algorithm.

Table1: Hardware Device Utilization Summary

S.NO	Logic Utilization	Used	Available
1	Number of Slice Flip Flops	166	3840
2	Number of Slices	117	1920
3	Number of 4 input LUTs	180	3840
4	Number of bonded IOBs	34	173
5	Number of BUFGMUXs	1	8

Timing reports include total time delay for output to appear after giving input. At speed grade of -5, design operates at maximum frequency of 186.712ns.The minimum period require is 5.356ns.

4.1 Comparison of present Pipelined CORDIC Sine and Cosine wave generator with previous work.

Present work is implemented on XILINX SPARTAN xc3s205ft256 FPGA device. Thus, finally comparison of present work with previous work is done as shown in Table 2.

Table 2: Comparison of Present work with previous Reference paper

Parameters	Present work	Reference Paper[4]
No. of Slices Flip Flops	166	321
No. of Slices	117	196
No. of 4 input LUTs	180	361
No. of bonded IOBs	34	37
Maximum Frequency MHz	186.712	161.65

From Table 2, it is cleared that present design shows an improvement in speed with reduction in used resources on target device.

5. CONCLUSION

In this paper, pipelining technique is implemented on CORDIC digital wave generator which shows the better results in terms of speed and area utilization. The design operates on maximum frequency of 186.712MHz.A Considerable increase in speed made the design suitable for many wireless applications like SDR and GSM. In this work CORDIC has been implemented in pipeline in order to avoid iterative cycle that is output obtained at every clock cycle.

REFERENCES

[1] J.E Volder. “The CORDIC Trigonometric Computing Technique”, IRETrans.Electronic Computers, EC-8:330-334, September 1959.

[2] J.S Walther. “A Unified Algorithm for Elementary Functions”, AIFS Spring Joint Computer Conference, pp.375-385,1971.

[3] P. K. Meher, J. Valls, T. B. Juang, K.Sridhan and K. Maharatna, “50 Years of CORDIC: Algorithms, Architectures, and Applications”, IEEE Transactions on Circuits and Systems, Vol. 56, No.9, pp.1893-1907, 2009.

[4] Rajesh Mehra, Bindiya kamboj“FPGA Based Design of Digital Wave Generator Using CORDIC: Algorithms”,Int .J.Comp.Tech.Appl,Vol1 (1)54-58.

[5] Dobbs, M. Pascale, “Using CORDIC methods for Computations in micro-controllers”, No. 9, 2000.

[6] R. Andraka, “A survey of CORDIC algorithm for FPGA based computers”, 6th International Symposium on Field Programmable Gate Arrays, No. 2, pp. 191-200, 1998

[7] J. Duprat and J. M. Muller, "The CORDIC Algorithm: New Results for Fast VLSI Implementation", IEEE Transactions on Computers, Vol. 42, No. 2, pp.168-178, 1993.

[8] M. Ercegovac and L. Tomas, "Redundant and On-Line CORDIC: Application to Matrix Triangularization and SVD", IEEE Transactions on Computers, Vol. 39, No.6, pp. 725-740, 1990.

[9] HerberiDawid, Heinrich Meyr."VLSI Implementation of the CORDIC algorithm using redundant arithmetic," IEEE International symposium on circuits and systems, 1992. ISCAS '92. Proceedings. 1992.

[10] B.Das, S.Banerjee."Unified CORDIC based chip to realize DFT/DHT/DCT/DST," IEE Proc-Comput. Digit. Tech., Vol. 149, No. 4, July 2002.

[11] J. G. Proakis and D. G. Manolakis,.Digital signal processing principles, algorithms and applications, Delhi: Prentice Hall, ed. 2nd, 2008.

[12] J. A. Lee, T. Lang, "SVD by constant –factor Redundant CORDIC",10th IEEE Symposium on Computer Arithmetic, No. 6, pp.264-271, 1991.

Authors Biography

Balwinder Singh has obtained his Bachelor of Technology degree from National Institute of Technology, Jalandhar and Master of Technology degree from University Centre for Inst. & Microelectronics (UCIM), Punjab University, Chandigah in 2002 and 2004 respectively. He is currently serving as Senior Engineer in Centre for Development of Advanced Computing (CDAC), Mohali and is a part of the teaching faculty and also pursuing Phd from GNDU Amritsar. He has 7+ years of teaching experience to both undergraduate and postgraduate students. Singh has published three books and many papers in the International & National Journal and Conferences. His current interest includes Genetic algorithms, Low Power techniques, VLSI Design & Testing, and System on Chip.

Navdeep Prashar received the B.Tech. (Electronics and Communication Engineering) degree from the CT Institute of Engineering, Management and Technology, Jalandhar affiliated to Punjab Technical University, Jalandhar in 2010, and presently he is doing M.Tech. (VLSI design) degree from Centre for Development of Advanced Computing (CDAC), Mohali and working on his thesis work. His area of interest is Embedded Systems, VLSI Design and Digital Signal Processing etc.