International Journal of Electronics Signals and Systems

Volume 3 | Issue 1 Article 9

July 2013

FPGA IMPLEMENTATION OF CORDIC ALGORITHM FOR FINGER PRINT RECOGNITION APPLICATIONS

POGIRI REVATHI
Dept. of ECE, GMR Institute of Technology, Rajam, India, grlocharla89@gmail.com

M.V. NAGESWARA RAO
Dept. of ECE, GMR Institute of Technology, Rajam, India, nageswararao.mv@gmrit.org

GOVINDA RAO LOCHARLA
Dept. of ECE, GMR Institute of Technology, Rajam, India, govindarao.l@gmrit.org

FPGA IMPLEMENTATION OF CORDIC ALGORITHM FOR FINGER PRINT RECOGNITION APPLICATIONS

POGIRI REVATHI[1], M.V.NAGESWARA RAO[2] & GOVINDA RAO LOCHARLA[3]

[1,2&3]Dept. of ECE, GMR Institute of Technology, Rajam, India
E-mail : grlocharla89@gmail.com, nageswararao.mv@gmrit.org & govindarao.l@gmrit.org

Abstract - Finger print recognition process involves many trigonometric evolutions. CORDIC based evolution of trigonometric functions is possible. In this paper CORDIC based architecture is proposed to evaluate the trigonometric or inverse trigonometric functions like sinθ, cosθ. Proposed architecture is partitioned into two main blocks: Magnitude generator and CORDIC processor. Magnitude generator outputs the value of the trigonometric function by driving the CORDIC processor. CORDIC processor executes the CORDIC rotations. Architecture Proposed in this paper is implemented using XILINX 13.2. Performance of the architecture is analyzed by calculating the relative error and resource utilization summery is reported.

Keywords- *CORDIC, FPGA, SIN, COS, XILINX 13.2*

I. INTRODUCTION

In a fingerprint, small local neighborhood ridges, valleys structure presents a well-defined orientation and a spatial local frequency between two consecutives ridges. These properties are the key for removing the undesired noise preserving the original ridge structure. Gabor filters are employed for removing the noise.

A two-dimensional Gabor filter consists of a sinusoidal wave with a particular orientation and frequency modulated by a Gaussian envelope. Evolution of the trigonometric functions can be done in terms of additions and shifts according to the CORDIC Algorithm.

II. REVIEW OF CORDIC ALGORITHM

CORDIC is an acronym for Coordinate Rotation Digital Computer. It is a trigonometric algorithm, computes the trigonometric functions using vector rotations. Inter conversion between polar to rectangular and rectangular to polar coordinates is possible with vector rotation. This algorithm [1] derived from the general rotation transform.

$$x' = x\cos\phi - y\sin\phi \quad \ldots (1)$$

$$y' = y\cos\phi + x\sin\phi \quad \ldots (2)$$

Above equations calculates the transformed Cartesian coordinates after the vector rotation by angle of ϕ degrees. Equation (1) and (2) can be written as follows

$$x' = \cos\phi[x - y\tan\phi] \quad \ldots (3)$$

$$y' = \cos\phi[y + x\tan\phi] \quad \ldots (4)$$

If the rotation angle can be restricted so that $\tan\phi = \pm 2^{-i}$, the multiplication by $\tan\phi$ can be reduced to a simple shift operation. Arbitrary angle can also be expressed in terms of these known restricted angles. So a vector rotation by an arbitrary angle can be achieved by successive elementary rotations by known angles. In equation (3) and (4), $\cos\phi$ is constant and these can be rewritten as follows:

$$x_{i+1} = K_i[x_i - y_i * d_i * 2^{-i}] \quad \ldots (5)$$

$$y_{i+1} = K_i[y_i + x_i * d_i * 2^{-i}] \quad \ldots (6)$$

$$\phi_{i+1} = \phi_i - d_i * \tan^{-1}(2^{-i}) \quad \ldots (7)$$

where

$$K_i = \cos(\tan^{-1}2^{-i}) = 1/\sqrt{1+2^{-2^i}} \quad \ldots (8)$$

$$d_i = sign(\phi_i) \quad \ldots (9)$$

III. IMPLEMENTATION OF CORDIC AlGORITHM

A. Phase angle mapping

Phase angle and sine magnitude are represented in this implementation with 16 bits, scaled value of $360°\ is\ 2^{16}$

$$\therefore 1° = 2^{16}/360° = 182. \quad \ldots (10)$$

B. Magnitude scaling

Magnitude is represented with 16 bits, out of these MSB is used to represent sign and rest 15 bits are used to represent the magnitude. Output represents the values in the 0 to +1 range. The results can be derived as follows:

$\sin 0 = 0$ is represented by the minimum values possible with 15 bits.

$\sin 90 = 1.0$ is represented by the maximum values possible with 15 bits, i.e., 2^{15}.

Therefore output scaling factor $=1/2^{15}$

IV. ARCHITECTURE

CORDIC processor, Fig.2, consists of CORDIC pipe and CORDIC rotation module. Pipe line processor executes the iterations specified in the pseudo code and CORDIC rotation module performs the vector rotation by the angle specified by the pipeline processor.

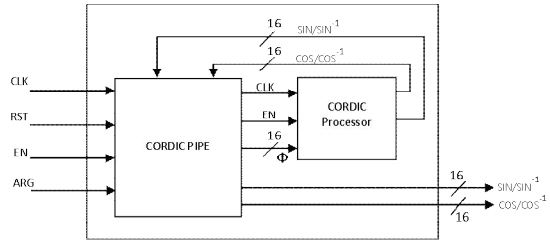

Fig.2 CORDIC Processor

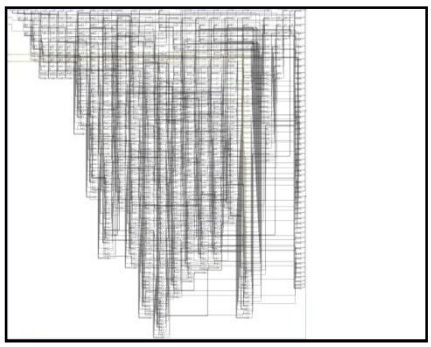

Fig.3 Technology Schematic of CORDIC Processor

V. PERFORMANCE ANALYSIS

Performance of the architecture is measured in terms of the relative error. We define the error as the distance between the ideal value and the practical value divided by the ideal value:

$$\varepsilon = \left| \frac{function_{pc} - function_{cordic}}{function_{pc}} \right|$$

From relative errors calculated for few points are tabulated in TABLE I, It can be observed that the trigonometric evolution done by the proposed architecture is very closer to the idle values.

TABLE.I RELATIVE ERROR MEASUREMENT

S.No	Performance measure for sinθ evolution			
	θ	$sin\theta_{Theory}$	$Sin\theta_{FPGA}$	ε
1	22.5	0.38268	0.37162	0.289
2	45	0.70710	0.70699	0.0001
3	67.5	0.92388	0.92308	0.0008
4	90	1.00000	0.97831	0.0216
	Performance measure for Cosθ evolution			
	θ	$cos\theta_{Theory}$	$cos\theta_{FPGA}$	ε
1	0	1.00000	0.98867	0.0113
2	22.5	0.92388	0.91478	0.0091
3	45	0.7071	0.7022	0.0049
4	67.5	0.38268	0.38157	0.0011
	Performance measure for $sin^{-1}x$ evolution			
	x	$Sin^{-1}x_{Theory}$	$Sin^{-1}x_{FPGA}$	ε
1	0.38268	22.5	21	0.2222
2	0.70710	45	45.8	0.0177
3	0.92388	67.5	66.9	0.0088
4	1.00000	90	89.6	0.4
	Performance measure for $Cos^{-1}x$ evolution			
	x	$Cos^{-1}x_{Theory}$	$Cos^{-1}x_{FPGA}$	ε
1	1.15286	0.02012	0.02034	0.019
2	0.92388	22.5	22.6021	0.00453
3	0.7071	45	45.0135	0.0003
4	0.38268	67.5	67.4872	0.0128

CORDIC architecture evaluates the trigonometric functions in terms of additions and shifts. Resource utilization of CORDIC architecture is summarized in TABLE II.

TABLE. II DEVICE UTILIZATION SUMMERY

ARCITECTURE FOR SIN'X/COS'XEVOLUTION		
S.NO.	LOGIC UTILIZATION	UTILIZED HARDWARE
1.	NUMBER OF SLICES	232
2.	NUMBER OF SLICE FLIP FLOPS	109
3.	NUMBER OF 4 INPUT LUTS	441
4.	NUMBER OF BONDED IOBS	87
ARCITECTURE FOR SINΘ/COSΘ EVOLUTION		
S.NO.	LOGIC UTILIZATION	UTILIZED HARDWARE
1.	NUMBER OF SLICES	386
2.	NUMBER OF SLICE FLIP FLOPS	750
3.	NUMBER OF 4 INPUT LUTS	746
4.	NUMBER OF BONDED IOBS	70

VI. CONCLUSION

Performance metrics and resource utilization counts are loudly saying that CORDIC can evaluate the trigonometric functions accurately with minimal hardware. Therefore, CORDIC architecture can be adapted in the ASIC implementations used for the applications like finger print recognition.

REFERENCES

[1] Ray Andraka, "A Survey of CORDIC Algorithms for FPGA based Computers", Andraka Consulting Group, Inc, North Kingstown, RI02852.

[2] Nihel NEJI, Anis BOUDABOUS, Wajdi KHARRAT, Nouri MASMOUDI, TUNISIA, "Architecture And Fpga Implementation of the CORDIC algorithm For Fingerprints Recognition Systems", 2011 8th International Multi-Conference on Systems, Signals & Devices,2011,IEEE

[3] P.K. Meher, J. Valls, T.B. Juan, K. Sridharan and K. Maharatna, "50 Years of CORDIC: Algorithms, Architectures and Applications", IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—I: REGULAR PAPERS, vol. 56, no. 10, pp. 9, Sept. 2009

[4] M. D. Ercegovac, and T. Lang, "Redundant and On- Line CORDIC: Application to Matrix Triangularization and SVD", IEEE Trans. Comput., vol. 39, no. 6, pp. 725{740, June 1990

[5] E. Antelo, J. Villalba, J. D. Bruguera, and E. L.Zapata, "High performance rotation architectures basedon the radix-4 CORDIC algorithm", IEEE Trans. Comput., vol. 46, no. 8, pp. 855-870, Aug. 1997

[6] H. Dawid, and H. Meyr, "The differential CORDIC algorithm:constant scale factor redundant implementation without correcting iterations", IEEE Trans. Comput. vol. 45, no. 3, pp.307-318, Mar. 1996

[7] Mariano López, Enrique Cantó and Mariano Fons, SPAIN, "Hardware-Software Co-design of a Fingerprint Image Enhancement Algorithm"

[8] K.Kota, J.R. Callaro, "Numarical accuray and Hardware Tradeoffs for CORDIC Arithmatic for Special purpose Processors", IEEE Transactions on Computers, Vol. 42, No.7, 1993, pp769-779

[9] Stefan Lachowicz and Hans-Jorg Pfleiderer, "Fast evaluation of the square root and other nonlinearfunctions in FPGA", *4th IEEE international symposium on electronic design*, 2008

[10] Anis BOUDABOUS, Fahmi GHOZZI, M. Wajdi KHARRAT, Nouri MASMOUDI, "Function Generator Based CORDIC Algorithm", *Third international conference on systems, signals & devices*, vol. 4, March2005.