International Journal of Modern Electronics and Communication Engineering (IJMECE) ISSN: 2321-2152
Volume No.-7, Issue No.-3, May, 2019

Implementation of Enhanced Range Exponent function for Xilinx CORDIC IP Core

Nalini Bodasingi 1st
Department of ECE
JNTUK-UCEV
Vizianagaram, India
E-mail: nalinib.ece@jntukucev.ac.in

Balaji Narayanam 2nd
Department of ECE
JNTU-UCEN
Vizianagaram,India
E-mail: narayanambalaji@rediffmail.com

Abstract: In this paper, Hardware architecture of Exponential function using CORDIC algorithm is proposed with extended range. The main application of the proposed exponential function is its use in different KERNAL functions of machine learning algorithms. This IP core is user configurable for different input ranges to match different KERNAL function implementations in hardware. The proposed Exponential function with extended range (EFER) is tested in this paper for an radial basis (RBF) KERNAL function of Support Vector Machine (SVM) Classifier. This architecture also supports different output accuracies by providing 7Q32, 7Q64, and 7Q128 format. The latency of the proposed architectures is observed to be very less with the values 250ns for multiple values of input with 0.75 when compared to conventional CORDIC IP Core. For other values, the latency slightly increased to support a wider range.

Keywords: About six key words separated by commas (Minimum 4 key words)

I. INTRODUCTION

Computation of the exponential value has become a key operation in digital signal processing(DSP), artificial neural networks, computer graphics, multimedia and real time applications [1] etc. In order to compute the exponential function, there is a need of dedicated hardware blocks to meet the timing constraints.

In the hardware implementation of the exponential block using the CORDIC algorithm, many works were done. The increase in the extra iterations will expand the convergence range of the CORDIC algorithm [2]. But, one disadvantage is that the computation time will be increased. With the use of arrays and trees, the algorithm is implemented by the parallel approach [3]. Here, the resource utilization is a major drawback to implement the approach, the Dadda and Wallace multipliers are to be used for the implementation of any elementary function in CORDIC algorithm.

A new algorithm was proposed called BKM algorithm [4], which is suitable for redundant number system for complex elementary functions like sin cos exp and ln functions. Hardware architecture of BKM was more complicated. To obtain 16 significant bit, requires 16 iterations, requires 30 (2n-2) position shifts and needs the storage of 72 9/2(n) constraints.

This hardware architecture is more compatible with high frequency applications [11], drawback is that little bit of error occurs, when compared with ideal value. To expand the hyperbolic (exponential) function by using CORDIC algorithm for a fixed point implementation [11], requires additional hardware. Disadvantage is the clock rates are relatively high.

In this work, the implementation of an exponential block in hardware is described for the computation of the Radial Basis Function (RBF) kernel in SVM classification algorithm [10]. In order to compute this kernel, there is a necessity to consider a large range of inputs which are negative indices. This laid the foundation for the development of exponential function architecture for the computation of an extended range of input argument. Fixed point number format is used for the present implementation so that resource utilization is minimized on target FPGA [3,9].

The preparing of this paper is organized as follows: In section 2, The description of the CORDIC algorithm and hyperbolic function are discussed, the enhanced range of CORDIC range hyperbolic function along with the digital architecture and its outcome are presented in section 3. In the following section 4, the simulation results have been presented with latency analysis. Finally, in section 5 conclusion of this work is presented.

II. CORDIC ALGORITHM

Coordinate **R**otation **DI**gital **C**omputer is shortened as CORDIC. J.E.Volder is the first person to give this computational algorithm[1]. To resolve a simple shift-add iterative procedures is the key role of CORDIC algorithm. There are different types of equations that can be implemented using CORDIC algorithm such as rectangular (polar conversion), trigonometric, hyperbolic, square wave.

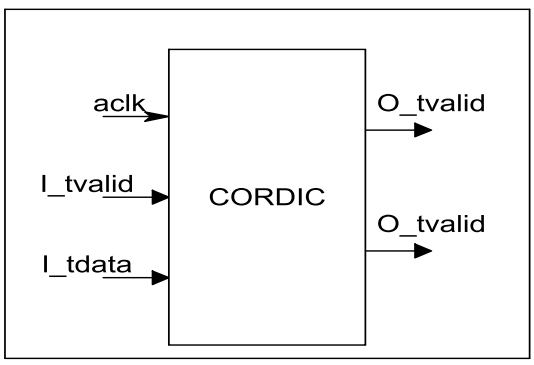

Figure1. CORDIC Block.

Table 1: The pin Description of CORDIC

Port	In/Out	Description
Aclk	In	Synchronous clock with rising edge.
I_valid	In	Handshake signal for channel input
I_data	In	This port has one subfield, input. The subfield is Input width bits wide, padded to the next byte width.
O_valid	Out	Handshake signal for channel output.
O_data	Out	Each subfield is Output width bits wide, padded to the next byte width before concatenation.

The pin Diagram of Xilinx CORDIC IP Core is shown in Figure 1. This CORDIC block consists of aclk, I_tvalid, I_tdata, O_tvalid, O_tdata ports as described in Table 1. The architectural configuration of CORDIC core are available in following ways.

- A fully parallel configuration with single cycle data throughput at the expense of silicon area.
- A word serial implementation with multiple cycle throughput but occupying a small silicon area.

III. HYPERBOLIC FUNCTION

The Sinh and Cosh functional configuration in CORDIC algorithm are used to represent the movement of the vector through hyperbolic angle P. In order to define the area under the vector with the logarithm we use the hyperbolic angle and it is not related with the trigonometric angle which generates the output vector. The input of hyperbolic function, Pin, is expressed as fixed point 2's complement number with the format of 2QN (3 integer width and (N-3) fraction bit width, in 3 integer width 1st bit represent sign). The output of hyperbolic function is the summation of Sinh (X_Out) and Cosh (Y_Out). Output is expressed as a pair of fixed point 2's complement number in the format of 1QN (2 integer width and (N-2) fraction width, in 2 integer width 1st bit represented as sign).

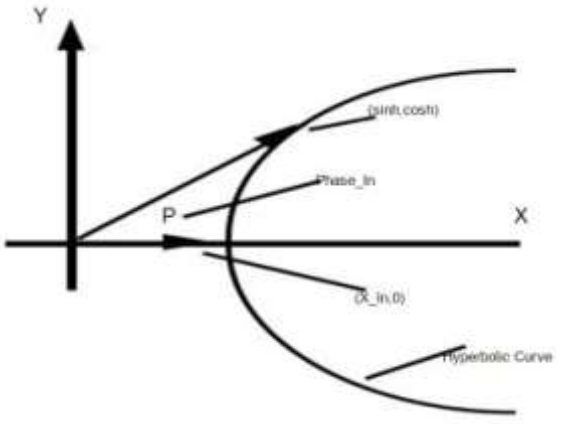

Figure2. Hyperbolic Function Representation of sinh and cosh function

The input range of CORDIC of Xilinx core IP is in between $-\pi/4$ to $\pi/4$, that means The input hyperbolic angle phase in signal range is $-\pi/4$ to $\pi/4$. The output is the combination of Sinh and Cosh. The output range of Cosh is in between 1 to 2, and the output range of Sinh is in between -2 to +2. The output of exponential function in CORDIC is in the form of Eq. (1).

$$Exp(X) = Sinh(X) + Cosh(X) \qquad (1)$$

IV. ENHANCED RANGE OF EXPONENTIAL FUNCTION

The design of efficient parallel hardware architecture of RBF kernel based SVM classifier is done by proposing its pin diagram and pin discerption as presented in Figure 3 and Table 3. Here Table 3 gives the details about the ports Clock, Reset, X_input, Xin_En, Exp_out, expout_En.

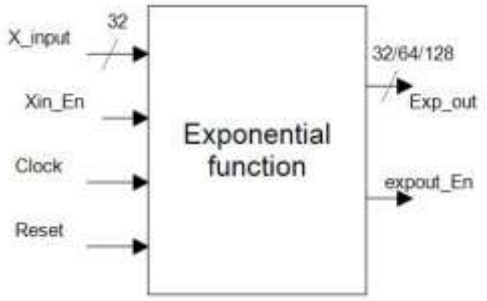

Figure3. The Pin Diagram of EFER

Table 2: The Pin Description of EFER Function

Port	In/ out	Description
X_input	In	This port has one subfield, input. The subfield is Input width bits wide, padded to the next byte width.
Xin_En	In	Handshake signal for channel input
Clock	In	Active rising edge clock for synchronization.
Reset	In	Initialize the hardware
Exp_out	Out	Final Exponential extended output
Expout_ En	Out	Handshake signal for channel output.

V. PROPOSED VLSI ARCHITECTURE OF EFER BLOCK

The input of the enhanced range of exponential function is positive (range in between 0 to 66.75), The architecture works to get both negative and positive values of exponential with its sign bit (that means the range is either exp $^{(0 \text{ to } -66.75)}$ or exp $^{(0 \text{ to } -66.75)}$). Figure 4 presents the proposed EFER hardware architecture, Where, the design enhances the range from 0 to Value 'X'. EFER consists following blocks subtractor, counter, RAM, multiplier, CORDIC, and shifter. Here, the input X_input is given to the subtractor which is of 32-bit word length. The input value is subtracted from 0.75. Here, two possible cases are considered for subtraction. Figure 5 presents algorithmic flow for remainder equal to 0 (Case I) and Figure 6 provides the flow for remainder not equal to 0 (Case II). If the remainder is 0 the counter counts the number of

times the input values is subtracted from 0.75 in the subtractor block. In the RAM, the multiples of 0.75 are stored up to exponential value of 'X'. This count of multiple values of 0.75 is given as input to the RAM where, the exponent values stored for multiple of 0.75. For example, if the counter value is 7, the output of RAM is the exponential value of (0.75x7=5.25). The output is in the format of 7Q32. 32 is the word length, 8 is the integer length, 24 is the fractional length. In the 8-bit integer, 1st bit is the sign bit.

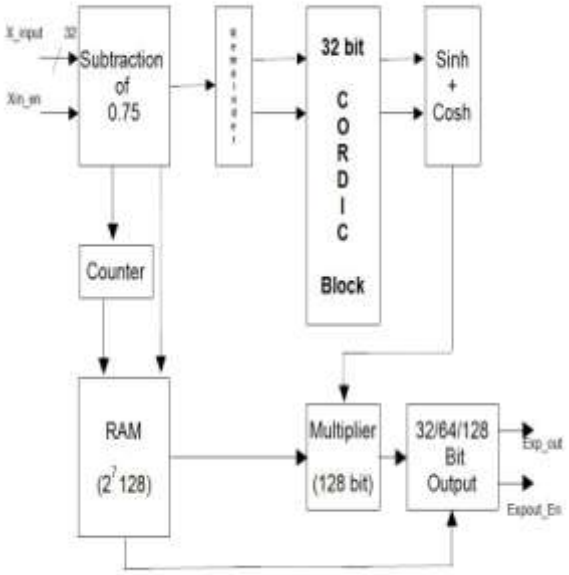

Figure4. VLSI Architecture of Extended Exponential Function.

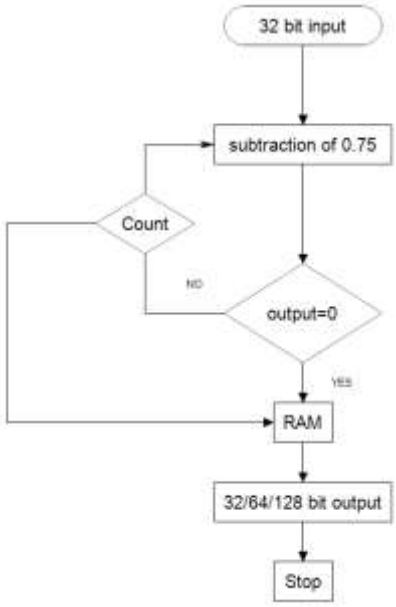

Figure5. Hardware flow to implement multiples of 0.75.

If the remainder is not equal to 0, the above procedure repeats except the remainder value. This remainder value goes as an input CORDIC block, the output from CORDIC is multiplied with exponent value from RAM to give the final value for extended range. The input of CORDIC is the 2's complement of remainder in the format of 2Q32 (3 integer width and 29 fraction bit width, 1st bit represents sign).

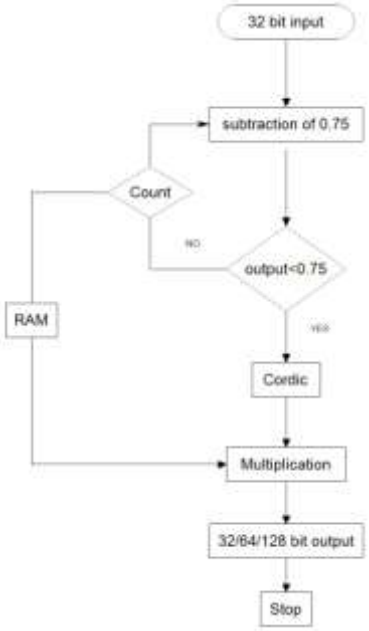

Figure5. Hardware flow to implement multiples other than 0.75.

The output of CORDIC is the summation of Sinh and Cosh which is the exponential of fixed point 2's complement number of remainder block. The output of CORDIC is multiplied with the output of RAM. The output can be taken from the RAM in the format of 7Q32, 7Q64 and 7Q128 bit.

$$RAM = exp^{-n(0.75)} \quad n = 1,2, \dots \dots \dots n \qquad (2)$$

For example, if the input value is 3, four times 0.75 is subtracted from input, and the count stores the value of four and saved in RAM. The output of the RAM shown in equation 3.

For n=4

$$exp^{-n(0.75)} = \exp^{-3} \qquad (3)$$

The output of extended range of exponential function is taken in 3 formats.

A. *7Q32 Format*

In this format, it is in the form of fixed point data type mode. 32 is the word length, 8is the integer length. In the 8-bit integer bit, the first bit is the sign bit.

B. *7Q64 Format*

In this format, it is in the form of fixed point data type mode. 64 is the word length, 8is the integer length. In the 8-bit integer bit, the first left most bit is the sign bit.

C. *7Q128 Format*

In this format, it is in the form of fixed point data type mode. 128 is the word length, 8is the integer length. In the 8-bit integer bit, the first bit is the sign bit.

VI. EXPERIMENTAL RESULTS

The input and output of the CORDIC block shown in Figure 5 and Figure 6 respectively. The input Data (-0.75) is in the format of 2Q32, in this format 32 represent as word length, and in the 32-bit word length, the left most 1st bit represent as sign, and 3 bits are integer part, and remaining 29 bits are fraction bit length.

The output data exp (-0.75)) is the combination of Sinh and Cosh. The output has 64-bit length, 32-bits for Cosh and another 32 bits for Sinh. The Cosh and Sinh outputs are in the format of 1Q32, in this format 32 represent as word length, and in the 32-bit word length, the left most 1st bit represent as sign, and 2 bits are integer part, and remaining 30 bits are fraction bit length. The output value of CORDIC is arrived after 16 clocks.

Figure5. Test Vector to show the input X Value

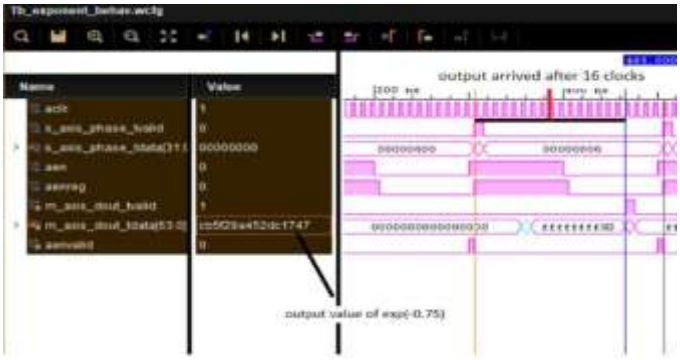

Figure6. Test Vector to show the Output for the value -0.75 Value

The input and output of the enhanced range of Exponential Block Shown in figure (6) and figure (7) respectively. The input data (1.7) in the format of 1Q32, in this format 32 represent as word length, and in the 32-bit word length, the left most 1st bit represent as sign, and 8 bits are integer part, and remaining 24 bits are fraction bit length. Positive value (+1.7) is given to the enhanced range of exponential block, the output of this block gives negative value of the exponential function($e^{-1.7}$).

The output value ($e^{-1.7}$) have 32-bit length in the format of 7Q32, in this format 32 represent as word length, and in the 32-bit word length, the left most 1st bit represent as sign, and 8 bits are integer part, and remaining 24 bits are fraction bit length. The output value of CORDIC is arrived after 39 clocks.

The input and output of the enhanced range of Exponential Block Shown in figure (8) and figure (9) respectively. The input data (22.12548673) in the format of 1Q32, in this format 32 represent as word length, and in the 32-bit word length, the left most 1st bit represent as sign, and 8 bits are integer part, and remaining 24 bits are fraction bit length. Positive value (+22.12548673) is given to the enhanced range of exponential block, the output of this block gives negative value of the exponential function($e^{-1.7}$).

The output value ($e^{-22.12548673}$) have 64-bit length in the format of 7Q64, in this format 64 represent as word length, and in the 64-bit word length, the left most 1st bit represent as sign, and 8 bits are integer part, and remaining 56 bits are fraction bit length. The output value of CORDIC is arrived after 71 clocks.

Table 3 gives the hardware utilization of Xilinx Artix 7 FPGA for 32, 64, and 128-bit formats. It shows the area increase slightly in the architecture for higher precision. For a 100MHz clock the latency report is depicted in Table 4, Here the multiples of 0.75 value always have less latency and for others, it increases to support extended values.

Figure7. Test Vector to show the Output for the value -1.7 Value

Figure8. Test Vector to show the Output for the value -22.12548673 Value

Table 3: Hardware Utilization of extended range of exponential function

Logic Utilization	32-bit	64-bit	128-bit
Number of Slice LUTs	5699	5731	5797
Number of Slice Registers	4556	4582	4650
Number of Bonded IOB's	68	100	164

Table 4: Latency report for different Phase input values

Bit size	32 bit	64-bit	128-bit
Cycles(clocks) (exp(1.7))	44	44	44
Latency(ns)	440	440	440

(exp(1.7))			
Cycles(clocks) (exp(15))	24	24	24
Latency (ns) (exp(15))	240	240	240

VII. CONCLUSION

Minimizing the area of the hardware architecture and accelerating the computation time are two prominent features in computing the monadic function such as exponential function. In order to fulfil these two features, hardware implementation should be properly maintained. The proposed work, for achieved good results in reduction of the hardware architecture of exponential function using the CORDIC algorithm, the input range is very limited, which is not suitable for hardware architecture like RBF kernel of SVM classifier. In our proposed work, we augmented the input range of CORDIC algorithm from 0 to -66.75 to implement exponential function.

In our proposed method, RBF kernel based SVM classifier is used to construct the hardware architecture of exponential function with large range input. Subsequently, we attained the elevated input range of CORDIC algorithm for exponential function implementation and the area of the architecture is reduced.

ACKNOWLEDGMENT

The authors would like to thank department of ECE, JNTUK UCEV, Vizianagram, India for providing material and required Xilinx Tools.

REFERENCES

[1] J.-A. Pineiro, M. D. Ercegovac, and J.D. Bruguera,"Algorithm and Architecture for Logarithm, Exponential, and Powering computation," IEEE Transactions on Computers, vol. 53, no. 9, 2004.

[2] X. Hu, R. G. Harber, and Steven C.Bass, "Expanding the range of convergence of the CORDIC algorithm" IEEE Transactions on Computers, vol. 40, no. 1, 1991.

[3] Shradha Gupta, Sumeet Saurav, Sanjay Singh, Anil K Saini, Ravi Saini, "VLSI architecture of exponential block for non-linear SVM classification," IEEE International Conference on Advances in Computing, Communications and Informatics (ICACCI), 2015.

[4] J.-C. Bajard, S. Kla, and J.-M. Muller,"BKM: A New Hardware Algorithm for Complex Elementary Functions," in Proceedings of the 11th IEEE Symposium on Computer Arithmetic, 1993.

[5] V. Kantabutra,"On Hardware for Computing Exponential and Trigonometric Function," IEEE Transactions on Computers, vol. 45, no.3, 1996.

[6] A. Naseem, and P. David Fisher,"The Modified CORDIC Algorithm," in Proceedings of the 7th IEEE Symposium on Computer Arithmetic, 1985.

[7] M. Arora, R. S. Chauhan, and L. Bagga,"FPGA Prototyping of Hardware Implementation of CORDIC Algorithm," International Journal of Scientific and Engineering Research, vol. 3, no. 1, 2012.

[8] J. E. Volder, "The CORDIC trigonometric computing technique," IRE Trans. Electron. Comput., vol. EC-8, no. 3, 1959.

[9] Jammu BR, Patra SK, Mahapatra KK, "VLSI architecture of reduced rule base inference for run-time configurable fuzzy logic controllers". 7th International conference on bio-inspired computing: theories and applications. Springer Verlag, Heidelberg, pp 77–88, 2012.

[10] N. Bodasingi and N. Balaji, "Classification of multiple diseases based on wavelet features," in *The Journal of Engineering*, vol. 17, no. 4, pp. 110-118, 4 2017.

[11] Anurup Saha, K Gaurav Kumar, Archisman Ghosh, and Mrinal Kanti Naskar, "Area efficient architecture of Hyperbolic functions for high frequency applications," International Conference on Circuits, Controls, and Communications (CCUBE), 2017.

AUTHOR'S BIOGRAPHIES

Mrs. Nalini Bodasingi (b. 1983) did Graduation, Post-Gradulation Degree in ECE from JNTU Kakinada, Kakinada. At present she had been working as an assistant professor in JNTUK- University College of Engineering, Vizianagaram. She has published about 10 research papers and articles in the area of biomedical image processing and it VLSI implementations. Her areas of interest are digital signal, image processing and FPGA implementations.

Dr. Narayanam Balaji (b.1975) obtained his B.E (ECE) Degree from Andhra University, Visakhapatnam, M.E in Systems and Signal Processing and Ph.D in Radar Signal processing from Osmania University, Hyderabad. He started his career as a Lecturer in the Department of ECE in VNR Vignana Jyothi Institute of Engineering and Technology, Hyderabad from August 1997 to December 2003. Then he was promoted as an Associate Professor in the same department of ECE from 2003 to 2011, at VNRVJIET. Later he was elevated as a Head of Research and Consultancy Centre on 1st April 2004 and promoted as professor from October 2011 to December 2012.

He had joined as a professor of ECE in JNTU Kakinada University College of Engineering Vizianagaram (UCEV) in January 2013. He was promoted as Head of ECE Department in January 2016 at UCEV. In June 2016, he was elevated as Vice- Principal (Administration) of JNTU Kakinada University College of Engineering Narasaraopet (UCEN) till date.

He is a member of ISTE, and Fellow of IETE, Executive Member of AP state Knowledge mission, Government of Andhra Pradesh. He has presented and published 42 research papers in National and International Conferences/Journals. He is presently guiding 6 research scholars. His areas of research interest are VLSI, Signal and Image Processing. He received grants of worth 48 lakhs from various funding agencies like DeitY and AICTE. Under his guidance research scholars submitted various proposals and got grants. Three patents of Dr.N.Balaji got published in "The Patent Office Journal", Issue No. 08/2015 dated 20/02/2015 and are still awaiting examination.