International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

COPY RIGHT

ELSEVIER
SSRN

IJIEMR Transactions, online available on 13th Apr 2022. Link :http://www.ijiemr.org/downloads.php?vol=Volume-11&issue=Issue 04

DOI: 10.48047/IJIEMR/V11/ISSUE 04/99

Title **FPGA REALIZATION OF TRIGONOMETRIC FUNCTIONS USING CORDIC ALGORITHM**

Volume 11, ISSUE 04, Pages: 638-647

Paper Authors
T.KAVYA SREE ,S.NIROSHA ,SK.FARVEEN ,V.PUSHPA ,P.SANDHYA ,Mrs.Dr.D.RAJANI
Mr. Rajesh Pasupuleti

USE THIS BARCODE TO ACCESS YOUR ONLINE PAPER

To Secure Your Paper As Per UGC Guidelines We Are Providing A Electronic Bar Code

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

FPGA REALIZATION OF TRIGONOMETRIC FUNCTIONS USING CORDIC ALGORITHM

First Author:

T.KAVYA SREE B.Tech (ECE) From Sree Venkateswara College Of Engineering, Nellore.

S.NIROSHA B.Tech (ECE) From Sree Venkateswara College Of Engineering, Nellore.

SK.FARVEEN B.Tech (ECE) From Sree Venkateswara College Of Engineering, Nellore.

V.PUSHPA B.Tech (ECE) From Sree Venkateswara College Of Engineering, Nellore.

P.SANDHYA B.Tech (ECE) From Sree Venkateswara College Of Engineering, Nellore.

Second Author:

Mrs.Dr.D.RAJANI, M.Tech,Ph.D, Associate Professor, Department Of ECE, Sree Venkateswara College Of Engineering, Nellore.

Third Author:

Mr. Rajesh Pasupuleti (Ph.D), HOD & Associate Professor, Department Of ECE, Sree Venkateswara College Of Engineering, Nellore

ABSTRACT

CORDIC stands for Coordinate Rotation Digital Computer. The CORDIC algorithm was introduced for the computation of Trigonometric functions, Multiplication, Division, Data type conversion, Square Root and Logarithms. It is a highly efficient, low complexity, hardware efficient algorithm giving a robust technique to compute the elementary functions. The present work focuses on the design principle.FPGA implementation of various trigonometric functions such as Sine, Cosine, Exponential, Inverse Exponential, Arc Tangent, Logarithm, and Polar to Rectangular conversion using the standard coordinate rotation digital computer (CORDIC) algorithm.Traditions implementation of these functions on a FPGA consumes a lot of area and the results of these functions are floating point which is difficult to design. Hence CORDIC algorithm with IEEE 32 bit floating point representation is used in this paper for implementation.CORDIC is an iterative algorithm which can perform the complex functions using the shift and add approach. The serial and pipelined CORDIC architectures configured on a Cyclone IV E Device are compared in terms of Area, Delay, and Power dissipation. Serial CORDIC architecture design has low area whereas; pipeline CORDIC architecture has low latency. It finds the application in graphic processors, digital synchronizer, Real time image processing, and scientific calculators and so on.

1.INTRODUCTION:

The advances in the very large scale integration(VLSI)technology and the advent of ZAADS electronic design automation(EDA)tools have been directing the current research in the are as of digital signal processing(DSP), communications, etc in terms of the style of fast-speed VLSI architectures for real-time algorithms and systems which have applications in the above mentioned areas. The development rate of VLSI technology was

predicted by Gordon Moore and since 1965 newer technologies have been developed by the industry fitting his predicted curve, which was introduced as the so called Moore's law. These advances have provided momentum to the designers for transforming algorithm into architecture.

Many DSP algorithms use elementary functions like logarithmic, trigonometric, exponential, division and multiplication. Two of the ways of implementing these functions are by using table lookup method and through polynomial expansions. The above mentioned methods require large number of multiplications/divisions and additions/subtractions. CO-ordinate Rotation Digital Computer (CORDIC), a special purpose computer to compute many non-linear and transcendental functions, was proposed by Volderin1959.

The functions that can be computed using a CORDIC computer include trigonometric,logarithmic, exponential, hyperbolic, multiplication, division, square root, etc. Though it initially served the purpose of navigation systems, it later became a popular tool to implement several digital systems especially in the areas of digital signal processing, communications, computer graphics, etc.

The simplicity of CORDIC is that it can compute any of the above mentioned functions using shifts and additions which are of the form x±2- i*y. The operating mode and the coordinate system chosen are two key factors to compute the desired functions in the CORDIC. Many signal processing and communication systems operate CORDIC in circular coordinate system and in either of rotation or vectoring modes.

CORDIC APPLICATIONS:

Applications of CORDIC can mainly be seen in the following areas:

1. Matrix decomposition
2. Signal and image processing
3. Communications
4. Computer graphics
5. Robotics

In matrix decomposition, CORDIC Is used to compute QR decomposition (QRD), singular value decomposition (SVD), and Eigen value estimation. In signal and image processing, CORDIC Is used in fixed/adaptive filtering, computing discrete transforms of Fourier basis, image enhancement operation, etc. In communications, CORDIC is mostly used in direct digital synthesis, digital and analog modulation, envelope detection, etc.. In computer graphics and robotics, CORDIC is used in solving direct and inverse kinematics for robot manipulators, 3D vector rotation in computer graphics, etc.

ADVANTAGES:

1. Components need and cost of CORDIC processer is less as only move signs up adders and look-up desk (ROM) are required
2. Variety of gateways needed in hardware execution, such as on an FPGA, is lowest as hardware complexness is decreased in comparison to other processer chips such as DSP multipliers
3. It is relatively easy in design
4. No multiplication and only inclusion, subtraction and bit-shifting function guarantees easy VLSI execution.

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

DISADVANTAGES:

1. Large amount of versions needed for precise outcomes and thus the rate is low and time delay is high.

2. Energy intake is great in some structure types.

3. Whenever a hardware multiplier is available, e.g. in a DSP micro-processor, desk lookup techniques and good old-fashioned power sequence techniques are usually faster than this CORDIC criteria.

2. LITERATURE OF SURVEY

CORDIC BACKGROUND:

Coordinate Rotation Digital Computer (CORDIC) unit has become an essential and inevitable hardware block in modern engineering and scientific applications. It serves many applications such as solving trigonometric and transcendental equations, in digital signal processing (DSP) for Fourier basis based orthogonal transforms, in computer technology for 3D graphics, in digital communication systems for modulation and demodulation of the signals, etc. The conventional CORDIC was first implemented by Volder, in 1959.CORDIC perform by spinning the organize program through continuous prefixed perspectives until the position is decreased to zero. Figure2.1 shows the coordinate rotation of a vector in 2D circular coordinate system.

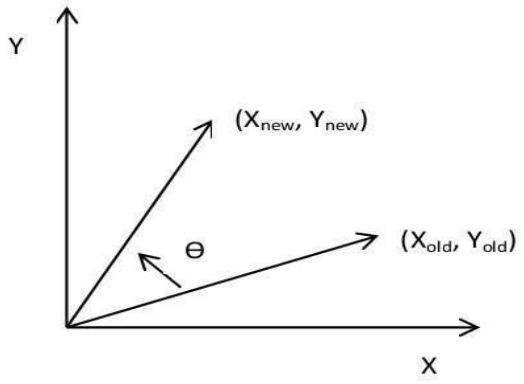

fig. rotation of a vector by an angle in 2d circular coordinate system

Frequency synthesizers, sometimes also called oscillators, are an essential unit of many communication systems. With the maturity of digital systems, communication systems are employing digital subsystems in their units, thus making the usage of digital system more ubiquitous. Direct digital synthesizers (DDS) area class of frequency synthesizers in digital domain, which generate waveforms of desired frequencies. Sometimes also called numerically controlled oscillators (NCO), these generate waveforms like sine, cosine, triangular, square or rectangular, saw tooth, etc. As mentioned earlier, these have wide applications in satellite communication systems, RF signal processing etc.

Many communication systems require quadrature inputs, for example both sine and cosine, for their systems thus bringing in the need of design of DDS which can generate quadrature outputs.

OVERVIEW OF CORDIC:

CORDIC acronym of CO-ordinate Rotation Digital Computer, which is also known as Volder's algorithm and digit-by-digit method, is a simple and efficient method to calculate many functions including trigonometric, hyperbolic, logarithmic, etc. Its main advantage lies in its less hardware over head and reduced latency. It performs complex multiplications using simple shifts and additions. Jack E. Volder described the modern CORDIC algorithm in 1959 for his navigation applications, which consisted of vector rotations.

It can be used in building low complexity finite state CPUs. Further in 1971,J.S.Walther generalized the algorithm by allowing it to

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

calculate functions such as hyperbolic, exponential, logarithms, multiplications, divisions and square-roots. The key usage of CORDIC lies in selecting its one of the operating modes, rotation or vectoring, and one of the coordinate systems, which being either circular, or linear or hyperbolic.

Usage of CORDIC in circular coordinate system results in direct outputs like and through which we can calculate other trigonometric identities. Implementing CORDIC in hyperbolic coordinate system results in functions such as and from which we can get other hyperbolic, logarithmic functions. CORDIC implementation in linear coordinate system results in calculating functions like multiplication, division. Several variants of CORDIC, both scaledand un-scaled, have been designed and among them, scale-free CORDIC architectures have gained popularity in terms of scale-factor compensation.

CO-ordinate Rotation Digital Computer (CORDIC) is an entire- transfer cordic architecture in the computer, containing a unique sequential mathematics unit made up of three shift signs up, three adder-sub tractors, and unique inter connections. It is mostly used to solve the trigonometric relationships that are invo lved in plane organ ize sp inning and

conversion from rectangle-shaped to complete harmonizes and vice versa. Using a pre-determined set of depending improvements or subtractions, the CORDIC

arithmetic can be controlled for solving either set of the equations given below.

$$X_{NEW}=K(X_{old}\cos\theta - Y_{old}\sin\theta)$$

$$Y_{NEW}=KX_{old}\sin\theta - Y_{old}\cos\theta)$$

$$R = K\sqrt{X^2_{old} + Y^2_{old}}$$

$$\theta = \tan^{-1} Y_{old} X_{old}$$

In above set of equations, K is an invariable constant. The above set of equations can be used in calculating the coordinates of the vector (Xnew, Ynew) from the vector (Xold,Yold) which is rotated by an angle of θ in a 2D circular coordinate system.

3. PROPOSED SYSTEM

IMPLEMENTATION EXPONENTIAL AND LOGARITHMIC FUNCTIONS

POWERING FUNCTION

The complexity of the powering function, xy (where x is the base and y the exponent),makes very difficult to implement an efficient and accurate operator in a direct way without any range reduction. However it can be reduced to a combination of other operations and calculated straight forward with the transformation:

$$Z = x^y = e^{y \times \ln x}$$

A direct implementation of this approach with three sub-operators (a logarithm, a multiplier and an exponential) presents three main problems that have to be effectively handled:

1. The enormous complexity of both exponential and logarithm functions. However, the use of table driven methods in combination with range reduction algorithms makes possible their implementation.

2. The computation with a negative base results in Not a Number even though the powering function is defined for negative bases and integer exponents.

3. Equation (7.1) can lead to a large error in the result. Although the sub-operators were almost exact the relative error from each sub-operator spreads through the equation generating the

final large relative error. Extending the precision of the partial results in an effective way to minimize these relative errors.

To the best of our knowledge there are only two previous works focused on the exponential function and only one for the logarithm function (from the same authors of). The first one employs an algorithm that does not exploit the FPGA characteristics, and consequently presents poor performance. The other two implementations are part of a common work and are designed suitings with FPGA flexibility (using internal tailored fixed arithmetic and exploiting the parallelism features of the FPGA) achieving much better results.

They are parameterizable implementations that, additionally to single f.p. format, also allow smaller exponent and mantissa bit-widths and are both based on input range reduction and table-driven methods to calculate the function in the reduced range. Our ex and ln x units, based on these units, include the following innovative features:

Single precision f.p. arithmetic extensions were designed considering only normalized numbers, not denormalized. Additional logic has been introduced to handle denormalized numbers at the output of ex and the input of ln x.

1. Redesign of units to deal only with single precision. The feature of bit-width configurability of the base designs has been removed. Thus, the resources needed have been reduced because specific units, just for single precision, have been developed to Simplification of constant multiplications.
2. As suggested in conventional multipliers have been removed where the multiplications involved constant coefficients, improving performance and reducing size.
3. Unsigned arithmetic. In internal fixed arithmetic with sign is used. However, some operations (like the ones involving range reduction and calculation of the exponent for the result in ex) are consecutive and related, and the sign of the result can be inferred from the Input sign. For such operations signed arithmetic has been replaced by unsigned arithmetic with the corresponding logic reduction.
4. Improved pipelining. The speed is enhanced by systematically introducing pipeline stages to the data path of the exponential and logarithm units and their subunits. The paper explains about the implementation of power and log function based on a simple modification of power series expansion of Taylor series. In power function implementation, the paper aims at reducing the exponent number to a smaller value. It requires a large amount of block ram and hardware multipliers as well. It becomes platform dependent and the clock frequency may vary from vendor to vendor. The degradation in throughput rate is due to the use of 18 X 18 embedded multipliers in it. The powering unitalso requires more number of stages which may be reduced further.

In the proposed method, we are going to reduce delay and improve the throughput rate by avoiding the embedded multipliers and block RAMs. In this paper, we are not completely avoid look up tables, but any value of logarithm or exponential can be calculated, by adjusting the look up table values to the desired number.

PROPOSED METHOD:

The proposed method avoids multiplication and division operations, and is thus appropriate for execution in application on processor chips that

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

absence such guidelines (or where the guidelines are slow) or in components on a automated reasoning system or devoted chip. This method is suitable when shifters are available in abundant. It is an extension to the implementation of sine and cosine explained in CORDIC. The proposed algorithm evaluates the power functions for both positive and negative values. There are some always the same by which it is simple to increase. For example, growing by 2n, where n is a beneficial or a damaging integer, can be carried out by basically moving a variety by n locations. The move will be to the remaining (division) if n is positive, to the right (multiplication) if n is adverse. It is nearly as simple to increase by variety of the form ±2n±1. These simply involve an add(or) subtract a shift.

table : lookup table for values of log function

K	Exp(k)
5.5452	256
2.7726	16
1.3863	4
0.6931	2
0.4055	3/2
0.2231	5/4
0.1178	9/8
0.0606	17/16
0.0308	33/32
0.0155	65/64
0.0078	129/128

IMPLEMENTATION OF EXP:

For implementing y = exp(x). The criteria is going to produce a series of principles

for x and y, and we are going to create sure that for each pair

$$y = \exp(4)$$

$$y' = \exp(4) \cdot \exp(-(x-k))$$

$$= \exp(4) \cdot \exp(-x) \cdot \exp(k)$$

$$= y \cdot \exp(k).$$

In other terms, if we deduct k from x, we have to increase y by exp (k). All we have to do now is create sure that exp (k) is an awesome variety, so we can increase by it quickly, and the relax is uncomplicated. Observe that k itself does not have to be awesome, as we are only subtracting that, not growing by it. Here are some awesome principles of exp (k) and the corresponding (not actually nice) principles of k. The flow of algorithm is as follows for positive powers of x:

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

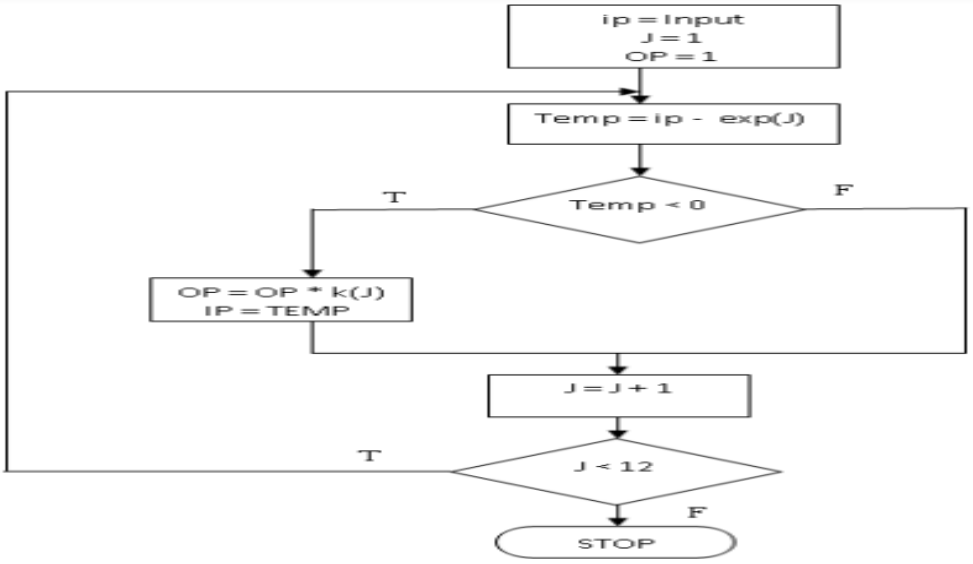

fig : flow chart for logarithmic function

Here in each iteration, we subtract the input from the nearest value of exp(k) as listed in the table. If the difference is negative, we multiply the output by corresponding exp(k). The process continues with more entities in our table of k, finally we get th result. In the same way the flow chart is mentioned for negative powers of x.

K	Exp(k)
5.5452	256
2.7726	16
1.3863	4
0.6931	2
0.2877	¾
0.1335	7/8
0.0645	15/16
0.0317	31/32
0.0157	63/64
0.0078	127/128
0.0039	255/256

table : lookup table for values of power function

The flow of algorthm is as follows for negative powers of x:

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

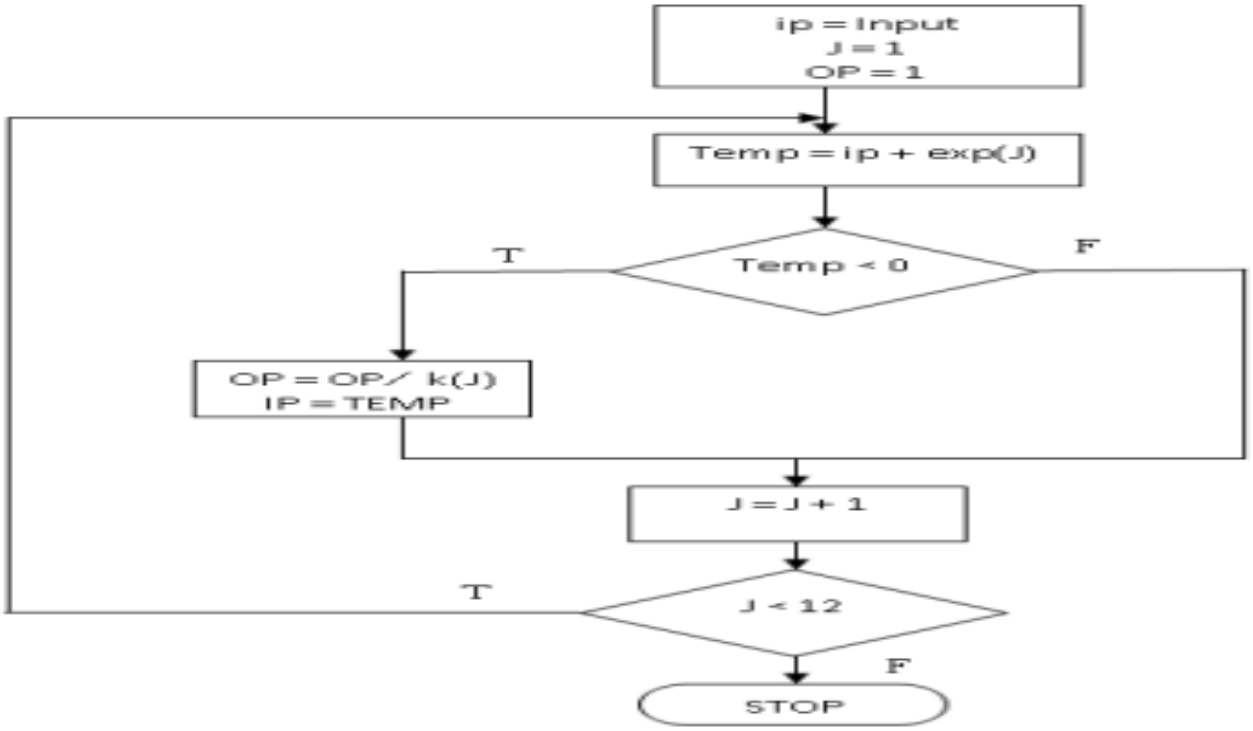

fig : flow chart for power function

Here in each iteration, we subtract the input from the nearest value of exp(k) as listed in the table. If the difference is positive, we divide the output by the corresponding exp(k). The process continues with more entities in our table of k, finally we get the result.

IMPLEMENTATION OF LOG :

For implementing Y=log (x), As for exp(), the requirements is a sequence of concepts for x and y. Now our invariant is

$$\exp(y)\cdot x=54$$

Or

$$y=\log(54/x).$$

Observe that y=log(54/x)=log(1)=0 as needed. Our aim is to get x to 1 while keeping the invariant. Then y will be given by

$$y=\log(54/1)=\log(54)$$

Suppose we increase x by some variety k. Then to sustain the invariant, the new the invariant, the new y value y′ will have to fulfil

$$\begin{aligned} y' &= \log(54/kx) \\ &= \log(54/x)+\log(1/k) \\ &= y-\log(k). \end{aligned}$$

All the k principles in this desk are higher than 1. We will therefore have to begin with x less than 1 so we begin by growing x by 1/256 (other awesome figures would perform too) after we multiply it by the minimum value which produces less than 1 and the corresponding output from the lookup table is added to the previous output. This process goes on till the end of the lookup table.

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

4. RESULTS

Simulation Results

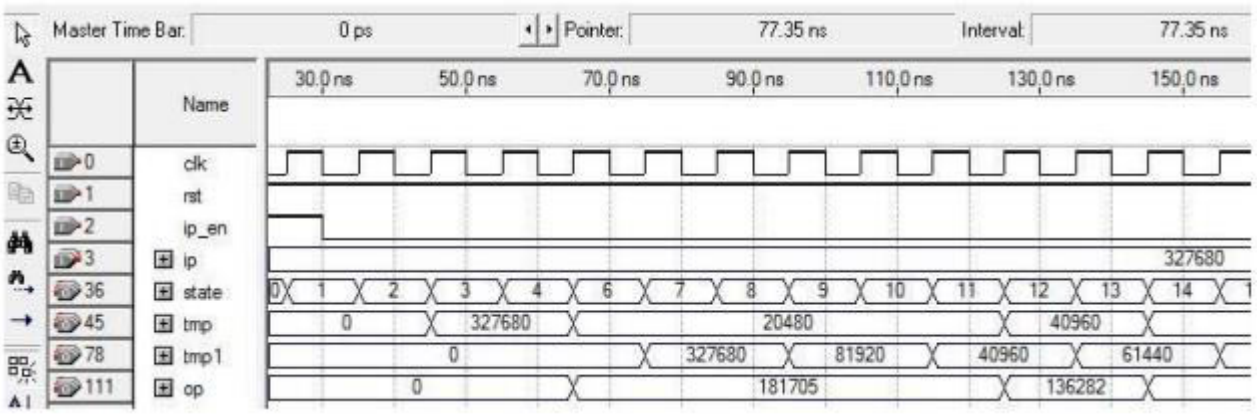

Fig: Simulation Result Of Log Function

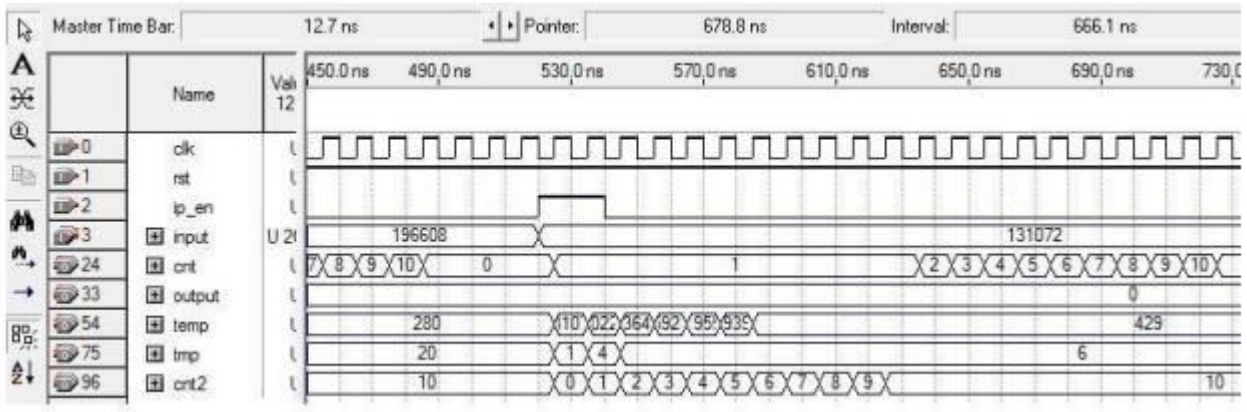

Fig: Simulation Result Of Exp Function

5. CONCLUSION

The advantage of the design proposed in this paper is that no DSP or multiplication blocks are used. As 16 bit precision is used, the accuracy of the design is high. The only disadvantage is that the number of iterations required is slightly more. This block can be used in few decoding algorithms in communication systems.

The design will be used in LDPC decoder sum product algorithm, IMAGE ENHANCEMENT ALGORITHMS the parallel architecture has high throughput (i.e. speed) as compared to serial architecture. While implementing exponential algorithm, we need to multiply input by 65536 to ensure the floating point number converts to a fixed point number. Here we are going to truncate the value to the nearest value.

6 .REFERENCES

[1] J. E. Volder, “The CORDIC trigonometric computing technique,” IRE Transactions on

Electronic Computers, vol. EC- 8, pp. 330–334, Sept. 1959.

[2] B. Gisuthan and T. Srikanthan, “Pipelining flat CORDIC based trigonometric function generators,” Microelectronics Journal, volume 33, Pp.77–89, 2002.

[3] E. Deprettere, P. Dewilde, and R. Udo, “Pipelined CORDIC architectures for fast VLSI filtering and array processing,” in IEEE International Conference on Acoustic,Speech, Signal Processing, ICASSP’84, March 1984, volume 9, pp.250–253.

[4] C. C. Doss and R. L. Riley, \FPGA-Based execution of a effective IEEE-754 rapid device," in IEEE Field-Programmable Custom Computing Machine2004,pp.229{238.

[5] J. Detrey and F. Dinechin, \A parameterized floating-point exponential function for FPGAs," in IEEE International Conference Field-Programmable Technology, 2005, pp.27{34.

[6] A parameterized Floating-point logarithm operator for FPGAs," in Signals, Systems and Computers, 2005. Conference Record of the Thirty-Ninth Asilomar Conference,2005, pp. 1186{1190.

[7] Pedro Echeverra, Marisa Lopez-Vallejo,”An FPGA Implementation of the Powering function with Single Precision Floating-Point Arithmetic”