Circuits and Systems for Receiving, Transmitting and Signal Processing
Устройства и системы передачи, приёма и обработки сигналов

Research article
DOI: https://doi.org/10.18721/JCSTCS.14401
UDC 621.3.049.774

ASIC IMPLEMENTATION OF HIGH-SPEED VECTOR MAGNITUDE & ARCTANGENT APPROXIMATOR

Ara Abdulsatar Assim[1,2] ✉

[1] Salahaddin University, Erbil, Iraq;
[2] Peter the Great St. Petersburg Polytechnic University,
St. Petersburg, Russian Federation

✉ araabdulsattar@gmail.com

Abstract. The quadrature processing techniques used in spectral analysis, computer graphics, and digital communications constantly demand high-speed calculation of the magnitude of a complex number (vector V) given its real and imaginary parts. The aim of this work is designing a digital signal processor (DSP processor) for approximating magnitude and arctangent (phase) of vectors (and/or complex numbers). This work can be divided into three main stages. Firstly, a mathematical model is designed in Simulink, then using that model. Secondly, Verilog description code is generated. The code is used to perform logic synthesis (converting the description code into logic gates) using XT018 technology (180 nm BCD-on-SOI) from X-FAB. Lastly, an ASIC (Application Specific Integrated Circuit) is created from the logic gates. The inputs and outputs of the device are fixed-point numbers, their length is equal to 16 bits and the fraction length is 8 bits. The proposed system can calculate magnitude and phase with an error of less than 1 and 0.35 % respectively.

Keywords: alpha max plus beta min algorithm, arctangent approximation, fast magnitude approximation, digital signal processing, DSP processor

Citation: Assim A.A. ASIC implementation of high-speed vector magnitude & arctangent approximator. Computing, Telecommunications and Control, 2021, Vol. 14, No. 4, Pp. 7–14. DOI: 10.18721/JCSTCS.14401

Научная статья
DOI: https://doi.org/10.18721/JCSTCS.14401
УДК 621.3.049.774

РЕАЛИЗАЦИЯ ИНТЕГРАЛЬНОЙ СХЕМЫ СПЕЦИАЛЬНОГО НАЗНАЧЕНИЯ ВЫСОКОСКОРОСТНОГО АППРОКСИМАТОРА ДЛЯ ВЕЛИЧИНЫ И АРКТАНГЕНСА ВЕКТОРОВ

А.А. Ассим[1,2] ✉

[1] Университет им. Салахаддина, Эрбиль, Ирак;
[2] Санкт-Петербургский политехнический университет Петра Великого,
Санкт-Петербург, Российская Федерация

✉ araabdulsattar@gmail.com

Аннотация. Методы квадратурной обработки, используемые в спектральном анализе, компьютерной графике и цифровой связи, постоянно требуют высокоскоростного вычисления величины комплексного числа (вектора V) с учетом его действительной и мнимой частей. Рассмотрен цифровой сигнальный процессор (DSP) для аппроксимации величины и арктангенса (фазы) векторов (и / или комплексных чисел). Работу можно разделить на три основных этапа. Сначала в Simulink создаётся математическая модель, затем с её помощью формируется код описания Verilog, используемый для выполнения логического синтеза (преобразования кода описания в логические элементы) с применением полупроводниковой технологии XT018 (180 нм BCD-on-SOI) от X-FAB. Наконец, из логических вентилей создаётся ASIC (специализированная интегральная схема). Входы и выходы устройства представляют собой числа с фиксированной точкой, их длина равна 16 битам, а дробная длина – 8 бит. Предлагаемая система может рассчитывать амплитуду и фазу с погрешностью менее 1 и 0,35 % соответственно.

Ключевые слова: алгоритм альфа макс плюс бета мин, приближение арктангенса, быстрое приближение величины, цифровая обработка сигналов, цифровой сигнальный процессор, ЦСП

Для цитирования: Assim A.A. ASIC implementation of high-speed vector magnitude & arctangent approximator // Computing, Telecommunications and Control. 2021. Т. 14, № 4. С. 7–14. DOI: 10.18721/JCSTCS.14401

Introduction

Calculating magnitude and phase of vectors or complex numbers is useful in many areas, including, but not limited to, AM demodulation, signal processing and image processing systems [1–9]. There is more than one method for that purpose, but choosing the optimal method depends on the required precision, hardware and software capabilities. For instance, calculating magnitude of a vector requires taking square root of the squared sum of the real and imaginary components [14], as in equation:

$$\text{Magnitude} = \sqrt{x^2 + y^2}, \tag{1}$$

while determining the phase requires solving arctan function [1, 4, 7], as given in

$$\varphi = \tan^{-1}\frac{y}{x}. \tag{2}$$

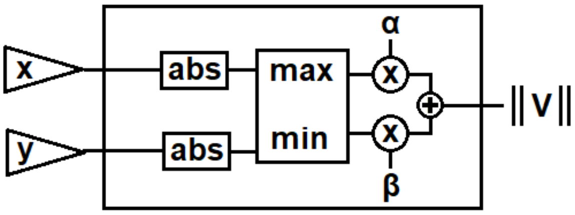

Fig. 1. Block diagram of the high speed magnitude approximator

Both operations require a lot of arithmetic computations with floating-point numbers. That is why approximation algorithms are introduced: they finish the same task much faster, and require less hardware and software resources [1, 5, 13–15]. In this paper, Alpha max plus beta min algorithm is used for fast magnitude approximation [10–15]. This algorithm can be defined by this formula [1, 3, 4]:

$$\text{Apprx. magnitude} \approx \alpha.\max + \beta.\min, \tag{3}$$

where max, min are the unsigned maximum and minimum values of the vector components respectively, α and β are constant values. The block diagram of this system is provided in Fig. 1.

The choice of α and β values depends on the desired precision as provided in Table 1. In this work, the last values are used, because they produce the most accurate results (maximum error is 1.0 %). The implemented arctan approximation equations used in this work are provided in Table 2. The proposed method mentioned here is quite efficient and convenient: it uses neither look-up tables nor very high-order polynomials. The only issue is that all the equations in Table 2 require division. Due to the fact that division is not a synthesizable operation in Verilog, it is not allowed to use a divider block in the mathematical model; instead, the division function (1/*x*) is approximated using Taylor series expansion with center 1, as provided in the following equation:

$$\text{Taylor series of } \frac{1}{x} \text{ with center } 1 \approx 1-(x-1)+(x-1)^2-(x-1)^3+\dots. \tag{4}$$

Table 1

Choice of alpha and beta values and corresponding error rates

α	β	Maximum error, %
1	0.5	11.8
1	0.25	11.6
1	0.375	6.8
0.9375	0.46875	6.3
0.9486	0.39293	5.1
1	0.4	7.7
$\alpha = 1$ if $\min < 0.375 \max$ $\alpha = 0.84375$ if $\min \geq 0.375 \max$	$\beta = 0.125$ if $\min < 0.375 \max$ $\beta = 1.1875$ if $\min \geq 0.375 \max$	1.8
$\alpha = 0.99$ if $\min < 0.4142135 \max$ $\alpha = 0.84$ if $\min \geq 4142135 \max$	$\beta = 0.197$ if $\min < 0.4142135 \max$ $\beta = 0.561$ if $\min \geq 4142135 \max$	1.0

Table 2

Used equations for approximating arctangent function

Octant	Arctan approximation formula
1st or 8th	$\theta = \dfrac{IQ}{I^2 + 0.28125Q^2}$
2nd or 3rd	$\theta = \dfrac{\pi}{2} - \dfrac{IQ}{Q^2 + 0.28125I^2}$
4th or 5th	$\theta = \operatorname{sign}(Q).\pi + \dfrac{IQ}{I^2 + 0.28125Q^2}$
6th or 7th	$\theta = -\dfrac{\pi}{2} - \dfrac{IQ}{I^2 + 0.28125Q^2}$

Implementation of the fast magnitude approximator

The fast magnitude approximator system is given in Fig. 2, it is based on α max plus β min algorithm [4]. An ideal (reference) model is needed to justify the validity of the proposed model. There is a special block in Simulink for that application, namely (Complex to Magnitude-Angle), but it is not synthesizable in practice.

An input signal is applied to the designed system in Fig. 2 for checking the performance of the mathematical model. The signal consists of the summation of three sinusoidal signals with different magnitudes and frequencies, as depicted in Fig. 3*a*. This input generates an output that is provided in Fig. 3*b*. We can see the difference between the ideal magnitude and the approximated magnitude is very little. Thus, we can conclude that the system's performance is valid.

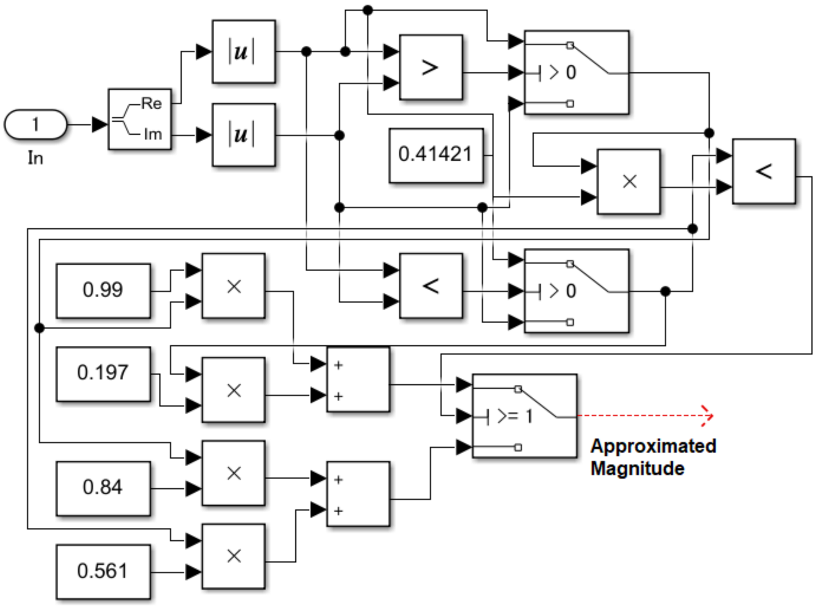

Fig. 2. Block diagram of the fast magnitude approximator (in Simulink)

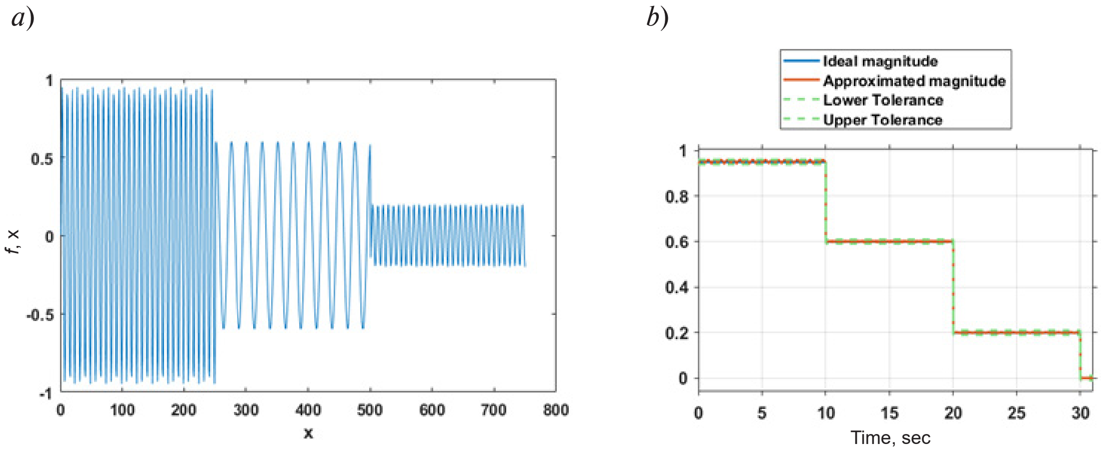

Fig. 3. The applied input signal (*a*) and output signal (*b*)

Implementation of the arctangent (phase) approximator

The structure of the arctangent function approximator system is provided in Fig. 4. The last block is a 4-to-1 multiplexer, because there are four different formulas for approximating arctangent function based on the octants (provided in Table 2). Based on the control signal's value, the multiplexer connects the output to one of its four inputs.

A comparison between the ideal arctangent signal and the output signal of the arctangent approximator system is shown in Fig. 5*a*, and the difference between the ideal and approximated arctangent function is shown in Fig. 5*b*, the maximum error is 0.0035.

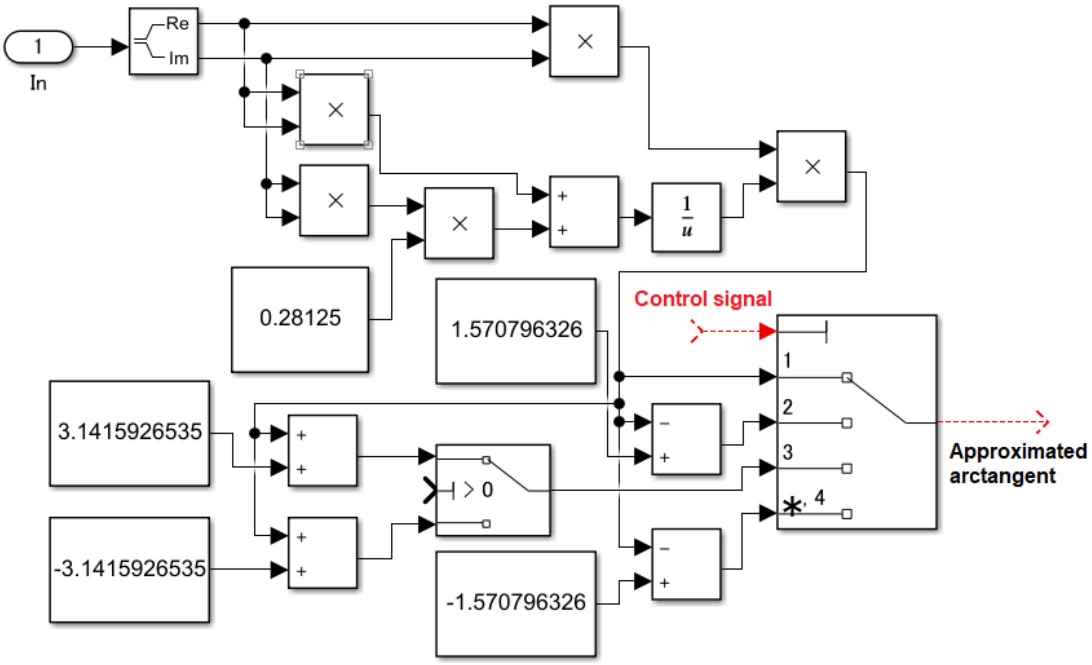

Fig. 4. Block diagram of the arctangent (phase) approximator (in Simulink)

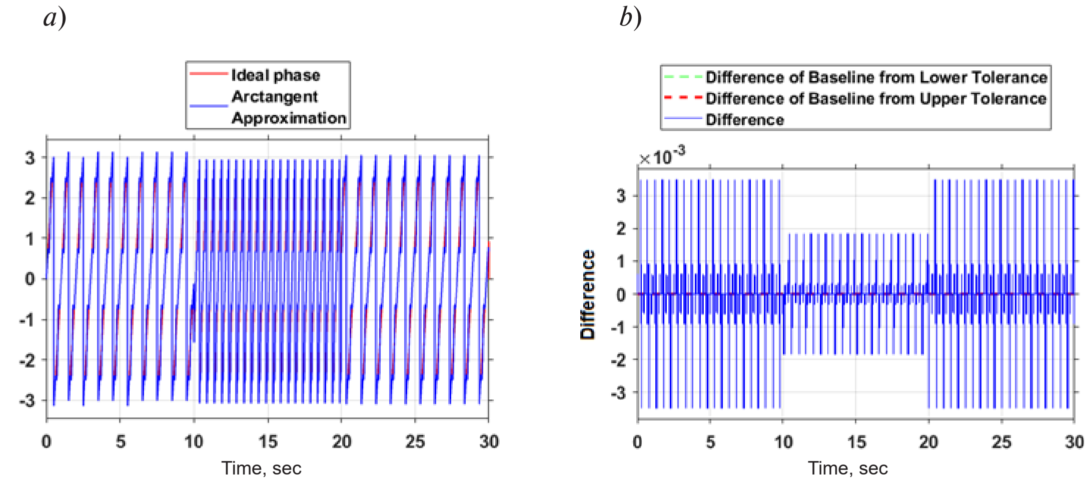

Fig. 5. Ideal vs. approximated arctangent function (*a*) and tolerance (error) signal (*b*)

Realization of the Application Specific Integrated Circuit (ASIC)

Before the realization of the ASIC, the entire blocks in the system must be converted into fixed-point numbers, so that later, the system can be defined in Verilog. The process is done by using the built-in tool in Simulink, known as fixed-point tool. Once all the blocks' date types are converted to a fixed-point, it can be used to generate the Verilog code by means of HDL coder, a MATLAB tool generating a Verilog description of the mathematical model. In addition to that, this tool transforms the input and output signal to an array of hexadecimal numbers. This later can be used as a reference to verify that the netlist functions correctly. The Verilog description code is used to create a netlist of the device (synthesis). In this work, Cadence Encounter RTL Compiler was used to synthesize the code. The netlist is shown in Fig. 6*a*. The same tool (Cadence Encounter) was used to create the layout from the netlist in Fig. 6*b*. (list of logical gates and interconnects obtained after logical synthesis) with reference to the technological library: the position of the input and output pins and the constraint file. The process of layout generation is automated, but there are many specifications that need to be carefully specified in the tools. The chosen clock frequency

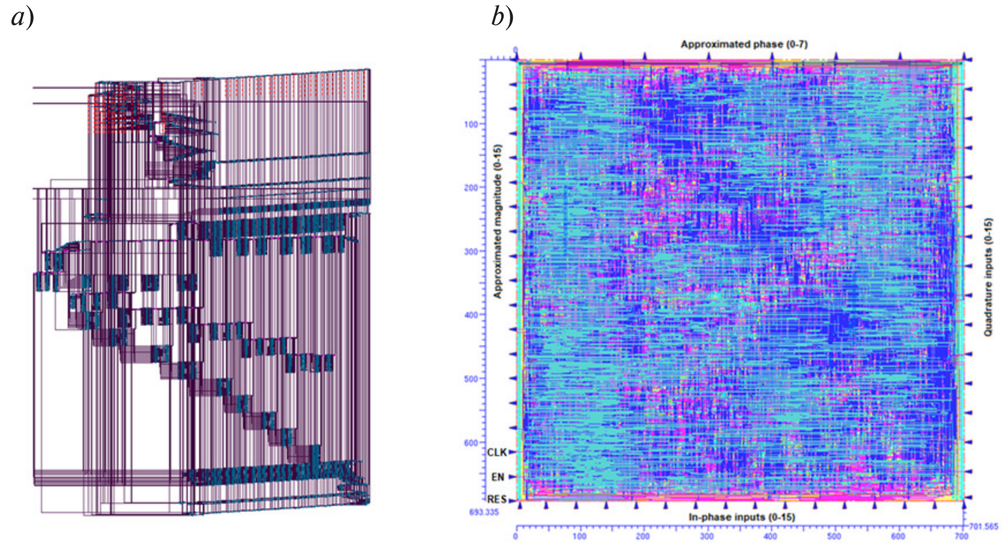

Fig. 6. Netlist of the device (*a*) and layout of the DSP processor (*b*)

is 10 MHz with an uncertainty of 0.05. The layout of the device is provided in Fig. 6*b*. Its dimensions are (701.565 μm × 693.335 μm), it requires an area of 486419.56 μm^2.

Conclusion

In conclusion, this research paper covers the entire process of the development of a digital device, from writing a Verilog code (system level) to creating a layout of the device (physical level). A system for approximating vector magnitude and arctangent using the FPGA was developed. All the main stages of development were passed: description of a digital device in Verilog HDL language, logical synthesis of a device in Cadence RTL Compiler, layout generation in Cadence Encounter. In addition to that, functional verification was carried out in Cadence Incisive at all the three stages: behavioral level, synthesis and layout generation. The timing diagram results confirm the correct operation of the device, and during the stage of layout generation, different verifications were carried out (time analyses for post-Route and SignOff stages for both cases of setup and hold). Verifications for DRC, connectivity and geometry were performed as well, all of them showing no violations. After its generation, the layout was imported to Cadence Virtuoso undergoing two checks, namely DRC and LVS, which it passed successfully. Therefore, we may conclude that the layout was generated correctly. The source codes are uploaded to GitHub, the link is provided in Appendix, in case someone is interested in repeating the same work.

Appendix

The Verilog codes can be found in this repository: https://github.com/AraAssim/AraAssim-Vector-magnitude-and-arctangent-approximation

Acknowledgment

I would like to express my deep gratitude to my parents for their boundless encouragement and love. I also thank my dearest friend (Nikolai Kirichenko) for helping me during my stay in Russia. Finally, I want to extend my thanks to the editors of this journal for taking time to review my work.

REFERENCES

1. **Ukil A., Shah V.H., Deck B.** Fast computation of arctangent functions for embedded applications: A comparative analysis. *2011 IEEE International Symposium on Industrial Electronics*, 2011, Pp. 1206–1211. DOI: 10.1109/ISIE.2011.5984330

2. **Hwang D.D., Dengwei Fu, Willson A.N.** A 400 MHz processor for the conversion of rectangular to polar coordinates in 0.25 μm CMOS. *IEEE Journal of Solid-State Circuits*, 2003, Vol. 38, No. 10, Pp. 1771–1775. DOI: 10.1109/JSSC.2003.817588

3. **Pilato L., Fanucci L., Saponara S.** Real-time and high-accuracy arctangent computation using CORDIC and fast magnitude estimation. *Electronics*, 2017, Vol. 6, No. 1, P. 22. DOI: 10.3390/electronics6010022

4. **Lyons R.** *Understanding digital signal processing.* 3rd ed. Upper Saddle River, N.J.: Prentice Hall, 2011, Pp. 550–612.

5. **Mikami N., Kobayashi M., Yokoyama Y.** A new DSP-oriented algorithm for calculation of the square root using a nonlinear digital filter. *IEEE Transactions on Signal Processing*, 1992, Vol. 40, No. 7, Pp. 1663–1669. DOI: 10.1109/78.143438

6. **Lyons R.** Another contender in the arctangent race. *IEEE Signal Processing Magazine*, 2004, Vol. 21, No. 1, Pp. 109–110. DOI: 10.1109/MSP.2004.1267054

7. **Rajan S., Sichun Wang, Inkol R., Joyal A.** Efficient approximations for the arctangent function. *IEEE Signal Processing Magazine*, 2006, Vol. 23, No. 3, Pp. 108–111. DOI: 10.1109/MSP.2006.1628884

8. **Adams W., Brady J.** Magnitude approximations for microprocessor implementation. *IEEE Micro*, 1983, Vol. 3, No. 5, Pp. 27–31. DOI: 10.1109/MM.1983.291163

9. **Wong W.F., Gogo E.** Fast hardware-based algorithms for elementary function computations using rectangular multipliers. *IEEE Transactions on Computers*, 1994, Vol. 43, No. 3, Pp. 278–294. DOI: 10.1109/12.272429

10. **Powell S.** Design and implementation issues of all digital broadband modems. *DSP World Workshop Proceedings*, Toronto, Canada, Sept. 13–16, 1998, Pp. 127–142.

11. **Frerking M.** *Digital signal processing in communications systems*. Chapman & Hall, New York, 1994, P. 330.

12. **Jacobsen E.** *Minister of Algorithms, Abineau Communications*, private communication, Sept. 11, 2003.

13. **Palacherls A.** *DSP-mP routine computes magnitude*. EDN, Oct. 26, 1989.

14. **Mikami N., Kobayashi M., Yokoyama Y.** A new DSP-oriented algorithm for calculation of the square root using a nonlinear digital filter. *IEEE Trans. on Signal Processing*, 1992, Vol. 40, No. 7, Pp. 1663–1669. DOI: 10.1109/78.143438

15. **Lyons R.** Turbocharge your graphics algorithm. *ESD: The Electronic System Design Magazine*, Oct. 1988.

THE AUTHOR / СВЕДЕНИЯ ОБ АВТОРЕ

Assim Ara Abdulsatar
Ассим Ара Абдулсатар
E-mail: araabdulsattar@gmail.com

The article was submitted 01.09.2021; approved after reviewing 02.12.2021; accepted for publication 22.12.2021.

Статья поступила в редакцию 01.09.2021; одобрена после рецензирования 02.12.2021; принята к публикации 22.12.2021.