VLSI ARCHITECTURE OF MAGNITUDE ESTIMATION ALGORITHM FOR SPEECH RECOGNITION SYSTEM

Nguyen Ly Thien Truong[1], *Hoang Trang*[1]

Abstract

In speech recognition system, speech signal is preprocessed, and then will be transferred to the frequency domain by Fast Fourier Transform (FFT). The output of FFT block is the complex numbers that will be estimated in magnitude to prepare for the calculation of Mel-frequency filter bank block. This paper proposes a novel "alpha max-beta min" algorithm for magnitude estimation whose hardware implementation is very simple. The time executes proposed algorithm in only one clock cycle, and average error of less than 1%. The influences of proposed algorithm on the whole speech recognition system are also given in calculation speed and recognition accuracy.

Trong hệ thống nhận dạng tiếng nói, tín hiệu tiếng nói được tiền xử lý, rồi qua khối FFT. Đầu ra sau đó là những số phức mà cần được tính xấp xỉ biên độ để tiếp đến được xử lý qua khối lọc tần số Mel. Bài báo này đề xuất cải tiến thuật toán "alpha max-beta min" mà có được sự thuận lợi, dễ dàng trong thực hiện kiến trúc phần cứng. Thời gian thực hiện thuật toán cải tiến này chỉ trong 1 xung clock, sai số trung bình của phép xấp xỉ nhỏ hơn 1%. Sự ảnh hưởng của thuật toán đề nghị lên độ chính xác nhận dạng của cả hệ thống cũng được xem xét.

Index terms

Alpha max-beta min, magnitude estimation, VLSI architecture.

1. Introduction

IN implementing the hardware, the problem is raised how to design hardware and to obtain small area, low power consumption, fast processing speed and high accuracy. All the above criteria couldn't be achieved simultaneously. Depending on a specific application, some criteria will be focused. The magnitude estimation block of complex number plays an important role in various applications such as speech recognition, coding, audio and image processing in medicine [1], [2], [3]. Speech recognition systems require fast processing speed with acceptable accuracy. The issue of square root is difficult to implement on hardware. Square root is calculated by software with high accuracy but slow processing speed [4]. In hardware implementation, many authors tried to implement and develop magnitude calculation algorithms [2]-[6]. To simplify the hardware implementation of the square root, approximation algorithms are used. There are many methods used to approximate the square root of a number in which the "alpha max-beta min" and CORDIC algorithms have been used commonly [1]. The work in [1] compared the accuracy of CORDIC and "alpha max-beta min" algorithms and showed the α and β coefficient for "alpha max-beta min" algorithm used for hearing aid

(1)Ho Chi Minh City University of Technology-Vietnam National University - Ho Chi Minh City

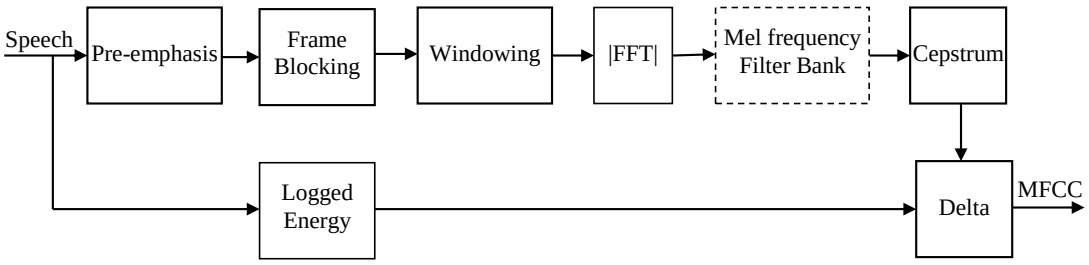

Figure 1. *Block diagram of the MFCC extraction algorithm [7]*

applications. To implement "alpha max-beta min" algorithm with these coefficients would be complex because they are floating point numbers format. The authors in [2] introduced equiripple error criteria to select the α, β coefficients for getting error as small as possible. Although this technique is correct, the hardware implementation of this technique is also complicated since it uses multiplier with floating point number. In this paper, the design and implementation of magnitude estimation for speech recognition system are presented. The novel algorithm using the α, β coefficients that correspond to the relationship between the real and the imaginary parts is also introduced. This novel technique has average error of less than 1% and easy hardware implementation.

The rest of this paper is structured as follows. Section II presents "alpha max-beta min" estimation algorithm and proposes novel coefficients of alpha, beta for the better approximation and easier implementation. The hardware implementation of the proposed technique for the magnitude estimation is given in section III. The comparison between proposed and conventional algorithms is also presented in this section. Simulation results are illustrated in section IV with different data lengths. Finally, section V presents the conclusion.

2. Alpha Max - Beta Min Algorithm: Review and Propose

For speech recognition, Fig. 1 illustrates a block diagram of the Mel Frequency Cepstral Coefficients (MFCC) extraction algorithm [7], in which magnitude of FFT block output in the complex number format will be approximated in the *Mel-frequency filter bank* block. This section presents "alpha max-beta min" algorithm which is commonly used to estimate magnitude of complex number and also gives the proposed coefficients of alpha, beta.

With the real part I and the imaginary part Q of the complex number C = I + jQ, the magnitude of the complex number C is defined:

$$|C| = |I + jQ| = \sqrt{I^2 + Q^2} \tag{1}$$

In equation (1), two problems need to be solved in implementing hardware are the square and the square root operations. The square operation would be easily implemented by multiplying a number with itself. In square root calculation, the raised problem is how to find algorithms with acceptable error and easy hardware implementation. One of the simple algorithms is "alpha max-beta min" that is linear approximation algorithm of a complex number magnitude. We define:

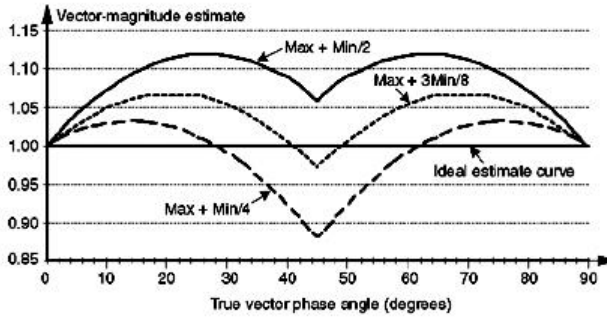

Figure 2. Normalized $\alpha.max + \beta.min$ estimates for $\alpha = 1, \beta = \frac{1}{2}, \beta = \frac{1}{4}$, and $\beta = \frac{3}{8}$ [4]

$$x = max(|I|, |Q|); y = min(|I|, |Q|) \tag{2}$$

Absolute and max, min operations will rotate the complex number such that its argument lies between 0^0 and 90^0 and between 0^0 and 45^0 respectively. Argument equals to 45^0 when max equals to min. In this region, the magnitude of the complex number is a linear approximation. Equation (1) is rewritten as:

$$|C| = |I + jQ| = \sqrt{I^2 + Q^2} = \sqrt{x^2 + y^2} \tag{3}$$

where $0^0 \leq \theta \leq 45^0$; $\theta = tan^{-1}(\frac{y}{x})$

In this region, $|C|$ is a linear approximation with $|\hat{C}|$ [2] as given in (4)

$$|\hat{C}| = \alpha.x + \beta.y \tag{4}$$

where α, β are positive constants. Relative error is defined as

$$|e\,(\theta)\,| = \frac{||C| - \left|\hat{C}\right|\,|}{|C|} = |1 - \frac{\left|\hat{C}\right|}{|C|}| = |1 - \alpha.\cos\theta - \beta.\sin\theta| \tag{5}$$

By taking the derivative of e(θ) in (5) with respect to θ then set the given derivative equals to zero, the maximum error $e_{max}(\theta)$ is determined at

$$\theta_{max} = tan^{-1}(\frac{\beta}{\alpha}) \tag{6}$$

Choosing the coefficients α, β will affect the accuracy and complexity of hardware implementation. The authors in [4] give methods predicting the accuracy of the estimated magnitude of the complex number. They use the ideal estimation curve with a unity magnitude over the phase angle range from 0^0 to 90^0; this means that the ideal curve has a mean value of one and an error standard deviation of zero. By plotting the magnitude estimation as a function of phase angle with various coefficients α and β, the average error, the maximum error and the standard deviation are calculated as in Fig. 2 [4]. Table 1 compares the maximum error, the average error and the standard deviation corresponding to the different coefficients α and β [4].

Table 1. *ALPHA MAX+ BETA MIN algorithm comparisons [4]*

Algorithm $\lvert C\rvert \approx$	**Largest error** (%)	**Largest error** (dB)	**Avg error** (%)	**Avg error** (dB)	**Standard deviation** σ_e	**Max** $\lvert V\rvert$ (%F.S)
Max+Min/2	11.8	0.97	8.6	0.71	0.032	89.4
Max+Min/4	-11.6	-1.07	-0.64	-0.06	0.041	97.0
Max+3Min/8	6.8	0.57	3.97	0.34	0.026	93.6
7(Max+Min/2)/8	-12.5	-1.16	-4.99	-0.45	0.028	100
15(Max+Min/2)/16	-6.25	-0.56	1.79	0.15	0.030	95.4

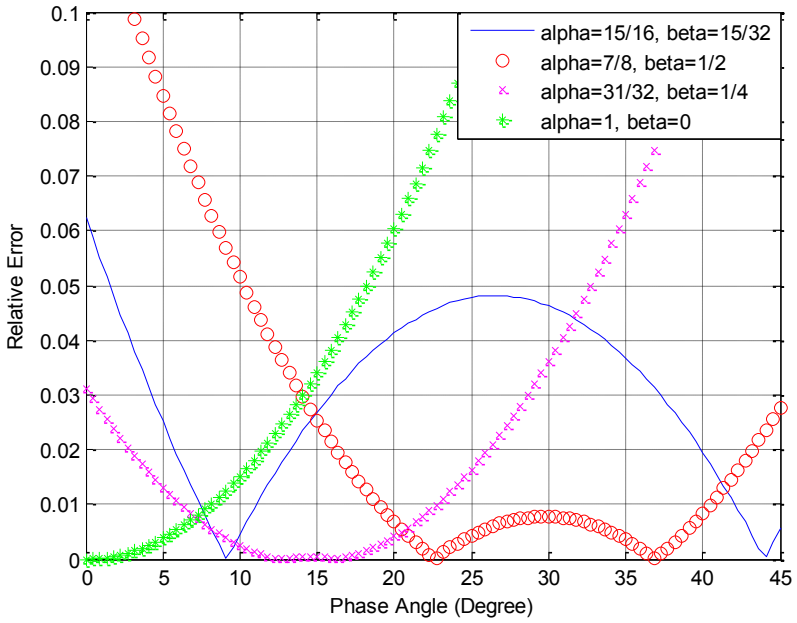

Figure 3. *Relationship between relative error* $|e(\theta)|$ *and phase angle* θ

From the illustrations and summary in Table 1, different coefficients α, β will result in different average errors and for each region (with phase angle in range from 0^0 to 90^0), different coefficients α, β will also lead to different average errors. In order to have more accurate approximation, depending on the relationship between max and min values, we should choose the coefficients α and β suitably.

Fig. 3 illustrates the relationship between relative error $|e(\theta)|$ and phase angle θ with θ from 0^0 to 45^0. In this range, we divide phase angle θ into four regions so that the relative error will be as small as possible.
Region I: $0^0 \leq \theta < 7^0$
Region II: $7^0 \leq \theta < 23^0$
Region III: $23^0 \leq \theta < 41^0$
Region IV: $41^0 \leq \theta < 45^0$
From $\theta = \tan^{-1}\left(\frac{\min}{\max}\right)$, we have $\frac{\min}{\max} = \tan(\theta)$. The relationship between max/min and phase angle θ is calculated as in Table 2. Ratio max/min is chosen of the form $a + b.1/2^n$

Table 2. Relationship between phase angle and MAX, MIN

Region	Coefficient α, β	Ideal	Proposal	Rewrite
I	$\alpha = 1, \beta = 0$	$\frac{\min}{\max} < 0.1228$	$\frac{\min}{\max} < 0.125$	$8min < max$
II	$\alpha = \frac{31}{32}, \beta = \frac{1}{4}$	$0.1228 \leq \frac{\min}{\max} < 0.4244$	$0.125 \leq \frac{\min}{\max} < 0.4444$	$\frac{9}{4}min < max \leq 8min$
III	$\alpha = \frac{7}{8}, \beta = \frac{1}{2}$	$0.4244 \leq \frac{\min}{\max} < 0.8693$	$0.4444 \leq \frac{\min}{\max} < 0.8$	$\frac{5}{4}min < max \leq \frac{9}{4}min$
IV	$\alpha = \frac{15}{16}, \beta = \frac{15}{32}$	$0.8693 \leq \frac{\min}{\max}$	$0.8 \leq \frac{\min}{\max}$	$max \leq \frac{5}{4}min$

Table 3. Relationships between MAX, MIN with proposed ALPHA, BETA

Relationship between max and min	Proposed coefficients α, β	
	α	β
$max \leq \frac{5}{4}.min$	15/16	15/32
$\frac{5}{4}.min < max \leq \frac{9}{4}.min$	7/8	1/2
$\frac{9}{4}.min < max \leq 8.min$	31/32	1/4
$8.min < max$	1	0

Table 4. Coefficient comparisons

I	Q	Software	[1]	[4]		This work
			$\alpha = 0.96043$ $\beta = 0.39782$	$\alpha = 1$ $\beta = 1/4$	$\alpha = 15/16$ $\beta = 15/32$	Table III
100	100	141.421	135.825	125	141	141
253	154	296.18	304.25	273	292	299
11	57	58.052	59.12	60	58	58
9	4	9.849	10.23	10	10	10

with *a*, *b* are positive integers. With this choice, when comparing the relationship between *max* and *min* we only use bit shifting and addition operations instead of using the multiplication and division operations.

The choice of coefficients α, β corresponding to *max* and *min* values is based on the experiment and easiness in hardware implementation. The α, β coefficients are proposed in Table 3. In the binary system, left or right shifting to *i* bits corresponds to the multiply or divide them by 2^i. Thus choosing the coefficients α and β having the reciprocal of a power of two form will lead to the easy hardware implementation.

Table 4 compares the results of the "alpha max-beta min" estimation algorithm used in [1], [4] and this paper with different α, β coefficients.

3. Implementation and Comparison of Conventional and Proposed Algorithms

3.1. Alpha max-beta min: conventional and proposed algorithm comparison

This section presents the conventional "alpha max-beta min" estimation algorithm and the proposed algorithm. In the conventional algorithm, there is only one comparator to find the max, min values. The output data of comparator will be right shifted and subtracted to create $\alpha.max$, $\beta.min$ values that will be added together. The block diagram of the conventional magnitude block is illustrated by Fig. 4.

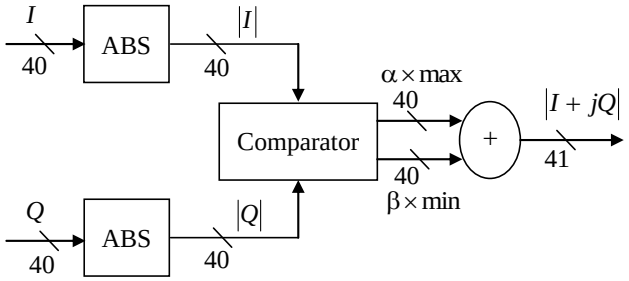

Figure 4. *Block diagram of the conventional magnitude computation block*

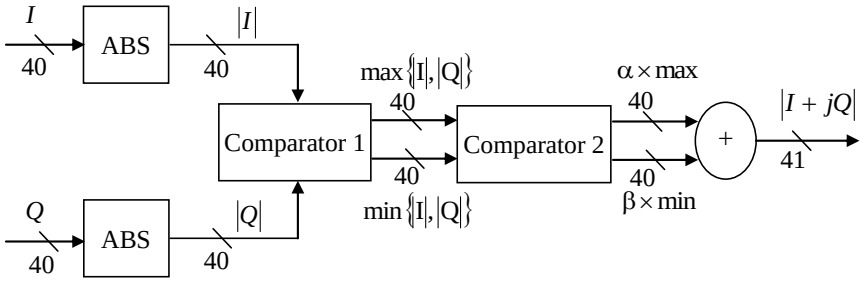

Figure 5. *Block diagram of the proposed magnitude computation block*

The real part I and imaginary part Q are the two's complement signed binary number with 40-bit length. The magnitude of real part I and the imaginary part Q are obtained by the ABS block (absolute calculation block). Then data are passed through the comparator to find the largest and smallest of the two numbers $|I|$ and $|Q|$. Next, the value of max, min will be right shifted and subtracted (if needed) to create $\alpha.max$, $\beta.min$ values. Finally, the output value of the adder is the magnitude of the complex number. The whole process is only performed in one clock cycle. For example, to calculate $\frac{15}{32}$.min, first min value is shifted to the right by 4 bits (equivalent to dividing the min value for $2^4 = 16$) then subtracted from the min value to obtain $\frac{15}{16}$.min. The $\frac{15}{32}$.min value could be obtained by right shifting $\frac{15}{16}$.min value to 1 bit. Using only bit shifting, addition and subtraction operations, the magnitude of complex number could be easily estimated.

Fig. 5 illustrates the block diagram of the proposed magnitude block. The difference between proposed and conventional algorithms is in the second comparator. The output data of the first comparator will be taken through the second comparator with the threshold mentioned in Table 3 to pick out the coefficients α and β respectively. Then the output value of the second comparator will be added together to calculate the magnitude of the complex number.

Table 5 shows the design parameters of these algorithms. The results are synthesized with FPGA EP2C35F672C6 Cyclone II device. The disadvantage of the proposed algorithm is a total logic element of the proposed algorithm is 3.75 times greater than the conventional algorithm in case $\alpha = 1, \beta = 1/4$ but accuracy is 3 times greater (see Table 9). Therefore, the proposed algorithm could be used in high accuracy applications.

Table 5. Summarizes the synthesized results

Parameter	Conventional algorithm		Proposed algorithm
	$\alpha = 1$ $\beta = 1/4$	$\alpha = 15/16$ $\beta = 15/32$	Table II
Total logic elements	371	670	1393
Total registers	41	41	41
Timing (clock cycles)	1	1	1
Total thermal power (mW)	124.42	124.38	124.38

Table 6. CORDIC and proposed algorithm comparison

I	Q	$\|I + jQ\|$			Linear Error	
		Software	CORDIC [1]	This paper	CORDIC [1]	This paper
100	-20	101.9804	101.4375	102	0.5429	-0.0196
10	-65	65.7647	65.6875	65	0.0772	0.7647
8	6	10	10	10	0	0
255	6	255.0706	255.125	255	-0.0544	0.0706
96	1	96.0052	96	96	0.0052	0.0052
7	0	7	7	7	0	0
0	26	26	26	26	0	0
423	78	430.1314	428.3125	429	1.8189	1.1314
-100	4	100.0800	100.1875	100	-0.1095	0.0800
-20	12	23.3238	23.4375	24	-0.1137	-0.6762

3.2. CORDIC and proposed algorithm comparison

CORDIC is one of the popular algorithms used to estimate the magnitude and phase angle of complex number. The introduced CORDIC algorithm was better than the conventional “alpha max-beta min” algorithm in terms of accuracy and hardware implementation [1]. In this section, the comparison between using CORDIC and our proposed algorithm in “alpha max-beta min” is shown in Table 6. The results in this table indicate that accuracy of the proposed algorithm could be compared with the CORDIC algorithm. Besides, the CORDIC algorithm would have one disadvantage of that magnitude value estimated must be multiplied by the CORDIC gain [1] of value ranges from 1.16 to 1.64. This gain multiplication is not needed in our proposed algorithm.

4. Simulaton Results

This section presents simulation results. The design is done by using Verilog HDL, results are simulated by using ModelSim version 6.4a. Fig. 6 illustrates simulation results for magnitude estimation where alpha_max is $\alpha.max$, beta_min is $\beta.min$; inr, ini and out are the real part, the imaginary part and the magnitude value of complex number I + jQ, respectively.

Tables 7 and 8 present the relative errors in system using our proposed algorithm with different data lengths.

Table 9 compares average error between conventional algorithm and proposed algorithm. These results are verified in block $|FFT|$ output with whole speech recognition system input of Vietnamese speech samples including the words "khong", "mot", "hai", "ba", "bon", "nam", "sau", "bay", "tam", "chin" (in English, "zero", "one", "two", "three", "four", "five", "six", "seven", "eight", "nine" respectively). They are 16-bit 2’s complement, each digit is digitized in average of 3630 values. Results in this table indicate that average error of proposed algorithm

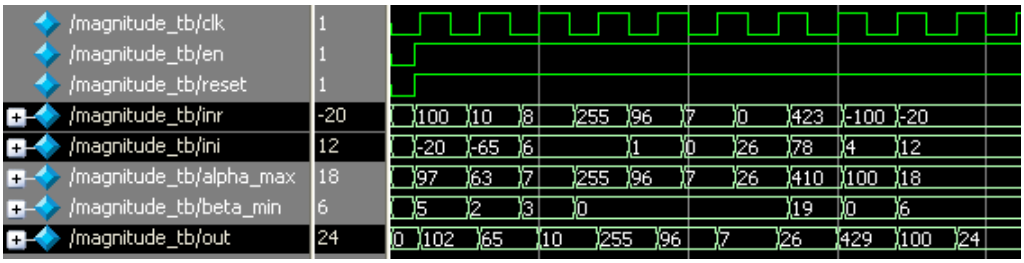

Figure 6. Simulation results for magnitude estimation using ModelSim

Table 7. Simulation results for an 8 bits system

Complex number: $C = I + jQ$		**Magnitude:** $\|I + jQ\|$		**Relative Error**
I	**Q**	**Software** $(\|C\|)$	**Hardware** $(\|\hat{C}\|)$	$\|e\| = \frac{\|\|C\|-\|\hat{C}\|\|}{\|C\|}(\%)$
89	-47	100.65	101	0.35
-45	13	46.84	47	0.34
88	4	88.09	88	0.10
-121	33	125.42	126	0.46
-78	109	134.03	135	0.72
181	-87	200.82	202	0.59

Table 8. Simulation results for an 16 bits system

Complex number: $C = I + jQ$		**Magnitude:** $\|I + jQ\|$		**Relative Error**
I	**Q**	**Software** $(\|C\|)$	**Hardware** $(\|\hat{C}\|)$	$\|e\| = \frac{\|\|C\|-\|\hat{C}\|\|}{\|C\|}(\%)$
30761	-2095	30832.26	30761	0.23
-26246	19725	32831.82	32828	0.01
9906	-18940	21374.11	21526	0.71
11483	3105	11895.39	11901	0.05
-29959	8956	31269.02	31262	0.02

Table 9. Comparison average error between conventional algorithm and proposed algorithm with Vietnamese speech signal

Word	**Conventional algorithm**			**Proposed algorithm**
	$\alpha = 0.96043$ $\beta = 0.39782$	$\alpha = 1$ $\beta = 1/4$	$\alpha = 15/16$ $\beta = 15/32$	Table III
khong	0.0249	0.0321	0.0306	0.0077
mot	0.0247	0.0319	0.0311	0.0077
hai	0.0251	0.0319	0.0312	0.0071
ba	0.0247	0.0320	0.0309	0.0074
bon	0.0248	0.0320	0.0307	0.0086
nam	0.0251	0.0312	0.0310	0.0083
sau	0.0250	0.0317	0.0310	0.0076
bay	0.0246	0.0326	0.0304	0.0080
tam	0.0242	0.0317	0.0301	0.0080
chin	0.0251	0.0318	0.0304	0.0093

is less than 1%. Average error could approach 0% once input data length increase. In our design, the magnitude computation block can operate with 40 bits input data.

Table 10. *Influence of the magnitude calculation block to the speech recognition system*

Parameter	Accurate calculation	Approximate calculation (by convention)	Approximate calculation (by proposal)
The number of clocks is used for recognizing a word including 2400 values	40206358	3731820	3731820
Recognition accuracy	95.4%	91.8%	92.2%

Table 10 illustrates the influences of the magnitude calculation block to the speech recognition system in calculation speed and recognition accuracy. Testing scenario in this table is as follows: 50 single words need to be recognized, 500 samples used for training (10 samples/a single word), 1000 samples used for testing the recognition (20 samples/a single word). These results show that the approximate calculation in amplitude could increase computational speed 10 times faster than the accurate one. Moreover, the proposed algorithm leads to the accuracy of the speech recognition system better than the conventional approximate calculation algorithm of 0.4%.

The whole speed recognition system including MFCC feature extraction and recognition is built on ASIC design flow, 130 nm technology. The operating frequency is obtained of 48 MHz.

5. Conclusion

In this paper, we propose the choice of coefficients α and β matching the max, min values. This novel technique gives easy hardware implementation and very small error. Error is as small as if the input data length is increased.

The hardware implementation for the proposed algorithm is very simple by using only the addition, subtraction and bit shifting operations instead of multiplications. This will lead to the reduction of power consumption. In this work, the proposed architecture in magnitude calculation with adder based on carry look ahead algorithm and subtractor based on the adder increases the processing speed. The magnitude calculation block is only implemented in one clock cycle with average error of less than 1%. Thus the whole system could be compared to CORDIC algorithm. The influences of proposed algorithm on the whole speech recognition system are also given in calculation speed and recognition accuracy.

Acknowledgments

This work is funded by the Ministry of Science and Technology, State-level key program, Research for application and development of information technology and communications, code KC.01.23/11-15.

References

[1] R. Naik, A. Stojcevski, V. Vibhute, J. Singh, Implementation of magnitude estimation algorithm for hearing aid, *IEEE International Workshop on Biomedical Circuits & Systems*, 2004

[2] A. Filip, Linear approximations to $\sqrt{x^2 + y^2}$ having equiripple error characteristics, *IEEE Trans. Audio Electroacoust*, pp. 554-556, Dec. 1973.

[3] Audhkhasi, K.; Zavou, A.M.; Georgiou, P.G.; Narayanan, S.S., Theoretical Analysis of Diversity in an Ensemble of Automatic Speech Recognition Systems , *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol.22, no. 3, pp.711-726, Mar. 2014 .
[4] R. Lyons, *Understanding Digital Signal Processing*, 2^{nd} ed. Upper Saddle River, NJ: Prentice-Hall, pp. 400-406, 2001.
[5] W. Adams and J. Brady, Magnitude approximations for microprocessor implementation, *IEEE Micro*, pp. 27-31, Oct. 1983.
[6] Mark Allie, Richard Lyons, A Root of Less Evil, *IEEE Signal Processing Magazine*, pp. 93-96, Mar. 2005.
[7] Wei HAN, Cheong-Fat CHAN, Chiu-Sing CHOY and Kong-Pang PUN, An Efficient MFCC Extraction Method in Speech Recognition, *ISCAS 2006. IEEE International Symposium on Circuits and Systems*, pp. 145-148, May 2006

Manuscript received 03-03-2014; accepted 07-11-2014.

Nguyen Ly Thien Truong was born in Tien Giang province, Vietnam. He received the Bachelor degree in Electronics Engineering and Master of Science degree Engineering in Electronics-Telecommunication Engineering from Ho Chi Minh City University of Technology, Vietnam in 2010 and 2012, respectively. Since 2010, he is lecturer at Faculty of Electrical and Electronic Engineering, Ho Chi Minh City University of Technology, Vietnam. He is interested in Digital system design, and Microcontroller system design. His research includes using HDL to design digital systems and using microcontrollers to develop automatic control systems.

Hoang Trang was born in NhaTrang city, Vietnam. He received the Bachelor of Engineering, and Master of Science degree in Electronics-Telecommunication Engineering from Ho Chi Minh City University of Technology in 2002 and 2004, respectively. He received the Ph.D. degree in Microelectronics-MEMS from CEA-LETI and University Joseph Fourier, France, in 2009. From 2009-2010, he did the post-doctorate research in Orange Lab-France Telecom. Since 2010, he is lecturer at Faculty of Electricals-Electronics Engineering, Ho Chi Minh City University of Technology. His field of research interest is in the domain of FPGA implementation, Speech Recognizer, MEMS, fabrication, integration of passive components and function for telecommunications, Embedded System, System-on Chip (SoC).