ISSN: 1226-7244 (Print)
ISSN: 2288-243X (Online)
논문번호 13-03-04

Journal of IKEEE.Vol.17,No.3,241~247,September 2013
http://dx.doi.org/10.7471/ikeee.2013.17.3.241
21

# Proposed Efficient Architectures and Design Choices in SoPC System for Speech Recognition

Hoang Trang★, Tran Van Hoang

**Abstract**

This paper presents the design of a System on Programmable Chip (SoPC) based on Field Programmable Gate Array (FPGA) for speech recognition in which Mel-Frequency Cepstral Coefficients (MFCC) for speech feature extraction and Vector Quantization for recognition are used. The implementing process of the speech recognition system undergoes the following steps: feature extraction, training codebook, recognition. In the first step of feature extraction, the input voice data will be transformed into spectral components and extracted to get the main features by using MFCC algorithm. In the recognition step, the obtained spectral features from the first step will be processed and compared with the trained components. The Vector Quantization (VQ) is applied in this step. In our experiment, Altera's DE2 board with Cyclone II FPGA is used to implement the recognition system which can recognize 64 words. The execution speed of the blocks in the speech recognition system is surveyed by calculating the number of clock cycles while executing each block. The recognition accuracies are also measured in different parameters of the system. These results in execution speed and recognition accuracy could help the designer to choose the best configurations in speech recognition on SoPC.

*Key words: Speech recognition, MFCC, VQ, SoPC, FPGA, Nios*

## 1. Introduction

Speech recognition system is applied in many application fields such as health care, military, human computer interaction, avionics technicians… [1], especially, the applications which support disabled people to communicate with the world in a better way. For that reason, there are many studies on software/hardware implementation of speech recognition systems for many years.

However, because of a large number of accents spoken around the world, there are still many challenges that need further research and development, for example, Vietnamese speech recognition.

The research on speech recognition is mainly in two directions, namely: the software runs on Personal Computers (PCs) and embedded systems. For the first direction, there exit many studies and software tools, which have been developed successfully. In particular, the Hidden Markov Model Toolkit (HTK) is a toolkit for building Hidden Markov Models (HMMs) used in speech recognition successfully [2]. There are also many tools running on the PC or smart phone aiming at controlling device via speech. For the second direction, embedded systems have many advantages of high performance, convenience, low cost, and great development potential. However, speech recognition research based on embedded systems has still many

★Department of Electrical-Electronics Engineering, University of Technology, HoChiMinh City, VietNam.
★Corresponding author: Hoang Trang, hoangtrang@hcmut.edu.vn.
※ Acknowledgment: This research is funded by The Department of Science and Technology HoChiMinh City (DOST HCM) from 12/2011-02/2013.
Manuscript received May,02.,2013; revised July. 10, 2013; accepted July, 12. 2013

challenges to overcome. This paper presents the implementation of a speech recognition system as an embedded system using FPGA technology.

In fact, the implementation of speech recognition systems has been done by using FPGA technology in recent years. In the previous work [3], speech recognition systems were implemented as hardware/software co-design systems using Hidden Markov Model (HMM). This work used Linear Predictive Coding (LPC) method for feature extraction, therefore, the recognition accuracy is not high compared with the MFCC method. In work of [4], the MFCC method was applied, but the optimization was not taken into account yet to increase performance. Another work, presented in [5] and [6], the authors proposed an efficient MFCC hardware implementation for feature extraction in speech recognition. However, this work has been done using ASIC technology and therefore less flexible than FPGA-based implementations. Other implementations for speech recognition systems were presented in [7-10]. Among these, the work presented in [8] proposed a hardware/software co-design method to tradeoff between the performance and the flexibility of the recognition system while [7] and [10] presented FPGA based implementations of the recognition systems. None of them discuss about the optimization method for MFCC algorithm.

In our work, the MFCC method is used with some proposed modifications to increase the performance of the system. The whole system has been implemented using Altera FPGA technology to be more flexible.

The paper is organized as follows. The overview of speech recognition system is briefly presented in Section 2. The design and implementation of the proposed speech recognition system as a SoPC (System on Programmable Chip) are mentioned and discussed in Section 3. Section 4 will show the achieved experimental results. Finally, conclusion is given in Section 5.

## 2. Overview of Speech Recognition

In this section, the overview of speech recognition is presented as shown in Fig. 1. Audio samples go through feature extraction block to retrieve the characteristics of sound. Through the Feature Extraction block presented in Section 2.1, the audio input will be transformed into the spectral coefficients. Then, these spectral characteristics go through Training block to create codebook for each word. In the recognition step, Recognition block using Vector quantization technique given in Section 2.2 will capture spectral features and will decide which words based on comparing spectral features with the codebooks already trained.

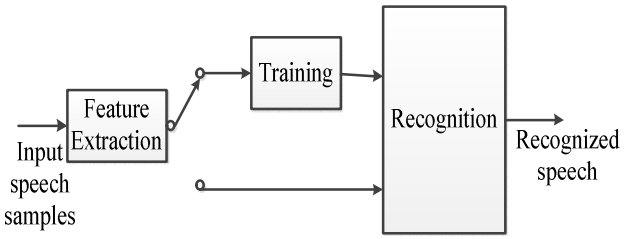


Fig. 1. Speech recognition system.

### 2.1 Feature extraction

In speech recognition, voice feature extraction is first step that will give the parameters used for recognition stage easily than the original speech signal. One of the most efficient algorithms used in feature extraction is MFCC algorithm. This method is based on the perceived sound of the human ear that is linear in the low frequencies and increases with logarithmic scales in the high frequencies. From this characteristic, the MFCC method gives the most important characteristics of the human voice. Fig. 2 presents the conventional MFCC algorithm for feature extraction.

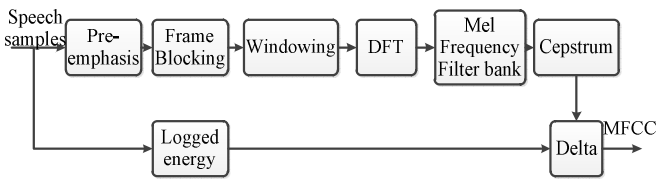


Fig. 2. Conventional MFCC feature extraction algorithm.

### 2.2 Vector Quantization process

The Vector Quantization process is described in Fig. 3. Audio signal after being extracted features

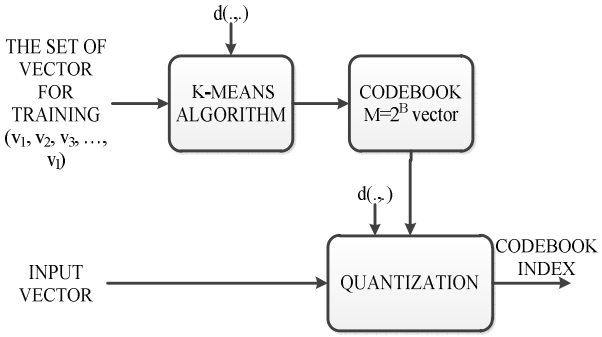


Fig. 3. Vector Quantization (VQ) training and classification.

will produce a series of feature vectors.

Then, feature vectors will be quantized and be split into different groups called as codebook (M=16/32/64/128/...) and each group will be labeled from 1 to M. K-Means algorithm is used for codebook generation.

In speech recognition, it is common to use the Euclidean distance as in (1) that will be used in the classification stage, labeled as feature vector and used in the recognition step

$$\text{-}(x,y) = \sqrt{\sum_{k=1}^{p} |x_k - y_k|^2} \quad (1)$$

## 3. Implementation

In our implementation process, some blocks will be adjusted, modified to obtain higher computing speed. In this section, we will present some improvements in a few blocks to optimize the computing speed. Evaluated results in terms of the number of clock cycles will be presented in the next section to show the improvements.

### 3.1 Feature Extraction implementation

#### 3.1.1 Voice Activation Detection (VAD)

Voice signal after recording through the microphone will gain a certain number of samples. In this work, the sampling frequency is 8 kHz, each time recording in 1 second, corresponding to 8000 samples. However, in the 8000 samples, not all are meaningful sound, much of them are silence. So, before the audio samples are extracted features, it requires the program to extract the significant audio and remove the silence.

As mentioned in previous section, audio signal is divided into M segments (i.e., blocks), L samples in each segment. In this work, we assigned that with , which means for each segment.

Then the energy function will be calculated for each segment by the following formula:

$$d(x,y) = \sqrt{\sum_{k=1}^{p} |x_k - y_k|^2} \quad (2)$$

VAD will reject segment if . In this work, we chose . The selection of TH is due to the test, go back and forth several times to select the appropriate value makes the correct signal clipping.

#### 3.1.2 Pre-emphasis

In pre-emphasis block, the coefficient "" has the value from 0.9 to 1. In theory, the normal value of "" is 0.97. Building the system on SoPC, we modify the value of "" to implement the system easily and increase the speed. The value 1, 15/16, 0.97 are surveyed in calculating speed through assessment of clock number.

Transfer function of the filter in pre-emphasis is described by Equation 3. In the time domain, the relationship between output and input is shown in Equation 4.

$$\text{-}(z) = 1 - a.z^{-1} \quad (3)$$

$$s_i' = s_i - a.s_{i-1} \quad (4)$$

With , Equation 4 will be simplified as: .

Advantage of using 15/16 as "" coefficient is expressed in Equation 5. can be realized in binary computation system by shifting 4 bits to the right. Using this value, the multiplication step in Equation (3), (4) is simplified to shift and subtract operations.

$$s_i' = s_i - a.s_{i-1}, \quad a = \frac{15}{16}$$
$$s_i' = s_i - \frac{15}{16}s_{i-1} = s_i - \left(s_{i-1} - \frac{1}{16}s_{i-1}\right) \quad (5)$$

#### 3.1.3 Discrete Fourier Transform (DFT)

In general, and are the complex numbers. N-point DFT can be calculated as follows:

$$X_R(k) = \sum_{n=0}^{N-1}\left[x_R(n)\cos\frac{2\pi kn}{N} + x_I(n)\sin\frac{2\pi kn}{N}\right] \quad (6)$$

$$X_I(k) = -\sum_{n=0}^{N-1}\left[x_R(n)\sin\frac{2\pi kn}{N} - x_I(n)\cos\frac{2\pi kn}{N}\right] \quad (7)$$

$$k = 0, 1, 2, \ldots, N-1.$$

If DFT transformation uses two equations 6 and 7 to calculate, it costs trigonometric calculations, real multiplications, and additions. This shows that when the direct calculation using the DFT formula above arise large computational cost, it will slow speed in program execution. Therefore, in this case, we use the Fast Fourier Transform (FFT) algorithm instead. In addition, the look-up table of coefficients cosine is proposed to be used, the computing speed is also increased.

### 3.1.4 Magnitude computation

If using the conventional formula for calculating the complex amplitude as Equation 8, then the calculation speed will be very slow.

$$M = \sqrt{I^2 + Q^2} \quad (8)$$

Therefore, the estimation algorithm is applied. This algorithm calculates very fast amplitude of a complex number almost exact compared to the normal range by taking the square root operation. For complex number , amplitude estimation algorithm as follows:

$$M \approx \alpha.\max\{|I|, |Q|\} + \beta.\min\{|I|, |Q|\} \quad (9)$$

In this system, as 1 and as ¼ are proposed to be used after our several tests. This approach reduces the number of calculations with acceptable error.

### 3.1.5 Mel frequency filter bank

The of power coefficient of the frame is calculated by (10) as

$$S'_{nk} = \sum_j S_{nj}.FC_{kj}, \quad k = 0, 1, \ldots, K \quad (10)$$

where, is the number of the filters, is the point of the frame's spectrum, and is the coefficient of the filter. When implementing the speech recognition system on SoPC, the rectangular filter bank is proposed in the new algorithm instead of the triangular filter bank in traditional approach. So, the Equation (10) becomes

$$S'_{nk} = \sum_j S_{nj}.FC_{kj}, \quad FC_{kj} = 0 \text{ or } 1 \quad (11)$$

The rectangular filters are proposed to be used instead of the triangular filters because the output characteristic of a rectangular filter is either a "1" or a "0", the multiply and sum operations can be simplified to simple "add" and "no add" operations. No multiplication step is required in the proposed approach and will help to increase calculation speed.

By adding time derivatives to the basic parameter in the Delta block, the performance of a speech recognition system can be greatly enhanced.

After Feature Extraction block, a 160-sample frame is converted to a vector composed of 26 elements, including 12 cepstral coefficients, 1 energy coefficient and their first order time derivatives.

### 3.2 Training and recognition implementation

In this work, codebook size of 128 is considered. We use K-Mean algorithm for training codebook. Firstly, vectors in the vectors are randomly chosen for training. Secondly, for each training vector , we find the codeword in the current codebook vectors with closest distance and assign it belongs to the group of the codeword. Thirdly, for each group, we update codeword using the focus of all training vectors in this group. Steps 2 and 3 are

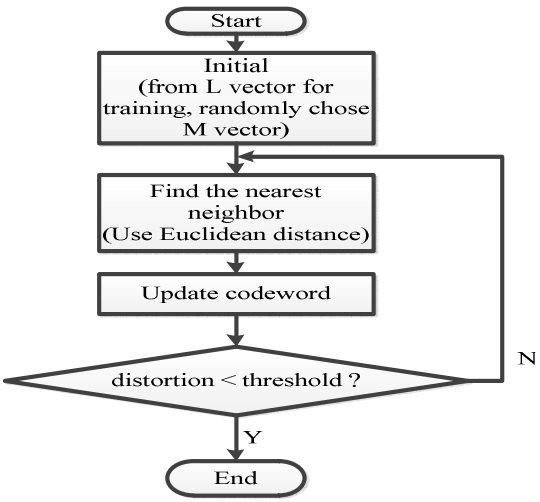


Fig. 4. Flowchart of the training codebook.

repeated until quantum error smaller than threshold value. The threshold value is obtained by doing experiments. The algorithm to implement this scheme is illustrated in Fig. 4.

The input speech sample is extracted the feature by the MFCC algorithm first. Then the feature vectors are calculated to find the VQ distortion for each codebook. The word having smallest distortion is the word which needs to be identified (Fig. 5).

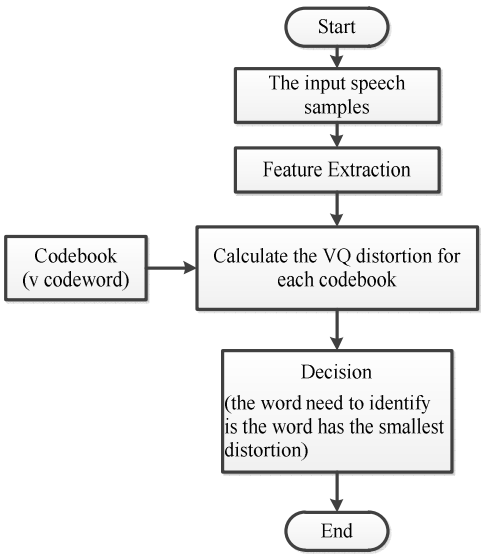


Fig. 5. Flowchart of recognition.

### 3.3 SoPC implementation

The proposed speech recognition has been intently implemented on Altera FPGAs for high performance. We propose a SoPC architecture for speech recognition system as described in Fig. 6.

In this architecture, Nios II Processor is the most important component of the system. SDRAM is used to store program. Flash memory is used to store the parameters of the codebook after training. The ADC interface is the part connected to the Audio Codec WM8731 chip which is responsible for data sampling of voices spoken into the microphone. LCD is used to show the recognition result. In particular, the Interval Timer is used to calculate the number of the pulse clocks when executing each block.

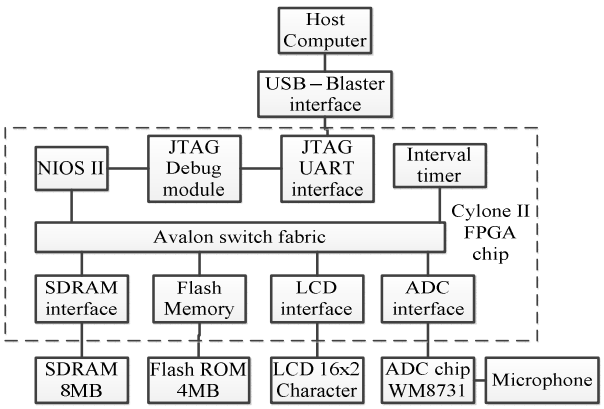


Fig. 6. Design of SoPC based on FPGA.

## 4. Experimental Results

As mentioned above, we use Interval Timer to survey the program execution speed of each functional block in speech recognition system. The input speech samples are used for system input is 2400 samples. The clock of the system is.

### 4.1 Feature Extraction

The proposed algorithms as presented in section 3 and traditional approachs are studied in program implementation speed and also the recognition accuracy. The obtained results are presented in Table 1.

As we can see in Table 1, the pre-emphasis block with  is executed fastest. The value  in the

Table 1. Results of program execution speed and recognition accuracy.

| Block | Parameter /Algorithm | Clock cycles (cycles) | Recognition accuracy (%) |
|---|---|---|---|
| Pre_emphasis | 1 | 2,078,463 | 85.10 |
| | 15/16 | 2,155,870 | 86.14 |
| | 0.97 | 2,156,018 | 86.29 |
| FFT/DFT | FFT with LUT | 94,874,620 | 86.14 |
| | DFT | 365,586,715 | 87.27 |
| Magnitude computation | Estimation amplitude | 7,463,640 | 86.14 |
| | Accuracy amplitude | 80,412,716 | 89.40 |
| Mel filter bank | Rectangle filters | 418,427 | 86.14 |
| | Triangle filters | 19,317,411 | 87.33 |

pre-emphasis block run faster than the pre-emphasis block with . These results compensate in recognition accuracy. From these results, the value 15/16 of coefficient "a" is proposed to be used in recognition system.

In the Fourier transform step, DFT algorithm is replaced by the FFT algorithm with LUT (Look-Up-Table) to increase the speed of execution. As shown in the Table 1, FFT algorithm runs faster than DFT algorithm very much but trade-off in recognition accuracy reduction of 1.13%. Therefore, the FFT with LUT is proposed to be used in the system.

In the magnitude computation step (the results in Table 1), the estimation algorithm calculates very fast amplitude of a complex number almost exact compared to the traditional algorithm by taking the square root operation. However, the recognition accuracy is reduced of 3.26% by using estimation amplitude algorithm in the system.

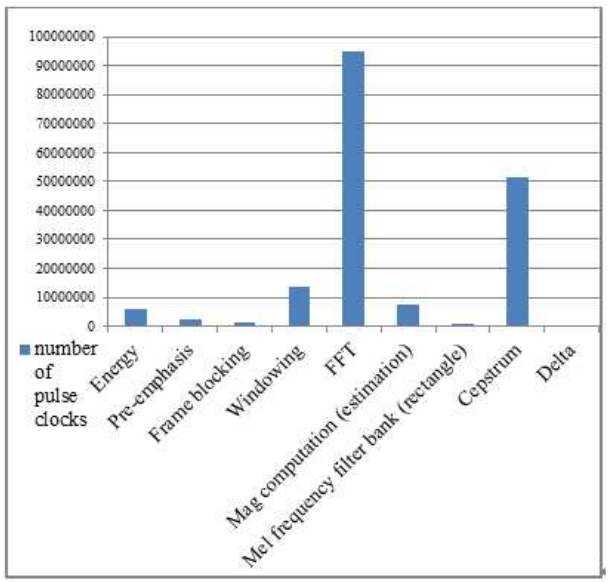


Fig. 7. Program execution speed of the blocks in MFCC based feature extraction.

By using the rectangle filters to replace the triangle filters, the program execution speed of the Mel Frequency Filter Bank block is increased larger 46 times than using the triangle filters, as in Table 1, but it gives lower recognition accuracy than using triangle filters about 1.19%. Therefore, we propose using rectangle filters in speech recognition system.

In Fig. 7, the program execution speed of all blocks in MFCC based feature extraction is shown to help the designer define which block consumes the longest time. The FFT block is the slowest, requires 94,874,620 clock cycles to complete the given input samples. The Cepstrum block also costs many clock cycles because in this block the logarithm has not been optimized.

### 4.2 Vector Quantization

With codebook size of 128, Vector Quantization is used in the recognition step and it costs 531,067,721 clock cycles.

### 4.3 Recognition accuracy

The whole recognition system with proposed architectures, parameters as stated above has the recognition accuracy of 87%, in which 7,416 utterances recorded from male and female adults in three regions of the North, Middle, and South of Vietnam are used.

## 5. Conclusion

In this paper, we propose efficient architectures and design choices for each part in MFCC-VQ-based speech recognition system to improve the processing speed. The recognition accuracy results are also presented in different parameters. The determination of design choices are based on the easiness in implementation and experimental results in execution speed, recognition accuracy of whole system. We propose to use value 15/16 of coefficient "a" in pre-emphasis, FFT with LUT, estimation amplitude algorithm, rectangle filters thanks to fast execution speed and acceptable recognition accuracy.

## References

[1] Lawrance Rabiner & Biing - Hwang Juang: "Fundamentals of Speech Recognition", Prentice Hall PTR, 1993.

[2] Thomas Hain, Gunnar Evermann, Dan Kershaw, Gareth Moore, Julian Odell, Dave Ollason,

Dan Povey, Valtcho Valtchev, Phil Woodland, Steve Young: “The Hidden Markov Model Toolkit (HTK) Book (for HTK version 3.2.1)”, Cambridge University. Available at: http://htk.eng.cam.ac.uk/ (1995 - 2002).

[3] V. Amudha, B.Venkataramani, R. Vinoth kumar, S. Ravishankar: “Software/Hardware Co-Design of HMM based Isolated Digit Recognition System.” In: Journal of Computers, VOL. 4, No. 2, pp. 154-159, (2009).

[4] Haitao Zhou, Xiaojun Han: “Design and Implementation of Speech Recognition System Based on Field Programmable Gate Array”. In: Modern Applied Science, Vol. 3, No. 8, pp. 106-111, August 2009.

[5] Wei Han, Cheong-Fat Chan, Chiu-Sing Choy, Kong-Pang Pun: “An Efficient MFCC Extraction Method in Speech Recognition.” In: the 2006 IEEE International Symposium on Circuits and Systems (ISCAS), pp. 145-148, Greece (2006).

[6] Wei Han: “A Speech Recognition IC with an Efficient MFCC Extraction Algorithm and Multi-mixture Models”, the Chinese University of Hong Kong, Doctor of philosophy thesis, September 2006.

[7] S.-T. Pan, C.-C. Lai and B.-Y. Tsai: “The implementation of speech recognition systems on FPGA - based embedded systems with SOC architecture”. In: International Journal of Innovative Computing, Information and Control, Volume 7, Number 10, October 2011.

[8] O. Cheng, W. Abdulla, Z. Salcic: “Hardware-Software Co-design of Automatic Speech Recognition System for Embedded Real-Time Applications”. In: IEEE Transactions on Industrial Electronics, pp. 850-859, March 2011.

[9] Weiqian Liang, Hui Geng: “Design of speech recognition co-processor with fast Gaussian likelihood computation”. In: the 3rd International Conference on Computer Research and Development (ICCRD), pp. 392-395, March 2011.

[10] Ge Zhang, Jinghua Yin, Qian Liu and Chao Yang: “A real-time speech recognition system based on the Implementation of FPGA”. In: Cross Strait Quad-Regional Radio Science and Wireless Technology Conference (CSQRWC), pp. 1375-1378, July 2011.

## BIOGRAPHY

### Hoang Trang

2002 : BS degree in Electronics Engineering, University of Technology HoChiMinh City, VietNam

2004 : MS degree in Electronics Engineering, University of Technology HoChiMinh City, VietNam.

2009: PhD degree in Microelectronics Engineering, CEA-LETI, France, University Grenoble 1, France.

2009-2010: Post-doctorate in France Telecom, France.

2010-present: lecturer at University of Technology HoChiMinh City, VietNam

### Tran Van Hoang

2012 : BS degree in Electronics Engineering, University of Technology HoChiMinh City, VietNam.

2012-present: Research Engineer at University of Technology HoChiMinh City, VietNam.