The Implementation of Automatic Dialing System Using MX96037 *

An-Nan Suen, Jhing-Fa Wang, and Horng-Jei Chang

Institute of Information Engineering
National Cheng Kung University, Tainan, Taiwan, R.O.C.
suenan@vlsi2.iie.ncku.edu.tw, wangjf@server2.iie.ncku.edu.tw

Abstract

In this paper, an automatic voice dialing system with MX96037 DSP using the speech recognition technology is presented. This merits of the proposed voice dialing system as follows, (1) low memory required, (2) real-time performance, (3) high recognition accuracy (4) low cost hardware required, and (5) high capacity of phonebook size. It can be a practical system for the automatic dialing especially for the mobile communication. In addition, the issues and accuracy study for the fixed-point realization of the word-based speech recognition system are also addressed in this paper.

1 Introduction

The automatic speech recognition (ASR) technology can be used in the telephone system especially the mobile communication. By porting the ASR to the general purpose DSP chips is the most widely used solution for the automatic dialing system (ADS) such as TMS320C50 of Texas Instruments [7], ADSP2181 of Analog Device [5] [6], and the MX96037 of MXIC [8]. In order to achieve a practical system , the cost and the performance of the DSP chip are the main considerable factors while selecting the body. In this research, low cost, high performance fixed-point (16-bit) MX96037 is selected as the body for porting the ASR algorithm. To implement a hands-free dialing system based on the speech recognition technology using MX96037 will meet the constrains of cost and performance.

We implement an automatic voice dialing system using MX96037 DSP. First, a word-based speech recognizer with floating-point operations on IBM PC is developed for the performance testing. In the training phase, 3 utterances for each keyword are recorded to be the training data. The first 8 of 12th order weighted cepstrum coefficients are taken to be feature vectors. In the next step, we use the fixed-point format variables instead of the floating-point one for porting the system on MX96037. In addition, the accuracy studies for finite word length are also addressed in this paper. The whole system is developed on MX96037 EVM and DAM board.

This paper is organized as follows. Section 2 briefly gives an overview of the ASR technology. Section 3 describes the issues of the fixed-point implementation of the ASR. Section 4 discusses the implementation on MX96037 DSP chip including the system architecture and optimal solution for the voice dialing system. In section 5, we evaluate the performance of the automatic voice dialing system and compare with other similar products. Finally, we make a conclusion in section 6.

2 Automatic Speech Recognition

A word-based speech recognizer is developed for the automatic dialing system . Four interleave factors should be considered while developing the word recognizer, real-time, low cost, small memory size, and high accuracy rate. Due to such a system is designed for the family's telephone system and mobile communication so the cost and accuracy rate are the most important factors while developing the automatic dialing system (ADS). According to the constrains of cost and memory size, the ASR employs the speaker-dependent (SD), word-based speech recognition technologies. The maximum phonebook size can be achieved to 200 keywords including basic and user define keywords. The 12 basic keywords are '0', '1', '2', ..., '9', 'wrong', and 'dial'. In addition, the user can also define the keywords in the phonebook such as name, or affiliation whom the user often dials to. These user keywords must be trained by uttering 3 times for each keyword before dialing.

The endpoint of input speech is automatically detected with short-term energy and zero crossing rate [?]. A fixed first-order filter is used as a preemphasizer with a preemphasis factor of 0.9375. The input signal is filtered by the preemphasizer. The preemphasized speech signal is blocked into frames of 256 samples(32 ms) in length with adjacent frames being separated by 128 samples(16 ms). Each frame of speech is weighted by a Hamming window. A 12th order linear predictive coding(LPC) coefficients are used to represent the short-time waveform. The weighted cepstrum coefficients are employed to be the feature parameters. The weighted cepstrum coefficients can be obtained by Eq. (1).

$$\hat{C}_n = W_n(-a_n - \sum_{m=1}^{n-1}(1 - m/n)a_m C_{n-m}) \quad 1 < n \leq p \tag{1}$$

*This work was supported by National Science Council, Republic of China, under Grant NSC84-2622-E-006-006

where a_i's are the LPC coefficients, p is order number and the weighting function W_n is expressed as Eq. (2),

$$W_n = 1 + \frac{h}{2} sin(\frac{n\pi}{h}) \quad 1 \leq n \leq p. \tag{2}$$

The feature parameters $\hat{C}_n$, are then used for the training data. In training phase, the key word is read triple to be training data. After extracting the feature factor of each frame, we represent each training data by feature vector series. For the similarity of speech signal in the short time, each series can be divided into 8 states. Then the mean and variance of all feature vectors can be calculated in every state. That is, we get 8 means and 8 variances to be the reference pattern for one keyword. In recognition phase, we get the feature vector series of the input speech. One feature vector maps to one state. We can get the observation probability to the mapped state of the feature vector using the state mean and variance in the ith reference pattern. Gaussian probability distribution function is applied to get the observation probability. Multiply the observation probabilities of feature vectors, we can get the overall probability to reference pattern i. Taking the reference pattern with the maximum m probability, we get the matched pattern. The training and recognition program is developed in C programming language on IBM compatible PC486DX-33. The program uses floating-point operations. The speech signal is 8-bit sampled with 8kHz sampling rate.

3 Fixed-point Implementaiton

According to MX96037 is a fixed-point DSP and has no floating-point arithmetic unit, so the word recognizer should be written in fixed-point. The interword length should be 32-bit or 16-bit with signed representation. The fractional representation is used to represent the fixed-point number. The number $m.n$ represents m-bit integer part and n-bit fractional part. In the two addition and subtraction operations, the radix point of both operands should be in the same position, that is, they must be the same format. In multiplication, these two operands need not be the same. The product of the two operands $m.n$ and $s.t$ is the format of $(m+s).(n+t)$. First, we divide the whole program into stages. Then observe the dynamic range of variables in first stage with variable inputs and determine the format of fixed-point variables. Once this stage is analyzed, feed the output into the next stage and continue analysis until all operations are fixed-point. PARCOR(partial autocorrelation) method, also called lattice method, is used for computing LPC coefficients often under finite word length computation environment. Although it is more complex and requires more memory than Durbin(autocorrelation) method, PARCOR method can guarantee the stability of LPC coefficients using finite bit length. After numerical analysis of PARCOR method, we can use 16-bit integer arithmetic operations to implement LPC analysis. The formats of variables in PARCOR method are listed in Table 1.

Variable	Maximum value	Minimum value	Format
$e^{(i)}(m)$	169.93	-205.83	9.7
$b^{(i)}(m)$	203.76	-174.39	9.7
k_i	0.995	-0.97	1.15
$\alpha_j^{(i)}$	2.99	-2.72	3.13
LPC coefficients	2.81	-2.59	3.13
Cepstrum coefficients	2.07	-1.13	3.13
Weighted Cepstrum coefficients	7.54	-7.02	4.12

Table 1: Dynamic range and the data format for the speech recognition system.

We first implement the speech recognizer using C language with floating-point for testing performance. Using the fixed-point format to instead of the floating-point version is the most important and hard step in the procedure of software simulation, because it should keep the performance as well as the latter under a shorter precision. In the next step, rewrite the fixed-point software to the MX96037 assembly program for porting the speech recognizer on MX96037 DSP chip. It's more an art than science for transforming a floating-point program into fixed-point one. The procedure of implementing the fixed-point program will be described as follows:

Step 1. Partition the speech recognition algorithm into n modules, $M_1, M_2, \ldots, M_n$: For example, we divided the whole system into data input , Hamming window, LPC, cepstrum, weighted cepstrum, etc.

Step 2. Determining the lower-bound and upper-bound for each module: The speech test database was used to determine the dynamic range and the upper-bound and lower-bound for each module. The internal word length is 16 bits. Using the fraction representation, $m.n$ where $m + n = 16$ to represent the data format with m-bit integer and n-bit fraction

Step 3. Overflow and underflow detection and avoidance: In some special case the overflow and underflow will happen, even if the data format is determined in step 2. So we add the ability of detecting the overflow/underflow and avoiding the quantization error propagating to next stage.

Step 4. Error measurement for each Stage (local test) : In the local test, SEGSNR is used for evaluating the quantization error by comparing their output with the output of the corresponding floating-point routines.

Step 5. System performance measurement (global test): In the global test, the accurate rate is used to evaluate the performance for the fixed-point word recognizer.

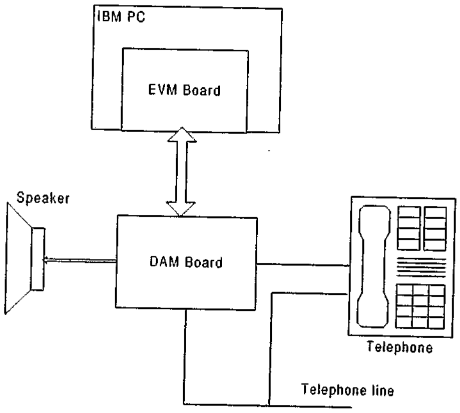

Figure 1: System block diagram of the automatic dialing system.

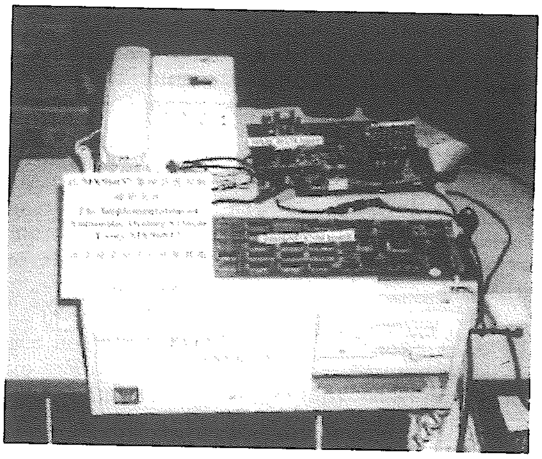

Figure 2: The photograph of the automatic dialing system.

4 Implementation on MX96037 DSP

The speech recognition system will be ported on the DSP chip MX96037 of MXIC for realizing the voice dialing system. Fig. 1 shows the system block of the automatic dialing system. The whole development environment consists of MX96027 EVM and DAM boards. MX96037 EVM is an emulation board which contains 32K words SRAM, ISA connector for downloading program, a μ-law (CCITT G.711 standard) CODEC and one mega*4bit DRAM for storing the massive data. The DAM board is a target board providing three I/O interfaces including microphone interface, speaker interface, and telephone line interface. A telephone handset is used for the voice input . Fig. 2 shows the whole development of the ASD. Two operation modes, training and recognition. In the training mode, the user can train the phonebook by three utterances for each keyword. After the training phase, user can enter the recognition mode to dial out by voice.

The DAM Board is noise sensitive and causes the

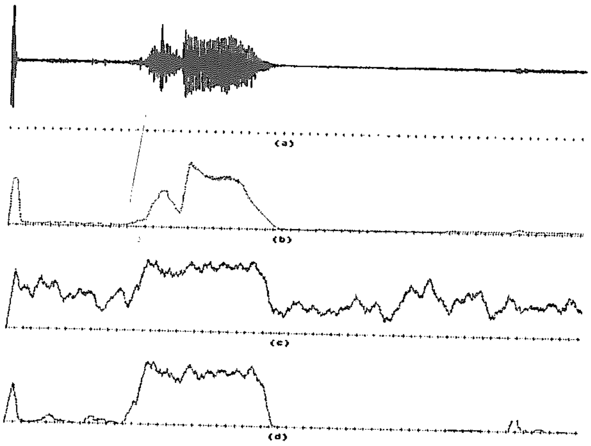

Figure 3: The improvement of the detecting the beginning and ending points by modifying the zero crossing rate equation. (a) is the original waveform, (b) is the energy contour, (c) and (d) are the zero crossing rate contour before and after improvement respectively.

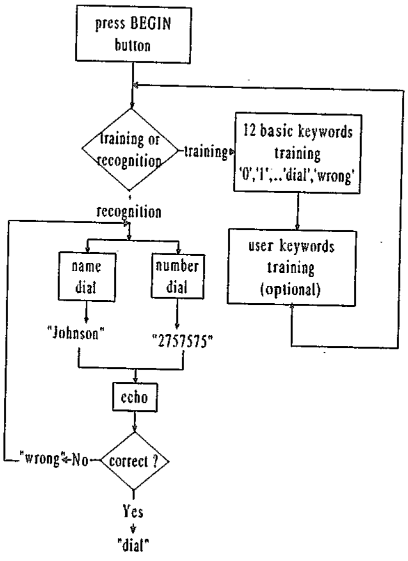

Figure 4: The user interface flow of the automatic dialing system.

high zero crossing rate (ZCR) even in the silence status. So the ZCR definition is to be modified for avoiding the noise influence on the accuracy rate. The original definition of ZCR is $S(n)S(n-1) \leq 0$, and the modified the ZCR definition is $S(n)S(n-1) \leq TH$, where TH is the threshold value depended on the environment and the input device. Fig. 3 shows the results of canceling the noise. The input speech waveform and the energy contour are shown in Fig. 3(a) and (b) respectively. Fig. 3 (c) and (d) shows the contour of the original and modified ZCR respectively with $TH = -4096$. This experiment results shows the modified ZCR will improve the effect while determining where is the beginning and ending points.

The word recognizer is ported on the MX96037. The major modules of ASR are: (1) training and recognition kernel, (2) CODEC interrupt service routine(ISR) (3) system timer ISR, and (4) system initialization and maintain module.

1. **Training and recognition kernel** : This kernel is ported from the fixed-point C language program. Using special instructions of MX96037 can drastically reduce the computation time. By using the **mxa** (multiply and accumulate) or **mxs** (multiply and subtract) for parallel operations can reduce the cycle number. Hardware supported loop instruction, **lup**, can reduce the loop overhead. Then the processes of feature extraction and recording can be done at the same time.

2. **CODEC ISR** : This interrupt is triggered at the frequency of 8kHz. As the interrupt is enable, the system will read the μ-law sampled speech signal from CEDEC and then expanding the samples to be linear for processing in MX96037. In a reverse process for playing the signal to the speaker, compressing the linear signal to be the μ-law samples and then sending to CODEC IC through the CODEC interface. Besides recording and playing the speech signal, the ISR can also be utilized for determining the ending points, sending the prompt sound, and generating the DTMF dial tone.

3. **System timer ISR** : This interrupt is triggered at every 1 ms, and can be used for keypad scanning, LED displaying and refreshing, and being a counter.

4. **System initialization and maintain module** : In the initial state, the wait state, control register (CTLR), interrupt mask register (IMR), will be set. The working mode of MX96037 is determined by the CTLR, including the working frequency, CODEC sampling rate, Each bit of the IMR is corresponding to one interrupt, and can set the interrupt enable or disable.

Frequency(Hz)	1209	1336	1477
697	1	2	3
770	4	5	6
852	7	8	9
941		0	#

Table 2: DTMF coding table corresponding to the telephone key pad.

	Proposed System	Name Caller	VoiceDialer	
			2030D	2060D
DSP Body	MX96037	-	TMS320C5X	
Phonebook Size	200	64	20	50
Accuracy Rate with 16 keywords	98.5	98.0	-	
Seek Time	0.03sec	real-time	real-time	

Table 3: The comparison list among our proposed dialing system with other automation dialing products.

Dual-Tone Multi-Freauency(DTMF) is the coding method for dialing. Two frequencies represent one keypad to generate the dialing tone. The DTMF coding table is shown in Table 2. The μ-law of the two sine value should be generated and set to the CODEC IC to generate one dial tone. The sine wave function can be obtained by using the polynomial approach expressed as Eq. (3). By the polynomial method will generate the correct dial tone.

$$\begin{aligned} sin(x) &= 3.140625x + 0.02026367x^2 - 5.325196x^3 \\ &\quad +0.5446778x^4 + 1.800293x^5 \end{aligned}$$

where $0^0 \leq x \leq 90^0$. Using the polynomial approach instead of the look -up table method will save much memory size.

The proposed ASR has two operation modes training and recognition modes. User needs train the 12 basic and the user keywords by uttering each keyword 3 times. After the training phase, then the user can enter the recognition mode to dial some one. Two dialing modes can be used. Number dial is for the user to call someone by uttering the number and another is name dial by speaking the name or the affiliation for dialing. For example, we want to dial the number "2757575",

- User : "2757575",
- ADS : echo the recognition result,
- User : "dial" or "wrong".

In this example "..." is uttered by user to the ADS, if the recognition result is correct than user speak "dial" to dial else speak "wrong" to cancel the process. Besides the number call, user also can specify the name tags (for example, office, home, school, etc) for calling out. The flow of the user interface is shown in Fig. 4. Finally, we make a comparison among the proposed ADS and other products and the comparison results are shown in Table 3.

	Accuracy Rate		
Tester	Floating-point C version	Fixed-point C version	MX96037 DSP version
a	94.0	90.0	82.5
b	90.0	84.0	80.5
c	92.5	85.5	81.5
d	89.3	83.0	81.0
e	93.0	87.5	84.0
mean	91.8	86.0	81.9

Table 4: The accuracy rate comparison list between the floating-point, fixed-point, and the MX96037 DSP versions with the microphone input.

	ASR Accuracy Rate on MX96037			
	Phonebook Size			
Tester	16 words	50 words	100 words	200 words
Male	98.2	96.8	93.8	80.6
Female	98.8	98.0	95.8	81.6
Mean	98.5	97.4	94.8	81.1

Table 5: The experiment results of the automatic dialing system with various phonebook size, 16, 50, 100, 150, and 200 keywords as indicated.

5 Experiments

There are 5 tester including 3 female and 2 male to test the proposed ADS . In the first experiment, we compare the accuracy rate between three versions of the ASR including floating-point, fixed-point, and MX96037 with the maximum phonebook size 200. The first tow versions are run on the IBM PC DX-33 with C language. The fixed-point version has high accuracy rate shows that the strategies of the converting from floating-point to fixed-point are effective and efficient. Table 4 shows the accuracy rate comparisons between three versions of ASR. The next experiment is the accuracy study on the MX96037 with various phonebook size 16, 50, 100, and 200 keywords. In this experiment we instead the microphone with the handset as the input device for more real simulation. The accuracy rate of the proposed ADS with various phonebook size is shown in Table 5. According to the above experimental results, the proposed ASD has high performance using the low cost MX96037 DSP chip.

6 Conclusion

The implementation of an automatic dialing system with maximum phonebook size 200 by using the DSP chip MX96037 is proposed in this paper. Here, low-cost, real-time and high performance hands-free dialer is implemented and can be applied to the family telephone system and the mobile communication for more easy and safe use. In this paper, the issues of the fixed-point implementation for the ASR technology are also discussed. The whole system can be added in the telephone set for hands-free dialing. The ASR can also be applied for controlling the consumer electronics by voice. Some further works about the ADS will be studied continuously including the speaker-independent keywords, and robust word recognizer under noise background.

References

[1] T. Watanabe, "Speaker-independent word recognition using dynamic programming matching with statistic time warping cost," *Proc. Int. Conf. Acoust., Speech, Signal Processing*, 1988, pp. 195-198.

[2] C. H. Wu, J. F. Wang, C. C. Haung, and Jau-Yien Lee, "Speaker independent recognition of isolated words using concatenated neural networks," *Int. Journal of Pattern Recognition and Artificial Intelligence, Vol. 5, No. 5*, 1991, pp. 693-714.

[3] R. C. Shyu, J. F. Wang, and C. H. Wu, "A robust connected mandarin digit recognizer based on Bayesian template," in *Proceedings of 1992 Int. Conf. on Computer Processing of Chinese and Oriental Languages*, Florida, USA, pp. 100-107.

[4] J. Makhoul, "Stable and efficient lattice methods for linear prediction," *IEEE Trans. Acoust. Speech, Signal Processing*, vol. ASSP-25, no. 5, Oct. 1977, pp.423-428.

[5] Analog Devices, Inc. Digital signal processing applications using the ADSP-2100 family volume 1, *Prentice-Hall*, New Jersey, 1992.

[6] Analog Devices, Inc. Digital signal processing applications using the ADSP-2100 family volume 2, *Prentice-Hall*, New Jersey, 1995.

[7] Texas Instruments, TMS320C5X user's guide, 1991.

[8] Macronix Inc., MX96037 user's guide, 1994.

[9] C. S. Myers, and L. R. Rabiner, "A level building dynamic time warping algorithm for connected word recognition," *IEEE Trans. Acoust. Speech, Signal Processing*, vol. ASSP-29, no. 2, Apr. 1981, pp. 284-296.

[10] C. S. Mayers, and L. R. Rabiner, "Connected digit recognition using a level-building DTW algorithm," *IEEE Trans. Acoust. Speech, Signal Processing*, vol. ASSP-29, no. 2, Jun. 1981, pp.351-363.

[11] R.C. Shyu, J. F. Wang, C. H. Wu, S. N. Tsay, and J. Y. Lee, "A connected Mandarin digit recognizer," *Proc. Int. Conf. Microwaves and Communications*, Nanjing, China, 1992.

[12] H. Ney, "The use of one-stage dynamic programming algorithm for connected word recognition," *IEEE Trans. Acoust. Speech, Signal Processing*, vol. ASSP-32, Apr. 1984, pp. 263-271.