Dual-ALU Processor for Speech Signal Processing

Gin-Der Wu and Kuei-Ting Kuo
Department of Electrical Engineering
National Chi Nan University
Puli, Taiwan, R.O. C.
ginderwu@ncnu.edu.tw

Abstract

This paper proposed a low-cost dual-ALU processor for speech recognition. It does not give an emphasis on sophistication but on low-cost solution. The dual-ALU architecture provides parallel calculation capability. For the consideration of chip size, the area of the second ALU is only half of the first ALU. We use hardware-software codesign method to implement the speech recognition. The feature extraction bases on LPC-cepstrum coefficients, and template matching employs Hidden Markov Models (HMM). The processor is designed to process the HMM and connect with the ASIC of LPC-cepstrum.

Keywords: dual-ALU, parallel calculation, hardware-software codesign, LPC-cepstrum, Hidden Markov Models.

1. Introduction

The implementation of the speech recognition can be classified as software design [1]-[3] and hardware-software codesign method [4][5]. Although the high-performance Digital Signal Processor (DSP) can speed up the recognition time, the cost is relatively high. The hardware-software codesign method can solve this problem [4]. We proposed the dual-ALU architecture processor to provide the parallel calculation capability. In addition, we utilize the ASIC of LPC-cepstrum. This ASIC of LPC-cepstrum can reduce the computation load of processor. To reduce the area of a chip, the resource sharing method is adopted into the design of LPC-cepstrum. In addition, since the proposed high speed Dual-ALU processor has the parallel calculation capability, it can run more complicated algorithm of speech recognition. For comparison, we obtain the recognition rate by the single-ALU processor and dual-ALU processor, respectively. From the experiments, since the dual-ALU can efficiently reduce the run time, the dual-ALU structure outperforms the single-ALU structure.

Proceedings of the 22nd Conference on Computational Linguistics and Speech Processing (ROCLING 2010), Pages 311-319, Puli, Nantou,Taiwan, September 2010.

2. Dual-ALU Architecture Processor

The flowchart of the speech recognition system is shown in Fig. 1.

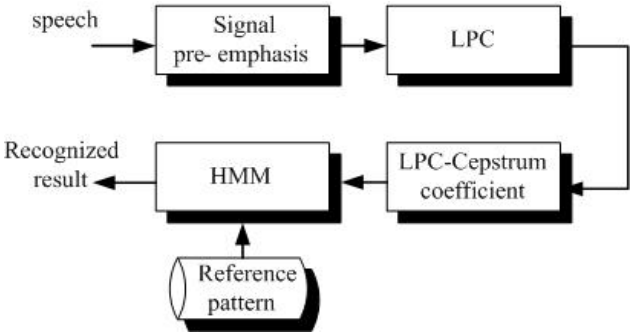

Figure 1. The flowchart of the speech recognition system

We proposed the dual-ALU DSP processor structural block diagram in Fig. 2.

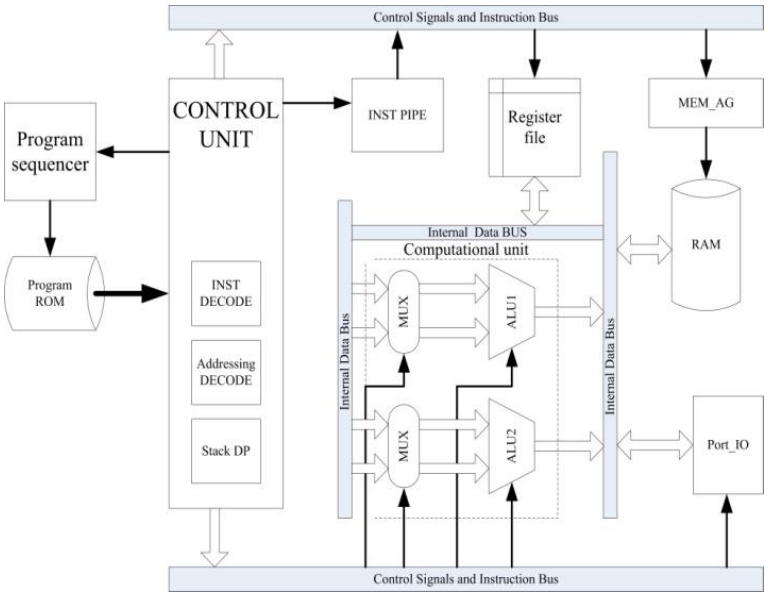

Figure 2. The dual-ALU processor structural block diagram.

It can be divided into the control unit, data register file, computational unit, memory address generator, embedded data memory and program sequencer. The computational unit performs the numeric processing of speech recognition. These paths get data from registers in the data register file. In addition, computational instructions provide fixed-point operations, and each instruction can completely execute in one clock cycle.

The architecture of the dual-ALU processor is shown in Fig. 3. It can operate two instructions at the same time.

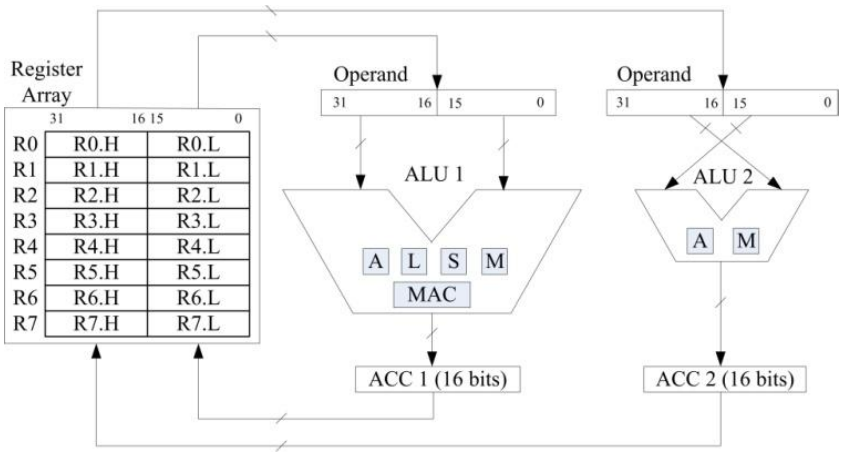

Figure 3. Dual-ALU architecture.

The first ALU (ALU1) performs arithmetic, logic, multiplication, shift, execute multiply/add and multiply/subtract operations. The second ALU (ALU2) performs arithmetic and multiplication operations. Data flow paths through the computational units are arranged in parallel. The output of any computational unit may serve as the input of any computational unit in the next instruction cycle. Data moving in and out of the computational units goes through a register file which consists of eight registers. By using the parallel data paths within the dual-ALU, this processor can support the parallel computation. Hence we can use single instruction to get two operation results from the dual-ALU. The usage is shown as follows:

```
MOV     #32,R0.L
MOV     #23,R0.H
MOV     #42,R1.L
MOV     #9, R1.H

ADAD    R0.HL, R1.HL, R2.H, R2.L      ; R2.H = R0.H  + R1.H = 31
                                      ; R2.L = R0.L  + R1.L = 74
ADSB    R0.HL, R1.HL, R3.H, R3.L      ; R3.H = R0.H  + R1.H = 31
                                      ; R3.L = R0.L  - R1.L = -10
ADMPY   R0.HL, R1.HL, R4.H, R4.L      ; R4.H = R0.H  + R1.H = 31
                                      ; R4.L = R0.L  * R1.L = 1444
SBAD    R0.HL, R1.HL, R5.H, R5.L      ; R5.H = R0.H  - R1.H  = 14
                                      ; R5.L = R0.L  + R1.L = 74
SBSB    R0.HL, R1.HL, R6.H, R6.L      ; R6.H = R0.H  - R1.H = 14
                                      ; R6.L = R0.L  - R1.L = -10
SBMPY   R0.HL, R1.HL, R7.H, R7.L      ; R7.H = R0.H  - R1.H = 14
                                      ; R7.L = R0.L  * R1.L = 1444
MPYAD   R0.HL, R1.HL, R8.H, R8.L      ; R8.H = R0.H  * R1.H = 207
                                      ; R8.L = R0.L  + R1.L = 74
MPYSB   R0.HL, R1.HL, R9.H, R9.L      ; R9.H = R0.H  * R1.H = 207
                                      ; R9.L = R0.L  - R1.L = -10
MPYMPY  R0.HL, R1.HL, R10.H, R10.L    ; R10.H = R0.H  * R1.H = 207
                                      ; R10.L = R0.L  * R1.L = 1444
```

The instruction set is the interface between the hardware and the software. The performance of the processor depends on the instructions. The instruction set is 32-bit formats. We define seven instruction types including Data-transfer, Boolean, Add/Sub, Mac, Dual-ALU, Shift/Rotate, and Control.

For the considerations of testability in circuit, there are two auxiliary circuits for DFT. First, we created Built-In-Self-Test (BIST) [6] circuit for the embedded memory of SRAM. The BIST controller uses the test algorithm to write test patterns into embedded memory. Then it reads each data from embedded memory in order to compare with the test pattern. Hence we can detect any error in the embedded memory. Secondly, one scan chain is inserted into each module of the processor by using DFT-compiler. In this part, all sequence logic includes BIST circuits. With the scan test, the chip can be tested easily.

3. Speech Recognition

In this section, we used the ASIC hardware to implement the lpc-cepstrum coefficient. This hardware can reduce the load of the proposed dual-ALU processor. The block diagram of this hardware is shown in Fig. 4.

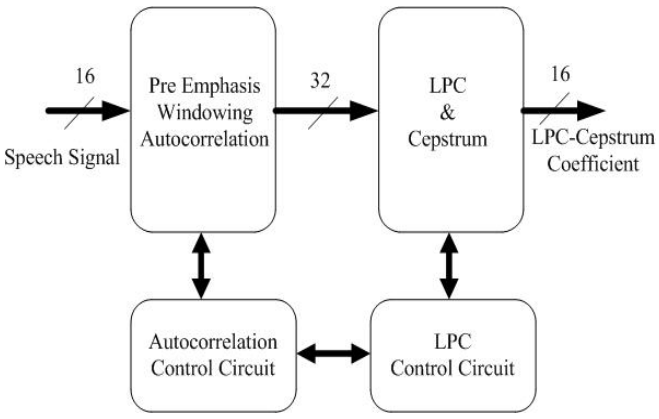

Figure 4. The block diagram of the ASIC of LPC-cepstrum.

In Fig. 5, the proposed LPC-cepstrum circuit contains three sections, the autocorrelation, the linear predictive coding, and the cepstrum. Each section has its corresponding control circuit which handles the transmission of data in the circuit.

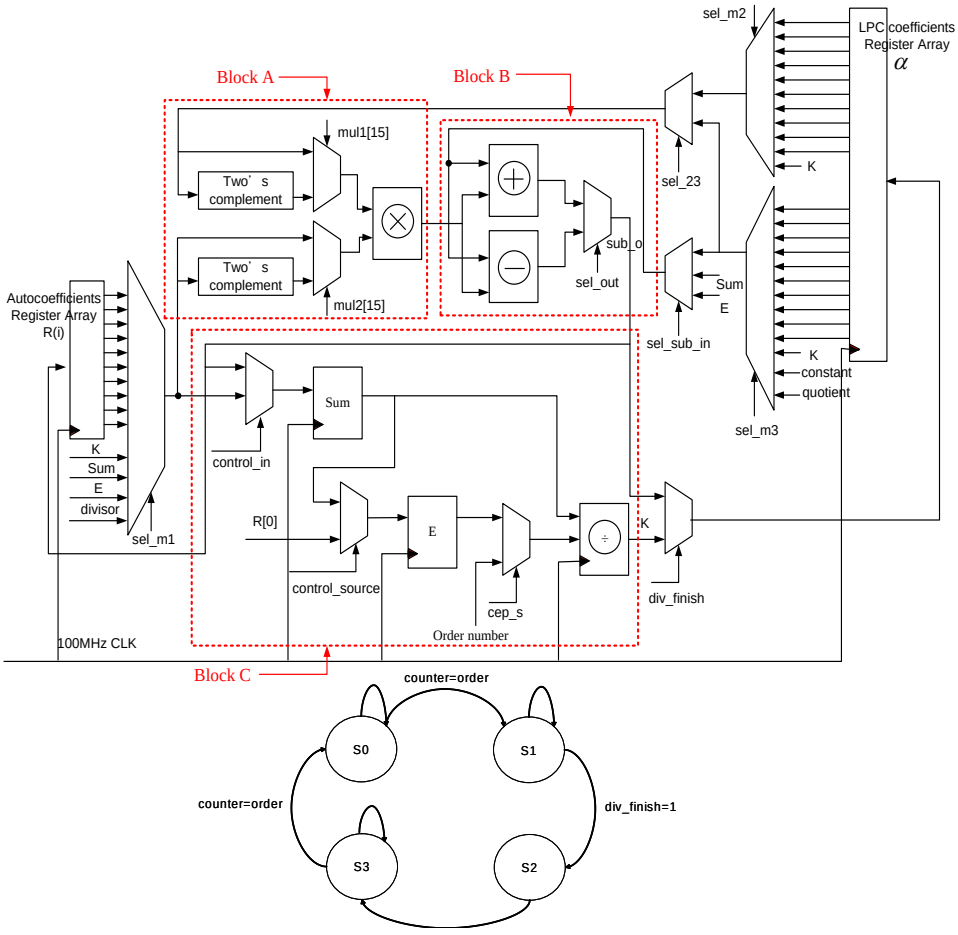

Figure 5. The hardware architecture of the proposed LPC-cepstrum.

The data type in the ASIC of LPC-cepstrum is shown in Fig. 6. In Fig. 7 (a), we build the protocol to connect the LPC-cepstrum circuit with the proposed dual-ALU processor. Fig 7 (b) shows the waveform of this protocol. When the port of “Busy” and “INT” are “High”, the LPC-cepstrum will be sent to “IN" port of the dual-ALU processor. After receiving the 10th transmission data, the port of “Busy” becomes “Low”. Then the dual-ALU processor begins to implement the HMM algorithm.

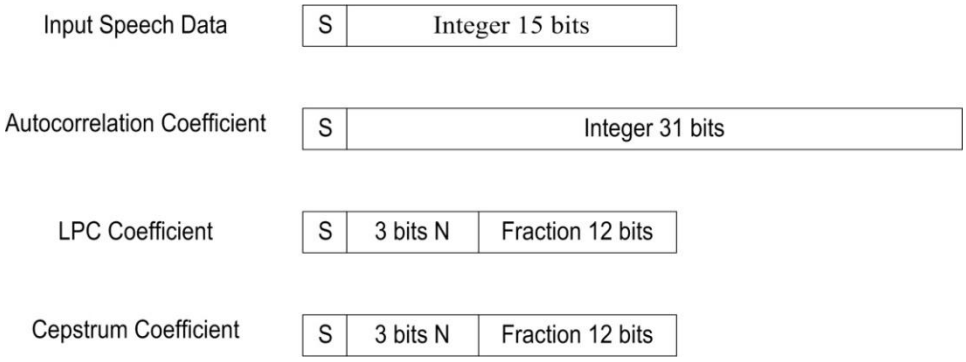

Figure 6. The data type in the ASIC of LPC-cepstrum.

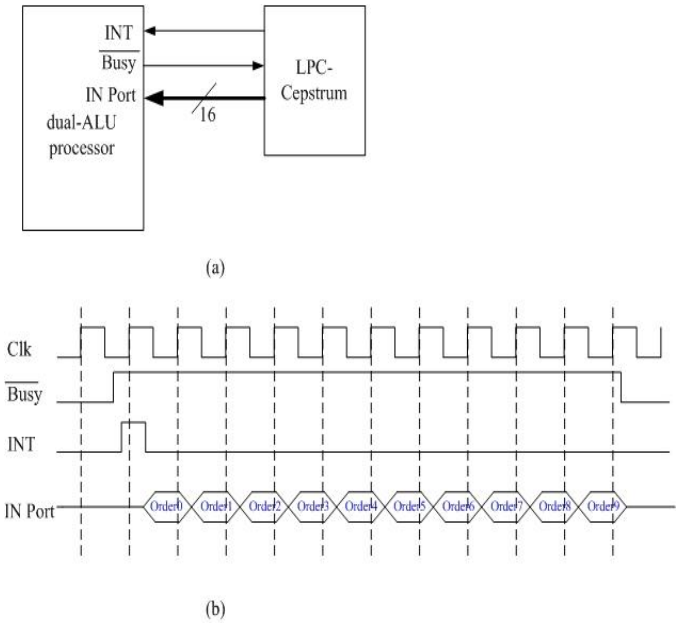

Figure 7. (a) Structure of the dual-ALU processor connect with the ASIC of LPC-cepstrum (b) Waveform of the protocol.

We use the HMM algorithm to execute the speech recognition. We define a new variable in (1). It is the best score along a single path at time t, which accounts for the first t observation and ends in state i. (2) and (3) are the initialization to the time and the state. (4) and (5) are the recursion to find the best state sequence. In addition, we use (7) and (8) to backtrack this sequence which has the biggest possibility.

$$\delta_i(i) = \max_{q^1,q^2,\cdots,q^{i-1}} P\left(q^1,q^2,\cdots,q^{i-1},q^i = s_i,o_1,o_2,\cdots,o_t|\lambda\right) \tag{1}$$

$$\delta_1(i) = \pi_i b_i(o_1), \quad 1 \le i \le N \tag{2}$$

$$\psi_1(i) = 0 \tag{3}$$

$$\delta_{t+1}(j) = \max_{1\le i\le N}\left[\delta_t(i)a_{ij}\right]b_j(o_{t+1}), \quad 1 \le j \le N \quad 1 \le t \le T \tag{4}$$

$$\psi_{t+1}(j) = \arg\max_{1\le i\le N}\left[\delta_t(i)a_{ij}\right] \tag{5}$$

$$p^* = \max_{1\le i\le N}\left[\delta_T(i)\right] \tag{6}$$

$$q_T{}^* = \arg\max_{1\le i\le N}\left[\delta_T(i)\right] \tag{7}$$

$$q_t{}^* = \psi_{t+1}(q_{t+1}), \quad t = T-1,T-2,\cdots,2,1 \tag{8}$$

We build up the speech model by using the Viterbi re-estimation algorithm. Finally, the speech model will be derived as shown in (9) and (10).

$$\overline{a}_{ij} = \frac{n(u_{ij})}{n(u_{j\bullet})} \tag{9}$$

$$\overline{b_j}(k) = \frac{n(u_{\bullet j}, o = v_k)}{n(u_{\bullet j})} \tag{10}$$

4. Experiments

The proposed dual-ALU structure processor is synthesized by UMC 0.18um 1P6M standard library. The chip works correctly in both RTL and gate-level simulation. The working frequency can reach 100MHz. The die size is 2.073 x 2.073 mm^2, and the 128-pin CQFP package is used. The core size of the processor is 1.481 x 1.483 mm^2. The gate counts is about 24K. For the speech recognition, the program performs 2849463 cycles (28.5 ms) in 100MHz. Since each frame size is 240 points (30 ms), this processor can run in real time.

We adopt chinese digit (0-9) words from 10 speakers. Everyone speaks 10 times. The speech signal is sampled in 8 KHz and 16-bit format. In Table 1, we obtain the recognition rate by the single-ALU processor and dual-ALU processor, respectively. For comparison, we use two different algorithms to implement the speech recognition. They are Dynamic Time Warping (DTW) and Hidden Markov Models (HMM). From the experiments, since the dual-ALU can efficiently reduce the run time, the dual-ALU structure outperforms the single-ALU structure. In addition, HMM has higher accuracy rate than DTW.

Table 1. Performance comparison between the single-ALU and dual-ALU processor.

	ROM(byte)	Cycle	Accuracy rate
Single-ALU(DTW)	4K	234189	90.83%
Dual-ALU(DTW)	4K	153476	91.5%
Single-ALU(HMM)	3.6K	4479279	94.95%
Dual-ALU (HMM)	3.6K	2849463	94.68%

5. Conclusion

This paper proposed a low-cost dual-ALU processor for speech recognition. We use hardware-software codesign method to implement the speech recognition. This dual-ALU architecture provides parallel calculation capability. In addition, the area of the second ALU is only half of the first ALU in order to reduce the chip size. The feature extraction bases on LPC-cepstrum coefficients, and template matching employs Hidden Markov Models (HMM). The dual-ALU processor is designed to connect with the ASIC of LPC-cepstrum. The processor can work at 100MHz. In our experiments, since the dual-ALU can efficiently reduce the run time, the dual-ALU structure outperforms the single-ALU structure.

References

[1] Phadke S., Limaye R., Verma S. and Subramanian K., "On Design and Implementation of an Embedded Automatic Speech Recognition System," *Proceedings of the 17th International Conference on VLSI Design*, pp.127-132, 2004。

[2] Shi Yuanyuan, Liu Jia and Liu Runsheng., "Single-chip speech recognition system based on 8051 microcontroller core," *IEEE Transactions on Consumer Electronics*, vol.47, Issue: 1, pp.149-153, Feb, 2001。

[3] Heungsuk Chin, Kim J. Kown Y., Lee K. and Sang-ll Yang, "Realization of speech recognition using DSP (digital signal processor)," *Proceedings of the IEEE international Symposium on Industrial Electronics*, vol.1, pp.508-512, June, 2001。

[4] Chow K.F., Liew S.C. and Lua K.T., "Thin client front-end processor for distributed speech recognition," *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, vol.2, pp.29-32, April, 2003。

[5] Rajeev Krishna, Mahlke S. and Austin T., "Memory system design space exploration for low-power, real-time speech recognition," *International Conference on Hardware/Software codesign and System Synthesis*, pp.140-145, 2004。

[6] TurboBIST-Memory SRAM Built-In-Self-Test Generator Reference Manual, *SynTest Technologies, Inc*. version 1.4, Jan 2004.