DSP를 이용한 음성인식기 구현

임 창 환, 문 철 홍, 전 경 남
광주대학교 컴퓨터 전자통신 공학부
전화 : 062-670-9923 / 핸드폰 : 018-625-7736

Implementation of Speech Recognizer using DSP(Digital Signal Processor)

Chang-Hawn Lim, Cheol-Hong Mun
Dept. of Computer and electronics Communication , Kwangju University
E-mail : our5778@hanmail.net

Abstract

In this paper, implementation of speech Recognizer system, Separated from Personal computer. By using DSP, this intends to extend the voice recognizing, limited into PC because of amount of data and calculations. For this performance The thesis uses the real time End point detector and organizes no additional device between human and the system, characteristic vector are that detects End point and voice from absolute energy and ZCR, that uses 12 difference Cepstrum from LPC, that uses the method to compensate the process of pattern separating and pre-calculated standard pattern limitation.

I. 서론

인간의 기본적인 수단인 음성을 기계에 인식하도록 하는 음성인식에 관한 연구가 근래에 들어 크게 활성화되고 있으며, PC와 자동차등의 다양한 분야에 사용 가능하도록 하는 실용화 연구가 진행 주에 있다. 본 논문에서는 PC로부터 분리하여 음성인식 시스템을 하드웨어로 구현하였다. 본 논문에서는 음성인식을 위한 특징 벡터로는 LPC로부터 유도한 12차의 Cepstrum계수를 사용하며 절대에너지와 영교차율을 이용하여 음성 구간 검출을 하여 인식률을 향상시키며, 인식 알고리즘은 동적 시간 정합(DTW)방식을 통해 시간 축에서 발생하는 차이를 보상하면서 두 음성간의 유사도를 측정하는 기법을 인식기에 사용하였다. 본 논문에서 사용된 프로세서는 TI사의 32비트 부동 소수점 프로세서인 TMS320C31이고 A/D컨버터로는 12비트의 AD678을 사용하여 인식 시스템을 구현하였다. 인식 단어로는 음성인식을 확인하기 위한 방법으로 모터를 구동하기 위한 단어를 선택하여 화자 독립으로 인식을 수행하였다. 본 논문의 구성은 Ⅱ장에서는 음성인식 시스템의 설명을 하였고, Ⅲ장에서는 음성인식 시스템의 하드웨어 구성을 다루고, Ⅳ장에서는 음성인식의 하드웨어적 구성에 따른 실험을 설명하며 Ⅴ장에서는 결론을 맺었다.

II. 음성인식 시스템

2.1 음성인식 시스템의 구성

본 논문에서는 실시간 음성인식 시스템을 구현하기 위하여 DSP(TMS320C31)을 사용하였다. DSP는 32비트 워드의 데이터 버스를 가지며 16M 어드레스 영역 중에서 부트로더 영역, 내부 주변 장치 영역, 내부 램 영역을 제외한 나머지는 메모리 및 I/O 확장영역으로 사용 할 수 있도록 구현하였다. 본 시스템에서는 외부 메모리를 256K바이트(ROM)와 160K 바이트(RAM)을 사용하도록 하였다. 어드레스 디코딩은 DSP의 실시간 처리에 맞게

빠른 동작 속도와 회로 수정이 용이한 EPLD를 사용하여 외부 메모리를 억세스 하였다. 음성 인식을 위한 전체 블록도는 그림 1에 보여주고 있다.

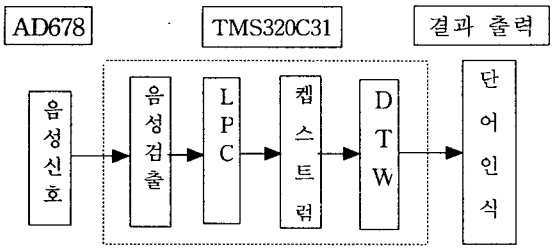

그림 1. 음성인식 시스템의 전체

2.1 실음성구간 검출

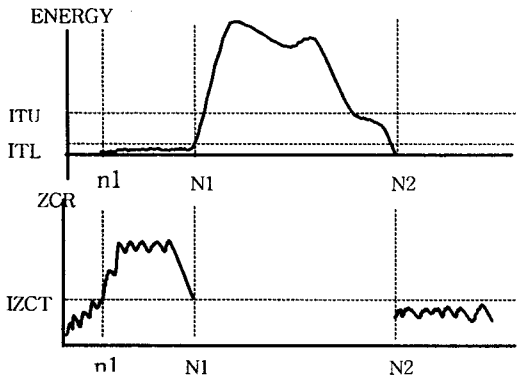

그림 2. 끝점 추출

$$E(n)=\sum_{m=n-N+1}^{n}|x(m)| \qquad \text{식(1)}$$

$$ZCR(n)=\sum_{m=n-N+1}^{n}|sgn[x(m)]-sgn[s(m-1)]| \qquad \text{식(2)}$$

그림 2는 식(1), 식(2)의 절대 에너지와 영교차율를 이용하여 끝점 추출하는 방법을 보여 주고 있다. 먼저 음성구간은 항상 ITU(threshold)를 초과한다. 따라서, 이 값 이상의 데이터를 갖는 것을 음성구간으로 여기고 시작점과 끝점은 이 간격 바깥에 위치한다고 가정한다. 시작점과 끝점은 영교차율를 이용하여 시작점은 N1앞에서 구하고 끝점은 N2 뒤에서 구한다. 영교차율이 정해진 문턱값 IZCT(rate이 3이상이다)보다 크기 시작하는 곳이 시작점이 되고, 같은 방법으로 N2이후에서 IZCT보다 값이 작아지기 시작하는 곳이 끝점이 된다.

2.2 특징벡터

음성을 인식하기 위해서는 음성이 가지는 특징을 나타내는 파라메터를 추출하여 이 특징 파라메터로부터 학습과 비교과정을 거쳐서 인식을 수행하게 된다. 따라서 음성인식에 관한 연구는 효과적인 음성 파라메터를 추출하는 연구와 음성특징 파라메터로부터 효과적으로 학습, 비교하는 연구로 나뉘어 질 수 있다. 그 중에서 음성특징 파라메터의 추출은 음성을 되도록 적은양의 수치데이터를 가지고 효과적으로 묘사하는 것이 매우 중요하다. 본 논문에서는 12차의 LPC계수를 이용하여 Cepstrum계수를 유도하여 인식 파라메터로 이용하였다.

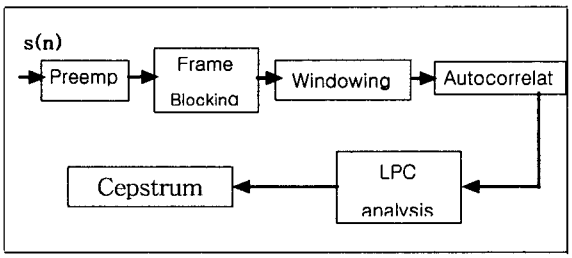

그림 3. 특징 파라메터를 구하는 과정

그림 3과 같은 구성으로 음성신호가 입력되면 음성신호에 대하여 프리엠퍼시스를 하고 프리엠퍼시스된 신호를 프레임 블록으로 나뉘어 램에 음성신호를 저장한다. 저장된 데이터를 프레임 블록 단위로 윈도우를 씌우고 자기 상관계수를 구하여 12차의 LPC계수를 구하고 구해진 LPC계수에 의해 Cepstrum계수를 유도하여 특징 파라메터를 구하였다.

본 논문에서는 LPC계수를 비교적 연산량이 적은 Durbin의 알고리즘을 이용하였다. 표1에 Durbin의 알고리즘을 나타내었다.

표 1. Durbin의 알고리즘

$$E_n^{0}=R_n(0)$$

$$k_i=\frac{R_n(i)-\sum_{j=1}^{i-1}a_j^{i-1}R_n(i-j)}{E_n^{i-1}}, \qquad 1\le j\le p$$

$$a_j^{(i)}=k_i$$

$$a_j^{(i)}=a_j^{(i-1)}-k_i\,a_{i-j}^{(i-1)}, \qquad 1\le j\le i-1$$

$$E_n^{i}=(1-k_i^{2})\cdot E_n^{(i-1)}$$

$$a_j=a_j^{(p)}, \qquad 1\le j\le p$$

2.3 DTW

DTW는 기준이 되는 음성신호의 패턴과 입력된 음성신호간의 유사도를 동적 프로그래밍을 이용해 구하는 방법이다. 즉, 시간 축에서의 차이를 보상하는 방법이다. 표2는 DTW의 알고리즘을 보이고 있다.

표 2. DTW의 알고리즘

1. Initialization : $D_A(1,1) = d(1,1)m(1)$
2. Recursion

$1 \le i_x \le T_x$, $1 \le i_y \le T_y$ 이면,

$$D_A(i_x,i_y) = \min_{(i'_x\, i'_y)} [D_A(i'_x,i'_y) + \zeta((i'_x,i'_y),(i_x,i_y))]$$

3. Termination

$$d(\chi,y) = \frac{D_A(T_x, T_y)}{M_m}$$

패턴 분류 과정은 분석과정에서 구한 특징 벡터와 미리 구하여둔 기준 패턴과 개별적으로 DTW해 유사도를 비교하는 것으로 가장 많이 사용되는 DTW의 패턴 매칭을 이용하였으며, 그림 4에 보이고 있다.

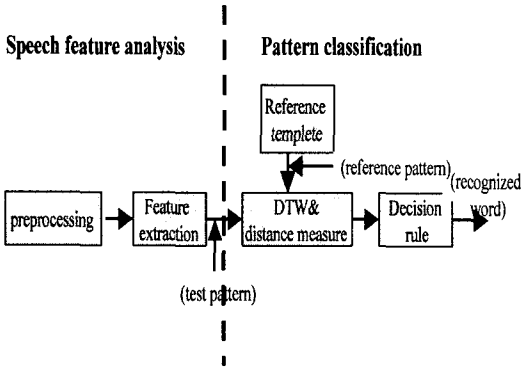

그림 4. DTW를 이용한 단독음 음성인식 시스템

III. 음성인식 시스템의 하드웨어 구성

3.1 음성보드 설계

본 논문에서 음성인식 독립시스템을 구현하기 위하여 32비트 부동소수점 연산방식인 TMS320C31의 DSP을 사용하였다 . TMS320C31의 구조는 기본적인 CPU부분, 레지스터 파일, 메모리, 내부 주변장치, DMA 제어기, 그리고 내부 버스 등으로 나눌수 있다.

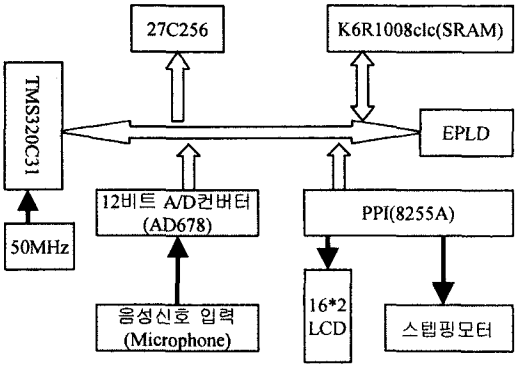

그림 5. 음성인식 하드웨어 구성 블록도

그림 5는 음성인식기의 구성도를 보이고 있다. 음성을 입력받기 위한 12비트 AD컨버터, 음성데이터 저장을 위한 128kbyte의 외부 SRAM, 음성프로그램 및 데이터 저장의 64Kbyte ROM, 어드레스 디코딩을 위한 EPLD, II장에서 설명된 특징벡터와 DTW의 알고리즘에 의한 인식 결과의 출력을 위한 스텝핑 모터와 LCD로 구성되어 있다.

3.2 메모리 맵과 어드레스 디코딩

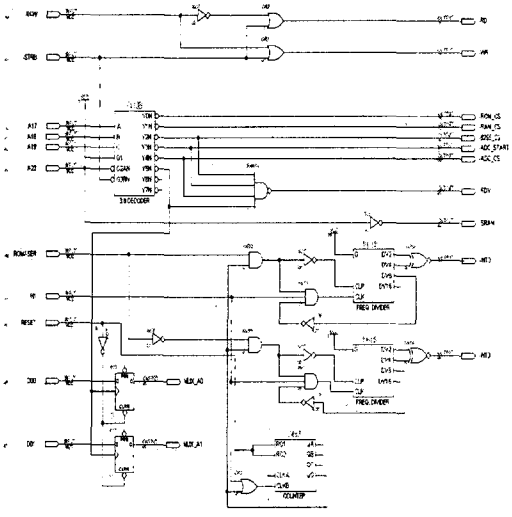

그림 6. EPLD 회로도

TMS320C31은 24개의 어드레스 핀이 있기 때문에 16M의 메모리 공간을 가지며, 내부 메모리는 RAM0, RAM1의 각각 1K 워드의 공간을 갖지만, 인식 시스템을 구현하기 위해서는 메모리가 부족하다. 그래서 본 시스템에서는 512K 워드(ROM)와 160K 워드(SRAM)까지 사용할 수 있도록 설계를 하였다. 외부 메모리를 억세스하기 위해 스트로브(STRB) 신호를 사용해서 어드레스 디코딩을 하였다.

어드레스 디코딩 소자로는 Altera의 EPLD를 사용하여 회로 설계의 복잡성을 줄이고, 회로 수정이 매우 용이하다는 점과 동작 속도가 빠른 DSP와 동작 속도가 느린 외부 소자와의 액세스가 용이한 장점을 가진 EPM7064SLC-44을 사용하였다. 그림 6은 음성인식기의 어드레스 디코딩을 위한 EPLD회로 구현을 보여주고 있다. 메모리 맵은 그림 7에 나타내었다. 인식에 필요한 단어들의 모델과 프로그램에 소스등 모든 데이터를 저장하기 위한 ROM은 000000h-00FFFFh 영역(512K)에 할당하였고, 실제로 프로그램이 수행되는 영역인 RAM은 020000h-027FFFh 영역(32k)와 900000h-9FFFFFh 영역(128k)에 배치 하였다.

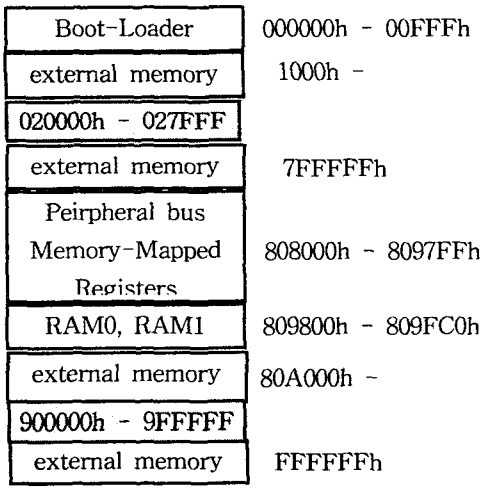

그림 7. 메모리 맵

Ⅳ. 음성인식 실험

음성인식에 대한 출력 값을 얻기 위해 본 논문에서는 1.8의 분해능을 갖고 오차가 누적되지 않는 스텝핑 모터를 구동하여 인식 결과에 대한 인식률을 살펴 보았다. 인식 단어는 모터를 구동할 수 있는 5가지의 단어를 사용하였으며 표 8에 나타내었다.

표8. 실험 대상 단어 및 동작

(a)

대상단어	인식횟수
정지	20
전진	19
좌로	19
우로	19
후진	20

(b)

명 령 어	동 작
전 진	40cm 전진
후 진	40cm 후진
좌 로	25 °좌회전
우 로	25 °우회전
정 지	정지(대기)

표 8(a)에 실험 대상 단어와 같이 음성인식 결과에 대한 모터의 동작을 하도록 프로그램을 구성하였으며 인식 결과에 대한 인식률은 표 8(b)에 나타내었다. 이 실험을 위한 발음 횟수는 총 21회로 하였다. 그림 8은 구현된 음성인식기를 보이고 있다.

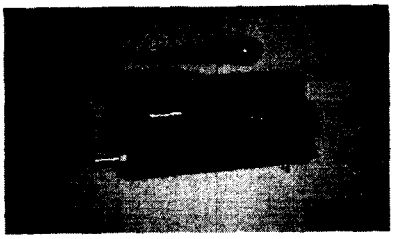

그림 8. 구현된 음성인식기

Ⅴ. 결론

본 논문에서 TMS320C31의 DSP을 사용하여 음성인식기를 구현하였으며 구현된 인식기의 특징 벡터로는 LPC 켑스트럼을 사용하여 DTW의 알고리즘에 의해 유클리드 거리 계산식을 이용하여 음성인식기를 구현하여 보았다. 음성인식결과는 스텝핑 모터로 구동하여 음성인식률을 살펴 본 결과 92%이상의 인식률을 보였다. 인식률을 좀더 높이기 위해서는 LPC 필터계수, 프레임 구간, 잡음제거의 적절한 방법을 찾아 구현한다면 더욱 인식률이 높아지게 될 것으로 생각된다.

참고문헌

[1] L.R.Rabiner, B.H.Juang, "Fundmentals of speech recognition", prentice-Hall, New Jersey, 1993.

[2] A.M.Kondoz, "Digital Speech Coding for Low Bit Rate Communications Systems", 1994.

[3] 오영환, " 음성언어 정보 처리 ", 홍릉과학출판사. 1998.

[4] 윤덕용, " TMS320C31 마스터",ohm사.1999.