

# KT 증권정보 서비스 시스템의 구현과 시스템 자원의 효율적 활용을 위한 방법 고찰

박성준\*, 구명완\*, 전주식\*\*  
한국통신 멀티미디어연구소\*  
서울대학교 컴퓨터공학과\*\*

## Implementation of Stock Information System and Methods for Efficient Use of System Resources

Sung-Joon Park\*, Myong-Wan Koo\*, Chu-Shik Jhon\*\*  
Multimedia Technology Research Laboratory, Korea Telecom\*  
Dept. of Computer Engineering, Seoul National Univ.\*\*

E-mail : sjpak@kt.co.kr

### 요 약

본 논문에서는 한국통신에서 음성인식을 이용한 전  
파정보 서비스의 일환으로 개발해 온 증권정보 시스템  
의 구조와 기능을 설명하고, 시스템을 다채널로 확장함  
에 있어서 시스템의 자원을 효율적으로 활용하기 위하  
여 적용한 방법에 대하여 기술하였다.

이 시스템에서는 음성특징을 추출하는 프로세서  
(DSP0)들과 단어검색을 하는 프로세서(DSP1)들이 분리  
되어 있으며, 이 둘 간의 개수 비율을 조절함으로써 실  
시간적 처리 효과를 유지하면서도 시스템의 전체 프로  
세서의 개수는 줄였다.

DSP0와 DSP1 간의 음성 특징 데이터 전송에 있어  
서는 DSP0에서 발생하는 데이터를 음성이 입력되는  
중에 전송할 수 있게 함으로써, DSP1에서는 DSP0과  
병렬적으로 작업을 수행시킬 수 있으며, 결과적으로 시  
스템의 응답 속도를 빠르게 하였다.

### 1. 서론

한국통신에서는 지난 몇 년간에 걸쳐 음성인식을  
이용한 시스템을 개발하고, 성능 개선을 위한 작업을  
수행해 왔다. 그 중의 대표적인 예가 음성인식 증권정  
보 시스템으로서 1995년에 내부 시험을 거친 후, 그 해  
말부터 증권 회사에 설치하여 시험 운용을 하였다  
[1,2,3]. 그 후 시스템을 상용화하기 위한 준비를 거치  
이름 다채널 시스템으로 확장하게 되었다. 이 과정에서  
시스템의 개발 비용을 절감하기 위해 시스템의 구조를  
개선하였으며 시스템의 운용 상황 파악 및 보완점 발견  
등을 위하여 올해 초부터 시험 서비스를 하고 있다.

이 시스템에서 음성 인식 기능과 관련하여 구현된  
부분은 8 개의 TMS320C30 을 내장하고 있는 DSP 보드

(DSP0 라고 명명하였다.) 와 6 개의 TMS320C30 을 내장  
하고 있는 보드 (DSP1 이라고 명명하였다.) 인데, 각각  
특징 추출과 비터비 (Viterbi) 검색을 담당한다. 현재 이  
시스템에 사용된 음성 인식 관련 기술은 기존의 시험용  
시스템에 적용되었던 기술을 큰 변화 없이 적용시킨 상  
태이지만, 시스템 개발 및 시험 서비스 기간 중에 개  
선 또는 추가된 기술을 차후 적용시킬 예정이다.

이 논문에서는 시스템의 구조를 설명하고, 시스템이  
실제 작동할 때에 발생하는 자원의 분배와 시스템의 성  
능에 대하여 설명한다.

### 2. 시스템 개요

이 시스템은 전화망을 통해 일반인이 회사 이름을  
말하여, 그 이름을 인식하여 관련된 주식 정보를 음성  
으로 알려 주는 서비스를 제공하는 화자독립 고립단어  
음성 인식 시스템이다.

시스템의 구조는 그림 1과 같이 여러 개의 보드가  
시스템 버스로 연결된 형태이다.

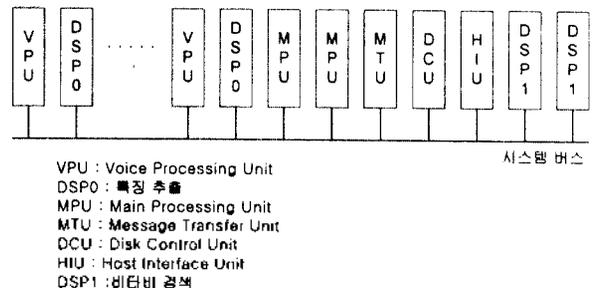


그림 1. 시스템의 구조

DSP0에서는 VPU로 입력되는 데이터 중에서 음성 데이터만을 걸러 내어 특징 추출을 수행하고 그 결과를 MPU로 보내는 작업을 수행하며, DSP1에서는 MPU를 기저 넘겨오는 특징 추출 데이터를 이용하여 비터비 검색을 한다. 비터비 검색을 하고 난 후, 후보 단어가 결정되면 이를 MPU로 보낸다.

MPU는 시스템의 전체적인 관리를 담당하며, 음성 인식과 관련해서는 DSP0로부터의 결과를 DSP1로 보내고 최종적으로 DSP1에서 보내 오는 인식 결과를 이용하여 안내 방송을 내보내는 작업을 한다.

특징 추출은 입력되는 음성 데이터에 대해 매 프레임 단위로 수행되기 때문에 최대 계산 시간이 한 프레임 시간보다 적어야 하며, 이를 위해서 매 채널마다 별도의 DSP가 할당되어 있다 [그림 2].

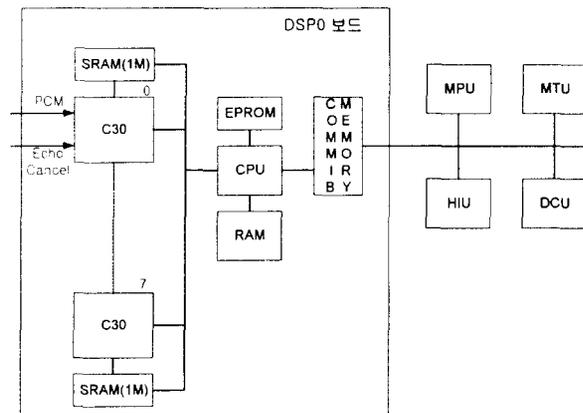


그림 2. DSP0의 블록도

하나의 DSP0 보드에는 모두 8 채널이 연결되어 있으며, 채널별로 8개의 DSP가 장착되어 있다. DSP를 제외한 하나의 프로세서(CPU)가 추가되었는데, 이것은 각각의 DSP로부터 나오는 특징 추출 데이터를 MPU로 전달하는 과정을 중재하기 위한 것이다.

DSP1 보드의 구조도 DSP0과 같은 형태로 되어 있다. 단, DSP1은 한 보드당 6개의 DSP가 있으며, 보드의 수도 적다. DSP1의 CPU는 DSP0로부터 오는 특징 추출 데이터를 받아서 유휴 상태에 있는 DSP로 넘겨 준다.

DSP0와 DSP1을 분리시킨 이유는 이러한 동적 연결이 사용자에게 빠른 응답을 제공하면서도 시스템의 효율적으로 활용할 수 있기 때문이다. 일단 사용자가 입력을 한 후, 시스템이 내보내고 있는 안내 방송을 듣고 있을 동안에는 DSP1 보드의 DSP에서는 해당 채널에 대하여 해 줘야 할 작업이 없다. 따라서 이 동안에 DSP0 보드의 다른 DSP로부터 넘어 온 특징 추출 데이터에 대하여 비터비 검색을 수행할 수 있다. 반면에 DSP0에서는 안내 방송 중에도 사용자 음성의 시작점 검출을 위한 작업이 계속적으로 수행되어야 하기 때문에 사용자가 전화를 끊기 전까지는 다른 작업을 할 수 없다. 결과적으로 DSP0의 DSP들은 채널 수만큼 있어야 하지만, DSP1에서는 시스템의 부하에 따라 그 수를

조절할 수 있다. 그 비율은 이 시스템의 개발 이전에, 시범 서비스를 통해 수집한 자료를 분석하여 구해 보았으며, 평균 11대 1까지 나왔다 [4]. 그러나 현재 구현된 시스템에서는 8대 3의 비율로 맞추어 놓았으며, DSP0 보드를 더 장착하면 비율을 증가시킬 수 있다.

DSP0와 DSP1을 분리시킨 다음에는 이 둘의 연결 방법이 문제가 된다. 이 시스템에서는 버스 구조가 사용되었는데 그 이유는 버스 구조가 가지는 간단성과 낮은 비용이라는 장점 때문이었다 [5].

버스 구조가 가지는 문제점 중의 하나가 버스 트래픽이 증가하면 시스템의 성능이 떨어진다는 것인데, 이 시스템에서는 DSP0와 DSP1의 트래픽 부하가 심하지 않기 때문에 큰 문제가 되지 않는다. 이미 기술한 바와 같이, DSP0에서는 특징 추출을 하는데, DSP1로 넘겨주는 특징 추출 데이터는 하나의 프레임에 대하여 벡터 양자화에 따른 4바이트의 코드북 인덱스이다. 따라서 몇 초 정도 되는 음성에 대하여도 전송할 데이터가 얼마 되지 않는다. DSP0와 DSP1 간에 형성되는 일종의 파이프라인을 버스 시스템에서도 그대로 유지할 수 있다. 음성이 입력되면 DSP0에서는 특징 추출을 하면서 프레임별로 생기는 특징 추출 데이터를 다 모았다가 한꺼번에 DSP1로 전송하는 것이 아니라, 하나씩 또는 일부를 모아서 DSP1로 전송한다. 따라서 음성이 입력되는 동안 비터비 검색이 수행될 수 있다. 이것은 DSP0에서 DSP1로 특징 추출 데이터를 보낼 때, 해당되는 채널 번호도 같이 보냄으로써 가능하다.

### 3. 특징 추출과 비터비 검색

시스템에서 전화망과 연결된 보드는 VPU인데, 이 보드로 들어 오는 입력에 대하여 PCM으로 표시된 음성 데이터만이 DSP0로 들어 온다 [그림 3].

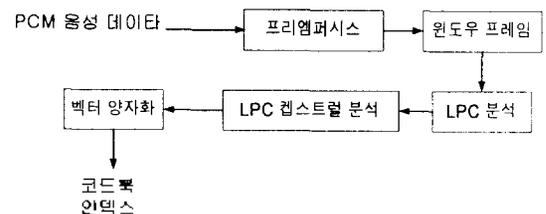


그림 3. 특징 추출 단계

입력된 음성은 프리엠퍼시스 과정을 거쳐 20msec 단위의 프레임으로 분할되며 10msec마다 중첩된다. 그리고 10msec 단위로 LPC (linear predictive coding) 분석과 켈스트럼 분석을 한다. 분석 결과 켈스트럼 계수를 비롯하여 이 계수의 1차, 2차 차이값 및 로그 파워의 1차, 2차 차이값 등이 얻어지고 벡터 양자화를 거쳐 코드북 인덱스가 생성된다. 코드북의 구성은 로그 파워(logged power)의 경우에는 1.2차 차이값을 함께 표현한 64\*2의 크기를 가지는 배열로 되어 있으며, 나머지는 256\*12 크기의 배열이다. 참조되는 코드북은 시스템의 초기화 때 각각의 DSP로 적재된다.

음성 데이터는 DSP의 직렬 포트에 들어오는데, 데이터가 들어올 때마다 인터럽트가 발생하여 인터럽트 루틴에서 이를 버퍼에 저장시킨다. 그리고 안내 방송

중인 음성도 다른 직렬 포트를 통해 들어오는데, 이는 반향 제거를 위해 사용된다. 반향 제거된 데이터는 시작점과 끝점 검출 과정을 거치는데, 시작점이 검출되면 그 때부터 특징 추출을 하여 코드북 인덱스를 생성하게 되며 이것을 DSP1로 넘겨 준다.

코드북 인덱스를 DSP1로 넘겨주는 개수는 프로그램 내에서 변경할 수 있도록 하였는데, 매 10msec마다 나오는 데이터를 바로 DSP1로 넘기거나 또는 데이터를 몇 개씩 묶어서 보낼 수 있다. 첫째 방법은 DSP0와 DSP1 사이의 데이터 전송 시간이 시스템의 성능에 별 영향을 미치지 않을 때 사용할 수 있으며 둘째 방법은 데이터 전송에 걸리는 시간이 무시하기 어려울 때 사용할 수 있다. 데이터 전송에 시간이 걸리는 이유는 시스템 버스의 속도가 느리거나 MPU가 다른 작업을 수행함으로써 인해 데이터 전송을 제대로 못해 줄 수 있기 때문이다. 이와 같이 데이터를 묶을 때, 그 묶는 개수는 일정한 값으로 고정할 수도 있고, 또는 시스템의 동작 중에 두 단계간의 데이터 전송 시간의 변동에 따라 그 단위를 변화시킬 수도 있다.

현재의 시스템에 적용되는 방법은 데이터를 여러 개 묶어서 보내는 두 번째 방법으로서 특징 추출 데이터를 10개씩 묶어서 보내도록 설정되어 있다.

DSP1로 넘어온 코드북 인덱스들은 비터비 검색에 사용되는데 이 시스템에 적용된 음성 인식 알고리즘은 비터비 검색 알고리즘으로서 HMM (hidden Markov model) 에 바탕을 두고 있다. 사용되는 기본 단위는 약 300개 정도의 문맥 종속 음소로서 화자 독립적인 훈련 과정을 거쳐 생성되었다.

#### 4. 데이터 전송 방법

이 시스템 개발 이전에 PC를 이용하여 만들어 증권 회사에서 시험 운용했던 시스템에서는 특징 추출과 비터비 검색에 각각 하나씩의 DSP를 할당하였다. 즉, 한 채널에 대하여 2개의 DSP가 할당되었으며, 둘 간의 코드북 인덱스 전송은 DSP끼리 직접 연결된 병렬 포트를 이용하여 이루어졌다. 이 경우에는 데이터 전송과 관련하여 DSP끼리의 데이터 전송 중재 문제는 없었다. DSP0와 DSP1을 분리시킬 때에는 데이터 중재 문제가 발생하게 되는데 이것은 DSP1 내의 CPU에서 처리하도록 하였다. 그 처리 과정을 그림 4에 나타내었다.

먼저 DSP0 쪽을 살펴 보면, DSP0의 한 DSP가 음성의 시작점을 검출하면 DATA\_START로 정의된 값을 메모리에 쓴다. 그리고 매 10msec마다 특징 추출을 수행하여 그 데이터를 메모리에 쓴다. 메모리에 쓴 값의 개수가 10개가 되면 CPU로 인터럽트를 걸어서 알린다. CPU는 DSP의 메모리로부터 데이터를 읽어서 공통 메모리에 설정되어 있는 특징 추출 데이터 버퍼로 복사한 다음, 커맨드 버퍼 (command buffer) 에 데이터 전송 명령을 쓴다. MPU의 운영체제는 이 명령을 보고, 쓰여진 개수만큼 특징 데이터를 읽어 간다. DSP는 그 다음의 음성에 대해 계속적으로 특징 추출을 하여 10개가 되면 다시 인터럽트를 발생시키며, 앞에서와 같은 순서에 의해 MPU 쪽으로 데이터가 넘어 간다. 이 과정은 음성의 끝점이 검출되거나 제한된 음성 입력 시간이 될 때까지 반복되며, 마지막으로 데이터를 보낼 때에는 DATA\_END로 정의된 값을 주가로 쓰고, 전송되는 데

이터의 개수가 10개가 되지 않더라도 전송을 한다. 한편 MPU는 VRU로부터 읽은 데이터를 DSP1의 공통 메모리로 쓰고, 데이터 전송 명령을 DSP1로 보낸다. DSP1의 CPU는 MPU로부터 데이터 전송 명령이 오면 유휴 상태에 있는 DSP를 하나 선택하여 특징 추출 데이터를 그 DSP의 메모리로 쓴다. 그 다음부터 넘어 오는 특징 추출 데이터는 이미 할당된 DSP의 메모리로 쓰여진다. DSP는 넘어 온 데이터를 이용하여 비터비 검색을 수행하고 검색이 끝났으면, 결과를 메모리에 쓰고 인터럽트를 발생시킨다. 인터럽트가 발생하면 CPU는 인터럽트를 발생시킨 DSP를 찾고, 해당 DSP의 메모리로부터 단 이 인덱스 값을 읽는다. 그리고 다음 인덱스 값을 공통 메모리에 복사한 다음, 결과가 나왔음을 알리는 명령을 공통 메모리의 커맨드 버퍼에 쓰고 방금 읽은 DSP의 상태를 유휴 상태로 만든다. MPU에서는 이 명령을 받고서는 MTU를 통해서 서버로 인덱스를 보내고 필요한 정보를 받는다. 그리고 이 정보를 이용하여 사용자에게 돌려 줄 메시지를 구성하고 이를 VPU로 보낸다. 사용자는 자신이 입력한 음성에 대한 응답을 VPU에서 나오는 메시지로 듣게 된다.

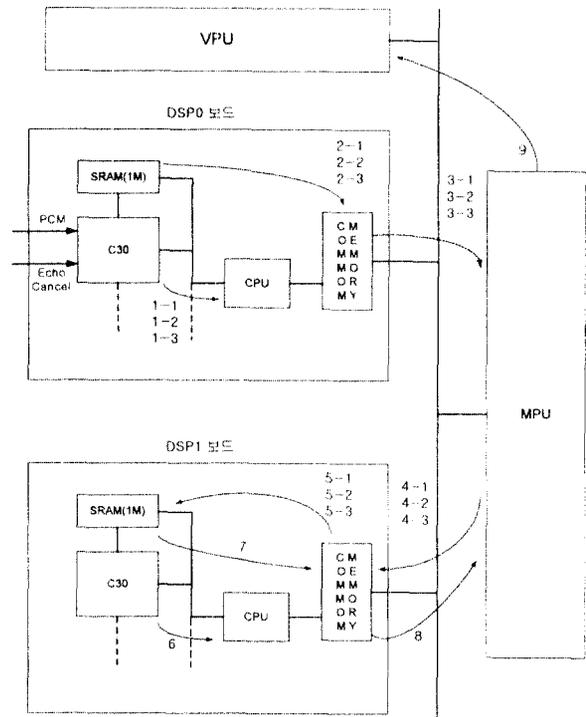


그림 4. DSP0와 DSP1간의 특징 추출 데이터의 전송

#### 5. 결론 및 앞으로의 과제

본 논문에서는 한국통신에서 개발한 음성인식 증권 정보 시스템의 전체적인 구조와 동작 과정을 살펴 보았다. 이 시스템을 구현하면서 초점을 맞춘 부분은 시스템을 다채널로 확장함에 있어서 시스템 자원의 효율적인 사용과 시스템의 응답 속도였으며, 결과적으로 음성의 특징 추출과 비터비 검색을 수행하는 부분이 분리된

비스 기반의 시스템을 만들게 되었다.

시스템의 구현 및 시험 시미스 기간 동안에 추가적

인 연구가 진행되었는데, 반향 제거기의 성능 향상과 음성 인식 거절 기능에 관련된 것이다 [6,7,8]. 이러한 작업은 모두 인식률을 높이기 위한 것으로서 앞으로 시스템에 적용시킬 예정이다.

## 참고 문헌

- [1] M. W. Koo et al., "KT-STOCK: A speaker-independent, large-vocabulary speech recognition system over the telephone," in *Proc. 1994 Int. Conf. on Spoken Lang. Processing*, pp. 1387-1390, Sep., 1994.
- [2] M. W. Koo et al., "An experimental field trial of a large vocabulary, speaker independent recognition system," in *Proc. Second IEEE Workshop on Interactive Voice Technology for telecomm. Applications*, Sep., 1994.
- [3] 김제인, 구명완, "음성인식 증권정보시스템의 개발 및 시험운용결과 분석," *음성통신 및 신호처리 워크샵 논문집*, 제13회, pp. 185-191, 1996.
- [4] 박성준, 구명완, 전주식 "음성인식 증권정보 검색 시스템의 개선 방향에 관한 고찰," *Proceeding of The 24<sup>th</sup> KISS Spring Conference*, pp. 639-642, 1997.
- [5] Sung-Joon Park, et al., "The architecture of a speech recognition system for cost reduction," in *Proc. Int. Conf. Of Speech Processing*, pp. 549-553, Aug., 1997.
- [6] 유창동, "이중 여진 음성 모델을 이용한 음질 개선," *한국음향학회 학술발표대회 논문집*, 제17회, pp. 175-178, 1998.
- [7] 강병구, 유창동, "입출력 신호의 상관계수를 이용한 반향 제거기," *한국음향학회 학술발표대회 논문집*, 제17회, pp. 189-192, 1998.
- [8] 김우성, 구명완, "발화 검증에 의한 음성인식 거절기능 연구," *한국음향학회 학술발표대회 논문집*, 제17회, pp. 67-70, 1998.