산업기술연구(강원대학교 산업기술연구소 논문집), 제29권 A호, 2009.
Journal of Industrial Technology, Kangwon Natl. Univ., Korea, No. 29 A, 2009.

TMS320F28335 DSP를 이용한 화자독립 음성인식기 구현

Implementation of a Speaker-independent Speech Recognizer Using the TMS320F28335 DSP

정 익 주*
Chung, Ik-Joo

Abstract

In this paper, we implemented a speaker-independent speech recognizer using the TMS320F28335 DSP which is optimized for control applications. For this implementation, we used a small-sized commercial DSP module and developed a peripheral board including a codec, signal conditioning circuits and I/O interfaces. The speech signal digitized by the TLV320AIC23 codec is analyzed based on MFCC feature extraction methed and recognized using the continuous-density HMM. Thanks to the internal SRAM and flash memory on the TMS320F28335 DSP, we did not need any external memory devices. The internal flash memory contains ADPCM data for voice response as well as HMM data.

Since the TMS320F28335 DSP is optimized for control applications, the recognizer may play a good role in the voice-activated control areas in aspect that it can integrate speech recognition capability and inherent control functions into the single DSP.

키워드 : *화자독립 음성인식기, TMS320F28335 DSP*
Keywords : *speaker-independent speech recognizer, TMS320F28335 DSP*

1. 서론

연구소 및 기업을 중심으로 음성인식 기술이 상당한 수준까지 발전하고, 이를 상용화하려는 노력이 꾸준히 지속되면서, 지난 수십 년 간 연구되어 온 음성인식 기술은 최근 들어 어느 정도 그 가시적인 성과가 나타나고 있다. 특히, 반도체 기술의 획기적인 발전으로 고성능의 마이크로프로세서와 고용량의 메모리가 해를 거듭할수록 저렴해지고 있으며, 이러한 고성능과 저가격은 연구실에서 이루어진 연구 성과의 실용화를 더욱 용이하게 하고 있다.

음성인식 시스템은 그 활용에 따라 크게 두 가지 분야로 나누어질 수 있는데, 첫 번째는 연산 성능이나 메모리와 같은 리소스의 제한을 받는 임베디드 시스템 기반의 인식기를 이용하는 응용 분야이고, 두 번째로는 위와 같은 리소스의 제한이 거의 없는 호스트 서버 기반의 인식기를 이용하는 응용 분야이다. 호스트 서버 기반의 인식기의 경우는 대용량 인식을 위주로 하는 전화망 또는 VoIP 기반의 음성인식 응용이 대표적인 예이며, 임베디드 시스템 기반의 응용 분야로는 단말기에 내장된 전용 하드웨어나 마이크로프로세서를 기반으로 한 이동형 단말기에서의 중소용량 음성인식 응용을 들 수 있다[1][2][3].

임베디드 음성인식기를 활용하는 분야를 좀 더 살펴보면, ASIC이나 저가형 DSP 칩을 기반으로 구현된 전용 하드웨어를 이용하는 방식과 내장된

*강원대학교 전기전자 공학부

CPU를 이용하는 두 가지 방식이 있을 수 있는데, 현재는 내장된 CPU를 이용하여 소프트웨어적으로 구현하는 방식을 선호하고 있다. 이는 이미 시스템에 포함된 기존의 CPU를 소프트웨어적으로 활용함으로서 별도의 추가 제조비가 발생하기 않기 때문이다. 이러한 장점에도 불구하고 이를 위해서는 기존의 CPU에서 음성인식 소프트웨어를 실행하기 위한 여분의 성능이 남아 있어야하며, 기존에 실행되고 있는 타 소프트웨어 모듈들과 함께 동작하는데 있어서 문제가 없는지도 검증되어야 하는 개발상의 단점이 있다. 한편, 전용 하드웨어를 이용할 경우, 추가의 제조비용이 발생하고, 대용량 인식과 같은 복잡한 음성인식 기술을 구현하는 데는 한계가 있기는 하지만, 기존의 시스템에 영향을 주지 않으면서 독립적으로 음성인식 기능을 추가할 수 있는 장점이 있다. 현재 시장에 출시되어 있는 전용 하드웨어는 ASIC 기반의 음성인식 칩과 DSP 또는 RISC 프로세서를 기반으로 한 하드웨어 모듈의 두 가지 형태가 있는데, 음성인식 칩의 경우는 가격이 저렴하다는 장점이 있는 반면, 기능이 매우 제한적이고, 프로세서를 기반으로한 하드웨어 모듈의 경우는 근본적으로는 소프트웨어적으로 구현되므로 사용되는 프로세서에 따라서는 고성능의 인식기 구현이 가능하나 가격이 음성인식 칩에 비하여 상대적으로 비싸기 때문에 대량으로 생산되는 저가의 제품에 적용되기에는 한계가 있다. 그러나 저가의 범용 프로세서를 이용할 경우 음성인식 칩에 비하여 customize가 용이하고 소량 생산에 유리하다는 장점을 가진다. 위에서 설명한 몇가지 장점으로 인하여 향후 하드웨어의 가격이 저렴해지고, 좀 더 발전된 음성인식 기술이 적용될 경우 전용 하드웨어를 이용한 음성인식 방식이 보다 널리 채택될 것으로 예상된다[3].

한편, [3]에서 TMS320C2000계열의 저가의 고정소수점 DSP를 이용하여 화자 종속 단일칩 음성인식기를 구현한 바 있다. [3]에서는 DSP가 내장하고 있는 A/D 변환기, 플래쉬 메모리 등의 내장 장치를 활용함으로서 추가의 부품 없이 단일칩 음성인식기를 구현할 수 있었다. 그러나, TMS320C2000 DSP의 장점은 제어 분야에 최적화되어 있다는 점인데, [3]에서는 저가의 단일 칩 구현에 초점이 맞추어져 있었기 때문에 음성인식과 연계된 제어 기능을 추가하기에는 부족함이 있었다. 또한, 내장된 12bit A/D 변환기를 사용하기 때문에 화자 종속 인식기 구현에 만족해야 했었다.

본 논문에서는 최근에 출시된 TMS320F28335 DSP를 이용하여 화자 독립 인식기를 구현하였다[7]. TMS320F28335 DSP는 TMS320C2000 계열 DSP이기는 하지만, [3]에서 사용하였던 DSP와는 달리 부동소수점 DSP이며, 음성인식에서 주로 사용되는 sigma-delta modulation 방식의 코덱을 연결할 수 있는 McBSP라는 직렬포트가 제공되기 때문에 과거 부동소수점 DSP로 구현하던 고성능의 화자 독립 인식기 구현이 가능하다. 뿐만 아니라, 내장된 충분한 양의 SRAM, 플래쉬 메모리 등을 활용할 수 있기 때문에, 최소 부품으로 음성 인식 기능을 수행하고 이를 바탕으로 기기를 제어하는 기능을 통합함으로서 음성 인식 제어 분야에 적합한 음성인식기를 구현할 수 있었다.

이후 논문의 구성은 2장에서 TMS320F28335 DSP의 소개 및 화자독립 인식기의 전체적인 구성에 대하여 설명하고, 3장 구현된 화자독립 음성인식 알고리즘, 4장 하드웨어 구현, 그리고 5장의 결론으로 이루어져 있다.

2. TMS320F28335 DSP 및 화자 독립 인식기 구조

TMS320C2000 계열 DSP는 DSP의 신호처리 기능과 범용 마이크로컨트롤러의 제어 기능을 모두 가지고 있기 때문에 Digital Signal Controller(DSC)라고도 불리운다.(그러나 경우에 따라서는 DSP보다는 MCU에 가깝게 분류하기도 한다.) 신호처리 연산을 수행하는 DSP core 외에도 PWM, UART, A/D 변환기, watchdog 타이머 등의 주변장치를 포함하고 있기 때문에 이런 주변장치들을 위하여 더 이상 마이로컨트롤러를 추가적으로 사용할 필요가 없다. 특히 TMS320C2000 계열 DSP는 디지털 모터 제어에 최적의 프로세서로 알려져 있다. 본 논문에서는 TMS320C2000 계열 DSP 중에서 TMS320F28335 DSP를 이용하였다. TMS320C2000 DSP는 기본적으로 고정소수점 DSP이나 최근 정밀제어의 필요성으로 인하여 부동수점 연산을 지원하는 TMS320F2833x 계열의 DSP가 출시되었다. 다음은 TMS320F28335 DSP의 특징을 간단히 요약한 것이다.

- 16x16 Dual MAC 및 32x32 MAC 연산을 지원하는 32bit CPU
- IEEE-754 단정도 부동소수점 연산 유닛
- Atomic 연산 및 매우 빠른 인터럽트 반응 지원
- Harvard Bus 구조 및 통합된 메모리 프로그래밍 모델
- 34K word의 SRAM 및 256K word의 플래쉬 메모리 내장
- SCI, SPI, CAN, PWM, McBSP 12bit A/D변환기, 타이머, GPIO등 다양한 주변장치 내장
- 확장을 위한 외부 메모리 버스 지원

그림 1은 본 논문에서 구현한 화자 독립 음성인식기의 블록도이다. 음성 검출과 동시에 MFCC 음성 분석을 통한 특징벡터 추출이 실시간으로 수행

된다. 유사도 측정은 연속확률분포 HMM을 이용하여 인식을 수행한다. 분석된 음성 특징벡터열은 미리 훈련된 명령어 모델들과의 Viterbi 디코딩을 통해 확률값을 계산하고 이렇게 얻어진 결과는 후처리 과정을 거쳐 인식 결과를 거부할지 아니면 최종 인식결과로 확정할지를 수행하게 된다.

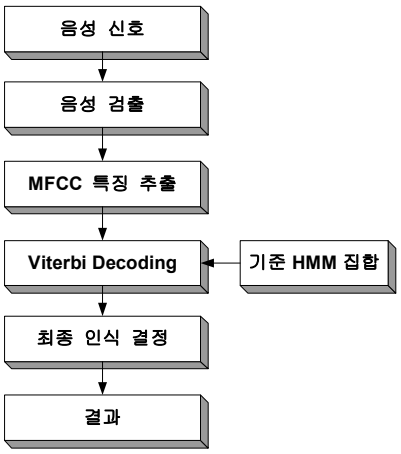


그림 1 HMM을 이용한 화자 독립 음성인식기

3. 화자독립 음성인식 알고리즘

3.1 끝점검출 및 음성 분석

16KHz로 샘플링된 음성신호는 우선 저주파 노이즈를 제거하기 위하여 150Hz를 cutoff 주파수를 가지는 고역 필터를 통과시킨 후, 끝점 검출기로 입력된다. 끝점 검출기는 연산량이 적으면서도 비교적 안정적인 끝점 검출을 하는 것으로 알려진 프레임 에너지와 프레임 ZCR을 변형한 LCR(Level Crossing Rate) 기반의 끝점 검출기를 이용하였다[4][5]. 그리고 주변 배경 잡음의 변화에 대응하도록 음성이 아니라고 판명된 프레임의 에너지 및 LCR을 이용하여 이들 파라메터의 문턱값을 적응시켰다. 끝점 검출을 위한 프레임의 길이는 200 샘플로 하였으나 분석 시에는 이전 프레임 200 샘플을 포함하여 400 샘플을 분석구간으로 하였다. 따라서 25msec 분석구간을 가지며 매 프레임 분석 시 마다 12.5msec의 오버랩을 하게 된다.

끝점 검출과 동시에 특징 추출을 실시간으로 수행한다. 해당 프레임이 음성프레임으로 판정된 경우 preemphasis 및 Hamming windowing를 수행한 후, 12차의 MFCC 분석을 수행하고 추출된 특징벡터는 bandpass lifter를 사용하여 liftering하였다. C0 파라메터 및 델타 파라메터도 사용하였기 때문에 특징벡터의 차수는 26차가 된다. 한편, HMM 모델을 훈련하기 위하여 여러 기관에서 배포된 음성 데이터베이스를 활용하였는데, 이들 음성 데이터베이스에 포함된 음성들은 대부분 다이나믹 마이크로폰을 이용하여 녹음된 것이다. 한편, 임베디드 음성인식기의 경우는 대부분 크기가 작은 콘덴서 마이크로폰을 사용하기 때문에, 이 두 종류의 마이크로폰 특성 차이로 인하여 모델 불일치(mismatch) 문제가 발생하게 된다[4]. 본 인식기 구현에서는 모델 불일치를 완화시키기 위하여 적은 연산량 만으로도 좋은 성능을 보이는 CMS 알고리즘을 적용하였다. 그림 2는 분석 과정을 보여주는 블록도이다.

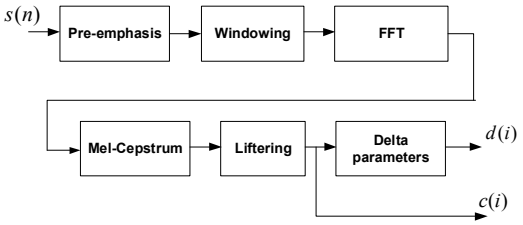


그림 2 MFCC 음성 특징벡터 추출

3.2 Viterbi 디코딩

본 논문에서 구현하는 인식기는 whole word 방식의 고정어휘 인식기이기 때문에 Viterbi 디코딩은 비교적 단순하다. 일반적으로 whole word 방식의 인식기에서는 Viterbi 디코더를 구현할 때, word 기반의 Viterbi 디코딩을 수행할 수 있는데, 본 논문에서는 향후 연결단어 인식기로의 확장을 위하여 frame-synchronous 방식의 Viterbi 디코딩을 수행하였다. 한편, 메모리의 사용량을 줄이기 위하여 3.1절에서 설명한 델타 파라메터의 경우, 미리 계산해 놓지 않고 Viterbi 디코딩 과정에서 직접 계산하여 사용하는데, frame-synchonous 방식의 경우는 입력된 테스트 음성에 대하여 한번만 델타 파라메터를 구하면 되기 때문에 연산량 측면에서 유리하다. 반면, word 기반의 Viterbi 디코딩을 수행할 경우에는 각각의 HMM 기준 모델과 비교할 때마다 매번 델타 파라메터를 구해야 한다.

인식을 위한 HMM 모델의 구조는 통상의 left-right 모델을 사용하였으며 상태 간 skip은 허용하지 않았다. whole word 방식의 인식기의 경우는 상태의 수를 충분히 허용함으로서 mixture를 최소화할 수 있기 때문에 일반적으로 하나의 mixture 만으로도 좋은 성능을 얻을 수 있다. 그러나 상태 간 skip을 허용하지 않을 경우에는 해당 단어의 발성 길이를 표현하는데 적절한 상태 수보다 많아질 수 있는 문제가 발생할 수 있다. 한편, mixture를 하나 이상 사용할 경우, 상태 확률 출력

을 계산하는데 log addition이 추가되며 특히 고정 소수점 DSP로 구현 시에는 상당한 연산량 증가를 초래하게 된다. 그러나 본 구현에서는 부동소수점 연산을 수행하고 내장된 플래쉬 메모리만을 활용하도록 구현했으므로 인식 어휘의 수도 제한이 있는 만큼 연산량 증가를 수용하는 것은 별 무리가 없기 때문에 상태 당 2개의 mixture를 사용하였다.

Viterbi 디코딩이 완료되면 최대 확률값을 주는 인식단어가 결정되며, 이를 인식결과로 출력하기 전에 out-of-vocabulary(OOV) 과정을 수행한다. 음성인식 기술에서 OOV 기술은 가장 도전적인 분야 중에 하나이다. OOV가 제대로 처리되기 위해서는 상당히 복잡한 처리를 요구한다. 현재 알려져 있는 반음소 방식[6]과 같은 OOV 알고리즘을 자원 사용을 최소로 하고자하는 임베디드 시스템에 적용하기에는 무리가 따른다. 본 연구에서는 [5]에서 제안한 단어 단위 기반의 가비지 방식 OOV 알고리즘을 적용하였다. 이 방식은 비교적 적은 자원을 사용하면서도 비교적 좋은 거부 성능을 발휘한다. 그림 3은 out-of-vocabulary 처리를 위한 구조를 보여주고 있다.

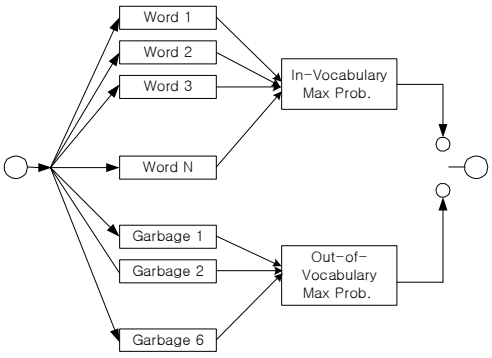


그림 3 Out-of-Vocabulary 처리를 위한 구조

가비지를 만들기 위해 Phone Balanced Word(PBW) 음성 데이터베이스에서 단음절, 2음절, 3음절 4음절, 5음절 그리고 5음절 이상으로 단어들을 음절 길이별로 분류하여 각각에 대하여 훈련을 통해 총 6개의 가비지를 만들어 OOV 처리에 사용하였다. 한편, 거부율을 조정하기 위하여 최종적으로 식(1) 같은 과정을 수행함으로서 최종 인식 결과를 얻게 된다.

$$P(W_{in} \mid O) - P(W_{oov} \mid O) > thr \qquad (1)$$

여기서 *thr*은 조정 가능한 거부율이며 $P(W_{in} \mid O)$은 In-Vocabulary에서 얻은 최대 Viterbi 확률, $P(W_{oov} \mid O)$은 가비지들에서 얻은 최대 Viterbi 확률이다.

4. TMS320F28335 DSP를 이용한 하드웨어 설계

다음은 TMS320F28335 DSP를 이용하여 구현된 음성인식 하드웨어 모듈의 블록도이다.

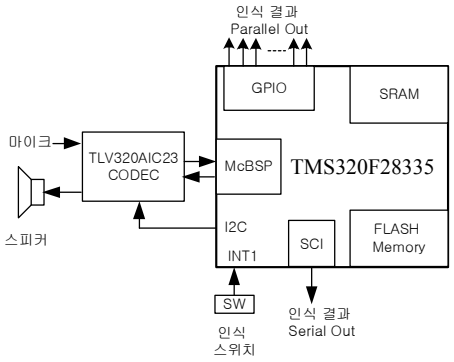


그림 4 음성인식 하드웨어 모듈 구조

1장에서 언급한 바와 같이 비록 TMS320F28335 DSP가 A/D 변환기를 내장하고 있지만, 12bit 해상도를 가지고 있기 때문에 이를 그대로 사용하기 위해서는 별도의 음성데이터를 구축해야 하고, 낮은 해상도로 인하여 충분한 인식률을 얻을 수 없기 때문에, 기존의 음성데이터베이스를 활용하고 충분한 해상도를 얻기 위하여 16bit 해상도를 지원하는 sigma-delta modulation 방식의 코덱인 TLV320AIC23을 TMS320F28335 DSP의 직렬포트인 McBSP에 연결하였다. 이를 위하여 인식기 동작에 필요한 몇몇 인터럽트 버튼을 포함하는 별도의 코덱 보드를 제작하여 사용하였다. 한편, TLV320AIC23는 A/D, D/A된 데이터 전송 외에 레지스터 설정 등의 제어를 위해 I2C 또는 SPI 프로토콜의 직렬포트를 별도로 필요로 하는데, 이를 위해 그림 4에 나와 있는 것처럼 TMS320F28335에 내장된 I2C 포트를 연결하여 사용하였다.

TMS320F28335 DSP는 256K 워드의 플래쉬 메모리와 34K 워드의 SRAM을 내장하고 있기 때문에 적당한 어휘 수의 고립단어 인식기를 구현하기에는 충분한 메모리를 내장하고 있다고 할 수 있다. 반면, TMS320F2833x 계열의 DSP는 다양한 메모리 구성을 하고 있는데, 예를 들어 TMS320F28332 DSP의 경우는 64K 워드의 플래쉬 메모리와 24K 워드의 SRAM을 내장하고 있다. 따라서, 인식 어휘 수에 따라 적당한 용량의 메모리

를 내장한 DSP를 선택하여 사용할 수 있다.

리셋 신호가 인가되면, 플래쉬 메모리에 저장된 음성인식 실행 코드가 플래쉬 메모리 자체에서 실행되는 것이 아니라, 우선 내장 SRAM으로 부팅이 이루어진다. 물론, 플래쉬 메모리에서 직접 실행될 수도 있으나, 내장된 플래쉬 메모리는 SRAM보다 읽기 속도가 느리기 때문에 실행속도를 향상시키기 위해서는 주요 실시간 처리 루틴은 0 wait인 내장 SRAM에서 실행하도록 하였다. 단, 실시간 실행과 관련이 없는 초기화 루틴들은 부팅 없이 플래쉬 메모리에서 직접 실행하도록 하여 SRAM 메모리의 사용량을 줄였다. 인식을 위한 HMM 모델 데이터의 경우는 이를 내부 SRAM으로 옮길 수 있을 만큼의 SRAM 용량이 충분하지 않기 때문에 플래쉬 메모리에서 곧바로 읽어 들이면서 실행이 되도록 하였다.

한편, [3]에서 사용한 고정소수점 DSP의 경우는 DMA를 지원하지 않기 때문에 A/D 변환기에서 발생하는 고속의 이벤트를 CPU가 인터럽트를 이용하여 처리할 수 밖에 없었다. 이는 스트림 데이터의 신호처리보다는 제어 응용에 초점이 맞추어져 있기 때문이다. 반면, TMS320F28335 DSP는 6채널의 DMA를 지원하기 때문에 제어 용도 뿐만 아니라 고속의 스트림 데이터를 신호처리하기에도 매우 적합하다. 일반적으로 DMA를 사용할 경우 sample-by-sample 직렬 포트 인터럽트가 아닌 frame-by-frame의 DMA 인터럽트에 기반함으로서 인터럽트 오버헤드를 대폭 줄일 수 있다. 특히, DMA의 기능에 따라서는 DMA 인터럽트 조차 사용하지 않음으로서 인터럽트 오버헤드를 완전히 없앨 수 있다. TMSF32028335의 DMA에서는 WRAP 관련 레지스터를 적절히 활용함으로서 인터럽트 없는 piong-pong 버퍼 구현이 가능하였다.

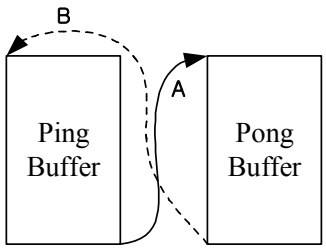


그림 5 DMA를 이용한 ping-pong 버퍼 구현

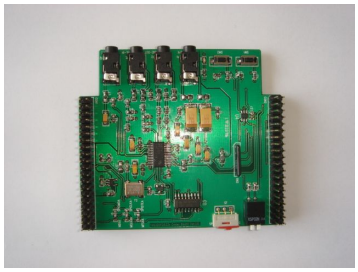
그림 5에서 DMA의 TRANSFER_SIZE는 ping 버버와 pong 버터를 합친 크기로 설정하고, DMA의 SHADOW 레지스터의 기능을 이용하면, ping 버퍼와 pong 버퍼의 모든 전송을 마쳤을 때, source address 또는 destination address가 그림 5의 B와 같이 자동으로 ping 버터의 시작번지로 자동 설정된다. 문제는 ping 버터의 전송을 모두 마쳤을 때 source address 또는 destination address가 그림 5의 A처럼 pong 버퍼의 시작 번지로 설정될 필요가 있는데, 이는 WRAP_SIZE를 ping 버퍼의 크기로 설정하고 WRAP이 발생했을 시 시작 어드레스를 pong 버퍼의 시작 번지로 설정함으로 해결할 수 있었다. 이 때 각각의 버퍼 전송이 완료되었는지는 TRANSFER_COUNTER와 WRAP_COUNTER 를 폴링함으로써 알 수 있다.
인식이 완료되면, 그림 4에 나와 있는 것처럼 충분한 수의 GPIO 단자를 통하여 원하는 제어 신호를 발생할 수 있다. 한편, 인식이 완료된 후 인식 결과를 쉽게 확인하고 보다 친근한 사용자 인터페이스를 위하여 인식 결과를 TLV320AIC23의 D/A를 통해 스피커로 출력하도록 하였다. 이를 위하여 각각의 음성명령어에 해당하는 음성을 별도로 녹음한 후 ADPCM으로 압축하여 플래쉬 메모리에 저장해 놓았다. 인식이 완료되면 해당 명령어의 ADPCM 음성 데이터를 복원하여 스피커를 통해 음성 출력한다.

본 구현에서는 음성 명령어로 TV 리모컨에서 주로 사용하는 15개의 명령어를 사용하였다. 출력을 위한 ADPCM 음성데이터를 제외할 경우, 총 플래쉬 메모리 사용량이 64K 워드 이하이기 때문에, 64K 워드 플래쉬 메모리를 내장한 TMS320F28332에서도 동작 가능하도록 하였다. 그림 6은 본 구현에서 제작한 코덱 보드와 이를 상용 TMS320F28335 DSP 프로세서 모듈에 연결한 모습이다.

5. 결론

본 논문에서는 TMS320F28335 DSP를 이용하여 화자 독립 음성인식기를 위한 하드웨어 설계 및 소프트웨어 구현 방법을 제시하였다. 이를 위하여 음성 분석을 위한 front-end 및 연속확률분포 HMM 알고리즘을 리소스가 제한된 임베디드 환경에 적합하도록 최적화하여, TMS320F2833x 계열 DSP의 내장 메모리만으로 동작하도록 하였다. 화자 독립 인식기에 적합하도록 내장된 12bit의 A/D 변환기을 사용하지 않고, 16bit 해상도의 별도의 코덱 보드를 설계하여 사용하였으며, McBSP로부터 들어오는 스트림 데이터를 인터럽트 없이 효과적으로 처리하기 위하여 DMA를 적절히 활용하였다.

제어 응용에 최적화된 TMS320F28335 DSP와 최소 부품으로 음성 인식 기능을 수행하도록 구현하였기 때문에 음성인식 기능과 기기를 제어하는 기능을 통합함으로서 음성 인식 제어 분야에 적합한 실용적인 음성인식기를 구현할 수 있었다.



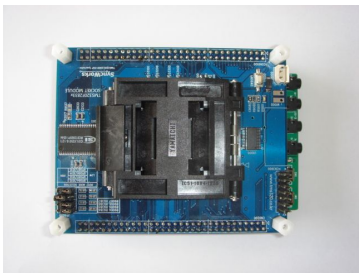


그림 6 구현된 음성인식기

참 고 문 헌

[1] *음성처리 시스템, 기술/시장 보고서*, 한국전자통신연구원, 2001.

[2] Jean-Claude Junqua. *Robust Speech Recognition in Embedded Systems and PC Applications*, Kluwer Academic Publishers, 2000.

[3] 정익주, "TMS320C2000 계열 DSP를 이용한 단일칩 음성인시기 구현", *음성과학*, 제14권, 4호, pp.157~167, 2007.

[4] L. Rabiner, B. Juang, *Fundamentals of Speech Recognition*, Prentice Hall, 1993.

[5] 정익주, "DSP를 이용한 가변어휘 음성인식기 구현에 관한 연구", *음성과학*, 제11권, 3호, pp.143~156, 2004.

[6] 김우성, 구명환, "반음소 모델링을 이용한 거절 기능에 대한 연구." *한국음향학회지*, 제18권 3호, pp. 3~9, 1999.

[7] *TMS320F2833x Digital Signal Controllers Data Sheet*, Texas Instruments, 2008.