

TMS320C5509 DSP를 이용한 AMR-WB 음성부호화기의 실시간 구현

Real-time Implementation of AMR-WB Speech Coder Using TMS320C5509 DSP

지 덕 구*, 최 송 인*
(Deock-Gu Jee*, Song-In Choi*)

* 한국전자통신연구원

(접수일자: 2004년 9월 14일; 수정일자: 2005년 1월 4일; 채택일자: 2005년 1월 12일)

AMR-WB 음성부호화기는 50~7000 Hz의 확장된 대역폭을 갖는 음성신호를 압축/복원하는 광대역 음성부호화기로서 6.60 kbit/s에서 23.85 kbit/s까지 9개의 전송 비트율을 가지고 있다. 본 논문에서는 2개의 MAC (Multiply and Accumulate) 유닛을 가진 TI의 16bit 고정소수점 DSP인 TMS320C5509 DSP를 이용한 AMR-WB 음성부호화기의 실시간 구현에 관하여 논한다. 실시간 구현은 intrinsic을 이용한 C 수준의 구현 및 어셈블리 코딩에 의한 구현을 수행하여 그 결과를 비교하였다. 어셈블리 코딩에 의하여 실시간 구현된 AMR-WB 음성부호화기는 23.85 kbit/s 모드에서 42.9 Mclock의 계산량을 가지며, 사용된 프로그램 메모리는 15.1 kword이고, 데이터 ROM 메모리는 9.2 kword이고 데이터 RAM 메모리는 13.9 kword이다.

핵심용어: 광대역 적응형 다중 비트율 (AMR-WB), 음성부호화기, 실시간 구현, TMS320C5509

투고분야: 음성처리 분야 (2.2)

The adaptive multirate wideband (AMR-WB) speech coder has an extended audio bandwidth from 50 Hz to 7 kHz and operates on nine speech coding bit-rates from 6.6 to 23.85 kbit/s. In this paper, we present the real-time implementation of AMR-WB speech coder using 16bit fixed-point TMS320C5509 that has dual MAC units. Firstly, We implemented AMR-WB speech coder in C language level using intrinsics, and then performed optimization in assembly language. The computational complexity of the implemented AMR-WB coder at 23.85 kbit/s is 42.9 Mclocks. And this coder needs the program memory of 15.1 kwords, data ROM of 9.2 kwords and data RAM of 13.9 kwords.

Keywords: AMR-WB, Speech Coder, Real-time Implementation, TMS320C5509, DSP

ASK subject classification: Speech Signal Processing (2.2)

I. 서론

현재까지 이동통신 시스템에 널리 사용되고 있는 협대역 음성부호화기는 200~3400 Hz 대역폭의 오디오 신호를 압축/복원한다. 이에 반하여 광대역 음성부호화기는 50~7000 Hz의 확장된 대역폭의 오디오 신호를 부호화함으로써 저주파 대역 (20~200 Hz)에서는 신호의 자연성, 현장감 및 편안함을 증가시켜주고, 고주파 대역

(3400~7000 Hz)에서는 명료성을 증가시킴으로써 협대역 음성부호화기에 비하여 월등한 음질을 제공한다[1].

Adaptive Multi-Rate Wideband (AMR-WB) 음성부호화기는 3rd Generation Partnership Project (3GPP)에서 2001년 3월 3세대 이동통신시스템의 광대역 음성부호화기로 승인되었다. AMR-WB 음성부호화기는 6.6 kbit/s에서 23.85 kbit/s까지 9개의 전송 비트율을 갖는 ACELP (Algebraic Code Excited Linear Prediction) 기반의 광대역 음성부호화기로 무선 채널 환경에 따라 적절한 비트율을 사용함으로써 고정 비트율을 사용하는 음성부호화기에 비하여 우수한 통화품질을 제공한다[1].

책임저자: 지 덕 구 (dgjee@etri.re.kr)
305-350 대전광역시 유성구 가정동 161
한국전자통신연구원 이동A/V연구팀
(전화: 042-860-1296; 팩스: 042-860-5452)

DSP에 의한 시스템의 개발은 ASIC에 의한 개발에 비하여 개발기간의 단축 및 수정이 용이하다는 장점이 있다. 또한 고속의 DSP가 개발되고 편리한 개발 환경 및 효율적인 C 컴파일러가 제공됨에 따라 C 언어 수준에서의 시스템 구현도 가능하다. 그러나 이동통신 단말기와 같이 battery로 동작하는 휴대기기에서는 전력소모량을 최소화하기 위하여 어셈블리 코딩에 의한 시스템 구현이 요구된다.

본 논문에서는 단일 클럭 사이클에 2개의 MAC 연산을 수행할 수 있는 TI의 16bit 고정소수점 DSP인 TMS 320C5509 DSP를 이용한 AMR-WB 음성부호화기의 실시간 구현에 관하여 논한다. 본 논문의 II장에서는 AMR-WB 음성부호화기의 알고리즘에 관하여 간략히 설명하고, III장에서는 AMR-WB의 C 언어 및 어셈블리 언어에 의한 구현 및 결과에 대하여 설명하고, IV장에서는 결론을 맺고자 한다.

II. AMR-WB 음성부호화기

AMR-WB 음성부호화기는 ACELP에 기반을 둔 광대역 음성부호화기로 전송 비트율에 따른 9개의 음성 부호화 모드 (23.85, 23.05, 19.85, 18.25, 15.85, 14.25, 12.65, 8.85, 6.6 kbit/s)와 1개의 배경잡음 부호화 모드로 동작한다[2].

AMR-WB 음성부호화기에서 샘플링 주파수는 16kHz이고 한 프레임의 크기는 320샘플에 해당하는 20ms이며 인코더의 블록도는 그림 1과 같다. 복잡도를 줄이고 더 중요한 주파수 영역에 비트를 할당하기 위하여 입력 신호는 저주파 대역 (50~6400Hz)과 고주파 대역 (6400~7000Hz)으로 나뉘어 각각 부호화된다. 16kHz로 샘플링된 입력신호는 12.8 kHz로 다운샘플링 후 고역 통과 필터와 프리엠퍼시스 필터를 사용한 전처리 과정을 거친다. 인코더는 매 프레임마다 LP 분석, 개루프 피치 검색, 적응 코드북 검색, 대수 코드북 검색의 과정을 통하여 추출된 CELP 모델의 파라미터를 디코더로 전송하고, 디코더는 전달된 파라미터를 사용하여 음성신호를 복원한다.

LP분석 단계에서는 16차의 LP (Linear Prediction) 계수를 20ms 프레임 단위로 구한다. LP 계수는 LSP 계수와 동일한 특성을 나타내며 양자화 성능이 좋은

immittance spectrum pair (ISP)[3]로 변환하여 split-multistage 벡터양자화기에 의해 양자화한다.

피치 분석은 피치 탐색 과정을 간략화하기 위하여 3단계로 나뉘어 진다. 먼저 첫번째 단계에서는 개루프 분석을 통하여 정수 피치값을 결정한다. 개루프 피치검색은 계산량을 감소시키기 위하여 가중화한 후 데시메이션 과정을 거친 음성신호 상에서 수행하며, 6.60 kbit/s 모드에서는 프레임 당 한번을 수행하고, 나머지 모드에서는 두 번을 수행한다. 두 번째 단계에서는 정수 피치 값 주변의 ± 7 샘플 범위에서 최적의 정수 피치 값을 검색하기 위한 페루프 검색을 수행한다. 세 번째 단계에서는 최적의 정수 피치 값 주변에 대하여 소수 값의 피치 지연을 검색하며, 모드와 피치의 범위 그리고 서브프레임에 따라 1/4 또는 1/2 샘플의 분해능을 사용한다. 2단계와 3단계의 페루프 검색은 입력신호와 합성신호의 평균 자승오차를 최소화하는 정수값 및 소수값의 피치를 결정한다.

광대역 음성부호화기는 고음질을 보장하기 위하여 협대역 음성부호화기 보다 훨씬 많은 수의 펄스 수를 가진 대수 코드북을 사용하며, 따라서 대수 코드북 검색에 많은 계산량을 요구한다. AMR-WB에서는 적당한 계산량에서 비교적 좋은 음질을 유지하기 위하여 검색 범위를 많이 한정된 depth-first 트리 검색방식을 사용한다.

AMR-WB의 대수 코드북은 interleaved single-pulse permutation (ISPP)을 기반으로 설계되었다. 64개의 펄스 위치를 갖는 코드벡터는 각각 16개의 펄스 위치를 갖는 4개의 트랙으로 나뉘어 진다. 전송률에 따라 각 코드북은 트랙마다 1개에서 6개까지의 펄스를 갖는다.

대수 코드북 검색은 페루프 피치 검색의 목표신호 $x(n)$ 에서 적응 코드북 성분을 제거한 신호 $x_2(n)$ 를 사용하여 검색한다.

$$x_2(n) = x(n) - g_p y(n), \quad n = 0, \dots, 63 \quad (1)$$

여기서 $y(n)$ 은 필터링된 적응 코드북 벡터이며 g_p 는 적응 코드북 이득이다.

대수 코드북 검색은 식 (2)의 검색 기준을 최대화 하는 코드 벡터 c_k 를 검색함으로써 가장 입력 신호와 가장 합성 신호의 평균 자승 오차를 최소화하는 펄스의 위치와 부호를 검색한다.

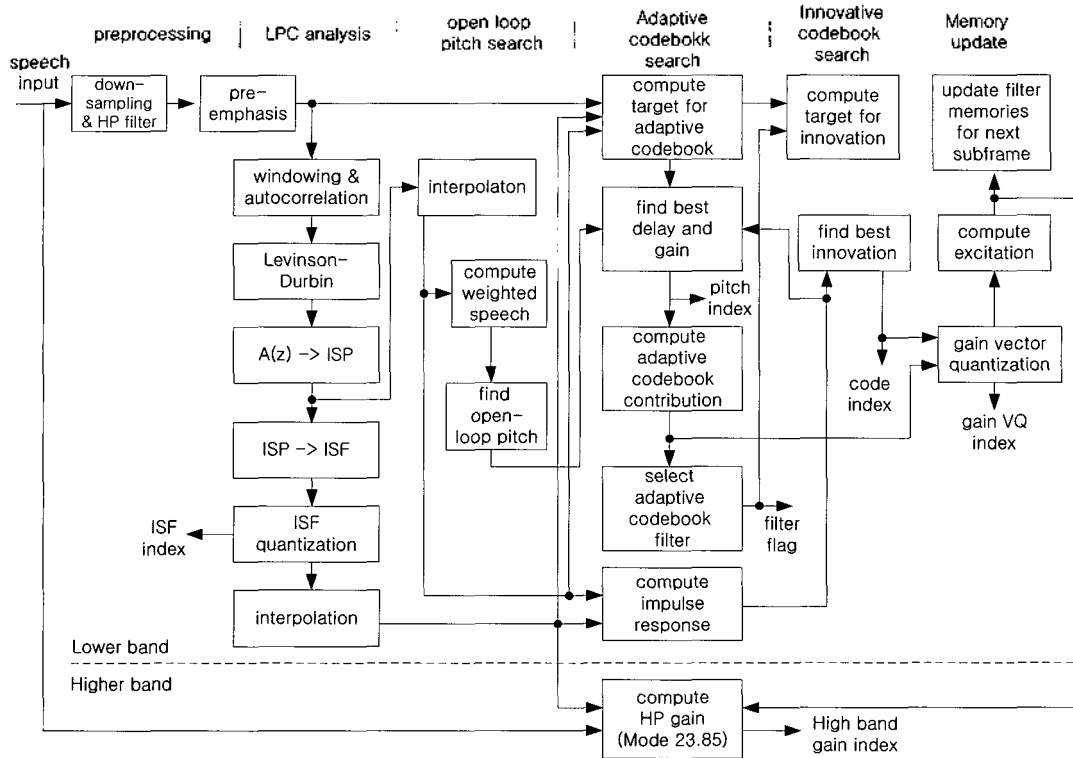


그림 1. AMR-WB 음성부호화기의 인코더 블록도
 Fig. 1. Block Diagram of AMR-WB encoder.

$$Q_k = \frac{(d'c_k)^2}{c_k' \Phi c_k} = \frac{(R_k)^2}{E_k} \quad (2)$$

여기서 $d = H'x_2$ 는 목표신호 $x_2(n)$ 과 임펄스응답 $h(n)$ 의 상관도이며 $\Phi = H'H$ 는 $h(n)$ 의 상관 행렬이다.

AMR-WB에서는 인접한 두 트랙에서 한번에 2개의 펄스를 검색하는 특별한 형태의 depth-first 트리 검색 방법을 사용한다. 메모리 요구량을 감소시키기 위하여 $N \times N$ 의 전체 매트릭스 대신 미리 계산되어 저장된 상관 매트릭스의 필요한 부분들만 선택하여 사용한다. 또한 복잡도를 감소시키기 위하여 두 개의 가능한 펄스 조합을 테스트하는데 있어서, 테스트되는 첫번째 펄스의 위치를 제한된 수로 고정시킨다.

III. AMR-WB의 실시간 구현

AMR-WB 음성부호화기는 TMS320C5509 DSP, 4 Mword의 DRAM, Flash ROM, USB, 8 kHz ~ 96 KHz

의 샘플링을 제공하는 TLV320AIC23 오디오 코덱을 장착하고 JTAG 인터페이스를 제공하는 Spectrum Digital사의 TMS320C5509 EVM 보드를 사용하여 구현하였다. TMS320C5509 DSP는 128 Kword (32 Kword의 DARAM (Dual Access RAM), 96 Kword의 SARAM (Single Access RAM))의 RAM을 내장하고 있으며, 2개의 MAC 유닛을 사용하여 1 clock cycle에 2개의 곱셈 연산이 가능하며, 최대 200 MHz로 동작한다.

AMR-WB 음성부호화기의 실시간 구현은 3GPP에서 제공되는 고정 소수점 C 프로그램[4]을 사용하여 실시간 구현하였다. 실시간 구현은 먼저 1단계에서 C 언어 수준의 실시간 구현 후 2단계에서 어셈블리 코딩에 의한 최적화 작업을 수행하였다. C 언어 수준의 실시간 구현은 어셈블리 구현을 용이하게 하고, 어셈블리 구현에 의한 계산량 감소 효과를 각 세부 기능별로 비교 평가하기 위하여 수행하였다. 2단계에서 어셈블리 언어 수준의 최적화 작업은 함수 호출 계층 구조를 바탕으로 하위 순위의 함수에서 상위 순위의 함수 순서로 수행하였다. 어셈블리 언어로 작성된 함수는 C 컴파일러의 함수 호출 규칙에 따라 코딩하여 C 코드에서 호출하여 사용할 수 있도록 하였다. 작성된 어셈블리 함수는 1단계의 C 레벨의 함수를 대체하여 테스트하는 과정을 거쳤다. 이러한 방

표 1. TMS320C55x 와 TMS320C54x DSP용 LP 합성 필터의 복잡도 비교

Table 1. Complexity comparison of TMS320C55x and TMS320C54x DSP code.

Subroutine	Complexity	
	C55x DSP	C54x DSP
Synth_filt	900 Clock	2555 Clock
Synth_filt 32	1860 Clock	3219 Clock

법은 바로 어셈블리 언어에 의한 최적화 작업을 수행하는 방법에 비하여 작성된 어셈블리 코드를 검증하기가 용이한 장점이 있다.

3.1. C 언어 level의 실시간 구현

일반적으로 범용의 고정소수점 DSP에서 수행되는 연산은 곱셈 연산을 포함하여 1개의 명령어에 의하여 1 cycle에 수행되며, 연산 결과는 16bit 또는 32bit 범위의 값을 가진다. AMR-WB의 고정 소수점 C 프로그램은 DSP 명령어에 대응되는 기본 연산자에 의해 수행된다. 기본 연산자는 연산 결과를 16bit 또는 32bit으로 제한하기 위한 루틴을 포함하는 함수로 이루어져 있다. 따라서 C 프로그램을 그대로 사용할 경우 빈번한 기본 연산 함수의 호출로 인하여 복잡도가 매우 높아진다. 따라서 이러한 문제점을 해결하기 위하여, C 언어 수준의 실시간 구현 단계에서 기본 연산 함수는 어셈블리 명령어와 1:1로 대응되는 C compiler intrinsic으로 대체하여 구현하였다. C compile시 최적화 옵션은 루프 최적화 및 루프 unrolling 등을 수행하는 최적화 옵션 o2를 사용하였다.

3.2. 어셈블리 level의 실시간 구현

어셈블리 코드의 최적화 작업은 각 서브루틴 단위로 수행하였다. 이때 서브루틴내에서 대부분의 계산량을 소요하는 부분은 반복 문이므로 반복 문 내의 복잡도를 감소시키기 위한 레지스터의 할당 및 명령어의 선택에 유의하였다. C55x DSP는 반복 문 구성을 위한 2개의 레지스터를 제공하므로, 이 레지스터를 이용하여 2중 반복 문을 구성하였다. AMR-WB 알고리즘의 전처리 루틴에서는 데이터 이동에 의한 복잡도를 감소시키기 위하여 circular 어드레싱을 적용하였다.

TMS320C55x DSP는 다중 버스 아키텍처, 듀얼 MAC 유닛, 그리고 분리된 Program unit (PU), Address unit (AU), Data computation unit (DU)을 사용하여 다양한 2개 명령어의 병렬처리를 제공한다. 따라서 어셈블리 코

표 2. AMR-WB 음성부호화기의 모드별 계산량

Table 2. Complexity of AMR-WB speech coder.

Bit rate (kbit/s)	Encoder (Mclock)	Decoder (Mclock)	Total (Mclock)
6.60	25.0	6.5	31.5
8.85	30.7	5.9	36.6
12.65	36.1	5.1	41.2
14.25	39.9	5.2	45.1
15.85	40.0	5.3	45.3
18.25	41.2	5.4	46.7
19.85	42.7	5.4	48.1
23.05	42.7	5.7	48.4
23.85	36.7	6.2	42.9

당과정에서 최대한 병렬 처리를 사용하여 복잡도를 감소시키도록 노력하였다. 특히 두개의 MAC 유닛은 1 cycle에 2개의 곱셈 연산을 병렬로 수행함으로써 산호처리에서 가장 많이 사용되는 multiply-and-add 연산을 효과적으로 처리 할 수 있다. 표 1은 병렬 MAC 연산에 의한 복잡도 감소효과를 비교하기 위하여 각각 C55x DSP와 단일 MAC 유닛을 가진 C54x DSP용으로 개발된 LP 합성 필터의 복잡도를 나타내었다. 함수 synth_filt()의 경우 병렬 MAC 명령어를 사용한 C55x 코드는 C54x 코드에 비하여 복잡도가 약 35%로 감소하였음을 알 수 있다.

일반적으로 한번의 곱셈 연산에는 2개의 피연산자가 필요하므로 2개의 곱셈연산을 위해서는 4개의 피연산자가 요구된다. 그러나 C5509 DSP는 3개의 읽기용 데이터 버스를 가지고 있다[5]. 따라서 2개의 곱셈 연산을 병렬로 수행하기 위해서는 1개의 피연산자를 2개의 MAC 연산에서 공통으로 사용하는 특별한 방법이 요구된다[6]. 따라서 어셈블리 구현 과정에서 대부분의 FIR 필터와 LP 합성필터, 기타 병렬 MAC 연산이 가능한 루틴은 참고문헌[6]의 방법을 사용하여 어셈블리 코딩하였다.

3.3. 실시간 구현 결과

어셈블리 코딩에 의한 최적화 과정을 통하여 실시간 구현된 AMR-WB 음성부호화기는 3GPP에서 제공하는 24개의 테스트 데이터와 5개의 DTX 테스트 데이터를[7] 이용하여 각 모드별로 검증하였으며, 인코더 및 디코더 모두 비트 단위로 일치함을 확인하였다.

구현된 AMR-WB 음성부호화기의 프로그램 크기는 15 kword이고 ROM 메모리는 9.2 kword이며 RAM 메모리는 13.8 kword이다. 각 전송 모드별 계산량은 표 2와 같다. 6.60 kbit/s 모드에서 31.5 Mclock으로 가장 낮은 복잡도를 나타내었으며 23.05 kbit/s 모드에서 48.4 Mclock 가장 높은 복잡도를 나타내었다. 따라서 구현된

표 3. TMS320C5509 DSP와 TeakLite DSP와의 계산량 비교
Table 3. Complexity comparison of TMS320C5509 DSP and TeakLite implementation results.

Function block (23.85kbit/s)	Complexity (Mclock)		
	C5509 DSP		TeakLite DSP
	C	ASM	
Pre processing	3.5	0.8	1.9
VAD	2.1	1.4	1.0
LPC analysis	17.5	7.1	7.8
Open loop pitch	2.7	1.5	3.2
Adaptive cb search	12.2	4.1	7.8
Algebraic cb search	21.1	17.2	21.6
Memory update	9.6	4.2	0.4
Decoder	18.4	6.2	8.6
Total	87.1	42.9	52.2

AMR-WB 음성부호화기는 전송률에 따라 200 MHz로 동작하는 TMS320C5509 DSP 성능의 16 ~ 25% 만을 사용하여 실시간 동작이 가능하다.

표 3은 23.85 kbit/s 모드에서 주요 기능별로 C 언어 수준에서 구현된 결과와 어셈블리 언어로 구현된 결과, 그리고 참고문헌[8]의 TeakLite DSP를 사용하여 구현한 결과를 비교하였다. 23.85 kbit/s 모드에서 TMS320C5509 DSP를 사용하여 구현한 결과는 TeakLite DSP를 사용하여 구현한 결과보다 9.3 Mclock이 감소하였음을 알 수 있다.

표3에서 C 언어 수준의 구현 복잡도는 87.1 Mclock으로 TMS320C5509 DSP 성능의 약 44%를 사용하여 실시간 동작함을 알 수 있다. 따라서 기본연산함수의 intrinsic 변환 및 C 컴파일 만으로도 짧은 기간 (약 일주일) 안에 실시간 구현이 가능함을 확인할 수 있었다. 어셈블리 코딩에 의한 최적화 결과 복잡도 감소율은 C 언어 수준의 구현결과와 비교하여 평균 49.3%를 나타내었다. 각 세부 기능별로는 Circular 어드레싱이 적용된 전처리 루틴의 복잡도 감소율이 23%로 가장 높았으며, 병렬 MAC 연산이 많이 사용된 LP 분석 및 적응 코드북 검색 루틴에서 감소율이 큼을 확인할 수 있었다. 인코더 전체 계산량의 약 40%를 차지하는 대수 코드북 검색 루틴의 계산량 감소율은 81.5%로 여전히 높은 복잡도를 나타내었다. 이는 다른 루틴과는 달리 대수 코드북 검색 루틴이 알고리즘상 병렬 MAC 연산의 적용이 어려웠기 때문으로 풀이된다.

IV. 결론

최근에는 DSP의 계산 능력이 향상되고 손쉬운 개발 툴 및 비교적 효율적인 C 컴파일러가 제공됨에 따라 어셈블리 코딩에 의한 계산량 감소의 필요성이 점차 줄어드는 추세이다. 그러나 더 복잡한 알고리즘의 등장과 휴대용 기기와 같이 전력 소모량을 최소화해야 하는 기기에서는 여전히 중요한 문제이다.

본 논문에서는 AMR-WB 음성부호화기를 TMS320C5509 DSP를 이용하여 실시간 구현하였다. 구현된 AMR-WB 음성부호화기의 계산량은 23.85 kbit/s 모드에서 42.9 M clock이고 프로그램 메모리 크기는 15 kword이며 데이터 메모리 크기는 23 kword이다.

실시간 구현은 먼저 계산량 분석 및 어셈블리 코드의 점증을 용이하게 하기 위하여 intrinsic을 사용한 C 언어 수준의 구현 후 어셈블리 코딩에 의한 최적화 작업을 수행하였다. 어셈블리 최적화 과정에서 계산량 감소를 위하여 특히 병렬 MAC 연산을 최대한 활용하도록 노력하였으며, 병렬 MAC 연산이 많이 적용된 부분에서의 계산량 감소 효과를 확인할 수 있었다. 그러나 인코더에서 가장 높은 복잡도를 나타내는 대수 코드북 검색 루틴은 여전히 높은 복잡도를 나타내고 있으며, 추후 이 부분의 최적화 작업을 더 진행할 예정이다.

참고 문헌

1. B. Besette, R. Salami, R. Lefebvre, M. Jelinek, J. Rotola-Pukkila, J. Vainio, H. Mikkola, K. Jarvinen, "The Adaptive Multirate Wideband Speech Codec (AMR-WB)", IEEE Trans. on speech and audio processing, 10 (8), 2002.
2. 3GPP TS 26.190, "AMR Speech Codec: Transcoding function", 2001.
3. Y. Bistriz and S. Pellerin, "Impedance Spectral Pairs (ISP) for speech coding", ICASSP93, pp. 11-9 - 1-12.
4. 3GPP TS 26.173, "ANSI-C code for the Adaptive Multi Rate Wideband speech codec", 2001.
5. Texas Instruments, "TMS320C55x DSP CPU reference Guide", 2001.
6. Texas Instruments, "Efficient Implementation of Real-Valued FIR filters on the TMS320C55x DSP", Application Report, SPRA655, 2000.
7. 3GPP TS 26.174, "AMR Wideband Speech Codec test sequences", 2001
8. 변경진, 정희범, 김경수, 현민수, "TeakLite DSP를 이용한 적응형 다중 비트율 광대역 (AMR-WB) 음성부호화기의 실시간 구현", 한국음향학회지, 23 (3), 262-267, 2004

저자 약력

• 지 덕 구 (Deock-Gu Jee)

1998년 2월: 충북대학교 전자공학과 졸업(공학석사)
2000년 2월: 충북대학교 전자공학과 졸업(공학석사)
2000년 8월~현재: 한국전자통신연구원 이동A/V연구팀 연구원
주관심분야: 이동통신, 음성/영상신호처리

• 최 송 인 (Song-In Choi)

1982년 2월: 광운대학교 응용 전자공학과 졸업(공학석사)
1987년 2월: 광운대학교 전자계산기공학과 졸업(공학석사)
1982년 7월~현재 : 한국전자통신연구원 이동A/V연구팀 책임연구원
주관심분야: 이동통신, 음성신호처리, 영상신호처리