論文 95-9-9-14

가변 저 비트율 음성 부호기 개발 및 실시간 구현

正會員 辛 哲 善*, 鄭 在 皓**

Development and Real-Time Implementation of a Variable Low Bit-Rate Speech Coder

Chul Sun Shin*, Jae Ho Chung** Regular Members

이 논문은 1993년도 한국학술진흥재단의 공모과제 연구비에 의하여 연구되었음

要 約

본 논문에서는, 가변 저 비트율 (variable low bit-rate)을 갖는 효율적인 음성 부호기 알고리듬을 제안하였고, 이를 한 개의 DSP 칩을 사용하여 실시간 구현 하였다. 최근 이동통신 (예를 들면, 디지틀 셀룰러 폰)과 음성 데이타 저장 (예를 들면, 자동 응답 전화기) 등의 응용분야에서, 효율적인 음성 부호기의 필요성이 고조되고 있다. 이와같은 응용분야에서는, 최소한의 정보로 최대한의 성능 (즉, 음질, 부호화 지연 등)을 유지하며 적은 비용으로 실시간 구현될 수 있는 효율적인 음성 부호기 시스템이 요구된다. 이에 본 논문에서는, 무성음에 대해서는 2.56Kbps 그리고 유성음에 대해서는 6.6Kbps의 가변 비트율을 갖는 음성 부호기 알고리듬을 제안 하였으며, TI 사 (Texas Instruments, Inc.)의 TMS320C30 DSP 칩 한 개와 60KB의 메모리를 이용하여 제안된 알고리듬을 실시간 구현하였다. 개발된 실시간 음성 부호기는 한국어 문장들과 영어 문장들에 대하여 각각 5.66Kbps와 5.43Kbps의 평균 전송율을 나타내었다. 음질면에서는, 한국어 문장들에 대하여 평균 11.144dB, 영어 문장들에 대하여 평균 10.047dB의 좋은 성능을 보였다. 또한 MOS 테스트를 통하여서는, 한국어 문장들에 대하여 3.25, 영어 문장들에 대하여서는 3.36의 좋은 점수를 얻었다.

ABSTRACT

In this paper, an efficient variable low bit-rate speech coder is developed, and real-time implemented using a single DSP chip. In the areas such as mobile cellulars (e.g., digital cellular phone) and speech data storage systems (e.g., automatic answering machine), efficient speech coding systems are required. In those application areas, a speech coder which can be real-time imple-

*LG전자 미디어통신 연구소
**인하대학교 전자공학과 디지탈 신호처리 연구실
Department of Electronic Engineering, Digital Signal Processing Laboratory, Inha University
論文番號 : 95091-0228
接受日字 : 1995年 2月 28日

mented with low cost while still maintaining its maximum performance (i.e., speech quality, coding delay, etc.) is necessary. In this paper, a variable low bit-rate speech coding system, having bit-rates of 2.56Kbps for unvoiced signal and 6.6Kbps for voiced signal, is developed and implemented using a single TMS320C30 DSP chip along with 60KB memory. The developed real-time speech coder has the average bit rates of 5.66Kbps and 5.43Kbps for Korean and English, respectively. The quality of the synthesized speeches were very good. The SNRs (Signal-to-Noise Ratios) of the synthesized Korean and English speeches were 11.144dB and 10.047dB in average, respectively. The subjective tests were also performed using the MOS tests. The developed real-time coder achieved 3.25 and 3.36 MOS scores for Korean and English speeches, respectively.

Ⅰ. 서 론

최근 이동 통신 분야에서는 급증하는 수요에 비해 제한된 주파수 대역으로 인하여, 기존의 아날로그 방식의 이동통신 시스템이 제공할 수 있는 통신 수용 용량은 포화 상태에 이르고 있다. 이에 미국, 유럽, 일본 등에서는 용량 증대를 확대하기 위해 디지틀 셀룰러 방식의 이동 통신 시스템을 연구 개발 중이며, 이에 효율적인 저 비트율 음성 부호기가 요구되어지고 있다. 또한, 저 비트율의 음성 부호기는 디지틀 저장 매체를 이용하여 녹음 및 재생하는 응용분야에서도 효율적으로 사용되어질 수 있다. 대표적인 예로는 자동 응답 전화기, 음성 사서함 서비스, 어학용 카세트 등이 있으며, 이미 몇몇 제품에서는 저장을 위해 사용되는 메모리의 양을 줄이기 위해 저 비트율 음성 부호기를 사용하고 있다. 이와 같이 여러 응용 분야에서 효율적이며, 적은 비용으로 구현할 수 있는 음성 부호기 개발이 요구되고 있다. 본 논문에서는 가변 비트율을 적용한 효율적인 음성 부호기 알고리듬을 제안하고 있으며, 이를 1개의 부동 소수점 (floating point) DSP 칩을 사용하여 실시간 구현하였다.

제안된 음성 부호기는 Atal과 Schroeder에 의하여 제안된 CELP (Code-Excited Linear Predictive) 부호기[1]에 기반을 두고 있으나, 가변 비트율을 적용하여 부호기의 성능 향상을 도모하였다. 또한, 새로운 여기신호 (residual signal) 분석 방법을 사용하여, 실시간 구현에 가장 큰 관건이 되는 연산량을 크게 줄였다. 이를 위하여, 제안된 부호기에서는 각각의 LPC (Linear Predictive Coding) 프레임을 유성음 프레임과 무성음 프레임 중 하나로 판별하여, 유성음 프레임의 경우 각각의 LPC 프레임에 4개의 long-term 서브 프레임과 8개의 short-term 서브 프레임을 두어 여기신호를 분석하며, 무성음 프레임의 경우 각각의 LPC 프레임에 4개의 short-term 서브 프레임만 두어 여기신호를 분석한다. 이와 같은 방법으로 여기신호를 분석할 때, 유성음 프레임에서는 short-term의 프레임 크기가 기존의 CELP에서 보다 반으로 작아져 연산량을 대폭 줄일 수 있으며, 무성음 프레임에서는 유성음의 주기적 성분을 나타내기 위해 사용되는 long-term 분석을 하지 않음으로 해서, 음질은 떨어뜨리지 않으면서 비트율을 대폭 낮출 수 있었다. 또한, 제안된 음성 부호기 알고리듬은, 가장 많은 연산량이 요구되어지는 long-term 분석과정과 short-term 분석과정에서는, 계속적으로 사용되는 합성 결과를 메모리에 저장시켰다 재사용하는 방법을 사용하였으며, 그 결과 TMS320C30 DSP 칩 (33MHz, 16.5Mips)[2] 한 개와 60KB의 메모리를 이용하여 부호기 및 복호기를 실시간 구현하였다.

개발된 실시간 음성 부호기는, 무성음 프레임에 대해서는 2.56Kbps 그리고 유성음 프레임에 대해서는 6.6Kbps의 비트율을 갖는다. 실험을 통하여, 개발된 실시간 부호기는 한글 문장들과 영어 문장들에 대하여 각각 5.66Kbps와 5.43Kbps의 평균 전송율을 나타냈으며, 음질에서는 평균 11.144dB와 10.047dB의 좋은 성능을 보였다.

2절에서는 제안된 알고리듬에 대하여 소개 하였으며, 3절에서는 알고리듬의 실시간 구현과정에 대하여 설명하였다. 4절에서는 실시간 구현한 음성 부호기의 성능평가를 위하여 행한 실험들과 그 결과들에 대하여 설명하였으며, 마지막으로 5절에서는 본 논문의 결론을 기술하였다.

Ⅱ. 제안된 음성 부호기의 기본구조

2.1 음성신호의 유/무성음 판별 알고리듬

제안된 음성 부호기는 분석/합성 여기신호 분석 방법(analysis-by-synthesis excitation analysis)을 사용하는데 프레임을 유성음과 무성음 두 종류로 구분하여, 이에따라 여기신호를 구성하는 코드북을 다르게 사용한다. 프레임이 유성음으로 결정되면 음성신호의 주기적 성분을 여기시켜 주기 위한 적응 코드북 (adaptive codebook, 또는 long-term predictive codebook)과 비주기적 성분을 여기시켜 주기 위한 고정 코드북 (fixed codebook, 또는 short-term predictive codebook)을 모두 사용하며, 프레임이 무성음 프레임으로 결정되면 무성음 프레임은 비주기적 성분만 가지고 있기 때문에 고정 코드북만 사용한다. 이때, 유성음 프레임과 무성음 프레임은 각각 비트율이 다르므로 음성 부호기의 비트율은 가변적으로 된다.

음성신호를 유성음과 무성음으로 구분하는 데는, 일반적으로, 에너지, 영교차비 (zero-crossing rate), 상관 계수 (correlation coefficient), 1차 반사 계수, 2차 반사 계수 등 여러 가지 측정 방법[3][4]을 사용한다. 본 논문에서는 음성 부호기의 실시간 처리를 위해 (즉, 연산량을 줄이기 위해) LPC 계수를 구하는 과정에서 얻어지는 에너지와 영교차수 만을 사용하여 다음과 같은 효율적인 방법으로 유성음/무성음 판별을 한다.

```
if (영교차수)100)      무성음 프레임
else if (ER1(0.05 or ER2(0.04)  무성음 프레임
else if (ER1)10 or ER2)15)  유성음 프레임
else      이전 프레임의 종류를 따른다
```

여기서, ER1은 현재 프레임 에너지와 이전 프레임 에너지와의 비이고, ER2는 현재 프레임 에너지와 두 프레임 이전의 프레임 에너지와의 비이다.

주파수가 8KHz인 음성신호에 대하여, 200 샘플씩을 한 프레임으로 하고, 각 프레임마다 유/무성음 판별을 시도 하였다. 먼저, 현재 프레임에서 영교차 수를 구하여 그 값이 100을 넘으면 현재 프레임을 무성음 프레임으로 판별한다. ER1이나 ER2가 매우 작을 때에는 한 프레임 혹은 두 프레임 이전의 프레임 에너지에 비해 현재 프레임의 프레임 에너지가 매우 작아서 유성음 상태에서 무성음 상태로 변이했다고 가정하여 현재 프레임을 무성음 프레임으로 판별하고, ER1이나 ER2가 매우 클 때에는 반대의 이유에서 현재 프레임을 유성음 프레임으로 판별한다. ER1이나 ER2의 값이 매우 크거나 작지 않은 경우에는 분석중인 프레임의 상태가 이전 프레임들의 상태와 비교해 크게 변하지 않았음을 의미하므로 현재 프레임은 이전 프레임의 상태를 따르게 한다. 이와 같이 에너지의 비에 의해 유성음/무성음을 판별하는 방법은 현재 입력되는 음성 신호의 에너지 레벨에 상관없이 유성음/무성음을 판별할 수 있다는 장점을 갖고 있다.

2.2 여기신호 분석

여기신호를 만들 때 사용되는 코드북은 프레임의 유/무성음 판별에 따라 다르게 사용되는데, 유성음일 경우 〈그림 1〉, 무성음일 경우 〈그림 2〉와 같은 구조를 갖는다.

LPC 프레임	200 samples							
적응 코드북	50		50		50		50	
고정 코드북	25	25	25	25	25	25	25	25

그림 1. 유성음 프레임에서 여기신호를 위한 코드북 구조
Figure 1. Codebook structure for excitation in voiced frame

LPC 프레임	200 samples			
고정 코드북	50	50	50	50

그림 2. 무성음 프레임에서 여기신호를 위한 코드북 구조
Figure 2. Codebook structure for excitation in unvoiced frame

제안된 음성 부호기에서는 여기신호 분석을 위하여 Atal과 Remde에 의하여 제안된 분석/합성 여기 신호 분석 방법[5][6]을 사용하였다. 유성음 프레임에서는 적응 코드북 검색 (adaptive codebook search 또는 long-term predictive search)과 고정 코드북 검색 (fixed codebook search 또는 short-term predictive search)을 모두 수행하며, 무성음 프레임에서는 고정 코드북 검색만을 수행한다[7]. 즉, 유성음 프레임은 식 (1), 무성음 프레임은 식 (2)에 의해 여기신호를 구

한다.

$$e(n) = b_0\, e(n-r_0)$$
$$+b_{10}\, f_{r_{10}}(n)\{u(n)-u(n-\mathrm{FCD})\} \quad (1)$$
$$+b_{11} f_{r_{11}}(n-\mathrm{FCD})\{u(n-\mathrm{FCD})-u(n-\mathrm{ACD})\}$$
$$e(n) = b_2 f_{r_2}(n) \quad (2)$$

여기서, $f_{r_{10}}(n)$과 $f_{r_{11}}(n)$은 코드북 디멘젼이 FCD(=25)인 가우스 코드북 (Gaussian codebook)이고, $f_{r_2}(n)$은 코드북 디멘젼이 ACD(=50)인 가우스 코드북이며, $u(n)$은 단위 계단 (unit step) 함수이다.

유성음 프레임에서 적응 코드북 검색[7]은 하나의 LPC 프레임 (LPC 프레임 크기=200 [샘플])에서 4 번을 수행하는데, 가중 입력 음성신호 (weighted speech signal)와 적응 코드북에 의한 가중 합성 음성신호와의 가중 평균자승 에러 (weighted mean-square error)를 최소화시키는 적응 코드북 인덱스 r_0와 적응 코드북 이득 b_0를 구하는 것으로 식 (3)과 식 (5)를 이용한다. 식 (3)의 $E_{LT}(r)$를 최소화하는 r이 적응 코드북 인덱스 r_0가 되며, 식 (5)에 r_0를 대입하여 적응 코드북 이득 b_0를 구한다. 식 (3)에서 $y_0(n)$은 필터 메모리 성분이 제거된 가중 입력 음성 신호이며, 식 (4)의 $x_r(n)$은 과거의 여기신호 $e(n)$과, LPC 계수로부터 얻어진 길이가 IRS인 가중 임펄스 응답 $g(n)$과의 컨벌루션에 의해서 구해진다. ACD(=50 [샘플])는 적응 코드북 디멘젼이고, ACS(=2^7=128 [샘플])는 적응 코드북의 크기이다.

$$E_{LT}(r) = \sum_{\mathrm{ACD}} y_0^2(n) - \frac{\{\sum_{\mathrm{ACD}} y_0(n)\, x_r(n)\}^2}{\sum_{\mathrm{ACD}} x_r^2(n)}$$

$$(\mathrm{ACD} \le r < \mathrm{ACD}+\mathrm{ACS}) \quad (3)$$

$$x_r(n) = g(n) * e(n-r), \quad (0 \le n < \mathrm{ACD}) \quad (4)$$

$$b_0 = \frac{\sum_{\mathrm{ACD}} y_0(n)\, x_{r_0}(n)}{\sum_{\mathrm{ACD}} x_{r_0}^2(n)} \quad (5)$$

고정 코드북 검색[7]은 프레임의 유/무성음 판별에 따라 다르게 수행되며, 기본적인 개념은 가중 입력 음성신호에서 적응 코드북에 의한 가중 합성 음성신호를 뺀 값과 고정 코드북에 의한 가중 합성 음성신호와의 가중 평균자승 에러를 최소화시키는 적응 코드북 인덱스 r_1 (즉, r_{10}와 r_{11})과 적응 코드북 이득 b_1 (즉, b_{10}와 b_{11})을 구하는 것이다.

유성음 프레임의 경우, 하나의 적응 코드북 서브 프레임에서 2번의 고정 코드북 검색을 수행한다. 가중 입력 음성신호 $y_0(n)$과 적응 코드북에 의한 가중 합성 음성신호 $b_0\, x_{r_0}(n)$과의 차 $y_{10}(n)$과 $y_{11}(n)$는 각각 식 (6)와 (7)을 이용하여 구해진다. 식 (8)의 $E_{ST_0}(r)$를 최소화하는 r은 적응 코드북 서브 프레임의 첫 번째 고정 코드북 인덱스 r_{10}가 되며, 식 (9)의 $E_{ST_1}(r)$를 최소화하는 r은 적응 코드북 서브 프레임의 두 번째 고정 코드북 인덱스 r_{11}이 된다. 식 (10)의 $x_r(n)$은 고정 코드북 $f_r(n)$과 $g(n)$과의 컨벌루션에 의해 구해진다. FCD(=25 [샘플])는 고정 코드북 디멘젼이고, FCS(=2^6=64 [샘플])는 고정 코드북 크기이다. 첫 번째 고정 코드북 이득 b_{10}과 두 번째 고정 코드북 이득 b_{11}는 각각 식 (11)과 식 (12)를 이용하여 구해진다.

$$y_{10}(n) = y_0(n)-b_0 x_{r_0}(n), \quad (0 \le n < \mathrm{FCD}) \quad (6)$$
$$y_{11}(n) = y_0(n)-b_0 x_{r_0}(n), \quad (\mathrm{FCD} \le n < \mathrm{ACD}) \quad (7)$$

$$E_{ST_0}(r) = \sum_{\mathrm{FCD}} y_{10}^2(n) - \frac{\{\sum_{\mathrm{FCD}} y_{10}(n)\, x_r(n)\}^2}{\sum_{\mathrm{FCD}} x_r^2(n)}, \quad (8)$$

$$(0 \le r < \mathrm{FCS})$$

$$E_{ST_1}(r) = \sum_{\mathrm{FCD}} y_{11}^2(n) - \frac{\{\sum_{\mathrm{FCD}} y_{11}(n)\, x_r(n)\}^2}{\sum_{\mathrm{FCD}} x_r^2(n)}, \quad (9)$$

$$(0 \le r < \mathrm{FCS})$$

$$x_r(n) = f_r(n) * g(n), \quad (0 \le n < \mathrm{FCD}) \quad (10)$$

$$b_{10} = \frac{\sum_{\mathrm{FCD}} y_{10}(n)\, x_{r_{10}}(n)}{\sum_{\mathrm{FCD}} x_{r_{10}}^2(n)} \quad (11)$$

$$b_{11} = \frac{\sum_{\mathrm{FCD}} y_{11}(n)\, x_{r_{11}}(n)}{\sum_{\mathrm{FCD}} x_{r_{11}}^2(n)} \quad (12)$$

무성음 프레임의 경우, 적응 코드북 검색은 하지 않고, 하나의 LPC 프레임에 대해 4 번의 고정 코드북 검색만 수행한다. 즉, 식 (13)의 $E_{ST}(r)$를 최소화하는 r이 고정 코드북 인덱스 r_2가 되며, 식 (15)으로부터 고정 코드북 이득 b_2를 구한다. 식 (13)에서 $y_0(n)$은 필터 메모리 성분이 제거된 가중 입력 음성 신호이며, 식 (14)의 $x_r(n)$은 고정 코드북 $f_r(n)$과 $g(n)$과의 컨벌루션에 의해 구해진다. 이때, FCD는 50 [샘플]이고,

FCS는 2^5=32[샘플]이다.

$$E_{ST}(r)=\sum_{FCD} y_0^2(n) - \frac{\{\sum_{FCD} y_0(n)\,x_r(n)\}^2}{\sum_{FCD} x_r^2(n)}, \quad (0\le r<\mathrm{FCS}) \tag{13}$$

$$x_r(n) = f_r(n) * g(n), \quad (0\le n<\mathrm{FCD}) \tag{14}$$

$$b_2 = \frac{\sum_{FCD} y_0(n)\,x_{r_2}(n)}{\sum_{FCD} x_{r_2}^2(n)} \tag{15}$$

2.3 음성 부호기의 비트율과 복호기

유성음 프레임과 무성음 프레임은 여기신호를 위해 사용되는 코드북의 구조가 다르므로 비트율도 각기 다르게 된다. 표 1에서 프레임 종류에 따른 비트율을 비교하였다. 유성음 프레임의 경우 6.6Kbps의 비트율을 갖고, 무성음 프레임의 경우 2.56Kbps의 비트율을 갖는다.

부호화시 양자화된 파라미터는 복호기에서 여기신호 파라미터 b_0, r_0, b, 그리고 r과 LPC 계수로 다시 복원되며, 복원된 계수를 이용하여 합성신호를 만들어 낸다. 〈그림 3〉에 본 논문에서 구현한 복호기 (decoder or synthesizer)의 블럭도를 나타내었다.

제안된 음성 부호기의 음성 스펙트럼 양자화를 위해서는, 구해진 LPC 계수를 LSP 계수로 변환하여 양자화 하였다. 일반적으로 스칼라 양자화에 비하여 벡터 양자화의 성능이 우수하나, 벡터 양자화는 연산량이 많다는 점과 코드북을 위한 많은 메모리가 필요하다는 단점을 갖고 있다. 따라서 본 논문에서는, 실시간 구현을 고려하여 스칼라 양자화기를 사용하였다. 구체적으로 본 논문에서는, 일차 DPCM (Differential Pulse Code Modulation)과 AQBW (Adaptive Quantizer for Backward Sequence)의 원리를 접목시킨 차등 AQBW (Differential AQBW)[8]를 사용하여 LSP 계수를 양자화 하였다.

표 1. 제안된 음성 부호화기의 유/무성음에 따른 비트 할당 내용
Table 1. Bit allocations of the speech coder according to voiced/unvoiced frames

파라미터	유성음 프레임	무성음 프레임
프레임 분류 비트	1 비트	1 비트
10개의 LSP 계수	32 비트	23 비트
적응 코드북 인덱스	7*4 비트	0 비트
적응 코드북 이득	4*4 비트	0 비트
고정 코드북 인덱스	6*8 비트	5*4 비트
고정 코드북 이득	5*8 비트	5*4 비트
프레임 당 비트수	165 비트	64 비트
비트율(bits per sec.)	165*40=6600 bps	64*40=2560 bps

Ⅲ. TMS320C30 DSP 칩을 이용한 실시간 구현

일반적으로, 저 전송율 음성 부호기 알고리듬은 부호화시 많은 연산량을 요구한다. 특히 분석/합성 여기신호 분석 부분의 연산량은 전체 알고리듬에서 필요한 연산량 중 대부분을 차지한다. 따라서, 여기신호 분석에 필요한 연산량을 줄이는 것이 실시간 구현[9][10]의 핵심이라 할 수 있다.

본 논문에서는, 여기신호 분석과정에서의 연산량을 줄이기 위하여, 적응코드북 검색시 다음과 같은 연산량 감소 기법을 사용하였다. 적응 코드북 검색은 식 (3)의 $E_{LT}(r)$을 최소화하는 적응 코드북 인덱스 r_0를 구함으로써 수행되는데, 이때 ACS 번의 코드벡터의 합성 $x_{r_0}(n)$이 요구된다. 코드벡터의 합성 $x_{r_0}(n)$은 컨벌루션에 의해 구해지므로 (ACD IRS) ACS 번의 곱셈과 덧셈 연산이 필요하다. 본 논문에서는 많은 연산량이 필요한 이 부분에서의 연산량을 줄이기 위해 ACS 번의 코드벡터의 합성을 모두 컨벌루션을 하는 대신에 〈그림 4〉와 같은 방법으로 코드벡터의 합성값을 구한다.

첫 번째 코드 벡터는 일반적인 컨벌루션에 의해 합성한다. 이때 필요한 연산량은 IRS ACD 번의 곱셈과 덧

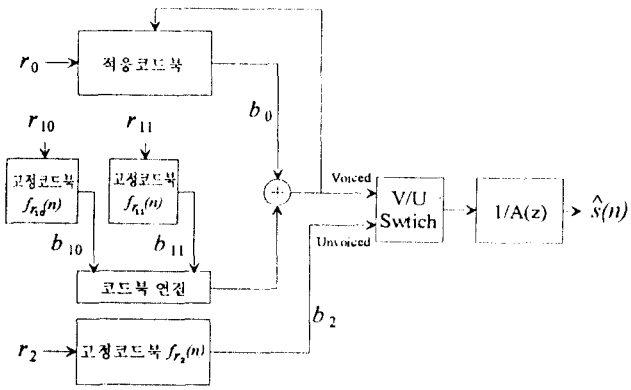
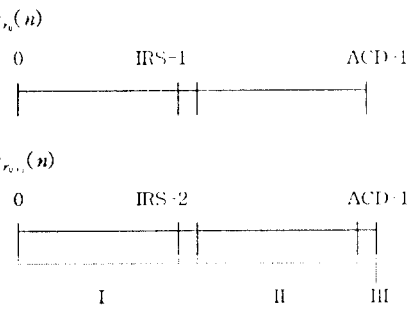


그림 3. 제안된 실시간 음성 부호기의 복호기
Figure 3. Decoder of proposed real-time speech system



Part I

$$x_{r_{0+1}}(n) = x_{r_0}(n+1) - c(-\mathrm{ACS}-\mathrm{ACD})\, g(n+1), \qquad 0 \le n < \mathrm{IRS}-1$$

Part II

$$x_{r_{0+1}}(n) = x_{r_0}(n+1), \qquad \mathrm{IRS}-1 \le n < \mathrm{ACD}-1$$

Part III

$$x_{r_{0+1}}(n) = c(n-r_{0+1}) * g(n), \qquad n = \mathrm{ACD}-1$$

그림 4. 연산량을 감소시킨 적응 코드북 벡터의 합성 방법
Figure 4. Synthesis of adaptive codebook vector reducing the computational complexity

셈이다. 두 번째 이후의 코드 벡터는 새로이 컨벌루션하지 않고, 바로 전 벡터의 합성값을 사용하는데 이때 part Ⅰ 에서는 IRS-1 번, part Ⅲ 에서는 새로운 컨벌루션 값을 구하기 위해 IRS 번의 곱셈, 덧셈의 연산이 필요하다. 따라서, 총 연산량은 IRS ACD+(2 IRS-1) ACS 번의 곱셈과 덧셈으로, 원래의 연산량 (ACD IRS) ACS 보다 크게 줄어든다.

다음은 고정 코드북 검색시 필요한 연산량을 알아본다. 기존의 CELP 음성 부호기에서는 LPC 프레임을 4 개의 고정 코드북 서브 프레임으로 나눈다. 이때 FCS가 2^{10}이라 하면, 한 프레임에서 고정 코드북으로 만들 수 있는 여기신호의 경우의 수는 $2^{10\cdot4}$이고, 각 프레임당 $(50\cdot\mathrm{IRS})\cdot2^{10\cdot4}$의 곱셈과 덧셈 연산이 필요하다. 본 연구에서는 한 프레임을 8 개의 고정 코드북 서브 프레임으로 나누었는데, $2^{10\cdot4}$ 개의 경우의 수를 만들기 위해서는 FCS가 25이어야 한다. 이때 요구되는 연산량은 프레임당 $(25\cdot\mathrm{IRS})\cdot2^{5\cdot8}$의 곱셈과 덧셈이다. 한 프레임 내에서 각 고정 코드북 서브 프레임은 같은 음성 스펙트럼을 갖고 있다. 따라서, 각 서브 프레임에서의 고정 코드북 벡터의 합성값은 모두 같다. 이는 첫 번째 프레임에서 구해진 코드벡터의 합성값을 메모리에 저장해 두고 나머지 7 프레임은 다시 코드벡터를 합성하지 않고 메모리에 저장된 합성 값을 읽어서 고정 코드북 분석을 수행할 수 있음을 의미한다. 이러한 방법을 사용할 때 요구되는 연산량은 프레임당 $(25\cdot\mathrm{IRS})\cdot2^5$의 곱셈과 덧셈에 불과하다. 따라서, CELP 음성 부호기의 고정 코드북 검색시 필요한 연산량과 비교 볼 때 $1/2^8$ 배로 연산량이 감소하였다.

이와같이 연산량을 감소시킨 알고리듬은 TI 사 (Texas Instrument Inc.)의 TMS320C30 개발 모듈 (evaluation module)[11]을 사용하여 실시간 구현되었다. 이 개발 모듈의 사양은 다음과 같다.

* TMS320C30 33-MFLOP(16.7 MIPS) 부동소수점 연산 DSP 칩 내장
* 16K word(64 KB)의 zero waite-state SRAM
* TLC32044 칩을 이용한 D/A, A/D 컨버터 내장
* 16-bit 양방향 PC host 통신 포트

개발 모듈에 내장된 TMS320C30 칩[3]은 범용 부동

소수점 연산 DSP 칩으로 최근에 가장 널리 사용되는 DSP 칩 중 하나이다. TMS320C30 개발 모듈은 C-compiler와 C-source optimizer[12][13]를 제공하는데 본 논문에서 제안된 모든 알고리듬은 C-언어로 구현되었으며 C-source optimizer를 사용하여 최적화 시켰다. 실시간 구현된 음성 부호기의 실시간 실험 블럭도를 〈그림 5〉에 나타내었다. 마이크를 통해 입력된 음성신호는 개발 모듈의 A/D 변환기를 통하여 8KHz의 주파수를 가지고 디지틀 신호로 바뀌어지며, TMS320C30 DSP CPU를 이용한 실시간 부호화 및 복호화 시뮬레이션을 거친 후, 합성 재생되어 D/A 변환기를 통하여 스피커로 출력된다.

Ⅳ. 실험 및 고찰

제안된 음성 부호기의 성능(음질)을 평가하기 위해 SNR (Signal-to-Noise Ratio) 측정과 MOS (Mean Opinion Score) 테스트를 실시하였다. SNR은 식 (16)에 주어져 있고, MOS 테스트에서 각 점수의 의미는 표 2와 같다.

$$SNR = 10\log_{10}\left\{\sum_{N} s^2(n) - \sum_{N}[s(n) - \hat{s}(n)]^2\right\} \text{ [dB]} \quad (16)$$

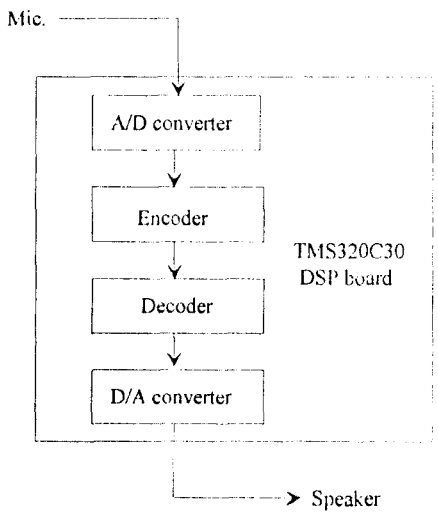


그림 5. 음성 부호기의 실시간 구현 블럭도
Figure 5. Block diagram for real-time implementation of the speech coder

여기서, N은 프레임 크기이고, $s(n)$은 입력 음성신호, $\hat{s}(n)$은 재생 합성된 음성신호이다.

성능 평가는 표준 CELP 알고리즘으로 구현된 부호기(이하, 표준 부호기)와 본 논문에서 제안된 부호기에 대해서 각각 실시하여 비교 하였다. 표준 부호기의 비트 할당은 표 3에 주어져 있다. 표준 부호기에서 사용된 프레임 크기는 160 샘플(20 ms)이고, 고정코드북은 평균이 0, 편차가 1 이며, 디멘젼이 512인 가우스 코드북을 사용하였다.

10개의 한글 문장과 10개의 영어 문장에 대하여, 표준 부호기와 제안된 부호기에서 측정한 SNR 값을 각각 표 4와 표 5에 나타내었다. 10개의 한글 문장에 대하여 표준 부호기는 평균 12.13dB, 제안된 부호기는 평균 11.13dB이고, 10개의 영어 문장에 대하여 표준 부호기는 평균 10.28dB, 제안된 부호기는 평균 10.42dB이다. 한글 문장에서는 제안된 부호기가 표준 부호기보다 1dB 낮고, 영어 문장에서는 0.14dB 높은 것으로 나타났다. MOS 테스트는 SNR 측정시 사용된 10개의 한글 문장과 10개의 영어 문장에 대하여 10명에게 얻은 MOS 테스트 점수를 평균하였으며, 그 결과는 각각 표 6과 표 7에 나타내었다. 한글 문장에 대하여 표준 부호기는 평균 3.36, 제안된 부호기는 평균 3.25이고, 영어 문장에 대하여 표준 부호기는 평균 3.61, 제안된 부호

표 2. MOS 테스트에서 각 점수의 의미.
Table 2. Meaning of each score in MOS test.

MOS (Mean Opinion Score)	
5	Excellent(Original Speech)
4	Good
3	Fair
2	Poor
1	Unacceptable

표 3. 표준 CELP 알고리즘으로 구현된 부호기의 비트 할당
Table 3. Bit allocations of the speech coder implemented with the standard CELP algorithm

파라미터	비트 할당
10개의 LSP 계수	32 비트
적응 코드북 인덱스	7*4 비트
적응 코드북 이득	4*4 비트
고정 코드북 인덱스	9*4 비트
고정 코드북 이득	5*8 비트
프레임 당 비트수	152 비트
비트율(bits per sec.)	152*50=7600 bps

기는 평균 3.36이다. 실험 결과를 살펴보면, 표준 부호기의 비트율은 7.6Kbps이고 제안된 부호기는 한글 문장에서 평균 5.6Kbps, 영어 문장에서 평균 5.43Kbps로, 제안된 부호기의 비트율이 표준 부호기 보다 낮으며, 제안된 부호기는 음질보다는 실시간 처리에 촛점을 두어 개발된 것을 고려할때, 제안된 부호기 성능은 매우 우수한 것으로 평가된다. 또한 무성음의 전체 신호에 대한 비율은 각각 23.23%와 28.06%로 나타났다. 대화체의 경우를 고려 할 경우, 실제의 평균 비트율은 더욱 낮아 질 것이다.

〈그림 6〉과 〈그림 7〉에서는 제안된 부호기의 유/무성음 판별에 및 그때의 여기신호를 보여주고 있다. 〈그림 6〉에서 윗 그림은 100ms 길이의 입력 음성신호를 보여주고 있으며, 이때 첫번째 프레임과 두번째 프레임은 무성음으로 판별되었고, 세번째 프레임과 네번째 프레임은 유성음으로 판별되었다. 세번째 프레임은 실제로는 전이 부분이지만, 본 부호기 알고리즘은 전이 부분도 유성음으로 판별하도록 설계되어있다. 아래 그림은 이때의 여기신호를 보여주고 있다. 〈그림 7〉은 첫번째 프레임, 두번째 프레임과 세번째 프레임은 유성음으로 판별하고,

표 4. 한글 문장에 대한 SNR 및 비트율
Table 4. SNRs and bit rates for Korean sentences

한 글 문 장	표준 부호기 (SNR [dB])	제안된 부호기 (SNR [dB])	무성음 프레임 비율[%]	평균 비트율 [Kbps]
아이들은 타고난 인격체입니다.	12.05	9.81	15.85	5.96
미는 피부 한 껍질 차입니다.	14.00	9.96	35.35	5.17
결혼을 인륜지 대사로 삼고 있는 한국	11.24	11.89	16.82	5.92
금나와라 뚝딱, 도깨비 방망이	12.11	11.07	22.35	5.70
알아서 남주나, 안 그래?	11.61	11.54	39.08	5.02
참으로 딱한 일이로다.11.99	14.55	21.21	5.74	
개구리는 파충류입니다.	14.35	12.82	17.54	5.89
이번 겨울은 예년과 달리 포근합니다.	12.31	11.53	8.60	6.25
오르지 못할 나무는 쳐다보지도 말랬다.	11.29	9.28	23.65	5.64
오스트리아 왕이 거만해서 기분 나쁘다 전쟁이다.	10.38	8.93	31.94	5.31
평 균	12.13	11.13	23.24	5.66

표 5. 영어 문장에 대한 SNR 및 비트율
Table 5. SNRs and bit rates for English sentences

한 글 문 장	표준 부호기 (SNR [dB])	제안된 부호기 (SNR [dB])	무성음 프레임 비율[%]	평 균 비트율 [Kbps]
The birch canoe slid on the smooth planks.	11.25	11.45	38.59	5.04
Glue the sheet to the dark blue background.	8.76	9.75	20.17	5.79
The pipe began to rust while new.	8.74	9.45	24.44	5.61
Oak is strong and also gives shade.	10.02	9.290	23.33	5.66
Thieves who rob friends deserve jail.	9.89	9.23	21.43	5.73
The boy was there when the sun rose.	11.69	9.22	31.25	5.34
A rod is used to catch pink salmon.	10.95	10.08	37.78	5.07
A pot of tea helps to pass the evening.	9.88	9.98	33.65	5.24
The soft cushion broke the man's fall.	11.29	10.25	25.71	5.16
Smoky fires lack flame and heat.	10.28	10.42	28.06	5.43
평 균	10.28	10.42	28.06	5.43

표 6. 한글 문장에 대한 MOS 테스트
Table 6. MOS test for Korean sentences

한 글 문 장	표준 부호기 (MOS)	제안된 부호기 (MOS)
아이들은 타고난 인격체입니다.	2.6	2.4
미는 피부 한 겹질 차입니다.	3.7	3.6
결혼을 인륜지 대사로 삼고 있는 한국	2.6	2.9
금나와라 뚝딱, 도깨비 방망이	3.1	2.7
알아서 남주나, 안 그래?	2.6	2.9
참으로 딱한 일이로다.	4.1	3.0
개구리는 파충류입니다.	4.1	4.0
이번 겨울은 예년과 달리 포근합니다.	3.4	3.7
오르지 못할 나무는 처다보지도 말랬다.	4.0	3.9
오스트리아 왕이 거만해서 기분 나쁘다. 전쟁이다.	3.4	3.4
평 균	3.36	3.25

표 7. 영어 문장에 대한 MOS 테스트
Table 7. MOS test for English sentences

영 어 문 장	표준 부호기 (MOS)	제안된 부호기 (MOS)
The birch canoe slid on the smooth planks.	3.7	3.6
Glue the sheet to the dark blue background.	3.7	3.3
The pipe began to rust while new.	2.6	3.0
Oak is strong and also gives shade.	4.3	4.1
Thieves who rob friends deserve jail.	4.0	3.7
The boy was there when the sun rose.	4.0	3.4
A rod is used to catch pink salmon.	3.4	3.4
A pot of tea helps to pass the evening.	4.1	3.3
The soft cushion broke the man's fall.	3.6	2.9
Smoky fires lack flame and heat.	2.7	2.9
평 균	3.61	3.36

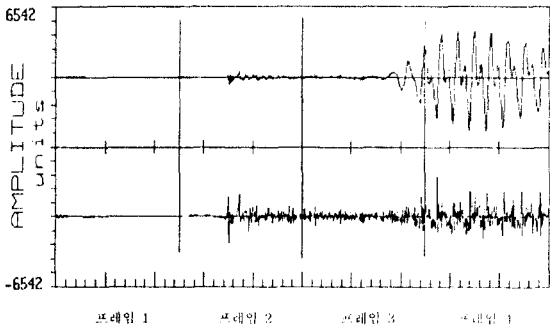


그림 6. 100 ms 길이의 음성신호의 유/무성음 판별 예 (Ⅰ)
(위 : 입력 음성신호, 아래 : 여기신호)
Figure 6. Voiced/unvoiced decision example of 100 ms long speech signals (Ⅰ)

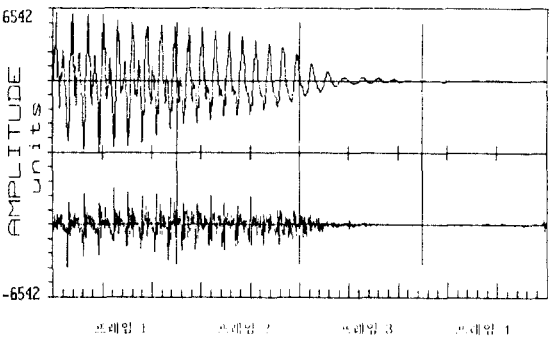


그림 7. 100 ms 길이의 음성신호의 유/무성음 판별 예 (Ⅱ)
(위 : 입력 음성신호, 아래 : 여기신호)
Figure 7. Voiced/unvoiced decision example of 100 ms long speech signals (Ⅱ)

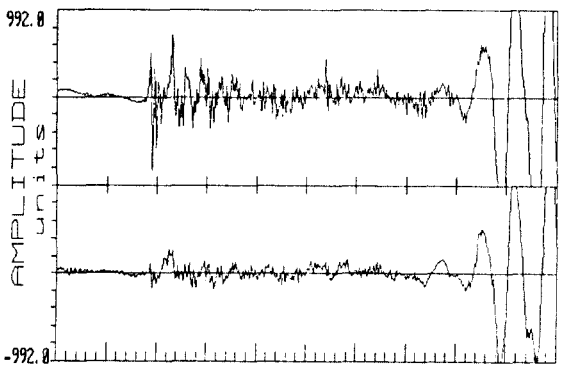
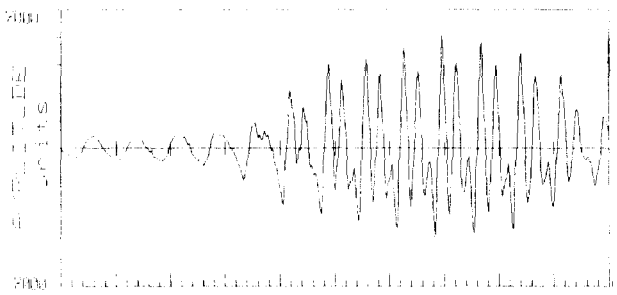


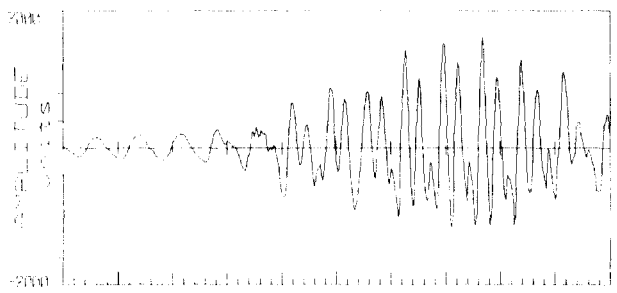
그림 8. 50 ms 길이의 무성음 음성신호 파형 비교
(위 : 원 음성신호, 아래 : 재생된 음성신호)
Figure 8. Waveform comparison of 50 ms long unvoiced speech signals (Ⅰ)

네번째 프레임은 무성음으로 판별한 경우를 보여주고 있다.

〈그림 8〉에서 부터 〈그림 12〉에서는 입력 음성신호와 재생 합성된 음성신호의 파형을 비교하고 있다. 〈그림 8〉에서는 50ms 길이의 무성음 음성 신호를, 〈그림 9〉와 〈그림 10〉에서는 75ms 길이의 유성음 음성신호들, 〈그림 11〉에서는 150ms 길이의 유성음 음성신호, 그리고 〈그림 12〉에서는 2.25sec 길이의 음성신호 파형을 비교하고 있다. 파형들을 살펴보며, 재생된 음성신호는 입력 음성신호를 충실히 재생해내고 있음을 알 수 있다.

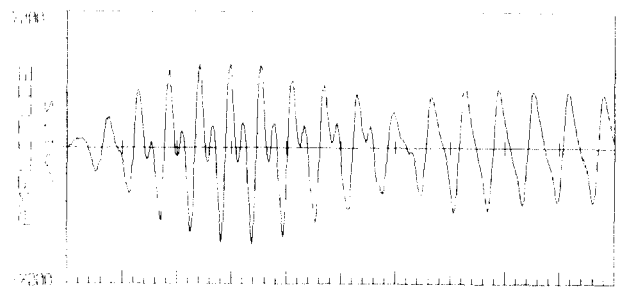


(가) 원 음성신호

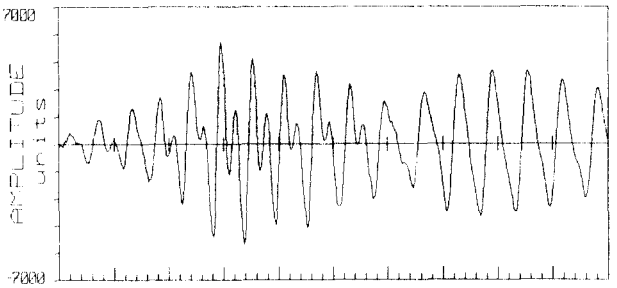


(나) 재생된 음성신호

그림 9. 75 ms 길이의 유성음 음성신호 파형 비교 (I)
Figure 9. Waveform comparison of 75 ms long voiced speech signals (I)

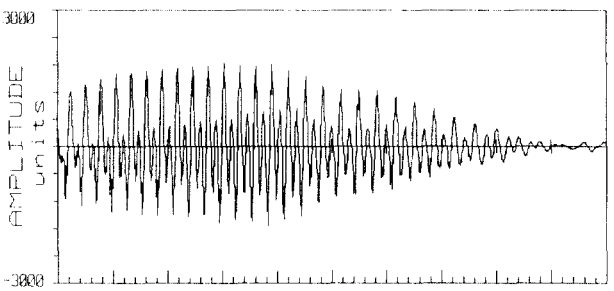


(가) 원 음성신호

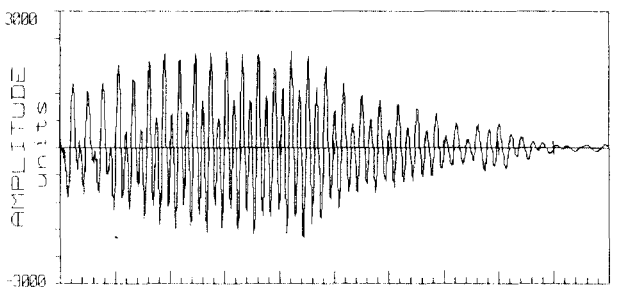


(나) 재생된 음성신호

그림 10. 75 ms 길이의 유성음 음성신호 파형 비교 (Ⅱ)
Figure 10. Waveform comparison of 75 ms long voiced speech signals (Ⅱ)

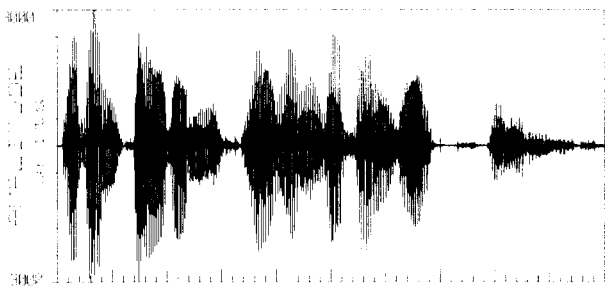


(가) 원 음성신호

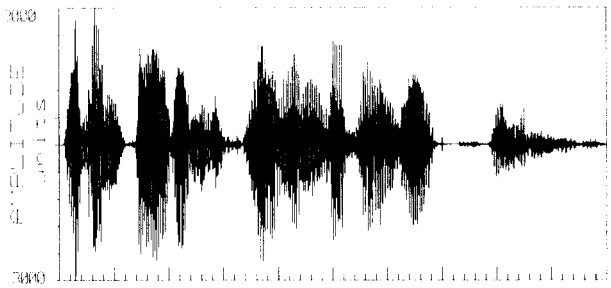


(나) 재생된 음성신호

그림 11. 150 ms 길이의 유성음 음성신호 파형 비교
Figure 11. Waveform comparison of 150 ms long voiced speech signals



(가) 원 음성신호



(나) 재생된 음성신호

그림 12. 2.25 sec. 길이의 음성신호 파형 비교
Figure 12. Waveform comparison of 2.25 sec. long speech signals

Ⅴ. 결 론

효율적인 음성 부호기는 많은 응용 분야를 갖고 있으며, 특히 이동 통신 분야에서는 음성 부호기에 대한 연구가 활발히 진행되고 있다. 본 논문에서는 음성 신호 중 유성음과 변이음에서는 많은 비트를 할당 (6.6 Kbps)하고, 무성음과 무음에서는 적은 비트를 할당 (2.56 Kbps)하는 효율적이면서도 연산량이 적은 음성 부호기 알고리듬을 개발 하였다. 또한, 새로운 여기신호 분석 방법을 사용하여, 실시간 구현에 가장 큰 관건이 되는 연산량을 크게 줄였으며, 이를 바탕으로 한개의 TMS320C30 DSP 칩 (33MHz, 16.7Mips)와 60KB의 적은 메모리를 이용하여 부호기 및 복호기를 실시간 구현하였다.

본 논문에서 구현한 실시간 음성 부호기는 한글 문장들과 영어 문장들에 대하여 각각 5.66Kbps와 5.43Kbps의 평균 전송율을 나타내었다. 개발된 음성 부호기는 음질면에서 한글에 대해 11.13dB(SNR)와 3.25(MOS), 영어에 대해 10.42dBd와 3.36(MOS)의 좋은 성능을 보였다. 또한 MOS 테스트를 통하여서는, 한국어 문장들에 대하여 3.25, 영어 문장들에 대하여 3.36의 좋은 점수를 얻었다.

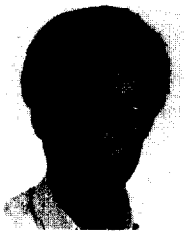
참고문헌

1. M. R. Schroeder and B. S. Atal, "Code-Excited Linear Prediction (CELP): High-quality speech at very low bit rates," *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pp.937-940, 1985.
2. *TMS320C30 user's guide*, Texas Instruments Inc., 1992.

3. Peter Lupini, Hisham Hassanein, and Vladimir Cuperman, "A 2.4 Kbps CELP speech codec with class-dependent structure," *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pp.143-146, 1993.
4. Chih-chung Kuo, Fu-Rong Jean, and Hsiao-Chuan Wang, "Speech classification embedded in adaptive codebook search for CELP coding," *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pp.147-150, 1993.
5. B. S. Atal and J. R. Remde, "A new model of LPC excitation for producing natural-sounding speech at low bit rates," *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pp.614-617, 1982.
6. B. S. Atal, "High quality speech at low bit rates: Multi-pulse and stochastically excited linear predictive codes", *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pp.1681-1684, 1986.
7. Jae H. Chung, *A new homomorphic vocoder framework using analysis-by-synthesis excitation analysis, Ph.D Thesis*, Geogia Institute of Technology, pp.19-26, 1991.
8. 정재호, 우홍체, 정대권, "이동 통신용 음성코딩 시스템 개발에 관한 연구", 최종 보고서, 한국전자통신 연구소, pp.43-63, 1994.
9. Juin-Hwey Chen and Allen Gersho, "Real-time vector APC speech coding at 4800 bps with adaptive postfiltering," *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pp.2185-2188, 1987.
10. Grant Davidson, Mei Yong, and Allen Gersho, "Real-time vector excitation coding of speech at 4800 bps," *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pp.2189-2192, 1987.
11. *Digital signal processing applications with the TMS320C30 Evaluation Module*, Texas Instruments Inc., 1991.
12. *TMS320C30 floating-point DSP Optimizing C compiler*, Texas Instruments Inc., 1991.
13. *TMS320C30 C* source debugger, Texas Instruments Inc., 1991.

辛 哲 善(Chul Sun Shin) 학생회원

1969년 7월 23일생
1993년 : 인하대학교 공과대학 전자공학과(학사)
1995년 : 인하대학교 공과대학 전자공학과(석사)
1995년-현재 : LG 전자 미디어통신 연구소 연구원
※주관심 분야 : 음성코딩, 디지틀 이동통신 및 PCS



鄭 在 皓(Jae Ho Chung) 정회원

1982년 : 美國 Univ. of Maryland(학사)
1984년 : 美國 Univ. of Maryland(석사)
1990년 : 美國 Georgia Institute of Technology(박사)
1984년-1985년 : 美國 Naval Surface Warfare Center, Electronic Engr.
1991년-1992년 : 美國 AT&T Bell Laboratories, 연구원 (MTS)
1992년-현재 : 인하대학교 공과대학 전자공학과, 조교수