

메모리 비트 에러에 강한 음성 부호화기의 구현

(Implementation of a Memory Bit-error Robust Coder for Digital Speech Storage)

成元鎔*, 金鎮栗**, 李滉洙***

(Wonyong Sung, Jin Yul Kim, and Hwang Soo Lee)

要 約

음성을 디지털화하여 메모리에 저장시, 일부 불량인 메모리를 이용할 수 있는 부호화 방법을 연구하였다. 이 방법에서는 음성데이터를 메모리에 기록한 후 즉시 다시 읽어서 에러가 발견되면 다음번 음성 샘플의 기록시 이를 보상한다. 따라서 ADM 이나 ADPCM과 같은 예측 부호화 방법에서, 메모리의 비트 에러에 의한 영향이 누적되지 않고 다음번 샘플에서 즉시 상쇄된다. 종래의 메모리 에러 정정 방식과 비교할때, 이 방법은 여분의 메모리를 필요로 하지 않는 장점이 있다. 이 방법을 ADM 부호화기에 적용시, 메모리의 비트 에러율이 10^2 에 이를 때까지 신호대 잡음비의 저하는 1dB 미만임을 시뮬레이션으로 확인하였다. 또한 본 방법을 bit-serial 산술 연산을 이용한 디지털 회로로 구현하여 만족스런 동작을 확인하였다.

Abstract

A speech coding method which can utilize bit-failed semiconductor memories is studied. This method detects memory bit-errors at the recording stage by reading back the data after writing it to memory. If an error is detected, the value of the predictor is adjusted to compensate the distortion due to the error. In comparison to memory bit-error correction methods, this approach not only requires a much simpler hardware but also needs no extra data for error compensation. The performance of the coder does not degrade much until the bit-error rate reaches to 10^2 . A hardware prototype using a bit-serial logic has been developed to examine the performance of the system, and a satisfactory result is obtained.

*正會員, *서울대학교 半導體共同研究所 및
制御計測工學科

(ISRC and Dept. of Control and Instrumentation,
Seoul Nat'l Univ.)

**正會員, 韓國科學技術院 서울 分院 情報 및 通信工學科
(Dept. of Infor. and Commu. Eng., Korea
Advanced Institute of Science and Technology)

接受日字 : 1992年 7月 23日

I. 서 론

반도체 메모리를 이용하는 디지털 녹음·재생 시스템은 기존의 기계식 녹음기보다 편리하고 다양한 기능을 쉽게 제공할 수 있으므로 자동응답전화기, 음성 우편, multi-media 등에 널리 응용되고 있다. 그러나, 장시간의 음성을 저장하고 재생하려면 대용량의 메모리 소자가 필요한데, 아직 반도체 메모리의 비트(bit)당 가격이 비

싼 것이 실용화의 커다란 장애요인이 되고 있다. 따라서 메모리 소자내의 cell이나 row, column의 일부에 결함이 있어서 상용가치가 거의 없는 메모리를 사용하면 시스템의 경제적인 구현이 가능하다. 메모리 소자의 비트 에러(bit-error)를 보상하는 직접적인 방법은 에러 정정 부호를 채용하는 것이지만, 이 방법은 복잡한 하드웨어를 필요로 하고 또 메모리의 에러 정정을 위한 별도의 정보를 더 기록해 놓아야하므로 필요한 메모리 소자의 용량이 늘어나는 결점을 갖는다.

본 논문에서는 이 문제를 해결하기 위하여서 새로운 비트 에러 보상 (compensation) 음성부호화 방식을 연구하였다. 디지털 녹음·재생 시스템은 디지털 음성 통신과는 달리 전송채널에 해당하는 메모리 소자가 시스템 내부에 포함되므로 비트 에러가 발생하면 기록후 바로 감지할 수 있다. 따라서 메모리의 비트 에러를 직접 정정하는 대신, 부호화기의 출력값을 녹음용 메모리 소자에 썼다가 즉시 다시 읽어서, 메모리 소자의 비트 에러가 발생했을 경우(즉, 메모리에 기록한 데이터와 읽어낸 값이 서로 다른 경우), 이를 다음번 부호화 단계에서 보상하면 찌그러짐을 크게 줄일 수 있다. 이 방법은 메모리 소자의 비트 에러를 보상하는데 별도의 정보를 기록해 둘 필요가 없고 또 복잡한 에러 정정 회로도 필요로 하지 않는다. 또한, 이 방식은 적응 델타 변조방식(ADM), 적응 차등 PCM(ADPCM) 방식등의 차등 예측 부호화 방식들에 두루 적용될 수 있다. 본 논문에서는 CVSD(Continuously Variable Slope Delta modulation) 방식의 ADM(Adaptive Delta Modulation)에 이 새로운 방식을 적용하고 컴퓨터 모의실험(simulation)과 평가보드의 제작을 통하여 그 성능을 보인다. 특히, 평가보드의 제작시 bit-serial logic을 이용한 디지털 방식을 이용하였기 때문에, 추후에 ASIC(Application Specific Integrated Circuit)으로 쉽게 제작이 가능하다.

II. 메모리의 비트 에러에 강한 음성 부호화 방식

본 논문에서 제안하는 비트 에러 보상 방법의 원리를 편의상 LDM(Linear Delta Modulation)^[1]에 적용한 예를 이용하여 설명한다. 그림 1은 기존의 방법으로 LDM을 구현하는 구성도이다. 이 LDM에서는 음성신호를 부호화하여 메모리에 저장하였다가 재생시에 저장된 데이터를 차레로 복호기(decoder)로 보내어 재생파형을 얻는다. 그림 2에 원래의 음성신호 파형과 저역필터(LPF)를 통과하기 전의 재생파형을 보였다. 메모리 소자에 비트 에러가 없는 경우, 그림 2에서 보인 바와 같이 LDM의 재생출력은 원래의 음성 신호를 충실히 좇아간다. 그러나, 메모리 소자가 비트 에러를 갖는 경우, LDM

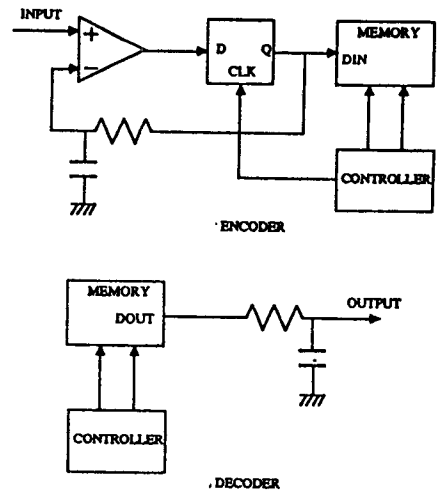


그림 1. 기존 LDM의 구성도
Fig. 1. A linear delta modulator

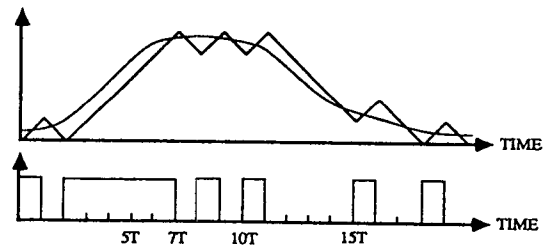


그림 2. LDM의 재생파형(정상인 메모리 사용시).
Fig. 2. Reproduced waveform of the LDM without a bit error.

의 재생파형은 원래의 신호파형을 벗어나 찌그러짐이 발생하며, 이 찌그러짐은 캐패시터에 저장되어 차후에 계속적으로 나쁜 영향을 미친다. 그림 3은 $t=7T$ 에서 메모리 소자의 비트 에러로 인하여 데이터가 0에서 1로 바뀐 경우이다. 이때 그림 3에 보이는 바와 같이 재생파형은 $t=7T$ 에서부터 원래의 파형과 크게 달라진다.

이제 본 논문에서 제안하는 메모리 소자 비트 에러 보상 방법이 어떻게 작용하는지 설명한다. 그림 4는 제안된 새로운 부호화법을 LDM에 적용한 구성도이다. 그림 1에서는 플립플롭의 출력(Q)이 예측기의 입력(저항)으로 연결되었으나, 그림 4에서는 메모리의 출력(DOUT)이 예측기로 연결되었다. 즉, 방법은 메모리 소자를 부호화기의 피드백 루프 내에 포함하여 에러 보상이 가능하게 하였다. 이때 제어기는 양자화기의 출력을 일단 메모리에 썼다가 즉시 읽어 내도록 제어신호를 발생시킨다. 그림 5는 $t=7T$ 에서 비트 에러로 인하여 데이터가 0에서 1로

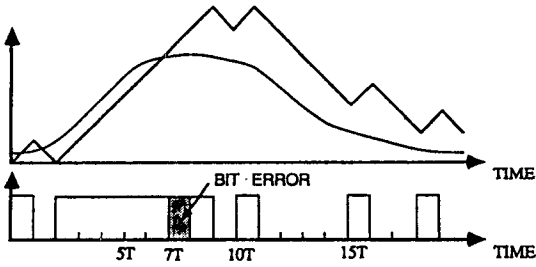


그림 3. 기존 LDM의 재생파형(비트 에러가 있는 메모리 사용시)

Fig. 3. Reproduced waveform of the LDM with a bit error.

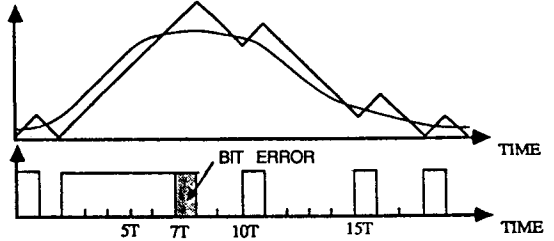
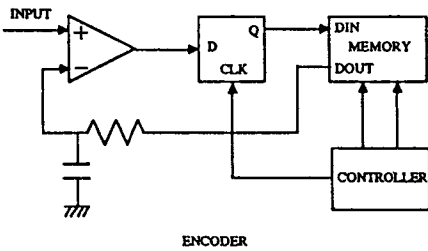
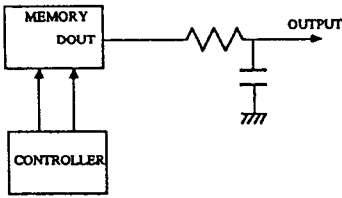


그림 5. 제안하는 LDM의 재생파형(비트 에러가 있는 메모리 사용시)

Fig. 5. Reproduced waveform of the proposed LDM with a bit error.



ENCODER



DECODER

그림 4. 제안하는 LDM의 구성도

Fig. 4. The LDM employing the proposed error compensation method.

바뀌는 메모리를 사용하는 경우의 재생파형이다. 이 경우 $t=7T$ 에서 생긴 에러의 영향은 그 다음 부호화 단계 $t=8T$ 에서 보상되어 재생파형의 찌그러짐이 그림 3에 비하여 월등히 줄어든다. 즉, 그림 3에서는 에러의 영향이 누적되어 계속 전파되었으나, 그림 5에서는 다음번 샘플에서 상쇄되어 없어졌다. 그림 5에서 $t=8T$ 의 데이터가 바로 앞 부호화 단계에서 생긴 비트 에러를 보상하기 위하여 1에서 0으로 바뀌어졌음을 알 수 있다. LDM에 대해 설명한 이 방법은 비슷한 원리로 적응델타 변조방식(ADM), 적응 차등 PCM(ADPCM) 등에 적용될 수 있다.

지금까지 단일 에러에 대한 보상에 대해 설명하였는데, 실제의 메모리 소자에서 비트 에러는 어떤 부분에서 연속적으로 발생하는 경우(예를 들어 메모리 소자의 column decoder에서 고장이 생긴 경우)가 많다. 이 경우 메모리 번지제어기(address controller)는 메모리의 비트 에러가 전체 메모리 공간에 균등하게 분포되도록 불규칙한 순서로 메모리 번지 신호를 발생시키도록 해야 한다.

III. 적응 델타 변조(ADM)에의 적용

이 장에서는 본 논문에서 제안하는 음성부호화 방법을 ADM에 적용한 결과를 설명한다.

1. CVSD 방식의 ADM

ADM은 양자화기의 step size를 입력신호의 변화에 따라 적응시킴으로써 LDM에 비해 넓은 dynamic range를 갖는다. ADM은 입력신호의 변화에 추종하여 1-비트 양자화기의 step size를 적응시키는 방식의 차이에 따라 몇가지 시스템으로 분류된다.^{[2][4]} 본 논문에서는 이중 널리 사용되는 CVSD 방식을 사용한다.

그림 6에 CVSD codec을 보였다. 이 시스템에서 양자화기의 step size Δ_n 은 출력 시퀀스 $\{b_n\}$ 에 따라 증감된다. 3개의 연속적인 비트들이 모두 같은 경우, 적응 논리(adaptation logic) 회로는 그 출력 신호 K_n 을 1로 만들어 Δ_n 을 최대값인 Δ_{max} 의 방향으로 증가시킨다. 이러한 경우는 step size가 너무 작아 입력 파형의 급한 기울기를 따르지 못할 때 일어난다. 반면, 입력 신호가 거의 일정한 크기이거나 천천히 변화하는 경우, 3개의 연속되는 비트들이 모두 같아지는 빈도는 적어진다. 이때 적응 논리 회로는 K_n 을 0으로 만들어 Δ_n 을 최소값인 Δ_{min} 의 방향으로 감소시킨다. 즉 syllabic 필터의 step size는 다음과 같이 적용된다.

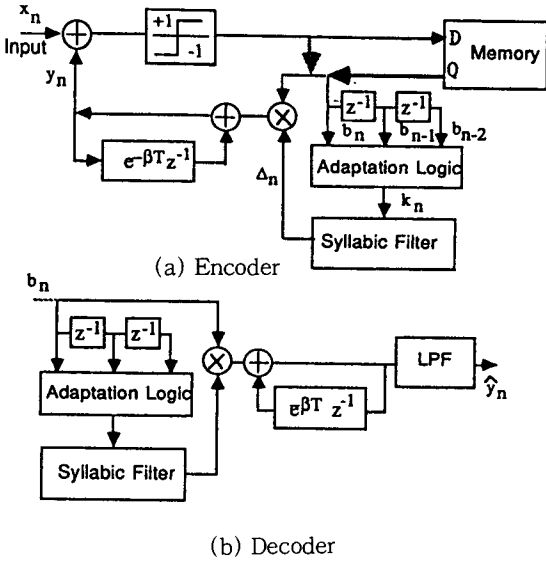


그림 6. CVSD 방식 ADM의 구성도
Fig. 6. Block diagram of the CVSD ADM.

을 메모리 소자에 일단 썼다가 즉시 읽어서 그 값을 피드백 시키면 된다.

2. 컴퓨터 모의실험

이제 본 논문에서 제안하는 음성부호화 방법을 CVSD에 적용하는 경우를 컴퓨터로 모의실험하기 위한 구성을 그림 7에 보였다. 그림에서 LPF 1은 입력 음성신호의 고주파 성분을 제거하기 위한 것이고 LPF 2는 ADM 복호기의 내부에 포함되어 있는 LPF 때문에 생기는 시간 지연(delay)을 보상해주어 찌그러짐의 양을 바르게 계산하기 위한 것이다. 모의실험은 기존의 CVSD 방식을 사용하는 경우와 본 논문에서 제안하는 방식에 대하여 각각 수행되었고, 이때 메모리 소자의 비트 에러율에 따른 음성부호화기의 성능을 비교하였다. 모의실험에 사용한 값들은 참고문헌[2]의 결과에 근거하여 선택되었다. 시스템의 샘플링 주파수가 16 KHz일때 prediction 필터와 syllabic 필터의 시정수는 각각 1.0 msec와 3.8 msec가 최적임이 위의 참고문헌에 나타나 있다. LPF의 차단 주파수는 2.5 KHz가 선택되었다. 또, step size의 비 $\Delta_{max}/\Delta_{min}$ 는 dB가 사용되었다. 시스템의 성능을 가늠하는 척도로 SQNR(signal-to-quantization-noise ratio)을 사용하였다. SQNR은 다음과 같이 정의된다[2].

$$\Delta_n = e^{-\alpha T} \Delta_{n-1} + G(1 - e^{-\alpha T}),$$

$$\text{if } b_n = b_{n-1} = b_{n-2} \quad (1a)$$

$$\Delta_n = e^{-\alpha T} \Delta_{n-1}$$

$$\text{otherwise,} \quad (1b)$$

여기서 T는 sampling 주기, G는 step size를 증가시키는 정도를 나타내는 상수이다. Step size의 적용은 시정수 $1/\alpha$ (syllabic 필터의 시정수)에 의하여 제어되는데 이 값은 보통 수 msec 정도이다. 한편 입력 신호의 예측은 시정수 $1/\beta$ (prediction 필터의 시정수)을 갖는 손실이 있는 적분기 (leaky integrator)에 의하여 수행된다. Prediction 필터는 방금 부호화된 b_n 의 부호값 (sign)과 step size Δ_n 을 곱한 값이 더해져서 새로운 값으로 적용된다. 즉,

$$y_n = e^{\beta T} y_{n-1} + \Delta_n,$$

$$\text{if } b_n = +1, \quad (2a)$$

$$y_n = e^{\beta T} y_{n-1} - \Delta_n,$$

$$\text{if } b_n = -1, \quad (2b)$$

이다.
제안하는 음성 부호화 방식을 CVSD ADM에 적용하기 위해서는 그림 4에서 설명했던 것처럼 양자화기의 출력 값을 그대로 피드백 시키는 대신, 메모리 소자를 부호화기의 피드백 루프 내에 포함하도록 양자화기의 출력 값

$$SQNR = \frac{\sum_{n=1}^N \bar{x}_n^2}{\sum_{n=1}^N (x_n - \hat{y}_n)^2} \quad (3)$$

여기서 x_n 대신 지연 보상된 음성 입력인 \hat{x}_n 을 사용해야 한다. \hat{y}_n 은 decoder의 출력신호를 나타내고 N은 입력으로 사용된 음성 신호의 샘플 수이다. 한편, 모의실험에서 메모리 소자의 비트 에러 분포는 전체 메모리 공간에 균등하다고 가정하여 비트 에러가 난 메모리 cell의 번지를 산수(random number) 발생기로 얻었다.

이렇게하여 메모리 소자의 비트 에러율에 따른 SQNR을 각 시스템마다 측정하였다. 모의실험에서는 먼저 입력 음성신호의 크기를 바꾸어 가며 SQNR이 최대가 되는 지점을 구한 다음, 이 지점을 기준으로 메모리 소자의 비트 에러에 따른 SQNR 변화를 측정하였다. 그림 8에 모의실험의 결과를 보였다. SQNR 값들은 기존의 방식(Conv.로 표시)과 제안된 방식(Prop.로 표시) 각각에 대하여, 식 (3)에 의해 구해졌다. (실선부분) 또 \hat{x}_n 과 \hat{y}_n 에서 300Hz 이하의 주파수 성분을 제거한 후에 같은 방법으로 구해졌다. (점선 부분) 이 결과로부터 본 논문에서 제안된 비트 에러 보상 부호화법이 기존의 방법에 비하여 메모리 소자의 비트 에러에 훨씬 더 강함을 알 수 있다. 즉 제안된 방식에서는 메모리 소자의 비트 에러율이

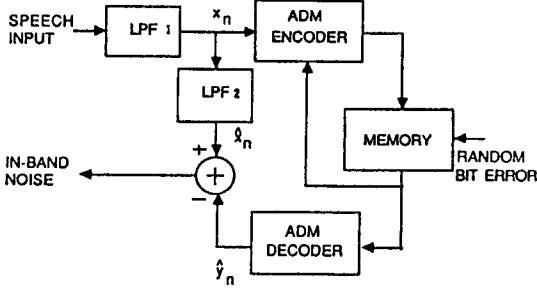


그림 7. 모의 실험을 위한 구성도
Fig. 7. The experimental setup for measuring the SQNR.

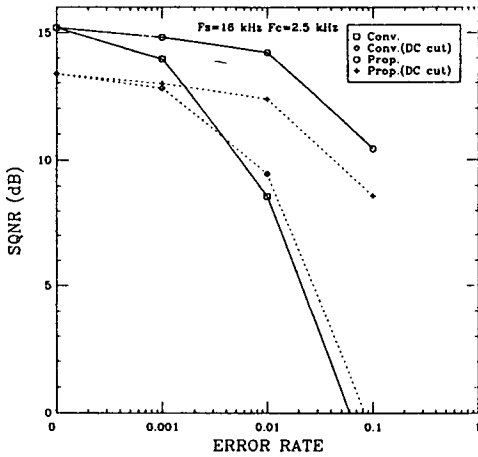


그림 8. 메모리의 비트 에러율에 대한 SQNR
Fig. 8. SQNR versus memory bit-error rate.

10^2 정도에서는 비트 에러가 없는 메모리를 이용한 경우에 비해 재생 음질의 SQNR이 약 1dB밖에 저하하지 않음을 볼 수 있다.

IV. 평가보드의 구현

본 평가 회로는 음성을 디지털 신호로 바꾸어 주는 ADM 부호기(coder)부, DRAM과의 interface부, 그리고 controller로 구성되어 있다. ADM 부호기를 구성하는 방법은 크게 아날로그와 디지털 방법이 있는데, 후자가 설계는 복잡하지만 VLSI 회로로 만들기에 간단하므로 본 연구에 채용되었다. 본 디지털 ADM 부호기는 그림 9에 보이는 것과 같은 구성을 가진다. 즉 입력의 anti-aliasing 필터와 DAC(디지털-아날로그 변환기),

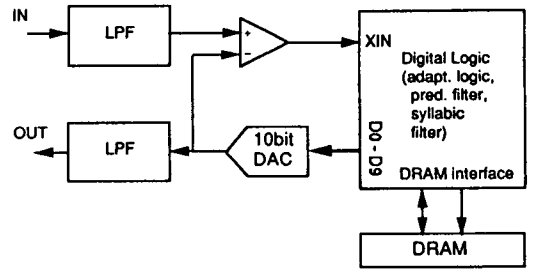


그림 9. 디지털 ADM 부호화의 구성도
Fig. 9. Block diagram of the digital ADM coder.

그리고 아날로그 비교기를 제외한 적응(adaptation) 논리회로, prediction이나 syllabic filter 등은 모두 디지털 회로로 구현된다. 이밖에 메모리 소자의 비트 에러가 전체 address 공간에 균등하게 분포하도록 번지를 바꾸는 address-shuffler, 메모리 제어부 등도 모두 디지털 회로로 구성된다. 각 부분의 동작설명은 다음과 같다.

1. 필터 연산부의 설계

ADM의 필터 연산 및 적응 알고리즘부는 앞의 식(1)과 (2)에서 보이는 것과 같이 승산과 가산 동작으로 구성되어 있다. 이를 경제적인 하드웨어로 구현하기 위해서는 정수 연산방법의 채용과 word length의 최소화, 그리고 bit-serial의 채용이 필요하다.

1) Word length의 결정

본 평가보드의 구현에는 정수 연산을 사용하였으며, 승산을 없애기 위하여서 덧셈과 shift만을 쓴 형태로 변형시켜 구현하였다. 우선 적응식에 필요한 승산을 없애기 위한 과정을 설명하면, 식 (1)에서 $0 < e^{xT} < 1$ 이고 또 e^{xT} 는 1에 가까운 값이므로 우리는 어떤 작은 실수 $\epsilon = 1 - e^{xT}$ 을 정의하여

$$\Delta_n = (1 - \epsilon)\Delta_{n-1} + G\epsilon, \tag{4}$$

를 얻는다. 여기서 ϵ 을 2^{-b} (b 는 양의 정수) 형태로 표현되는 값으로 근사화하면

$$\Delta_n = \Delta_{n-1} - 2^{-b}\Delta_{n-1} + 2^{-b}G \tag{5}$$

로 표현할 수 있다. 식 (5)는 b -비트 만큼 Δ_{n-1} 을 오른쪽으로 산술이동(arithmetic shift-right)하여 이를 Δ_{n-1} 에서 빼주고, 적응 논리 회로에서 syllabic 필터의 step-size를 증가시키라는 신호가 있을때만, $2^{-b}G$ 만큼의 gain 값을 더해주면 됨을 의미한다.

Prediction 필터에 대해서도 위에서 설명한 방법과 마찬가지로 곱셈을 산술이동(shift)으로 바꾸면, 식 (2)에서 $e^{\beta T}$ 는 1에 가까운 값이므로 $1 - e^{\beta T}$ 는 2^c (c 는 양의 정수)로 근사화할 수 있으며, 따라서

$$y_n = y_{n-1} - 2^c y_{n-1} + \Delta_n \tag{6}$$

을 얻을 수 있다.

샘플링 주파수 16kHz에 대하여 $1/\alpha$ 를 3.8msec, $1/\beta$ 를 1.0msec로 선택하면 $1 - e^{\beta T}$ 는 0.0163이 되므로 식 (5)의 b 는 6이 된다. 또 $1 - e^{\beta T}$ 는 0.0605이므로 식 (6)의 c 는 4를 얻는다.

위의 정수 연산식을 실제 회로로 구현시 내부 data 형식을 그림 10에 도시하였다. 그림에서 보듯 prediction 필터의 출력은 DAC의 입력으로 사용되는 부분과 연산 정도를 높이기 위해 필터 내부에서만 사용되는 가드 비트(guard bit)로 나누어진다. 위의 식에서 $b=6$ 이므로, 6 비트의 가드 비트가 필요하다.

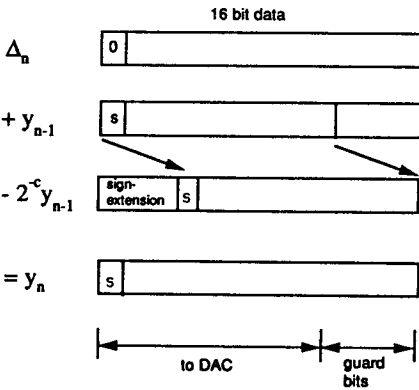


그림 10. 정수연산 디지털 필터의 내부 data 형식
Fig. 10. Internal data format of the digital filter.

필요한 DAC의 비트 수를 결정하기 위하여서, 가드 비트를 6비트로 둔채, DAC의 비트 수를 8 비트, 10 비트, 12 비트로 변화시키면서 각 경우의 SQNR을 정수 모의 실험에 의하여 측정하였다. 그림 11-(a)(b)(c)는 DAC 비트 수를 바꾸면서 메모리 소자의 비트 에러에 따른 SQNR을 측정한 것이다. 모의실험의 결과를 보면, DAC의 비트 수는 10 비트이상 이어야함을 알 수 있는데, 10 비트의 경우와 12비트의 경우를 비교해보면 SQNR의 차이가 거의 없으므로 10 비트가 선택되었다. 따라서, 전체의 내부연산 정도는 6 비트의 가드 비트를 포함하여 16 비트가 되었다.

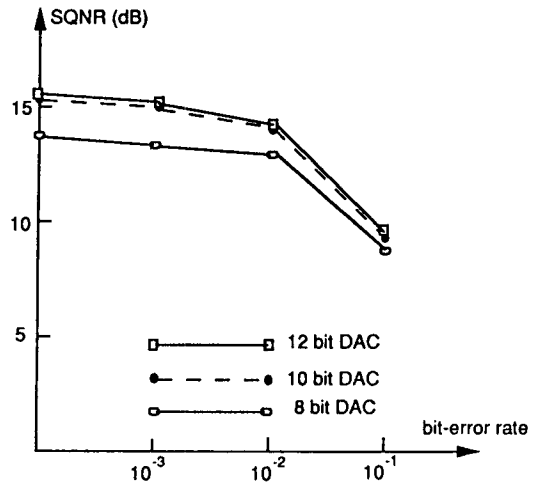


그림 11. DAC의 비트 수에 따른 SQNR
Fig. 11. SQNR versus the word length of the DAC.

2) Bit-serial 구조를 이용한 구현

본 연구에서는 필터 연산부에 bit-serial 구조를 채용하여 연산기(adder)의 숫자를 대폭 줄였다. 이 경우 연산에 필요한 시간이 그만큼 증가하고 연산의 제어가 좀더 복잡해지는 단점을 지니는데, 본 연구의 ADM IC 경우는 샘플링 주파수가 비교적 낮은 음성신호를 취급하는 회로이므로 충분한 연산시간을 확보할 수 있고, 또 마이크로 프로그램을 이용하여 연산 제어를 구현하여 위의 문제점을 해결하였다.

그림 12에 2's complement로 표현된 수를 덧셈/뺄셈할 수 있는 bit-serial 회로를 보이고 있다. A와 B는 각각 n 비트의 SHR(shift register)로 구성되어 있고 n 개의 clock이 들어 온 후에 한번의 산술 연산이 끝난다. 제어단자 F는 산술연산이 시작되는 첫 clock 간격동안만 0으로 유지되고 다른 때는 항상 1을 유지한다. 위에서 설명한 bit-serial 구조 덧셈/뺄셈기를 필터 연산부에 응용한다. 식 (5)와 (6)을 보면 각 필터가 필요로하는 연산 동작은

$$A = A - 2^d A + B \tag{7}$$

의 형태임을 알 수 있다. 그림 13은 필터 연산부의 전체를 보이고 있는데, 크게 adaptation logic부와 prediction 필터부, syllabic 필터부로 구성된다. 이 그림에서 x_n 은 양자화기(즉, 비교기)의 출력을 받아들이는 입력단자이고 DO와 DI는 이 데이터를 메모리에 썼다가 읽어내기 위한 단자이다. Prediction 필터부는 SHRO를 중심으로 FA0가 뺄셈기로, FA1이 덧셈/뺄셈기로 동작한다.

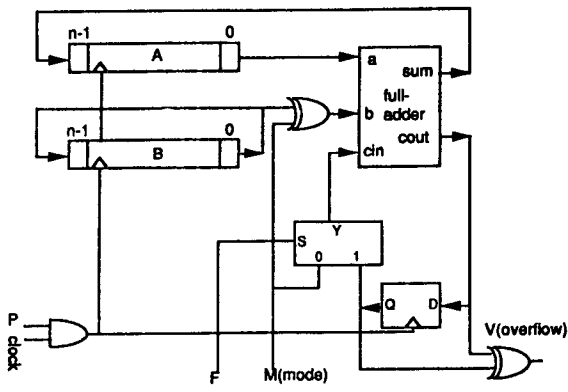


그림 12. Bit-serial 가/감산 회로
Fig. 12. Bit-serial adder /subtractor circuit.

다. 제어신호 PSIGN은 SHRO의 데이터를 산술 shift-right시 상위 4 비트를 sign으로 채우는 동작을 한다. FA1의 연산동작은 b_n 비트가 제어하고 있는데 b_n 은 +1과 -1대신 1과 0으로 표시된다. 그밖에 오버플로우를 조사하여 양의 최대값과(PMAX 신호) 음의 최대값(NMAX 신호)을 강제로 load하는 회로가 있다. Syllabic 필터부는 SHR1과 뿔셈기로 동작하는 FA2, 덧셈기로 동작하는 FA3로 구성된다. SSIGN의 역할은 PSIGN의 역할과 같은데 SHR1의 상위 6 비트를 sign으로 채운다. 연속된 출력 데이터를 검출하는 적응 논리

회로부는 exclusive-OR 게이트를 사용하여 간단히 구현한다. 필터 연산부가 한번 산술연산을 행하는데는 16개의 클럭(clock)이 필요하다. 필터연산이 끝나면 prediction 필터의 상위 10비트, 즉 SHRO의 비트 6부터 비트 15를 DAC 버퍼 레지스터로 보낸다.

2. 메모리 번지 사상(mapping) 함수의 설계

메모리 소자의 비트 에러는 어떤 부분에서 연속적으로 발생하는 경우가 많으므로 메모리의 번지 제어기는 메모리의 비트 에러가 전체 메모리 공간에 균등하게 분포되도록 불규칙한(random) 순서로 메모리 번지 신호를 발생시키도록 해야 한다. 보통의 경우 메모리 번지 제어기의 출력은 매 클럭마다 그 값이 순차적으로 하나씩 증가하는데, 이 값을 그대로 메모리의 번지 지정에 사용할 경우 메모리 소자에서 생기는 연속적인 비트에러 때문에 제안된 음성부호화방식이 효율적으로 작동하지 못할 수가 있다.

순차적으로 증가하는 메모리 번지를 일대일 대응관계를 유지하면서 불규칙한 새로운 번지로 사상해주는 함수를 구현하는 가장 쉬운 방법은 look-up table(즉, ROM)의 사용을 생각할 수 있지만, 회로의 크기 때문에 바람직하지 못하다. 여기서는 그림 14과 같이 EX-OR 게이트만으로 구성된 간단한 회로를 이용하여, 메모리 소자 내의 인접한 cell들이 연속해서 불려지는 횟수가 가능한 한 적고 또 균등하게 불려지도록 하였다. 이 회로는 연속

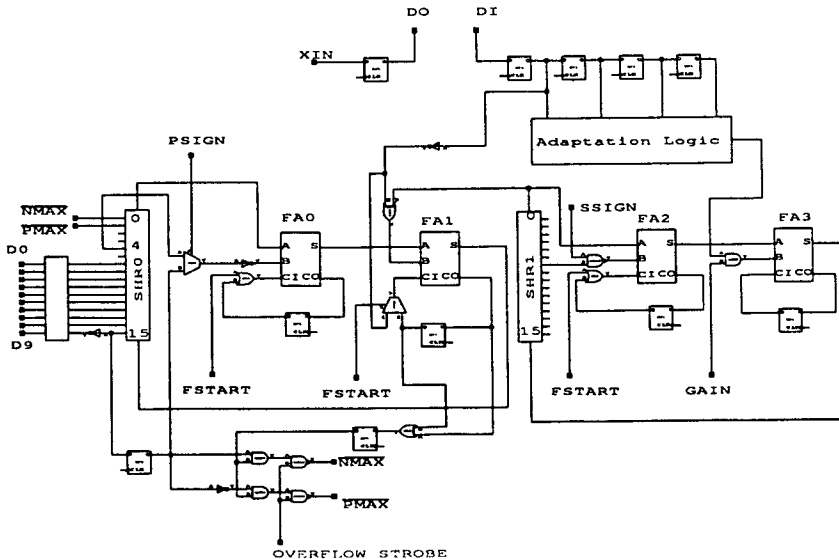


그림 13. ADM용 필터 연산부의 전체 회로
Fig. 13. Prediction and syllabic digital filter circuits.

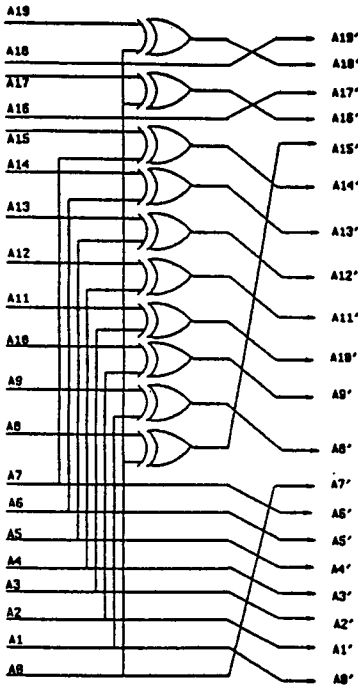


그림 14. 메모리번지 사상(mapping) 회로
 Fig. 14. Memory address mapping circuit.

되는 메모리 번지 지정때 마다 cell의 row와 column 위치가 바로 전 cell의 그것들과 서로 다르면서 두 cell간의 거리를 가급적 멀게 하는 특성을 가지고 있다. 이 회로의 성능을 컴퓨터 모의실험으로 확인하여 만족할만한 결과를 얻었다.

3. 구현 결과

본 논문에서 제안한 새로운 음성 부호화 방식을 실제로 TTL 논리소자와 마이크로 콘트롤러 그리고 DAC (Digital to Analog Converter)를 이용하여 평가 보드를 제작하고 그 성능을 확인하였다. Syllabic 필터와 prediction 필터는 식 (5)와 (6)에 의해 각각 구현되었고 모두 16-비트의 bit-serial 연산을 사용하였다. Prediction 필터의 상위 10-비트는 10-비트의 DAC로 전해져 그 출력 값이 비교기로 전달된다. 본 평가 보드는 1M DRAM과 256K DRAM을 사용할 수 있다. 약 10,000개 정도의 비트 에러를 갖는 1M DRAM(비트 에러율=약

10⁻²)을 사용하여도 음질의 저하가 느껴지지 않을 정도임을 커로 평가할 수 있었다.

V. 결 론

본 논문에서는 디지털 녹음재생 시스템을 경제적으로 구현하기 위하여, 비트 에러를 효율적으로 보상하는 새로운 음성 부호화 방식을 개발하였다. 본 개발된 방식에서는 부호화기의 출력을 메모리에 쓴 후 즉시 다시 읽어서 메모리의 비트 에러를 검출하며, 에러 검출시 다음번 샘플의 부호화 과정에서 피드백의 원리로 에러의 영향이 보상되도록 하였다. 따라서 본 개발된 방식은 에러 정정 방식처럼 추가의 메모리를 필요로 하지 않으면서, 기존의 방식보다 훨씬 메모리의 비트 에러에 강하다. 본 방식을 CVSD 방식의 ADM에 적용하여 컴퓨터 모의 실험으로 성능 확인후, 정수연산 모의 실험을 거쳐서 bit-serial logic을 이용하여 구현하였다. 본 구현된 프로토타입 보드에 약 10,000개의 비트 에러를 가지는 1M DRAM을 사용한 경우에도 만족스런 재생음을 커로 확인할 수 있었다.

參 考 文 獻

- [1] R. Steele, *Delta Modulation Systems*, Pentech Press, London, 1975.
- [2] C. K. Un and H. S. Lee, "A Study of the Comparative Performance of Adaptive Delta Modulation Systems", *IEEE Trans. Commun.*, vol.COM-28, no.1, Jan. 1980.
- [3] N. S. Jayant, "Adaptive Delta Modulation with a One-bit Memory", *Bell Syst. Tech. J.*, vol.49, pp.321-342. Mar. 1970.
- [4] C. K. Un, H. S. Lee and J. S. Song, "Hybrid Companding Delta Modulation", *Ieee Trans, Commun.*, vol.COM-29, no.9, Sep. 1981.
- [5] 성원용, 김진울, 이황수, "메모리 Bit-error에 Robust한 음성부호화 방식", 제2호 신호처리 합동학술대회, 서울, 1989.
- [6] W. Sung., J. Y. Kim, and H. S. Lee, "A Memory Bit-Error Robust Coding Method for Digital Speech Storage," *Proc. of the 9th International Kobe Symposium*, Kobe, Japan, 1991.

著 者 紹 介



成元鎔 (正會員)

1955年 4月 14日生. 1978年 2月 서울대학교 전자공학과 졸업(학사). 1980年 2月 한국과학기술원 전기 및 전자공학과 졸업(공학석사). 1980年 3월부터 1983年 7월까지 금성사 중앙연구소 근무. 1987年 7月 미국 Univ. of California Santa Barbara 졸업(공학박사). 1987年 부터 1988년까지 세방정밀(주) 상무이사. 1989年 2월부터 현재까지 서울대학교 반도체공동연구소 및 제어계측공학과 조교수. 주관심분야는 병렬처리 컴퓨터와 VLSI 등을 이용한 신호처리 시스템의 구현.



金 鎭 栗 (正會員)

1963年 3月 28日生. 1982年 3月 - 1986年 2月 서울대학교 공과대학 전자공학과(공학사). 1986年 3月 - 1988年 2月 한국과학기술원 전기 및 전자공학과(공학석사). 1988年 3月 - 현재 한국과학기술원 전기 및 전자공학과 박사과정. 주관심분야는 신호처리 알고리즘의 병렬처리 및 고속 신호처리를 위한 VLSI 구조 연구.

李 滉 洙 (正會員)

1952年 9月 19日生. 1971年 3月 - 1975年 2月 서울대학교 공과대학 전기공학과(공학사). 1976年 3月 - 1978年 8月 한국과학기술원 전기 및 전자공학과(공학석사). 1978年 9月 - 1983年 2月 한국과학기술원 전기 및 전자공학과(공학박사). 1975年 1月 - 1975年 10月 현대조선중공업(주) 설계부 사원. 1983年 3月 - 1989年 2月 한국과학기술원 전기 및 전자공학과 조교수. 1983年 3月 - 1992年 1月 한국과학기술원 전기 및 전자공학과 부교수. 1992年 2月 - 현재 한국과학기술원 서울분원 정보 및 통신공학과 부교수. 1984年 4月 - 1985年 5月 미국 Stanford 대학교 information Systems Lab. Post Doc. 연구원. 연구 분야는 디지털 통신, 이동통신, 신호처리(통신, 음성, 레이더) 등임.