

반복 복호수 감소에 의한 저전력 터보 복호기의 설계

정회원 백서영*, 김식*, 황선영*

Design of a Low Power Turbo Decoder by Reducing Decoding Iterations

Seo-Young Back*, Sik Kim*, Sun-Young Hwang* *Regular Members*

요 약

본 논문에서는 사용 전원이 제한적인 이동통신 기기에 사용되는 터보 복호기의 전력 소모 원인이 되는 반복 복호 횟수를 줄이기 위한 알고리즘을 제안한다. 기존의 반복 횟수를 제어하는 방법의 경우, CRC를 사용하는 방법은 하드웨어 복잡도가 낮은 반면 BER 성능의 감소가 큰 단점이 있으며 LLR을 이용하는 방법은 BER 성능이 임계값에 의존적이며 임계값을 계산하는 추가적인 하드웨어가 필요한 단점이 있다. 제안된 알고리즘은 터보 코드의 우수한 오류 정정 성능을 이용하여 하나의 데이터 프레임에 대한 연속된 두 번의 복호 출력이 동일한 경우 복호를 종료하는 방법으로 간단한 버퍼와 계수기를 이용하여 하드웨어의 부담을 최소화하는 구현이 가능하며 BER 성능의 감소 없이 전력 소모를 줄일 수 있음을 확인하였다. 실험 결과, 제안한 방법은 BER 성능의 감소 없이 반복 복호 횟수를 60% 정도 줄이는 것으로 나타났으며, 반복 복호 횟수의 감소 정도에 비례하여 소모 전력도 절약된다.

Key Words : 터보 코드, 반복 복호 제어, 저전력

ABSTRACT

This paper proposes a novel algorithm for a low power turbo decoder based on reduction of number of decoding iterations, targeting power-critical mobile communication devices. Previous researches that attempt to reduce number of decoding iterations, such as CRC-aided and LLR methods, either show degraded BER performance in return for reduced complexity or require additional hardware resources for controlling the number of iterations to meet BER performance, respectively. The proposed algorithm can reduce power consumption without degrading the BER performance, and it is achieved with minimal hardware overhead. The proposed algorithm achieves this by comparing consecutive hard decision results using a simple buffer and counter. Simulation results show that the number of decoding iterations can be reduced to about 60% without degrading the BER performance in the proposed decoder, and power consumption can be saved in proportion to the number of decoding iterations.

I. 서론

1993년 Berrou 등에 의해 제안된 터보 코드는 Shannon의 통신 채널의 전송률 한계에 관한 이론에

근접하는 오류 정정 성능을 나타내어 통신 분야에서 많은 연구가 이루어져 왔다[1]. 최근에는 차세대 이동 통신 시스템인 IMT-2000(International Mobile

*서강대학교 전자공학과 CAD & Computer System 연구실 (hwang@ccs.sogang.ac.kr)

논문번호 : 030050-0203, 접수일자 : 2003년 2월 3일

※본 연구는 한국과학재단 목적기초연구 (R01-2001-000-00321-0) 지원으로 수행되었습니다.

Telecommunication 2000)의 고속 데이터 전송용 채널 부호화 방식으로 채택됨에 따라 휴대용 응용 기기를 VLSI 시스템으로 구현하기 위한 저전력 시스템 설계가 중요한 과제로 부각되고 있다.

터보 부호기는 부호기 입력과 인터리버에 의해 재배열된 입력을 각각 구성 부호기(constituent encoder)를 통해 부호화하여, 입력 정보와 다중 패리티 정보를 출력하도록 구성된다. 독립적인 다중 패리티는 반복 복호를 가능하게 하고 인터리버는 연접 오류(burst error)를 효과적으로 분산시키는 역할을 한다. 인터리버는 프레임 단위로 정보의 재배열을 수행하며 프레임의 크기에 따라 인터리버의 메모리 크기가 결정된다. 인터리버에 사용되는 메모리 크기가 클수록 연접 오류를 효율적으로 분산할 수 있으나 이로 인한 지연 시간이 크게 증가하므로 실제 구현에 있어서는 인터리버 크기가 제한 될 수밖에 없다.

터보 복호기는 각 구성 부호기에 대응하는 SISO(Soft Input/Soft Output) 복호기와 경관정 블록으로 구성된다. SISO 복호기의 연관정 출력으로부터 복호기에 의해 추가된 부가 정보(extrinsic information)를 계산하여 다른 SISO 복호기의 사전 확률(APP: A Priori Probability) 입력으로 사용하는 과정의 반복으로 복호기 출력의 신뢰도를 증가시킨다. 터보 코드에 사용되는 SISO 알고리즘으로는 MAP(Maximum A Posteriori), log-MAP, max-log-MAP 및 Viterbi 디코더를 변형시킨 SOVA(Soft Output Viterbi Algorithm) 등이 알려져 있다[2][3][4]. 터보 코드의 반복 복호는 BER 성능을 향상시키는 반면 복잡성을 증가시키며 많은 전력 소모의 원인이 된다. 일반적으로 한번 반복 복호가 Viterbi 디코더의 4배 정도의 복잡성을 가지고 있기 때문에 이러한 구조를 반복해서 사용하는 터보 복호기의 전력 소모는 크게 증가한다. 저전력 터보 코드의 설계를 위해 BER 성능을 유지하는 한계 내에서 프레임 당 반복 복호의 횟수를 복호 단계에서 실시간으로 조절할 수 있는 방법이 필요하다. 일

반적으로 반복 복호의 횟수를 조절하기 위한 방안으로 CRC(Cyclic Redundancy Check)를 사용하는 방법과 LLR(Log Likelihood Ratio) 값의 변화 정도를 이용하는 방법이 보고되었다[2][4][5]. CRC를 적용하는 방법의 경우 BER 성능 감소가 크기 때문에 실제로 사용하기에는 한계가 있으며, LLR 값의 변화 정도를 이용하는 방법은 임계값(threshold)과 복호의 정확성에 대한 함수 관계가 명확히 밝혀진 바 없으며, 대략적인 LLR 특성에 기대어 반복 복호 횟수를 제어하는 방식은 BER 성능의 감소를 초래할 수 있다.

본 논문에서는 터보 복호기에 간단한 버퍼와 계수기를 추가하여 프레임당 반복 복호 횟수를 유동적으로 조절할 수 있는 알고리즘에 대해 제안하되, 추가된 버퍼의 크기를 최소화할 수 있는 방법을 제안한다. 제안된 구조의 성능 확인을 위해 Log MAP 알고리즘을 기반으로 한 터보 코드 환경에서 실험을 수행하였으며, 기존의 알고리즘과 비교하여 BER 성능과 반복 복호 횟수를 비교 실험하였다.

II. 반복 복호 제어 알고리즘

1. CRC를 이용하는 방법

CRC 방식은 데이터 전송시에 터보 코드의 부호기 측에서 각 데이터 프레임마다 수신단에서 전송된 프레임 에러를 검출할 수 있도록 여분의 정보 FCS(Frame Check Sequence)를 포함하여 전송한다. FCS는 나눗셈 연산을 수행한 나머지가 되며 수신단에서는 검출된 데이터를 원래의 데이터를 나눈 값으로 나누었을 때 나머지가 0인지 검사하여 오류를 검출한다.

그림 1은 CRC를 사용하여 반복 복호 횟수를 제한할 수 있는 복호기의 구조를 보인다. 복호기 측에서는 매번 반복 복호를 수행할 때마다 CRC를 체크하여 FCS가 0이면 그 상태에서 반복 복호를 끝내고, 0이 아니면 한번 더 반복 복호를 수행하되 미

리 정해 놓은 최대 반복 복호 횟수를 초과하지 않도록 하고 있다[4][5].

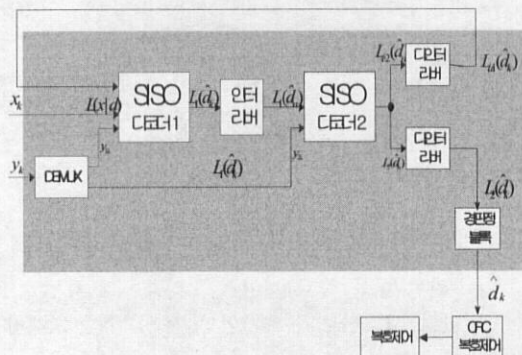


그림 1. CRC 반복 복호 제어 알고리즘을 적용한 터보 코드.

CRC를 적용하여 반복 복호 횟수를 제한하는 방법은 추가적인 버퍼가 필요하지 않고 현재 복호 결과만을 가지고 반복 복호의 수행 여부를 결정할 수 있다는 장점이 있으나, FCS 부분을 같이 전송하므로 데이터 전송 효율이 떨어지며 CRC 자체의 오류로 인하여 BER 효율이 떨어진다. 또한 CRC 체크를 수행 하는데 가장 중요한 역할을 하는 FSC 부분이 채널 에러로 손상되어 제대로 복호되지 못한 경우 CRC 체크를 수행하더라도 그 결과를 신뢰할 수 없는 등 정확한 반복 복호 지점을 결정하는 데에는 문제가 있다. CRC 방법은 이론적으로 제안되었으나 실제 CRC 방법만을 사용하여 반복 복호 횟수를 제한하는 경우 에러 정정 성능 감소가 크기 때문에 복호기에 직접 구현하는 것은 한계가 있다.

그림 2는 CRC를 적용하여 반복 복호 횟수를 제어하는 경우 CRC 길이에 따른 터보 복호기의 BER 성능을 나타낸다. CRC 코드 길이가 길어짐에 따라 BER 성능은 향상되는 특성을 나타낸다. 그러나 CRC를 체크하기 위한 코드 사이즈를 16으로 하여 반복 복호 횟수를 제한한다 해도 최대 반복 복호 횟수를 6으로 고정하여 반복 복호를 수행하는 터보 복호기의 BER 성능에 근접하지 못한다. 이때 최대 반복 복호 횟수는 임의로 정한 값이며, 많은 논문에서 6내지 8정도의 값으로 고정하여 BER 성능을 분석하는 실험을 수행하고 있다[6][7][8].

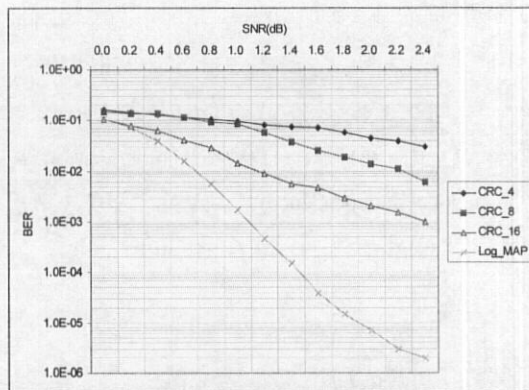


그림 2. CRC를 적용하여 반복 복호 횟수를 제어하는 터보 복호기의 BER 성능 비교.

CRC를 사용하여 반복 복호 횟수를 제어하는 방법은 여러 연구에서 보고되었다[7][9]. 참고 문헌 [7]에서는 CRC 방법에 관한 언급이 있으나 실제 시뮬레이션 결과에서는 CRC를 적용하지 않았으며, SNR이 증가할수록 더 적은 반복 복호 횟수로도 좋은 BER 성능을 나타냄을 확인하고 있을 뿐이다. 참고 문헌 [9]에서 프레임의 크기를 작게 하여 반복 복호를 수행하는 방법을 제안하고, 작은 프레임에 대하여 CRC를 적용하여 필요 없는 반복 복호 횟수를 줄이는 방법에 대하여 제안하였다. 작은 프레임에 대해 CRC를 적용할 경우 비교적 BER 성능의 감소 없이 효율적으로 반복 복호 횟수를 제어할 수 있으나, 데이터 전송 효율의 감소 정도가 상대적으로 크며, 기본적으로 대량 데이터 전송을 목적으로 하는 터보 코드에서 작은 프레임의 사용은 효율성이 떨어진다.

2. LLR 값의 변화를 이용하는 방법

하나의 SISO 복호기의 출력 값이 다음 SISO 복호기의 사전 정보 값의 바탕이 되는 터보 복호기의 구조 상 반복 복호가 진행됨에 따라 LLR(Log Likelihood Ratio)의 신뢰도는 증가한다. 두 개의 연속적인 디코더의 결과 LLR 값의 교차 엔트로피(cross entropy)의 값은 반복 복호가 진행될수록 감소된다. LLR의 값을 이용하여 반복 복호를 제어하는 방법은 매번 반복 복호를 수행하여 LLR 값들

사이의 교차 엔트로피 $T(i)$ 를 구하여 일정 임계값 이내인 경우 반복 복호를 멈추는 방식이다[6][10].

두 개의 확률 분포 값 $P(d_k)$, $Q(d_k)$ 사이의 교차 엔트로피[6]는 식 (1)과 같고 이때, 통계적 독립(statistical independence)이 이루어져 있다고 가정하면, 식 (2)와 같이 나타낼 수 있다[6].

$$E_b \left\{ \log \frac{P(d_k)}{Q(d_k)} \right\} \tag{1}$$

$$\log \left\{ \frac{P(d_k)}{Q(d_k)} \right\} = \sum_k \log \left\{ \frac{P(d_k)}{Q(d_k)} \right\} \tag{2}$$

터보 복호기의 두 개의 SISO 복호기가 연속적인 반복을 $(i-1)$, (i) 로 표시하고, 원래의 정보 순서값과 같은 순서의 LLR 결과 값을 ' \leftrightarrow '의 기호로 인터리버를 통과한 정보 순서를 ' \uparrow '라고 표현하여, 각각의 SISO 복호기를 통과한 LLR 결과 들 사이의 교차 엔트로피를 구하면 식 (3)과 같다 [6].

$$E_b \left\{ \log \frac{P(d_k)}{Q(d_k)} \right\} \approx \sum_k \frac{|\Delta L_e^{\leftrightarrow(i)}(d_k)|^2}{\exp(|L_Q^{(i)}(d_k)|)} \tag{3}$$

따라서 LLR 값의 분포 정도를 이용하여 반복 복호를 제어하는 기준으로 식 (4)와 같이 나타낼 수 있다.

$$T(i) = \sum_k \frac{|\Delta L_e^{\leftrightarrow(i)}(d_k)|^2}{\exp(|L_Q^{(i)}(d_k)|)} < threshold \tag{4}$$

LLR 값의 변화를 이용하는 방법에 있어서는 임계값을 결정하는 방법이 중요하다. 임계값이 작으면 복호 횟수는 크게 감소하나 BER 성능이 감소되며, 임계값이 너무 크면 BER 성능은 유지할 수 있으나 필요 없는 반복 복호로 인하여 반복 복호 횟수가 증가하게 된다. 이러한 임계값은 실험적으로 결정할 수밖에 없으며 모의 실험 결과 $T(i) < (10^{-2} \sim 10^{-4}) T(0)$ 을 만족하는 경우 BER

성능의 감소 없이 반복 복호를 제어할 수 있다고 알려져 있다[6].

III. 제안된 반복 복호 제어 알고리즘

본 절에서는 터보 코드에서 BER 성능의 감소 없이 프레임당 반복 복호 횟수를 유동적으로 조절할 수 있는 방법에 대하여 제안한다.

1. 제안된 반복 복호 제어 알고리즘

일반적으로 터보 복호기의 반복 복호는 프레임 단위로 수행이 되며, 프레임당 반복 복호를 수행할 경우 복호 상태는 다음과 같은 세 가지 경우로 나누어 볼 수 있다.

(1) 현재 반복 복호 상태에서 프레임 내에 오류가 전혀 없이 복호가 되어 더 이상의 반복 복호가 필요 없는 경우.

(2) 현재 반복 복호 상태에서 프레임 내에 약간의 오류가 존재 하나, 더 이상 반복 복호를 수행하더라도 수정될 수 없는 오류를 가지고 있는 경우.

(3) 현재 반복 복호 상태에서 프레임 내에 오류가 있으며, 더 이상의 반복 복호를 수행하면 완전히 혹은 부분적으로 오류가 수정될 수 있는 가능성이 있는 경우.

위의 세 가지 경우 중 (1), (2)의 경우는 그 상태에서 전원을 차단시키면 BER 성능 감소 없이 반복 복호 횟수를 제한할 수 있으나, (3)의 경우는 계속해서 전원을 공급하여 반복 복호를 수행하여야 BER 성능의 감소가 없다. 따라서 BER 성능의 감소 없이 반복 복호를 제한하는 방법을 제시하기 위해서는 반복 복호를 계속해서 수행해도 더 이상 오류가 수정될 수 없는 블록의 경우와 그렇지 않은 블록의 경우를 반복 복호를 수행하는 과정에서 찾아낼 수 있어야 한다.

그림 3은 제안한 반복 복호 제어 알고리즘을 보인다. 그림에 나타낸 바와 같이 제안한 방법은 각각의 복호를 수행하여 경관정을 거친 값(A)이 이전 복호기의 경관정을 거친 값(B)과 차이가 없다면 이를 더 이상 반복 복호를 수행해도 오류 정정이 불가

능한 상태(1)(2)의 경우로 보고 전원을 차단시킨다.

첫 번째 SISO 복호기를 통과 한 값은 경판정 블록을 거쳐 버퍼에 저장되고 다음 번 SISO 복호기의 경판정 결과 값과 비교하기 위한 대상이 된다. 두 번째 SISO 복호기의 경판정 결과 값은 버퍼에 저장된 값과 XOR 연산을 수행한 후 다시 버퍼에 저장된다. 이 때 XOR 연산의 결과 값은 카운터기에 의해서 합산이 되며, 합산 결과가 '0'인 경우에는 두 SISO 복호기의 경판정 결과가 같으므로 더 이상 반복 복호를 수행해도 오류 정정이 불가능한 상태로 보고 전력 공급을 차단하여 반복 복호를 멈추게 된다. 합산 결과가 '0'이 아니라면 계속 해서 전력을 제공해 주어야 하며 이러한 과정의 반복으로 프레임당 반복 복호의 횟수를 제한하게 된다.

```

/* Buffer : 프레임 크기의 버퍼
S_MAX : 가능한 최대 반복 복호 횟수
K : 프레임 크기
A : 현재 복호기의 복호 결과 경판정 값
B : Buffer에 저장된 경판정 값 */

Buffer = 0;
while ( i < S_MAX ) (
    복호 수행;
    nCount = 0;
    for ( j = 0; j < K; j++ ) (
        nCount += A (XOR) B;
    )
    if ( nCount = 0 ) (
        Break;
    ) else (
        현재 복호 결과값을 Buffer에 저장;
        i++;
    )
)
    
```

그림 3. 제한한 반복 복호 제어 알고리즘.

그림 4는 제안한 알고리즘을 구현한 하드웨어 구조를 보인다. 각각의 SISO 복호기의 경판정 결과 값은 버퍼에 저장되어 있는 값과 XOR 연산이 수행되며 XOR 연산 수행 결과 '1'의 개수가 카운터기에 의해서 계산된다. '1'의 개수는 이전 복호 결과 값과

현재 복호 결과 값의 차이 개수를 의미하므로 '1'의 개수가 하나라도 존재한다면 계속해서 전력을 공급하여 반복 복호가 수행될 수 있도록 한다. 버퍼에 저장된 값은 단지 현재의 복호 결과 값과 비교하기 위한 대상일 뿐이므로 XOR 연산 후에는 더 이상 필요 없는 값이 된다. 따라서 현재의 복호 결과 값을 다시 버퍼에 저장하여 다음 번 SISO 복호기의 결과 값과 비교하기 위한 대상 값이 존재하도록 한다.

이전 복호의 결과 값과 현재 복호의 결과 값이 같다고 하여 완전하게 복호가 되었다고 볼 수는 없으나, 터보 코드의 우수한 오류 정정 성능을 기반으로 유추해 볼 때 SISO 복호기를 통과한 값이 변화가 없다면 더 이상 복호를 하더라도 오류 정정이 불가능한 상태로 보고 반복 복호를 제어함으로써 복호 레이턴시와 전력 소모를 줄일 수 있는 알고리즘이다.

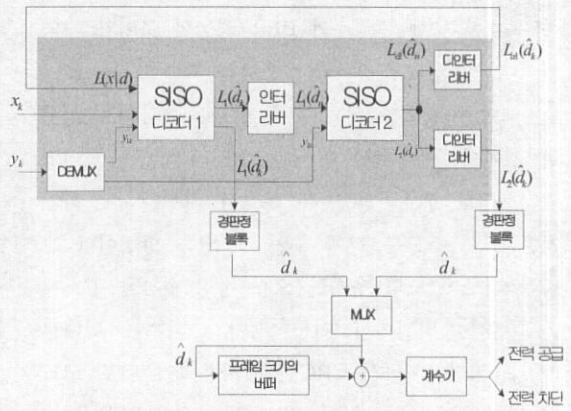


그림 4. 제한한 반복 복호 제어 알고리즘을 구현한 하드웨어 구조

IV. 실험 결과

시뮬레이션 시스템은 UNIX환경의 C언어를 사용하여 구현하였다. 실험에 사용된 공통된 파라미터는 표 1과 같다. 실험에 사용된 SISO 알고리즘은 블록 MAP 알고리즘을 사용하였으며, 블록 크기는 32로 하여 실험하였다[8].

1. BER 성능 및 프레임 당 평균 반복 복호 횟수
그림 5는 기존 반복 복호 제어 알고리즘인 CRC를 사용하는 방법, LLR 값의 변화 정도를 이용하는

방법과 제안한 반복 복호 제어 알고리즘의 SNR에 따른 BER 성능의 변화를 나타낸 그래프이다. 프레임당 CRC를 체크하기 위한 부가적인 정보 비트의 크기는 32으로 하여 실험을 수행 하였다. CRC 방법을 사용하여 반복 복호 횟수를 제한하는 경우 프레임당 CRC를 체크하기 위한 부가 정보를 포함하여야 하기 때문에 데이터 전송 효율이 떨어지며, SNR이 증가할수록 오류 정정 성능이 증가하기는 하나 원래의 터보 코드의 오류 정정 성능과는 차이를 보인다.

LLR 값의 변화 정도를 이용하여 반복 복호를 제어하는 경우 한번 반복 복호를 수행할 때마다 결과 LLR 값에 대한 교차 엔트로피를 구하여 임계값 ($T(i) < 10^{-3} T(0)$)을 초과하지 않는 경우 반복 복호를 멈추는 방식을 사용하여 반복 복호를 제어하였다. 실험 결과 반복 복호 횟수 6으로 고정하여 반복 복호를 수행한 경우와 BER 성능의 차이가 거의 없이 반복 복호가 제어되고 있음을 확인할 수 있다.

제안한 방법도 복호 횟수 6으로 고정하고 실험한 결과와 거의 차이가 없는 BER 성능을 보이고 있다. 터보 복호기의 반복 복호의 횟수를 제한하는 것은 전력 소모를 크게 줄일 수 있는 방법이나 그것은 터보 코드의 우수한 오류 정정 성능을 감소시키지 않는 범위에서 이루어져야만 의미가 있다고 할 수 있다. 따라서 CRC만을 사용하여 터보 코드의 반복 복호를 제어하는 방법은 원래의 BER 성능에 크게 미치지 못하므로 거의 효용성이 없다고 판단된다.

표 1. 실험에 사용된 파라미터.

파라미터	값
구속장 (K)	3
메모리의 길이(v)	2
생성 다항식(g_1/g_2)	7/3
프레임 크기(N)	1024
인터리버 사용	S 랜덤(821)
부호율(R)	1/3
채널, 변조 방식	AWGN, BPSK
입력 데이터 갯수	3,000,000개
E_b/N_0 (dB)	0.0 ~ 2.4(0.2 단위)

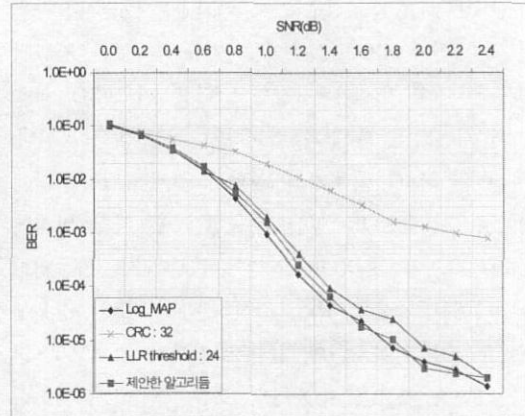


그림 5. 제안된 방식과 기존 반복 복호 제어 알고리즘의 BER 성능 비교.

표 2는 SNR에 따른 프레임당 평균 반복 복호 횟수를 나타낸다. 제안한 방법은 CRC 보다 반복 복호 횟수의 감소 정도가 적기는 하나 BER 성능의 감소 없이 반복 복호 횟수를 제어할 수 있다. 그리고, 반복 복호 제어 알고리즘을 사용하지 않는 방식 보다 SNR이 증가할수록 프레임당 반복 복호 횟수가 60% 이상 감소함을 확인할 수 있다.

표 2. SNR에 따른 프레임 당 평균 반복 복호 횟수 비교

SNR(dB)	반복 복호 횟수			고정 복호 대비 감소율 (6번 고정 반복 복호)
	CRC 적용	LLR 값 이용	제안한 방법	
0.0	6.0	6.0	6.0	0 %
0.4	5.9	6.0	5.7	5 %
0.8	5.1	5.9	4.7	22 %
1.2	3.7	5.6	3.5	42 %
1.6	2.7	4.9	2.8	53 %
2.0	2.1	4.4	2.4	60 %
2.4	1.8	3.9	2.1	65 %

LLR을 사용하여 반복 복호 횟수를 제어하는 알고리즘의 경우 BER 성능은 복호 횟수를 고정하여 반복 복호를 수행한 경우와 같게 나타낼 수 있으나, 복호 횟수가 제안된 방식에 비해 많은 것으로 나타났다. 이는 LLR의 교차 엔트로피가 반복 복호가 진행됨에 따라 감소하는 성향을 나타내기는 하나 복호의 정확성을 나타내는 지표가 되기에는 부

축함을 뜻한다. LLR 값의 변화 정도와 복호의 정확성을 나타내는 함수 관계는 아직까지 밝혀진 바가 없으며, 따라서 LLR 변화의 일반적인 성향만으로 반복 복호의 횟수를 제어하는 방법에는 한계가 있을 수밖에 없다. 현재 실험에서 교차 엔트로피의 임계값은 10^{-3} 의 값을 갖도록 설정하였으며 프레임 당 평균 반복 횟수가 제안한 방법보다 훨씬 큰 값을 가짐을 확인할 수 있다.

2. 제안된 구조의 면적 및 전력 소모 정도

면적 및 소모 전력을 비교하기 위해 VHDL 코딩을 이용하여 기존 구조와 제안된 구조를 구현하였으며, Synopsys 사의 Design Analyzer를 사용해 합성과정을 수행하여 면적과 소모 전력을 측정하였다. 하이닉스의 $0.35 \mu m$ 공정 라이브러리를 이용하였으며, 메모리의 면적과 전력은 UMC에서 제공하는 메모리 컴파일러를 이용해 측정을 하였다. 면적의 측정 단위는 2-input nand 게이트의 크기를 1의 단위로 본다.

표 3은 그림 5에 표현되어 있는 BER 성능을 얻기 위해 sub-block을 32로 했을 경우 하나의 프레임 수행하는 데 요구되는 clock 수를 나타내고 있다[8]. 제안된 구조는 기존 구조보다 SNR 값이 증가할수록 적은 clock 수로 원하는 BER 성능을 얻을 수 있음을 볼 수 있다. 주어진 동작 시간에서 원하는 BER 성능을 얻기 위해 요구되는 clock 수가 줄어든다면, 줄어든 시간만큼 주어진 시스템의 전력 공급을 차단함으로써 전력 소모를 줄일 수 있다.

표 3. 제안한 터보 복호기에 요구되는 clock 수.

SNR(dB)	요구되는 평균 clock 수		clock 수 감소율
	기존 구조	제안한 구조	
0.4	13,440	12,768	5.0 %
0.8	13,440	10,528	21.7 %
1.2	13,440	7,456	44.5 %
1.6	13,440	5,978	55.9 %
2.0	13,440	5,376	60.0 %
2.4	13,440	4,704	65.0 %

표 4는 기존의 터보 복호기와 제안된 구조와의 면적을 비교하여 나타내고 있다. 제안된 구조는 기존 구조와 거의 비슷한 면적을 나타내고 있으므로 제안한 반복 복호 제어 알고리즘을 사용하여 면적 측면의 손실이 거의 없이 저전력 터보 복호기의 설계를 이룰 수 있다.

표 4. 제안한 구조의 면적 비교.

	연산기 및 제어기	메모리	전체	면적 증가율
기존 구조	36,849.2	203,322.842	240,172.042	-
제안된 구조	37,308.3	203,322.842	240,631.142	0.2 %

V. 결론

본 논문에서는 Log-MAP 터보 복호기의 저전력 설계를 위해 BER 성능의 감소 없이 반복 복호 횟수를 효율적으로 제한하기 위한 알고리즘을 제안하였다. 기존의 반복 복호 제어 알고리즘은 반복 복호 횟수를 제어하기는 하나 BER 성능의 감소 정도가 크게 나타나고 있음에 반해, 제안한 반복 복호 횟수 제어 알고리즘은 비교적 간단한 구조로 하드웨어 면적을 기존 구조와 비슷하게 유지하면서 BER 성능의 감소 없이 프레임 당 반복 복호 횟수를 최대 60% 이상 줄일 수 있음을 확인하였으며, 이로 인하여 요구되는 전력 소모량도 반복 복호 횟수 감소량에 비례하여 줄일 수 있음을 확인하였다.

감사의 글

본 연구는 한국과학재단 목적기초연구 지원에 의해 수행되었습니다. (R01-2001-000-00321-0)

참고문헌

[1] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correction coding and decoding: Turbo codes," in Proc. ICC, Geneva,

pp. 1064-1070, May 1993.

- [2] L. Bahl, J. Cocke, F. Jelinek and J. Raviv, "Optimal decoding of linear codes for minimising symbol error rate," Diploma Thesis, University of Ulm, Sweden, Nov. 1997.
- [3] A. J. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," IEEE Transactions on Information Theory, vol. IT-13, pp. 260-269, Apr.1967.
- [4] J. Hagenauer, "Source-controlled channel decoding," IEEE Transactions on Communications, vol. 43, pp. 2449-2457, Sep. 1991.
- [5] 이문호, "비동기식 IMT-2000의 채널 부호화," 한국 통신 학회 학회지 제 14권, 9호, pp. 170-187, 1996년 9월.
- [6] J. Hagenauer, "Iterative decoding of binary block and convolutional codes," IEEE Trans, Inform. Theory, Vol. IT-42, No. 2, pp. 429-445, Mar. 1996.
- [7] Y. Chung-Wai, K. Letaief, S. Roger, R. Murch, "On the FER performance and decoding complexity of turbo codes," in Proc. IEEE 49th VTC., Houston, USA, Vol. 3, pp. 2214-2218, Feb. 1999.
- [8] G.Park, S. Yoon, C. Kang, and D. Hong, "An implementation method of a turbo-code decoder using a block-wise MAP algorithm," in Proc. VTC., Tokyo, Japan, pp.2956-2961, May 2000.
- [9] C. Zhipei, W. Zhongfeng, K. Parhi, "A study on the performance, power consumption tradeoffs of short frame turbo decoder design," in Proc. IEEE International Conference, Sydney, Australia, pp. 2637-2640, Jan. 2001.
- [10] 김동한, 황선영, "초기 임계값 설정에 의한 효율적인 터보 복호기 설계," 한국 통신학회 논문지, 제 26권, 5A호, pp. 582-591, 2001년 5월.

황 선 영 (Sun-Young Hwang)



1976년 2월 : 서울 대학교
전자공학과 졸업
1978년 2월 : 한국 과학원 전기
및 전자공학과 공학석사 취득
1986년 10월 : 미국 Stanford
대학 전자공학 박사학위 취득
1976~1981 : 삼성 반도체 주식회사 연구원, 팀장
1986~1989 : Stanford 대학 Center for Integrated
System 연구소 책임 연구원
Fairchild Semiconductor Palo Alto
Reaserch Center 기술 자문
1989~1992 : 삼성전자(주) 반도체 기술 자문
1989년 3월~현재 : 서강대학교 전자공학과 교수
2002년 4월~현재 : 서강대학교 정보통신대학원장
<관심분야> SoC 설계 및 framework 구성,
CAD시스템, Computer Architecture 및
DSP System Design 등

김 식 (Sik Kim)



1994년 2월 : 서강대학교 전자
공학과 졸업
1996년 2월 : 서강대학교 전자
공학과 석사학위 취득
1996년
3월~10월 : 현대전자(주)
연구원
2003년 8월 : 서강대학교 전자공학과 박사학위 취득
2003년 3월~현재 : 삼성전자 System LSI 사업부
CAE센터 책임연구원
<관심분야> 고속 저전력 digital 하드웨어 구조
설계, SoC 설계 및 CAD 시스템 등

백 서 영 (Seo-Young Back)



2001년 2월 : 서강대학교 전자
공학과 졸업
2003년 2월 : 서강대학교 전자
공학과 석사학위 취득
2003년 3월~현재 : LG 전자
DTV 연구소 연구원
<관심분야> SoC, Low power, DMB, Channel
coding