

# 병렬 SISO 복호기에 의한 저전력 터보 복호기의 설계

정회원 이희진\*, 황선영\*

## Design of a Low-Power Turbo Decoder Using Parallel SISO Decoders

Hee-jin Lee\*, Sun-young Hwang\* *Regular Members*

### 요 약

터보 코드는 Shannon의 한계 이론에 가까운 성능을 보이나 알고리즘의 특성상 반복 복호와 대용량 메모리의 사용으로 구현이 복잡하며 전력 소모가 크다. 본 논문에서는 병렬 구조 방식을 적용할 경우 출력 주파수가 감소되어 결과적으로 전력 소모가 감소된다는 사실을 이용하여 터보 부호기에 병렬 구조 방식을 적용한 설계를 제안하였다. 하드웨어의 복잡도를 줄이기 위해 Max-Log-MAP 방식을 사용하였으며, 병렬 구조가 적용된 터보 부호기를 위한 새로운 인터리버 구조를 제안하였다. 제안된 인터리버의 사용으로 병렬 구조화된 각각의 SISO 부호기들이 인터리버에 대해 읽기/쓰기 동작을 수행할 경우 발생할 수 있는 메모리 충돌을 방지할 수 있다. 실험 결과 기존의 구조와 비교하였을 경우 인터리버 제어기에 의한 면적의 증가는 무시할 정도이며, 전력 소모는 약 40% 정도 감소되었다.

Key Words : 터보 복호기, 저전력 설계

### ABSTRACT

Turbo code is popularly used for the reliable communication in the presence of burst errors. Even if it shows good error performance near to the Shannon limits, it requires a large amount of memories and exhibits long latency. This paper proposes an architecture for the low power implementation of the Turbo decoder adopting the Max-Log-Map algorithm. In the proposed design, two SISO decoders are designed to operate in parallel, and a novel interleaver is designed to prevent the collision of memory accesses by two SISO decoders. Experimental results show that power consumption has been reduced by about 40% in the proposed decoder compared to previous Turbo decoders. The area overhead due to the additional interleaver controller is negligible.

### I. 서론

제 3세대 이동통신 방식인 IMT-2000은 32 kbps 이상의 고속동작이 요구되는 통신 시스템으로, 고속 동작과 오류없는 전송을 위해 채널 부호화기/복호화기의 표준으로 터보 코드를 채택하였다[1]. 1993년

Berrou에 의해 제안되어 IMT-2000의 기본 채널 코딩 방식으로 채택된 터보 코드는 반복 복호와 확률 이론을 바탕으로 하여 Shannon의 한계 이론에 근접한 정확도 높은 복호 결과값을 보인다[2][3]. 전원 공급이 제한적인 이동통신 환경에서는 전력 소모를 최소화하여 이동 통신 기기의 사용시간을 증대시키

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

논문번호 : KICS2004-06-039, 접수일자 : 2004년 6월 3일

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

는 것이 중요하나, 반복 복호 과정의 채택과 대용량 메모리 모듈의 사용으로 전력 소모량이 증가한다. 따라서 저전력 터보 코드 설계를 위한 다양한 방법들이 연구되고 있다[4][5][6]. 본 연구에서는 터보 코드의 전체 동작 속도를 줄이면서 기존의 터보 복호기와 같은 성능을 낼 수 있는 병렬 구조를 적용한 터보 적용한 터보 복호기에 대한 연구를 진행하였다. 병렬 구조의 SISO 복호기를 사용하여 동작 주파수를 감소시켜 전체적인 전력소모를 줄일 수 있다. 2절에서는 터보 복호기의 구조와 알고리즘에 대해 설명하고, 3절에서는 병렬 구조가 적용된 터보 복호기의 구조와 이를 위한 인터리버를 제시한다. 4 절에서는 병렬 구조가 적용된 터보 복호기의 실험 결과를 MAP, Max-Log-MAP이 구현된 터보 복호기와 비교 결과를 제시한다.

## II. 터보 코드 복호기의 구조

터보 복호기는 2개의 SISO (Soft In-Soft Out) 복호기와 인터리버, 역인터리버, 경관정 블록으로 구성된다. 채널을 통해 함께 전송된 연속적인 정보 비트들과 터보 부호화기에서 발생된 패리티 비트들이 터보 복호기에 입력값으로 들어온다. 터보 복호기에서는 Trellis mux를 통해 정보 비트와 패리티 비트로 구분되어 SISO 복호기에 입력으로 들어가게 되며 정보 비트, 패리티 비트와 함께 이전 SISO 복호기에서 발생한 사전 정보 값을 함께 사용하여 복호가 수행되는 과정이 반복되고, 이전 복호기의 결과 값과 비교하여 추가된 부가 정보들에 대한 복호를 반복하여 결과 값에 대한 신뢰도를 증가시킨다. 이 과정에서 인터리버는 연립된 오류를 분산시키는 역할을 수행한다. 인터리버의 종류로는 블록 인터리버, Helical 인터리버, S-랜덤 인터리버 등이 사용되고 있다. 두 개의 SISO 복호기에서 지정된 횟수만큼 반복 복호로 얻는 결과 값은 경관정 과정을 거쳐 최종 결과 값으로 출력된다. 그림 1은 터보 복호기의 기본 구조를 나타낸다[7].

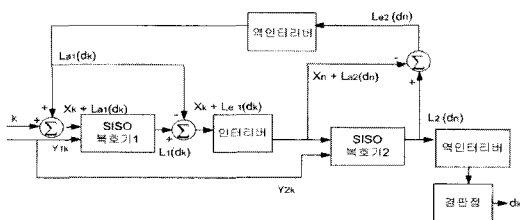


그림 1. 터보 복호기 구조

한편, SISO 복호기의 구현을 위하여 MAP 복호 알고리즘이 제안되었다. MAP 복호 알고리즘은 입력 데이터의 송신 확률에 따른 MAP 관정으로 복호를 수행한다. MAP 복호는 수신된 정보 비트의 LLR(Log Likelihood Ratio) 값으로 결과 값을 예측하는 방식으로, 식 (2-1)과 같이 상태 메트릭과 가지 메트릭의 조합으로 정의할 수 있다[8]. 식 (2-1)에서,  $\alpha_k(m)$ 은 현재 상태 이전의 입력 정보 비트들에 대한 확률이며 순방향 메트릭 (Forward State Metric)으로 정의 된다.  $\beta_{k+1}(S_j(m))$ 은 현재의 상태와 입력에 의해 결정되는 현재 상태 이전의 복호 상태에 대한 확률이며 역방향 메트릭(Backward State Metric)으로 정의된다. 이때  $S_j^i(m)$ 은 현재 상태  $S_k$ 와 입력  $d_k=i$ 에 의해 정의되는 다음 상태를 나타낸다.  $\delta_k(m)$ 은 현재 상태에서 얻을 수 있는 '0'과 '1'의 입력에 대한 확률로 가지 메트릭(Branch Metric)으로 정의된다.

$$L(d_k) = \log \frac{\sum_m \alpha_k(m) \delta_k^1(m) \beta_{k+1}(S_j^1(m))}{\sum_m \alpha_k(m) \delta_k^0(m) \beta_{k+1}(S_j^0(m))} \quad \text{식 (2-1)}$$

그림 2는 하나의 SISO 복호기에서 사용되는 입력과 출력 신호를 보인다. SISO 복호기는 입력으로 사용되는 사전정보  $L_a(d)$ 와 정보 비트  $x_k$ , 패리티 비트  $y_k$ 을 통해 가지 메트릭  $\delta_k(m)$ 을 먼저 구하고 순방향 상태 메트릭  $\alpha_k(m)$ 을 계산한다. 그리고 역방향 상태 매트릭  $\beta_{k+1}(m)$ 을 계산하면서 LLR 판정을 함께 수행한다.

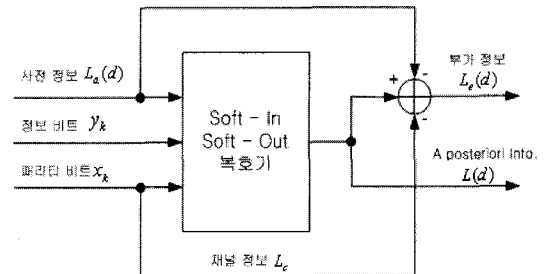


그림 2. SISO 복호기 입출력 값

식 (2-1)에서 보듯이 MAP 방식은 곱셈 연산이 많아 하드웨어로 구현할 경우 매우 복잡한 구조를 갖게 된다. Log-MAP 알고리즘은 로그 차원에서 곱셈 연산이 덧셈 연산으로 변환된다는 점을 이용

하여 MAP 알고리즘을 로그 차원의 계산으로 변환시킨 방법이다. 본 연구에서는 하드웨어 구현을 용이하게 하기 위해 Log-MAP 알고리즘을 식 (2-2)를 이용하여 보정항을 사용하는 Max-Log-MAP 알고리즘으로 변형하여 터보 복호기를 구현하였다[8].

$$\max^*(\delta_1, \delta_2) = \max(\delta_1, \delta_2) + \ln(1 + e^{-|\delta_2 - \delta_1|})$$

식(2-2)

### III. 병렬 구조 터보 복호기

저전력 시스템을 설계하기 위해서는 각각의 설계 단계마다 적용 가능한 설계 기법이 연구되었다. 일반적으로 시스템을 설계하는 과정은 시스템의 알고리즘 분석과 구조를 설계하는 단계와 논리 및 물리적 설계 단계로 나뉘어진다. 각 단계별 효과를 비교하였을 때, 알고리즘과 구조 분석 단계에서의 저전력 설계가 가장 큰 효과를 나타낸다고 알려져 있다[6]. 구조 수준의 저전력 기법 중 하나인 병렬 구조 방식은 효율적으로 동작 전압을 낮출 수 있다. 그림 3은 병렬 구조 방식을 도입하여 단일 입출력 구조를 다중 입출력 구조로 전환할 경우의 구조를 보인다[9].

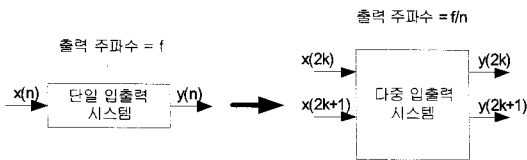


그림 3. 단일 입출력 시스템에서 다중 입출력 시스템으로의 변환.

그림 3의 병렬 구조 방식을 터보 복호기에 적용할 경우 전체 동작 주파수의 감소로 인해 전력 소모의 감소를 기대할 수 있다. SISO 복호기가 직렬로 연결된 기존의 터보 복호기에 병렬 구조를 적용시킬 경우, 동작 주파수를 반으로 낮출 수 있다. 결과적으로 1/2의 동작 주파수에서도 같은 throughput을 얻을 수 있다. 반복 복호에 필요한 SISO 연산기들을 병렬로 사용하는 경우, 각각의 SISO 복호기 동작을 위한 메모리 요구량의 증가에 대한 고려와 두 SISO 복호기에 의한 데이터 액세스 충돌을 방지해야 한다[10].

#### 3.1. 제안된 병렬 구조의 터보 복호기

그림 4는 본 논문에서 제안한 병렬 구조의 터보

복호기 구조를 보인다[10]. 2개의 복호기에 입력 정보와 패리티 정보가 동시에 입력된다. 직렬로 동작하는 SISO 복호기에서와 같이 인터리버와 역인터리버가 필요하며, 각각의 SISO 복호기 출력은 인터리버 또는 역인터리버에 저장이 가능하다. 기본적으로 한 프레임을 복호하기 위해서는 2회의 SISO 복호가 필요하기 때문에, 제한한 병렬구조 터보 복호기에서도 각각의 SISO 복호기는 한 프레임의 데이터에 대해 2회의 복호 복호를 수행한다.

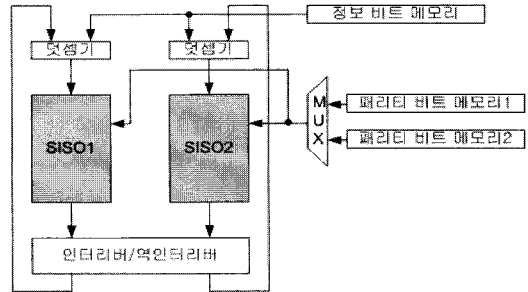


그림 4. 2단 병렬 구조의 터보 복호기.

그림 5는 제안된 병렬 구조 복호기에서 디코딩 과정을 보인다. 첫 번째 클럭 사이클에서는 (T = 0) k번째 입력 비트와 패리티 비트와 함께 이전 SISO 복호기에서 출력된 사전정보를 입력으로 하여 복호를 수행한다. SISO1과 SISO2는 복호를 수행한 뒤에 결과값을 인터리버에 저장하고 SISO 복호기와 역인터리버를 초기화한다. 두 번째 클럭 사이클에서는 (T = 1) 첫 번째 클럭 사이클에서 복호된 결과값과 패리티 비트를 이용하여 복호를 수행한다. SISO 복호기가 인터리버의 데이터를 모두 읽었을 경우 인터리버에서 종료 신호를 생성하며, 복호가 완료된 뒤에 SISO 복호기와 인터리버를 초기화한다. SISO1에서는 k번째 입력 비트와 패리티 비트가 사용되며, SISO2에서는 프레임 크기의 1/2인 l이 추가된 k+1번째 입력 비트와 패리티 비트가 사용된다.

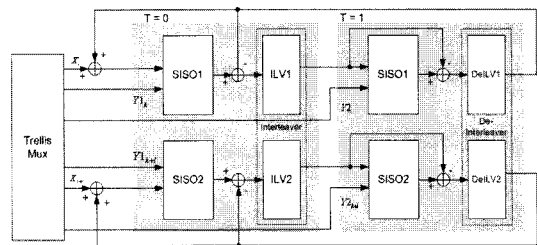


그림 5. 병렬 구조 터보 복호기의 데이터 흐름.

### 3.2. 병렬 구조를 위한 인터리버의 설계

병렬구조의 터보 복호기를 설계할 경우, 2개의 SISO 복호기가 동시에 인터리버에 대한 데이터 액세스가 가능해야 하므로 기존의 터보 복호기에서 사용되는 인터리버의 구조는 사용이 어렵다[11][12]. 2개의 SISO 복호기가 하나의 인터리버를 공유하기 위해서는 2개의 복호기가 동시에 인터리버에 데이터 액세스할 때 발생할 수 있는 충돌 상황을 방지해야 한다. 본 논문에서는 기존의 인터리버 메모리 크기가 동일하며 같은 오류 분산 효과를 나타낼 수 있는 새로운 인터리버 구조를 제안하였다. 그림 6은 제안된 인터리버의 개요를 보이며, SISO 복호기는 두 부분으로 분리된 인터리버에 데이터 액세스가 가능하다. 인터리버에 사용되는 주소 테이블은 s-random 알고리즘을 적용하여 추출되었으며, 인터리버1과 인터리버2의 동작 순서는 블록 사이즈가 2인 Helical 인터리버 동작과 같다. SISO 복호기와 인터리버 사이의 충돌을 방지하기 위해 제어가 추가적으로 요구된다.

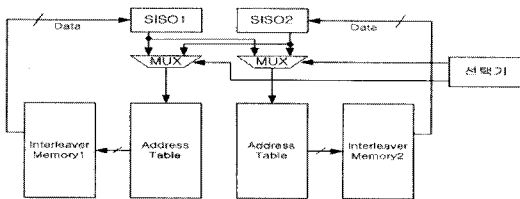


그림 6. 병렬 터보 복호기를 위한 인터리버 구조

그림 7은 인터리버의 상태와 SISO 복호기의 인터리버 데이터 액세스 패턴을 보인다. SISO1 복호기의 데이터 액세스 패턴은 {인터리버1, 인터리버2, 인터리버2, 인터리버1}의 형태를 갖으며, 데이터 액세스 주기는 4 사이클이 된다. 인터리버는 클럭의 positive edge에서 상태가 바뀌며, SISO 복호기는 입력 클럭의 negative edge에서 데이터 액세스를 하도록 설계되었다.

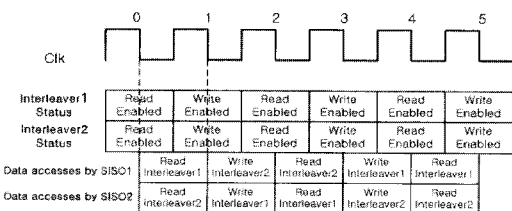


그림 7. 인터리버 - SISO 복호기의 상태 변화.

### IV. 실험 결과

복호기의 성능 측정을 위한 소프트웨어 시뮬레이션은 UNIX 환경에서 C 언어로 작성하여 수행하였다. 시스템의 입력은 프로그램 자체내에 랜덤 분포를 갖는 데이터 생성기를 설정하고 변조 방식으로는 BPSK(Binary Phase Shift Key)를 사용하였으며, 채널 모델링은 AWGN을 채택하였다. 하드웨어 모델 구현을 위해 Synopsys사의 Design Analyzer와 하이닉스의 0.35 $\mu$ m 공정 라이브러리를 이용하여 VHDL 코드를 합성하고 전력 소모를 측정하였다. 표 1은 본 실험에서 사용된 조건들을 보인다.

표 1. 실험 조건

파라미터	값
구축장 크기	3
메모리 길이	2
생성 다항식	7/5
프레임 크기	1024
부호율	1/3
채널 변조 방식	AWGN, BPSK

그림 8은 제안된 터보 복호기의 하드웨어 구조를 보인다. 2개의 SISO 복호기와 인터리버 제어기, 메모리, 주소 테이블과 최종 복호 된 결과 값을 경판정하기 위한 블록으로 구성되며 이들 블록은 전체 터보 복호 제어기에서 동작을 제어한다. 인터리버 제어기는 인터리버의 정상적 동작을 위한 제어 신호를 인가한다. 인터리버의 상태에 따라 읽기/쓰기를 제어하고, SISO 복호기에 전송되는 인터리버의 어드레스 출력을 제어한다. 인터리버 주소 테이블에서 얻게되는 어드레스 출력은 SISO 복호기로 전달되어, SISO 복호기에서 직접 인터리버 메모리에 데이터 액세스가 가능하도록 한다.

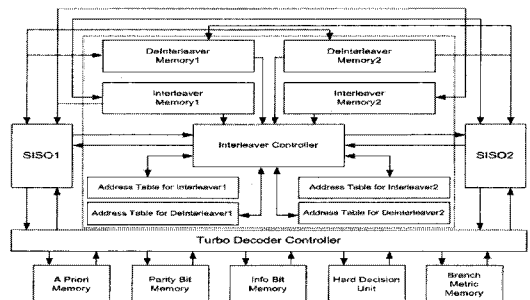


그림 8. VHDL로 구현된 제안된 터보 복호기.

표 2는 기존의 Max-Log-MAP을 적용한 터보 복호기와 제안된 터보 복호기를 면적과 전력 소모의 측면에서 비교한 결과값을 보인다. 제안된 터보 복호기에서 약 40 %정도 전력 소모가 감소되었고 전체 면적은 약 1% 증가하였다. 이는 병렬화에 따른 추가된 제어기 면적에서 비롯된다.

표 2. 면적 및 전력 소모 비교

	회로블록	기존방식	제안된 방식	비교 (%)
면적*	메모리	321,500	321,500	0
	연산기	35,392	39,396	+ 11.3
	전체	356,692	360,896	+ 1.1
전력소모 (mW)	메모리	107.5	60.5	- 43.7
	연산기	85.8	54.9	- 36.0
	전체	193.3	115.4	- 40.3

\* Unit gate : Two-input NAND (size : 1)

그림 9는 제안된 터보 복호기와 기존에 구현된 Log-MAP, Max-Log-MAP 터보 복호기의 BER 성능 결과를 보인다. 각각의 알고리즘은 총 1,000,000 회의 반복 복호과정을 수행하였다. Par\_MAP 으로 테이블이 붙은 플롯이 제안된 구조에서의 BER 성능으로 상대적으로 우수한 결과를 보인다.

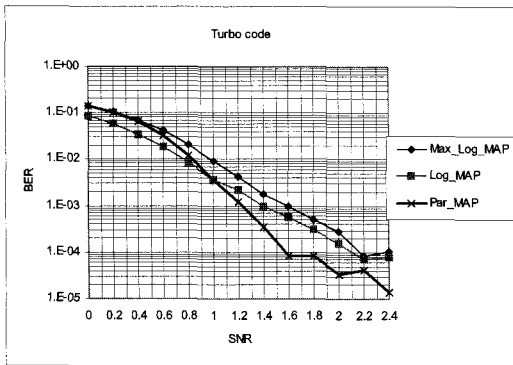


그림 9. Log-MAP, Max-Log-MAP과 제안된 터보 복호기의 복호 성능 비교

### V. 결론

본 논문에서는 저전력 터보 복호기 구현을 위해 SISO 복호기가 병렬로 동작하는 구조를 제시하였다. SISO 복호기를 병렬로 연결하였을 경우 추가되어야 하는 인터리버 메모리의 요구량을 최소화하기 위해 새로운 인터리버 구조를 제안하였으며, 제안된 인터리버와 병렬 구조의 채택 결과 연산기 면적은 인터리버 제어부의 추가 설계로 약 1% 정도 증가하여 무시할만한 반면 동작 주파수의 감소로 전체 동작에서의 전력은 감소된다. 메모리의 전체 면적에 대하여 SISO 복호기의 개수와 인터리버의 크기가 변하지 않았기 때문에 기본적인 Max-Log-MAP이 구현된 터보 복호기와 비교하여 면적 증가량이 크지 않으며, 기존의 방식과 비교하였을 경우 최종적으로 약 40% 정도의 전력 소모가 감소한다. 추후 과제로 완벽한 병렬 구조의 저전력 터보 복호기를 구현하기 위해서는 병렬 복호를 수행할 경우 입력단의 정보 비트와 패리티 비트를 효율적으로 처리하기 위한 방법에 대한 연구가 필요하며 제안된 인터리버의 오류 분산도에 대한 수학적 증명도 있어야 한다.

### 감사의 글

본 연구는 정보통신부 및 정보통신연구진흥원의 대학 IT 연구센터 육성지원사업의 연구 결과로 수행되었습니다. 설계에 있어 IDEC에서 지원한 CAD 툴을 사용하였습니다.

### 참고문헌

- [1] 이문호, “비동기식 IMT-2000의 채널 부호화,” 한국통신학회 학회지, 제 14권, 9호, pp.170-187, 1996년 9월.
- [2] C. Berrou, A. Glavieux, and P. Thitimajshima, “Near Shannon Limit Error-correcting Coding and Decoding: Turbo-Codes(1),” in Proc. ICC’93, Geneva, Switzerland, pp. 1064-1070, May 1993.
- [3] C. Shannon, “A Mathematical Theory of Information,” Bell System Technical Journal, Vol. 27, pp. 379-423, July 1948.
- [4] C. Su, C. Tsui, and A. Despain, “Low-Power Architecture Design and Compilation Technique for High Performance Processors,” in Proc. COMPCON’94, pp. 489-498, Feb. 1994.
- [5] Y. Rose, L. Shu, and P. Marc, “Two Simple Stopping Criteria for Turbo Decoding,” IEEE Trans. Communications, Vol. 47, No. 8, pp. 1117-1120, Aug. 1999.
- [6] S. Hong and W. Stark, “Design and Implementation of a Low Complexity VLSI

Turbo-Code Decoder Architecture for Low Energy Mobile Wireless Communications,” Journal of VLSI Signal Processing Systems, Vol. 24, pp. 43-57, Feb. 2000.

- [7] S. Barbulescu and S. Pietrobon, “Turbo Codes: A Tutorial on a New Class of Powerful Error Correcting Coding Schemes, Part2: Decoder Design and Performance,” Journal of Electrical & Electronics Engineering Australia Vol. 19, No. 3, pp. 143-152, Sep. 1999.
- [8] P. Jason and H. Lajos, “Comparative Study of Turbo Decoding Techniques: An Overview,” IEEE Trans. Vehicular Technology, Vol. 49, No. 6, pp. 2208-2238, Nov. 2000.
- [9] K. Parhi, VLSI Digital Signal Processing Systems. Wiley-Interscience Pub., 1999.
- [10] Z. Wang, H. Suzuki, and K. Parhi, “VLSI Implementation Issues of Turbo Decoder Design for Wireless Applications,” in Proc. IEEE Workshop on Signal Processing Systems, Taipei, Taiwan, pp. 503-512, Oct. 1999.
- [11] A. Giulietti, L. Perre, and M. Strum, “Parallel Turbo Coding Interleavers: Avoiding Collisions in Accesses to Storage Elements,” Electronics Letters, IEE, Vol. 38, No. 5, pp. 232-234, Feb. 2002.
- [12] S. Pietrobon, “Efficient Implementation of Continuous MAP Decoders and a New Synchronisation Technique for Turbo Decoders,” in Proc. Int. Symp. Info. Theory and it’s Applications, pp. 586-589, Victoria, BC, Canada, Sep. 1996.

**황 선 영(Sun-young Hwang)**

정회원



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 주식

회사 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

2002년 4월~2004년 2월 서강대학교 정보통신대학 원장

1989년 3월~현재 서강대학교 전자공학과 교수

<관심분야> 컴퓨터 시스템, SoC 설계 및 framework 구성, Embedded System 설계 등

**이 희 진(Hee-jin Lee)**

정회원



2002년 2월 서강대학교 전자공학과 졸업

2004년 8월 서강대학교 전자공학과 석사학위 취득

2004년 9월~현재 삼성전기 근무중

<관심분야> 전자공학, 통신공학