

논문 2010-47SD-5-5

IEEE 802.11n 무선 랜 표준용 LDPC 복호기 설계

(A Design of LDPC Decoder for IEEE 802.11n Wireless LAN)

정 상 혁*, 신 경 욱**

(Sang-Hyeok Jung and Kyung-Wook Shin)

요 약

본 논문에서는 IEEE 802.11n 무선 랜 표준용 LDPC 복호기 프로세서를 설계하였다. 설계된 프로세서는 IEEE 802.11n 표준의 블록길이 1,944와 부호화율 1/2의 패리티 검사 행렬을 지원하며, 하드웨어 감소를 위해 최소합 알고리즘과 layered 구조를 적용하였다. 최소합 알고리즘의 특징을 이용한 검사노드 메모리 최소화 방법을 고안하여 적용하였으며, 이를 통해 기존방법의 메모리 크기의 25%만을 사용하여 구현하였다. 설계된 프로세서를 0.35- μ m CMOS 셀 라이브러리로 합성한 결과, 200,400 게이트와 19,400 비트의 메모리로 구현되었으며, 80 MHz@2.5V로 동작하여 약 135 Mbps의 성능을 갖는다. 설계된 회로는 FPGA 구현을 통해 하드웨어 동작 검증과 복호성능을 분석하였으며, 이를 통해 설계된 LDPC 복호기의 유용성을 입증하였다.

Abstract

This paper describes a LDPC decoder for IEEE 802.11n wireless LAN standard. The designed processor supports parity check matrix for block length of 1,944 and code rate of 1/2 in IEEE 802.11n standard. To reduce hardware complexity, the min-sum algorithm and layered decoding architecture are adopted. A novel memory reduction technique suitable for min-sum algorithm was devised, and our design reduces memory size to 25% of conventional method. The LDPC decoder processor synthesized with a 0.35- μ m CMOS cell library has 200,400 gates and memory of 19,400 bits, and the estimated throughput is about 135 Mbps at 80 MHz@2.5v. The designed processor is verified by FPGA implementation and BER evaluation to validate the usefulness as a LDPC decoder.

Keywords : LDPC(Low-Density Parity-Check), IEEE 802.11n, WLAN, min-sum algorithm

I. 서 론

무선 및 이동통신 시스템에서는 무선채널의 페이딩, 음영현상, 경로손실 등 다양한 원인에 의한 채널잡음이 발생하며, 이러한 잡음에 대응하기 위해 다양한 채널 부호화 기법들이 사용된다. 기존의 시스템에서는 오류

정정(error correction)을 위해 길쌈부호와 터보부호를 사용하여 왔으나, 차세대 고속 이동통신 시스템에서는 채널용량의 한계에 근접한 높은 성능과 함께 고속 복호가 가능한 새로운 채널코딩 기법이 요구된다.

최근, 차세대 오류정정 부호화 방식으로 LDPC(Low-Density Parity-Check) 부호가 많은 관심을 받고 있다. LDPC 부호는 1962년 로버트 갤러거(R. Gallager)에 의해 제안된 채널부호화 기법이다.^[1] 당시의 기술로는 구현이 어려워 관심을 받지 못하였으나, 그 후 정보기술과 반도체 기술의 발달에 힘입어 1990년대부터 재조명되고 있으며, 반복적 복호를 사용하면서도 복잡도가 크게 증가하지 않는 LDPC 부호의 특성 및 생성방법에 대한 연구가 활발히 진행되고 있다.^[2-4] LDPC 부호는

* 정회원, 고등기술연구원 로봇생산기술센터
(Center for Robot Technology and Manufacturing,
Institute for Advanced Engineering)

** 정회원-교신저자, 금오공과대학교 전자공학부
(School of Electronic Engineering, Kumoh National
Institute of Technology)

※ 본 논문은 ETRI 시스템반도체진흥센터의 시스템반
도체 융복합형 설계인재양성사업의 결과임

접수일자: 2010년1월26일, 수정완료일: 2010년4월14일

유럽의 디지털 위성방송 규격 DVB-S2^[5], 차세대 무선 랜 규격 IEEE 802.11n^[6], 모바일 WiMAX 규격 IEEE 802.16e^[7] 그리고 10-Gbps 이더넷 표준 IEEE 802.3an^[8] 등에서 채널 부호화 방법으로 채택되고 있다.

LDPC 부호의 복호는 태너 그래프(Tanner graph) 상에서 변수노드(variable node)와 검사노드(check node) 사이의 반복적인 메시지 전달로 수행될 수 있다^[9]. 반복적인 메시지 전달을 통한 복호방법은 합곱 알고리즘(Sum-Product Algorithm; SPA)^[10], Log-Likelihood Ratio-합곱 알고리즘(LLR-SPA)^[11], 최소합 알고리즘(Min-Sum Algorithm; MSA)^[12] 등이 있다. 변수노드와 검사노드 사이의 메시지 전달과정에서 각 노드는 새로운 정보를 저장할 메모리를 필요로 하며, 블록길이가 긴 부호일수록 큰 용량의 메모리가 필요하여 복호기 면적의 많은 부분을 차지하게 된다.

본 논문에서는 IEEE 802.11n 무선 랜 표준용 LDPC 복호기를 최소합 알고리즘을 이용하여 설계하였으며, 검사노드 메모리를 최소화할 수 있는 새로운 방법을 고안하여 설계에 적용하였다.

II. LDPC 복호 알고리즘

LDPC 부호의 복호는 패리티 검사 행렬(Parity Check Matrix; PCM)에 의해 구성되는 그림 1과 같은 태너 그래프 상의 변수노드와 검사노드 사이에서 반복적인 메시지 전달과정에 의해 이루어진다. 대표적인 LDPC 복호 알고리즘으로 SPA^[10], SPA의 계산 복잡도를 감소시킨 LLR-SPA^[11], LLR-SPA를 근사화하여 계산 복잡도를 더욱 감소시킨 MSA^[12] 등이 있다.

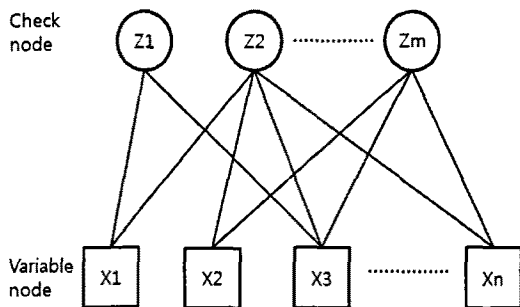


그림 1. 태너 그래프
Fig. 1. Tanner graph.

1. 합-곱 알고리즘(SPA)^[10]

SPA는 검사노드와 변수노드 사이의 메시지 전달과

정에서 합과 곱의 연산이 이루어지며, 가장 널리 이용되는 성능이 우수한 LDPC 복호 알고리즘이다. SPA의 검사노드와 변수노드에서의 관계식은 다음과 같다.

$$L_{j \rightarrow i} = 2 \tanh^{-1} \cdot \prod_{i' \in \omega(j) \setminus \{i\}} \tanh\left(\frac{1}{2} L_{i' \rightarrow j}\right) \quad (1)$$

$$L_{i \rightarrow j} = F_i + \sum_{j' \in \Omega(i) \setminus \{j\}} L_{j' \rightarrow i} \quad (2)$$

각각의 변수노드 i 에 대하여

$$z_i = F_i + \sum_{j \in \Omega(i)} L_{j \rightarrow i} \quad (3)$$

$$\hat{c} = \begin{cases} 0, & z_i \geq 0 \\ 1, & z_i < 0 \end{cases} \quad (4)$$

으로 $\hat{c} = [\hat{c}_i]$ 을 구하고 $\hat{c} \cdot H^T$ 를 계산한다.

$\hat{c} \cdot H^T = 0$ 이면 \hat{c} 를 유효한 부호어로 인식하여 복호를 종료하며, $\hat{c} \cdot H^T \neq 0$ 이면 지금까지 수행된 반복복호 횟수를 지정된 최대 반복복호 횟수와 비교하여 작으면 반복복호 과정을 계속하고, 그렇지 않으면 복호 실패를 선언하고 종료한다.

2. LLR-SPA^[11]

SPA의 연산과정에는 $\tanh()$ 함수와 곱셈이 포함되어 있으므로 연산 복잡도가 크다. LLR-SPA는 근사화를 통해 연산복잡도를 줄인 알고리즘이다. LLR-SPA의 검사노드와 변수노드에서의 관계식은 식(5)와 식(6)과 같으며, α_{ij} , β_{ij} , 함수 $\phi(x)$ 는 식(7)~(9)와 같이 정의된다. 함수 $\phi(x)$ 는 일반적으로 LUT(Look-Up Table)로 구현된다.

$$L_{j \rightarrow i} = \left(\prod_{i' \in \omega(j) \setminus \{i\}} \alpha_{i'j} \right) \cdot \phi\left(\sum_{i' \in \omega(j) \setminus \{i\}} \phi(\beta_{i'j}) \right) \quad (5)$$

$$L_{i \rightarrow j} = F_i + \sum_{j' \in \Omega(i) \setminus \{j\}} L_{j' \rightarrow i} \quad (6)$$

$$\alpha_{ij} = \text{sign}(L_{i \rightarrow j}) \quad (7)$$

$$\beta_{ij} = |L_{i \rightarrow j}| \quad (8)$$

$$\phi(x) = -\ln \left[\tanh\left(\frac{1}{2}x\right) \right] = \ln \frac{e^x + 1}{e^x - 1} \quad (9)$$

3. 최소합 알고리즘(MSA)^[12]

함수 $\phi(x)$ 는 x 가 작을 때 큰 값을 가지며, x 가 커지면 거의 0에 근접하는 특성을 가진다. 식(5)에서 함수 $\phi(x)$ 의 합은 x 값이 매우 작을 때 가장 큰 영향을 주므로 $\sum \phi(X) \approx \phi(\min(X))$ 로 근사화될 수 있다. 또한, 식(9)의 함수 $\phi(x)$ 는 x 가 0 보다 클 때 역함수와 본 함수의 값이 같으므로 $\phi(\phi(\min(X))) = \min(X)$ 로 변환되어 식(10)과 같이 근사화될 수 있다.

$$\phi\left(\sum_{i \in \omega(j)(i)} \phi(\beta_{ij})\right) \approx \phi\left(\phi\left(\min_{i \in \omega(j)(i)} \beta_{ij}\right)\right) = \min_{i \in \omega(j)(i)} \beta_{ij} \quad (10)$$

따라서 식(10)에 의해 식(5)는 식(11)과 같이 근사화될 수 있으며, 이를 최소합 알고리즘이라 한다. MSA는 복잡한 $\phi(x)$ 의 연산과정을 피할 수 있고, LUT를 사용하지 않으므로 연산 복잡도와 하드웨어 면적이 감소하는 장점을 갖는다.

$$L_{j \rightarrow i} \approx \left(\prod_{i' \in (j) \setminus (i)} \alpha_{ij'}\right) \cdot \min_{i' \in \omega(j) \setminus i} \beta_{ij'} \quad (11)$$

III. IEEE 802.11n용 LDPC 복호기 설계

IEEE 802.11n 표준의 블록길이 1,944와 부호화율 1/2을 지원하는 LDPC 복호기 프로세서 LDPC_D11n을 최소합 알고리즘 기반의 layered 복호방법과 검사노드 메모리 최소화 방법을 적용하여 설계하였다.

1. 전체 구조

설계된 LDPC_D11n 프로세서의 내부 구조는 그림 2와 같으며, 최소합 알고리즘의 복호연산을 수행하는 81개의 DFU(Decoding Function Unit), PCM을 저장하는 H-ROM, PCM에 따라 데이터를 분배하는 permuter, 검사노드와 변수노드의 메시지를 저장하는 메모리, 그리고 전체 동작을 제어하는 제어블록으로 구성된다.

IEEE 802.11n의 블록길이 1,944와 부호화율 1/2의 PCM은 부행렬(sub-matrix)의 크기가 81×81이며, QC(Quasi-Cyclic)-LDPC 방식으로 구성된 PCM의 대부분은 영(zero) 행렬로 이루어져 있다. 영 행렬은 검사노드와 변수노드 사이에 어떠한 연결도 형성하지 않으며 연산도 존재하지 않는다. 따라서 LDPC 복호기는 영 행렬을 무시하고 연산을 수행할 수 있다. 본 논문에서는 영 행렬을 효율적으로 무시하기 위해 부행렬 단위로

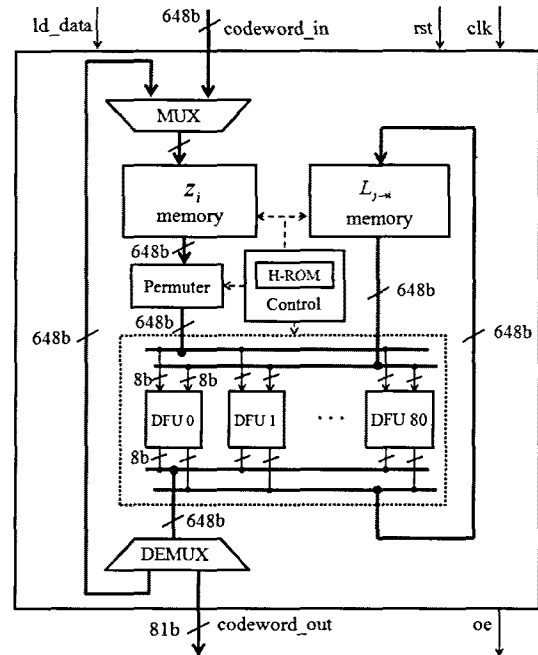


그림 2. 설계된 LDPC_D11n 프로세서의 전체 구조
Fig. 2. Architecture of designed LDPC_D11n processor.

연산을 수행하는 블록-시리얼(block-serial) 구조로 설계하여 81개의 LLR이 동시에 연산되도록 하였다.

변수노드에서 계산된 메시지를 저장하는 검사노드 메모리($L_{j \rightarrow i}$ memory)는 듀얼포트 메모리로 구현된다. 본 논문에서 사용된 PCM은 블록길이 1,944이며, LLR을 8 비트로 표현한다. 따라서 변수노드 메모리(z_i memory)의 용량은 1,944×8=15,552비트이며, 검사노드 메모리도 변수노드 메모리와 동일한 15,552 비트가 필요하다. 본 논문에서는 최소합 알고리즘의 특징을 이용하여 검사노드 메모리 용량을 1/4로 감소시키는 방법을 고안하여 적용하였으며, 이에 대해서는 3절에서 상세히 설명된다.

그림 3은 LDPC_D11n 프로세서의 동작 타이밍도이다. rst 신호에 의해 메모리와 레지스터가 초기화되며, ld_data 신호가 인가되는 24 클록 주기 동안 부호어가 입력되어 메모리에 저장된다. 하나의 레이어 복호에 12 클록 주기가 소요되며, 총 144 클록 주기에 걸쳐 12개의 레이어에 대한 복호가 완료되면 1회의 반복복호가 완료된다. 반복복호는 총 8회 진행되어 오류가 정정되며, 복호가 완료되어 오류가 정정된 부호어는 24 클록 주기에 걸쳐 출력된다. 그림 3의 동작 타이밍도에서 8회의 반복복호에 1,164(= 144×8 + 12) 클록 주기가 소요되며, 추가적으로 발생하는 12 클록 주기는 1개 레

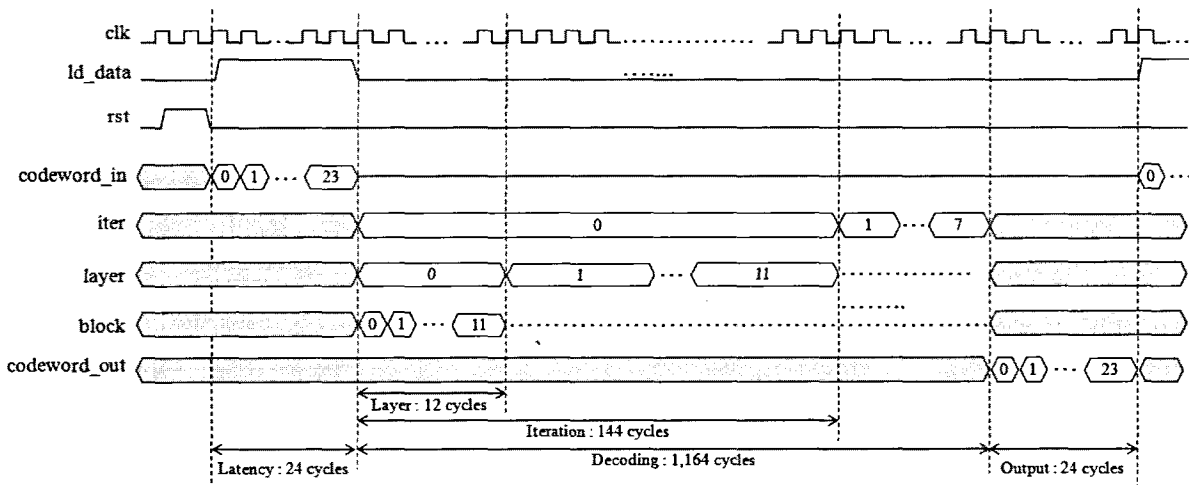


그림 3. LDPC_D11n 프로세서의 동작 타이밍도
Fig. 3. Timing diagram of LDPC_D11n processor.

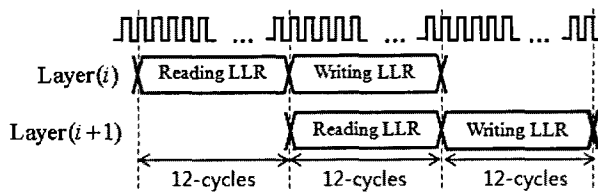


그림 4. DFU의 읽기와 쓰기 동작 타이밍도
Fig. 4. Timing diagram of DFU reading and writing.

이러한 데이터 읽기/쓰기에 소요되는 사이클과 동일한 값이다. DFU는 각 노드에 해당하는 정보를 메모리로부터 읽어와 복호연산을 수행하고 다시 메모리에 쓰는 동작을 수행하며, 읽기동작과 쓰기동작을 중첩하여 수행한다. DFU의 읽기와 쓰기동작의 타이밍은 그림 4와 같으며, i 번째 레이어의 쓰기동작과 $(i+1)$ 번째 레이어의 읽기동작은 동시에 이루어진다.

2. DFU의 구조 및 동작

DFU는 오류가 포함되어 있는 입력 부호어의 오류를 정정하기 위해 자신을 제외한 나머지 부호어들의 LLR을 취합하여 원래의 부호어에 가장 가까운 LLR을 예측하는 기능을 수행한다. 본 논문에서는 최소합 알고리즘을 기반으로 한 layered 구조의 DFU를 설계하였다.

설계된 DFU의 구조는 그림 5와 같으며, 최소값 검출기(Min_det), 부호비트 누산기(XOR), 데이터의 지연을 위한 FIFO(First-In First-Out) 메모리, 덧셈기, 뺄셈기, 비교기 그리고 수체계 변환기(TC_SM, SM_TC) 등으로 구성된다. DFU 내부의 LLR 값은 8비트로 근사화되고 부호와 크기가 분리되어 계산된다.

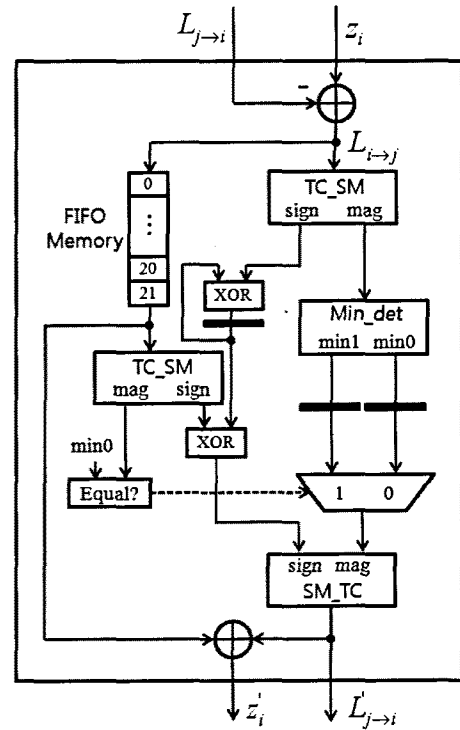


그림 5. 설계된 DFU의 구조
Fig. 5. Architecture of designed DFU.

DFU의 연산과정은 다음과 같다. 결정변수 z_i 와 LLR 값 $L_{j \rightarrow i}$ 를 입력받아 $L_{i \rightarrow j}$ 를 계산한 후, 계산된 $L_{i \rightarrow j}$ 는 부호와 크기로 분리되어 부호는 순차적으로 계산되는 $L_{i \rightarrow j}$ 의 부호들과 누적하여 곱셈(XOR 연산) 된다. 분리된 크기는 순차적으로 입력되어 계산되는 $|L_{i \rightarrow j}|$ 들과 비교되어 최소값과 준최소값이 검출된 후 레지스터에 저장된다.

순차적으로 계산된 $L_{i \rightarrow j}$ 는 FIFO에 누적되어 입력된 수만큼 지연된 후 부호와 크기로 분리된다. 분리된 크기 $|L_{i \rightarrow j}|$ 에서 검출된 최소값 $\min 0$ 과 비교하여 두 값이 같으면 준최소값 $\min 1$ 을 새로운 $|L_{j \rightarrow i}|$ 으로 결정하고, 다르면 최소값 $\min 0$ 을 새로운 $|L_{j \rightarrow i}|$ 으로 결정한다. 식 (11)에 따르면, i 번째 $|L_{j \rightarrow i}|$ 를 결정할 때에는 i 번째 $|L_{i \rightarrow j}|$ 를 제외한 나머지 $i' \in \omega(j) \setminus \{i\}$ 번째 $|L_{i \rightarrow j}|$ 의 크기 중 가장 작은 값을 선택하므로, 전체 $i \in \omega(j)$ 중 가장 작은 $|L_{i \rightarrow j}|$ 가 발생한 위치에는 준최소값이 갱신되고 나머지 $i' \in \omega(j) \setminus \{i\}$ 번째 $|L_{j \rightarrow i}|$ 에는 최소값이 갱신된다. 또한 부호의 누적 곱셈도 동일한 원리로 모든 부호를 누적 곱셈한 다음 i 번째 $L_{j \rightarrow i}$ 의 부호를 계산하기 위해 i 번째 $L_{i \rightarrow j}$ 의 부호를 한번 더 $i' \in \omega(j) \setminus \{i\}$ 번째 부호들만 누적 곱셈한 결과와 동일하게 만든다. 결정된 $|L_{j \rightarrow i}|$ 의 크기와 부호를 이용하여 2의 보수 수체계의 $L_{j \rightarrow i}$ 를 계산하여 출력한다. 결정변수 z_i 는 FIFO에 의해 지연된 $L_{i \rightarrow j}$ 와 DFU 연산을 통해 계산된 $L_{j \rightarrow i}$ 를 더해서 계산된다.

DFU의 동작은 메모리로부터 z_i 와 $L_{j \rightarrow i}$ 를 읽어오는 읽기동작, 연산된 z_i 와 $L_{j \rightarrow i}$ 를 메모리에 저장하는 쓰기동작, 그리고 연산과정으로 이루어진다. 읽기동작과 쓰기동작은 최초 1회를 제외하고는 항상 중첩되어 일어나며, $(i-1)$ 번째 레이어의 쓰기동작은 i 번째 레이어의 읽기동작과 동시에 일어난다. 연산과정은 읽기동작 동안 결정된 부호 누적 곱셈과 최소값/준최소값 검출의 결과를 결과값 레지스터로 저장하는 과정이다. 이와 같은 읽기/쓰기 중첩 동작에 의해 DFU 연산이 평균 12 사이클에 처리되어 새로운 z_i 와 $L_{j \rightarrow i}$ 가 계산되어 메모리에 갱신된다.

(1) 최소값 검출기

최소값 검출기는 순차적으로 입력되는 무부호(unsigned) 정수값들을 비교하여 가장 작은 최소값과 두 번째로 작은 준최소값을 검출해 낸다. 설계된 최소값 검출기는 그림 6과 같으며, 두 개의 비교기, 검출된 최소값/준최소값을 저장하는 레지스터 $\min 0$ 과 $\min 1$ 그리고 검출기를 제어하는 MUX로 구성된다. 두 개의 비교기는 검출기로 입력되는 값을 레지스터에 저장된 최소값 및 준최소값과 비교하여 en_m0 , en_m1 신호를 발생시킨다. en_m0 신호는 MUX를 제어하여 새롭게 갱

신될 값이 최소값인지 준최소값인지를 결정한다.

(2) 2의 보수/부호-크기(TC_SM) 변환기

TC_SM 변환기는 2의 보수 수체계의 값을 부호비트와 크기로 분리한다. 2의 보수 입력값이 양수인지 음수인지 판단하고, 만약 양수라면 부호는 0이 되고 크기는 입력된 2의 보수 값을 그대로 출력한다. 한편, 입력된 값이 음수라면 부호는 1이 되고 입력된 2의 보수를 식 (12)과 같이 반전하고 1을 더하여 크기 값을 계산한다.

$$|a| = \bar{a} + 1 \quad \text{단, } a < 0 \quad (12)$$

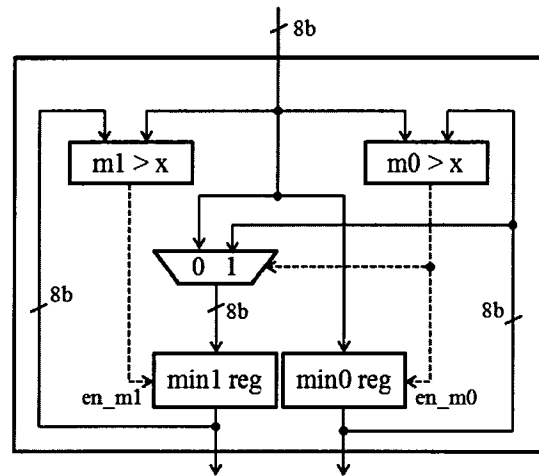


그림 6. 최소값 검출기
Fig. 6. Minimum value detector.

(3) 부호/크기-2의 보수 (SM_TC) 변환기

SM_TC 변환기는 부호-크기의 입력 받아 2의 보수 수체계로 변환한다. 입력된 부호가 양수인지 음수인지 판단하고, 부호가 양수라면 2의 보수 수체계의 MSB는 0이 되고 크기는 입력된 7비트의 크기를 그대로 출력한다. 한편, 입력된 부호가 음수라면 2의 보수 수체계의 MSB는 1이 되고 입력된 크기는 식(13)과 같이 반전하고 1을 더하여 변환된다.

$$a = \overline{|a|} + 1 \quad \text{단, } a < 0 \text{인 경우} \quad (13)$$

(4) H-ROM

H-ROM은 LDPC 복호에 사용되는 PCM의 정보를 저장하는 메모리이다. PCM은 IEEE 802.11n의 블록길이 1,944와 부호화율 1/2을 지원하며, 81×81 크기의 부행렬 288개로 이루어진 QC-LDPC 방식의 행렬이다.

288개의 부행렬은 202개의 영 행렬을 포함하고 있으며, 실제로 노드간의 연결을 형성하는 부행렬은 86개만 존재한다. 따라서 전체 행렬의 정보를 저장하지 않고 영 행렬을 제외한 86개 행렬의 오른쪽 순환 시프트 정보와 해당 행렬의 위치 정보만 저장하면 ROM의 크기를 줄일 수 있다.

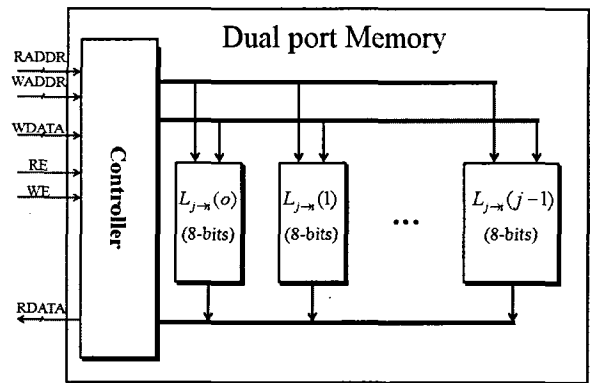
부행렬의 최대 오른쪽 순환 시프트는 79이며, 시프트 정보는 7 비트 ($2^6 < 79 < 2^7$)로 표현할 수 있다. PCM은 12개의 레이어로 구성되고, 각 레이어는 24개의 부행렬로 구성된다. 따라서 288개의 부행렬 정보를 모두 저장하는 직접적인 방법에서는 2,016 비트가 필요하다. 본 논문에서는 non-zero 부행렬들을 시프트 정보와 부행렬 번호만으로 식별하는 방법을 적용하였으며, 이를 통해 PCM이 1,032 (= 12 × 86) 비트로 저장되도록 하였다. 이와 같은 본 논문의 PCM 저장방법은 직접적인 방법에 비해 약 49%의 ROM 용량을 감소시켰다.

3. MSA기반 LDPC 복호기의 메모리 최소화 방법

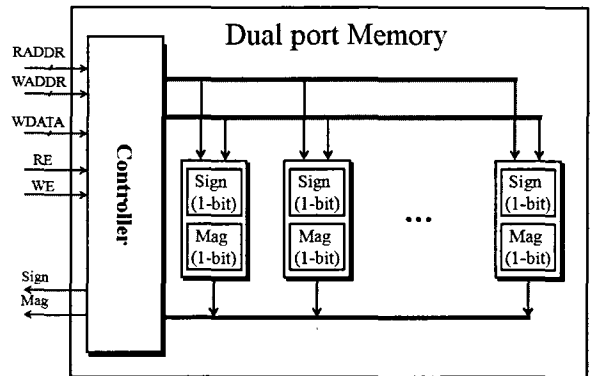
LDPC 복호기의 검사노드 메모리에는 수평단계의 연결과 $L_{j \rightarrow i}$ 가 저장된다. 이 때, $L_{j \rightarrow i}$ 의 크기 $|L_{j \rightarrow i}|$ 는 식(11)에 의해 β_{ij} 의 최소값으로 갱신되며, 최소값이 발생한 i 번째 $|L_{j \rightarrow i}|$ 은 최소값을 제외한 나머지 β_{ij} 의 최소값으로 갱신된다. 이는 $|L_{j \rightarrow i}|$ 가 다수의 최소값과 하나의 준최소값으로 갱신됨을 의미한다. 또한, 식(11)에서 $\prod_{i' \in \omega(j) \setminus \{i\}} \alpha_{i'j}$ 은 $L_{j \rightarrow i}$ 의 부호 비트로 사용되어 양수와 음수를 나타낸다. 따라서 $L_{j \rightarrow i}$ 은 최소값과 준최소값 그리고 각각의 부호만으로 갱신될 수 있으며, 식 (14)와 같은 4가지 경우로 정리될 수 있다.

$$L_{j \rightarrow i} = \begin{cases} + \min_{i' \in \omega(j) \setminus \{i\}} \beta_{i'j} \\ - \min_{i' \in \omega(j) \setminus \{i\}} \beta_{i'j} \\ + \min_{i' \in \omega(j)} \beta_{i'j} \\ - \min_{i' \in \omega(j)} \beta_{i'j} \end{cases} \quad (14)$$

본 논문에서는 이와 같은 고찰을 토대로 검사노드 메모리(그림 2에서 $L_{j \rightarrow i}$ 메모리)의 크기를 줄일 수 있는 효과적인 방법을 고안하여 적용하였다. 즉, 검출된 최소값과 준최소값을 그대로 저장하는 대신에, 최소값/준최소값의 여부와 부호를 2 비트로 표현하여 저장하는 방



(a) 기존의 검사노드 메모리 저장방식



(b) 제안된 검사노드 메모리 저장방식

그림 7. 검사노드 메모리 저장방식의 비교
Fig. 7. Comparison of check node memory contents.

법을 적용하였다. 그림 7은 기존의 검사노드 메모리 저장방식과 본 논문의 방식을 비교한 것이다. $L_{j \rightarrow i}$ 의 정보를 직접 저장하는 기존의 방식(그림 7(a)에 비해, 그림 7(b)의 제안된 방식에서는 $L_{j \rightarrow i}$ 의 부호비트인 Sign과 $|L_{j \rightarrow i}|$ 의 최소값 여부를 표현하는 Mag를 2 비트로 저장한다.

블록길이 1,944의 PCM을 이용하는 복호기에서 $L_{j \rightarrow i}$ 이 8비트이면, 기존의 방법은 $1,944 \times 8 = 15,552$ 비트의 메모리가 필요하다. 반면에, 본 논문에서 제안한 구조는 $1,944 \times 2 = 3,888$ 비트의 메모리만 사용되며, 따라서 검사노드 메모리의 크기를 1/4로 감소시킬 수 있다. 본 논문의 방법은 근사화 비트 폭에 무관하게 검사노드 메모리의 용량을 일정하게 유지할 수 있다.

한편, 복호연산을 수행하기 위해서는 2의 보수 수체계로 표현된 $L_{j \rightarrow i}$ 값이 필요하므로, Sign과 Mag로 구성된 신호를 $L_{j \rightarrow i}$ 로 변환해야 한다. 그림 8은 Sign과 Mag를 이용해 $L_{j \rightarrow i}$ 값을 생성하는 회로이다. Sign은 변환될 $L_{j \rightarrow i}$ 의 부호비트로 사용되고, Mag에 의해 최소

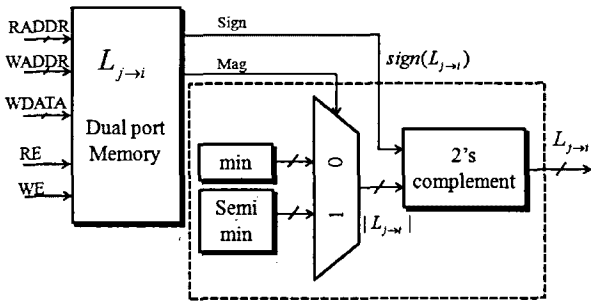


그림 8. $L_{j \to i}$ 값 생성 회로
Fig. 8. Architecture of $L_{j \to i}$ value generator.

값과 준최소값 중 하나가 선택된다. Mag가 0이면 $|L_{j \to i}|$ 은 최소값이고, 1이면 $|L_{j \to i}|$ 은 준최소값이 된다. 2's complement 블록은 부호-크기 값을 입력받아 2의 보수 수체계로 변환한다. 이 회로는 동시에 갱신되는 검사노드의 수만큼 필요하며, 반병렬 구조에서는 메모리 면적에 비해 작은 하드웨어로 구현될 수 있다.

IV. 설계검증 및 성능평가

LDPC 복호기 프로세서 LDPC_D11n은 Verilog HDL로 설계되어 ModelSim을 이용하여 기능 검증과 FPGA 구현을 통해 하드웨어 동작을 확인하였다. 검증에 사용된 테스트 벡터는 Matlab을 이용하여 다음과 같은 과정으로 생성하였다. 즉, 2진 랜덤 신호를 생성하여 LDPC 부호화와 QPSK 변조한 후, AWGN 채널을 거쳐 QPSK 복조하고 이를 8비트로 양자화하여 검증벡터로 사용하였다. 한편, LDPC 복호기를 Matlab으로 모델링하여 구현한 후, ModelSim 시뮬레이션 결과 및 FPGA 구현 검증결과와 비교하여 BER 성능을 분석하였으며, LDPC_D11n 프로세서의 복호성능을 평가하였다.

1. 설계검증

(1) 기능검증 및 성능분석

설계된 LDPC_D11n 프로세서는 그림 9의 과정으로 기능검증과 복호성능 평가가 이루어졌다. Matlab을 이용하여 랜덤 소스벡터 생성, LDPC 부호화, 변조, 채널 잡음 삽입 그리고 복조 등을 통해 $E_b/N_0 = 0.5 \text{ dB} \sim 3 \text{ dB}$ (0.5 dB 간격)의 시뮬레이션 벡터를 생성하였다. 8비트로 양자화된 시뮬레이션 벡터를 이용하여 설계된 복호기의 ModelSim 시뮬레이션을 수행하였으며, 시뮬

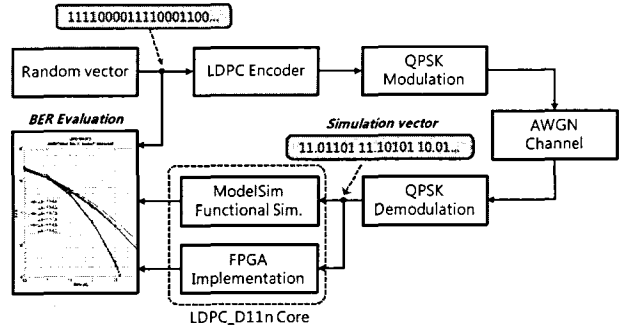


그림 9. 설계된 복호기의 기능검증 및 성능평가 방법
Fig. 9. Functional verification and performance evaluation method of designed decoder.

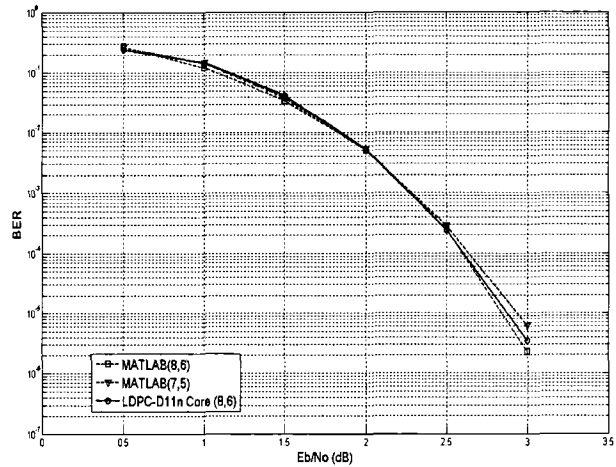


그림 10. 설계된 복호기와 Matlab 시뮬레이션 결과 비교
Fig. 10. BER performance comparison of designed decoder and Matlab simulation.

레이션 결과와 소스벡터의 XOR 연산을 통해 정정되지 못한 오류 비트들을 분석하고 BER 성능을 분석하였다.

그림 10은 ModelSim 시뮬레이션 결과로부터 얻어진 복호기 BER 성능과 Matlab 고정 소수점(fixed-point) 시뮬레이션 결과로부터 얻어진 BER 성능을 비교한 것이다. 10 Mbits 이상의 부호어에서 40 비트 이상의 에러가 발생할 때까지 복호를 수행하여 얻어진 결과이며, 설계된 복호기는 비트 폭 (8, 6)이고, 비교 대상인 Matlab 고정 소수점 모델링에는 비트 폭 (8, 6)과 (7, 5)가 사용되었다. (n, m)의 표현에서 n은 전체 비트 폭을 나타내고, m은 소수점 이하의 비트 폭을 나타낸다. 그림 10의 BER 성능 평가로부터, 본 논문에서 설계된 LDPC 복호기는 Matlab 고정 소수점 시뮬레이션 결과와 거의 일치되는 성능을 보였다.

(2) FPGA를 이용한 하드웨어 구현 검증

설계된 복호기 프로세서는 FPGA 구현을 통해 하드웨어 동작을 검증하였다. 검증 시스템의 구성도는 그림 11과 같다. FPGA 디바이스는 Xilinx Spartan XC3S400이 사용되었으며, RS-232c 직렬통신 프로토콜을 이용하여 PC와 연결된다. 검증에 사용된 테스트 벡터는 PC에서 Matlab으로 생성되어 RS-232c 통신을 통해 FPGA 보드로 인가되며, FPGA 보드에서는 수신된 데이터를 UART 코어를 이용하여 8비트 병렬로 변환한다. UART 코어에 의해 변환된 데이터는 Wrapper를 이용하여 648 비트로 변환되어 FPGA에 구현된 LDPC 복호기로 인가된다. LDPC 복호기에 입력된 테스트 벡터는 복호화되어 그 결과가 다시 Wrapper로 보내지고, Wrapper는 입력받은 648비트의 복호결과를 8비트로 분할하여 UART 코어로 보낸다. UART 코어는 8비트 데이터를 다시 RS-232c 통신을 통해 PC로 보내며, PC에서는 복호결과로부터 정정되지 못한 오류에 대한 분석을 거쳐 결과를 모니터에 출력한다.

FPGA 구현 검증결과는 그림 12와 같으며, E_b/N_0 값에 따라 복호기 입력신호의 정상도와 복호기에 의해 정정되지 않은 오류 비트를 보이고 있다. $E_b/N_0=0.5$ dB인 그림 12(a)의 경우는 복호기에 의해 정정되지 못한 오류 비트들이 많이 존재하며, $E_b/N_0=3.0$ dB인 그림 12(b)의 경우는 QPSK 변조된 신호의 정상도의 패턴이 비교적 뚜렷하며, 복호기에 의해 정정되지 못한 오류 비트들이 매우 적음을 확인할 수 있다. 이와 같은 FPGA 구현 검증결과는 ModelSim 시뮬레이션 결과와 유사한 BER 성능을 보였으며, 따라서 설계된 LDPC 복호기 프로세서가 정상 동작함을 확인할 수 있다.

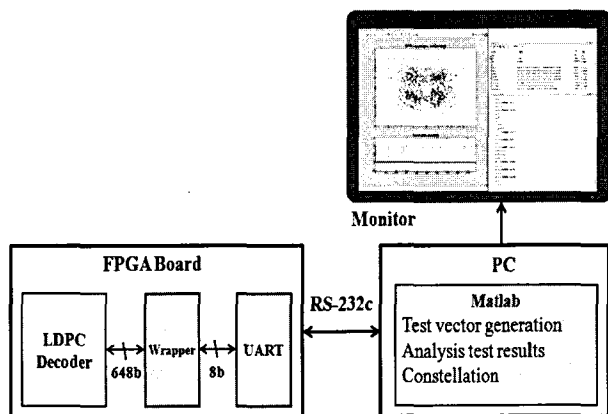
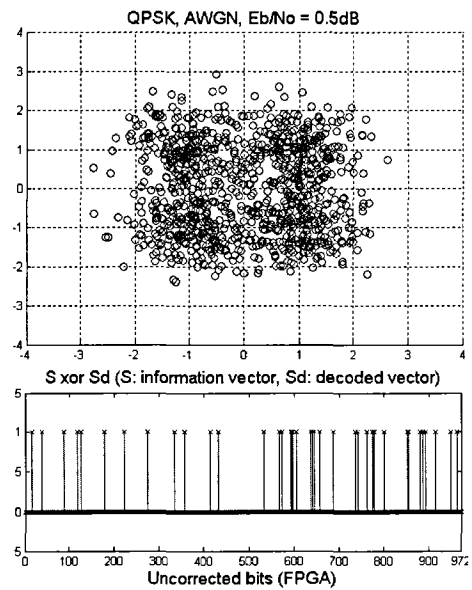


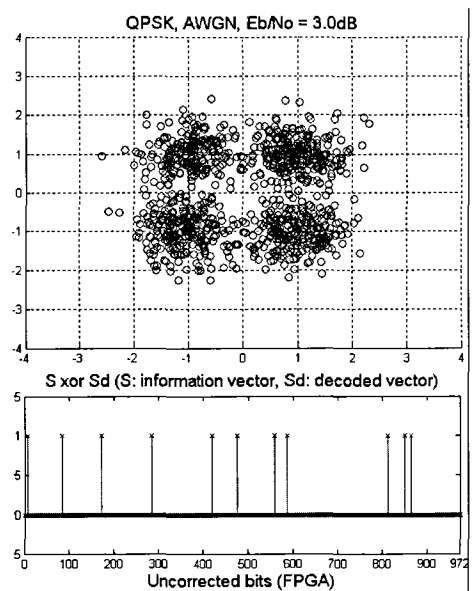
그림 11. FPGA 구현 검증 시스템 구성도
Fig. 11. Configuration of FPGA verification system.

2. 논리합성 및 레이아웃 설계

기능검증이 완료된 LDPC 복호기 프로세서는 $0.35 \mu\text{m}$ CMOS 셀 라이브러리를 이용하여 합성되었다. 논리합성 결과 약 200,400개의 게이트와 19,440 비트의 RAM으로 구현되었으며, 동일한 PCM을 갖는 LDPC 복호기 프로세서^[13]와 비교하여 게이트 수는 비슷하지만, 약 25%의 메모리만으로 구현되었다. 합성결과로부터 설계된 LDPC 복호기 프로세서의 성능을 평가한 결과는 표



(a) $E_b/N_0 = 0.5$ dB인 경우



(b) $E_b/N_0 = 3.0$ dB인 경우

그림 12. 설계된 LDPC 복호기의 FPGA 구현 검증 결과
Fig. 12. FPGA verification results of designed decoder.

표 1. LDPC 복호기 프로세서의 비교

Table 1. Comparison of LDPC decoder processors.

	[15]	[16]	[13]	본 논문
블록길이	1,024	5,114	1,944	1,944
부호화율	1/2	1/3	1/2	1/2
면적 (mm ²)	52.5	14.5	N/A	13.84(코어)
게이트 수	N/A	N/A	195k	200k
메모리(bits)	34,816	460,800	77,760	19,440
동작 주파수 (MHz)	64	145	400	80
출력을(bps)	1 G	24 M	1 G	135 M
공정 (μm)	0.16	0.18	0.13	0.35

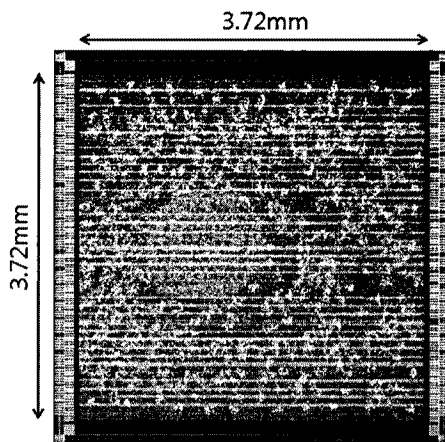


그림 13. LDPC_D11n 복호기의 레이아웃 도면 (코어 부분)

Fig. 13. Layout of LDPC_D11n (core only).

1과 같다. 레이아웃 설계 후의 배선에 의한 지연 30%를 추가적으로 고려하더라도 최대 지연시간은 12.5 ns를 넘지 않을 것으로 판단되며, 따라서 80 MHz로 안전하게 동작 가능할 것으로 평가된다. 본 논문의 LDPC 복호기는 문헌에 발표된 사례들과 비교하여 사용된 공정 기술의 한계로 인해 동작 주파수가 작게 평가되었으나, 0.13 μm 이하의 최신 공정기술을 사용한다면 훨씬 우수한 성능을 보일 것으로 예상된다.

LDPC_D11n 프로세서는 Synopsys의 Auto P&R 툴인 Astro를 이용하여 레이아웃 설계되었다. 메모리를 제외한 코어의 면적은 3.73×3.73 mm²이며, 메모리를 제외한 코어부분의 레이아웃 도면은 그림 13과 같다.

V. 결 론

차세대 무선 랜 표준 IEEE 802.11n용 블록길이 1,944와 부호화율 1/2을 지원하는 LDPC 복호기 프로세서 코어를 설계하고 검증하였다. 설계된 복호기는 최소합 알고리즘을 기반으로 한 layered 복호구조로 설계되었으며, 레이어간의 읽기와 쓰기 동작이 동시에 수행되도록 하여 연산성능을 최적화하였다. 또한, 검사노드 메모리에 저장되는 LLR 근사화 값을 근사화 비트 폭에 무관하게 2 비트의 정보만으로 저장하는 새로운 방법을 고안하여 적용함으로써 기존의 방법에 비하여 검사노드 메모리를 75% 감소시켰으며, 또한 PCM을 저장하는 ROM의 크기도 최소화되도록 하여 하드웨어 최소화를 이루었다. 설계된 LDPC 복호기의 FPGA 구현을 통해 하드웨어 동작을 검증하였으며, 복호성능 분석을 통해 실용성을 입증하였다.

본 논문의 LDPC 복호기 설계최적화 기법들은 유럽 디지털 위성방송 규격 (DVB-S2), 모바일 WiMAX 표준(IEEE 802.16e), 이더넷 표준(IEEE 802.3an), 고속 저장장치 등 다른 시스템의 LDPC 복호기 설계에도 폭넓게 적용될 수 있을 것이다.

감사의 글

본 연구는 2009년도 반도체설계교육센터(IDEC)의 CAD Tool 지원에 의하여 이루어진 연구로서, 관계부처에 감사 드립니다.

참 고 문 헌

- [1] R. Gallager, "Low-Density Parity-Check Codes," IRE Trans. Info. Theory, vol. IT-8, pp. 21-28, Jan. 1962.
- [2] D. Mackay and R. Neal, "Near Shannon Limit Performance of Low Density Parity Check Codes," Electro. Lett., vol. 32, no. 6, pp. 1645-1646, Aug. 1962.
- [3] T. Rihardson and R. Urbanke, "Efficient Encoding of Low Density Parity-Check Codes," IEEE Trans. Inform. Theory, vol. 47, pp. 638-656, Feb. 2001.
- [4] J. W. Bond, S. Hui, and H. Schmidt, "Constructing low-density parity-check codes," EUROCOMM 2000. Information System for

Enhanced Public Safety and Security. IEEE/AFCEA, pp. 260-262, 2000.

[5] DVB-S2 Draft ETSI EN 302 307 V1.1.1 (2004-06), ETSI

[6] IEEE P802.11n/D3.07, Draft Amendment to Standard for Information Technology - Telecommunications and information exchange between systems-Local and Metropolitan networks-Specific requirements- Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Enhancements for Higher Throughput, IEEE Std. 802.11n, 2008.

[7] IEEE Standard for Local and metropolitan area networks Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems Amendment 2: Physical and Medium Access Control Layers for Combined Fixed and Mobile Operation in Licensed Bands and Corrigendum 1, IEEE Std. 802.16e, 2005.

[8] IEEE Standard for Information Technology Tele- communications and Information Exchange Between Systems Local and Metropolitan Area Networks - Specific Requirements Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/ CD) Access Method and Physical Layer and Management Parameters for 10 Gb/s Operation, Type 10GBASE-T, IEEE Std 802.2an, 2006.

[9] D.J.C. MacKay and R. M. Neal. "Near Shannon limit performance of low density parity check codes," IEE Electronic Letter, vol. 32, no. 18, pp. 1645- 1646, Aug. 1996.

[10] F.R. Kschischang, B.J. Frey, and H.A. Loeliger, "Factor graphs and the sum product algorithm," IEEE Transaction on Information Theory, vol. 47, pp. 498-519, Feb., 2001.

[11] J. Chen and M. Fossorier, "Density evolution for two improved BP-based decoding algorithms of LDPC codes," IEEE Commun. Lett., vol. 6, pp. 208-210, May, 2002.

[12] M. Fossorier, M. Mihaljevic and H. Imai, "Reduced complexity iterative decoding of low-density parity check codes based on belief propagation," IEEE Trans. Commun., vol. 47, pp. 673-680, May 1999.

[13] Yang Sun, M. Karkooti and J. R. Cavallaro, "High Throughput, Parallel, Scalable LDPC Encoder/ Decoder Architecture for OFDM Systems," in IEEE Dallas Circuits and Systems Workshop (DCAS'06), Oct 2006.

[14] Mohammad M. Mansour and Naresh RShanbhag, "A Novel Design Methodology for High-Perfor- mance Programmable Decoder Cores for AA- LDPC Codes," Journal of VLSI Signal Processing Systems, 2005.

[15] C. Howland and A. Blanksby, "Parallel decoding architectures for low density parity check codes," Proc. IEEE Inter. Symp. Circuits Syst.(ISCAS), Sydney, Australia, pp. 742-745, May 2001.

[16] M. Bikerstaff et al., "A 24 Mb/s radix-4 LogMAP turbo decoder for 3GPP-HSDPA mobile wireless," IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers, pp. 150 - 151, 2003.

저 자 소 개



정 상 혁(정회원)
 2008년 2월 금오공과대학교
 전자공학부(공학사)
 2010년 2월 금오공과대학원
 전자공학과(공학석사)
 2010년 1월~현재 고등기술연구
 원 로봇생산기술센터

<주관심분야 : 암호 알고리즘, 시스템 및 네트워크 보안, LDPC>



신 경 옥(정회원)-교신저자
 1984년 2월 한국항공대학교
 전자공학과(공학사)
 1986년 2월 연세대학교 대학원
 전자공학과(공학석사)
 1990년 8월 연세대학교 대학원
 (공학박사)

1990년 9월~1991년 6월 한국전자통신연구소
 반도체연구단(선임연구원)
 1991년 7월~현재 금오공과대학교 전자공학부
 (교수)
 1995년 8월~1996년 7월 University of Illinois
 at Urbana-Champaign(방문교수)
 2003년 1월~2004년 1월 University of
 California at San Diego(방문교수)

<주관심분야 : 통신 및 신호처리용 SoC 설계, 정보보호 SoC 설계, 반도체 IP 설계>