

논문 2008-45SD-2-16

DVB-S2 기반에서 다양한 부호화율을 지원하는 LDPC 복호기

(A LDPC Decoder for DVB-S2 Standard Supporting Multiple Code Rates)

류 혜 진*, 이 중 열**

(Hye-Jin Ryu and Jong-Yeol Lee)

요 약

디지털 비디오 방송표준 (DVB-S2)은 순방향 에러 코딩방법으로 BCH와 LDPC를 연결한 시스템을 내부코딩으로 사용한다. DVB-S2에서 LDPC 코드는 11개의 서로 다른 부호화율을 정의하고 있기 때문에, DVB-S2 LDPC 복호기는 다양한 부호화율을 지원해야 한다. 11개의 부호화율 중에서 7가지 (3/5, 2/3, 3/4, 4/5, 5/6, 8/9, 9/10)는 균일한 부호화율이고, 나머지 4가지 (1/4, 1/3, 2/5, 1/2)는 비균일 부호화율이다. 본 논문에서는 균일한 LDPC 코드를 위한 유연한 복호기를 제시한다. 제안된 복호기는 칩의 면적, 메모리의 효율, 처리속도 등에서 많은 장점을 갖는 반 병렬 복호 구조와 변수노드와 체크노드의 내부 연결선을 줄이고 다양한 부호화율을 지원할 수 있도록 Benes 네트워크를 결합하여 블록크기가 64,800까지 사용가능하도록 설계하였다. 제안하는 복호기는 200MHz에서 193.2Mbps의 처리속도를 갖으며, 면적은 16.261mm²이고, 전력은 공급전압이 1.5V에서 198mW의 소모를 보인다.

Abstract

For forward error correction, DVB-S2, which is the digital video broadcasting forward error coding and modulation standard for satellite television, uses a system based the concatenation of BCH with LDPC inner coding. In DVB-S2 the LDPC codes are defined for 11 different code rates, which means that a DVB-S2 LDPC decoder should support multiple code rates. Seven of the 11 code rates, 3/5, 2/3, 3/4, 4/5, 5/6, 8/9, and 9/10, are regular and the rest four code rates, 1/4, 1/3, 2/5, and 1/2, are irregular. In this paper we propose a flexible decoder for the regular LDPC codes. We combined the partially parallel decoding architecture that has the advantages in the chip size, the memory efficiency, and the processing rate with Benes network to implement a DVB-S2 LDPC decoder that can support multiple code rates with a block size of 64,800 and can configure the interconnection between the variable nodes and the check nodes according to the parity-check matrix. The proposed decoder runs correctly at the frequency of 200MHz enabling 193.2Mbps decoding throughput. The area of the proposed decoder is 16.261mm² and the power dissipation is 198mW at a power supply voltage of 1.5V.

Keywords : DVB-S2, LDPC, Partially parallel architecture

I. Introduction

Low density parity check (LDPC) codes have been

proposed as the inner code for forward error correction in DVB-S2. LDPC code is linear block code described with a binary sparse $(n-k) \times k$ parity-check matrix. Each row of the matrix H corresponds to a parity check and each column represents a symbol. A regular (n, γ, ρ) LDPC code consists of code word of the length n and every row and column of a parity-check matrix has the same

* 학생회원, ** 정회원, 전북대학교 전자정보공학부
(Division of Electronics and Information
Engineering, Chonbuk National University)

※ This work was supported in part by the second
stage of Brain Korea 21 Project and the IDEC.

접수일자: 2007년11월13일, 수정완료일: 2007년2월5일

number of 1's. Otherwise it is an irregular code^[1~2].

The DVB-S2 standard requires LDPC codes for 11 different code rates, 1/4, 1/3, 2/5, 1/2, 3/5, 2/3, 3/4, 4/5, 5/6, 8/9, and 9/10 with a codeword length up to 64,800 bits. Seven of code rates, 3/5, 2/3, 3/4, 4/5, 5/6, 8/9, and 9/10 are regular and the rest four code rates, 1/4, 1/3, 2/5, and 1/2 are irregular. It requires that these codes to perform within 1dB of the Shannon limit.

DVB-S2 applies concatenated code that it combines LDPC code and BCH code. In order to comply with the DVB-S2 which allows multiple code rates an LDPC decoder should support various code rates.

In this paper, we propose a partially parallel LDPC decoder architecture with broadcasting^[4] that can reduce the area of interconnection. The proposed decoder also uses Benes network^[5] and can support multiple code rates and variable check matrices.

The rest of this paper is organized as follows. Section II briefly introduces the LDPC code and the sum-product decoding algorithm. Section III proposes a partially parallel decoding architecture with Benes network. Section IV shows the ASIC implementation results and performance comparison with the related works and Section V concludes the paper.

II. Low Density Parity Check Codes

1. Sum-Product Algorithm

The proposed decoder implements the sum-product algorithm. Each row and column of the parity-check

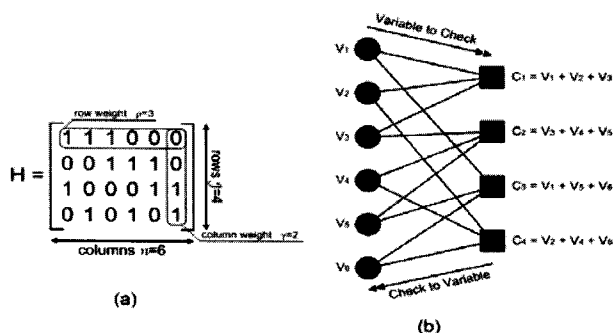


그림 1. 균일 (6, 2, 3) LDPC 코드 (a) 패리티 체크 매트릭스, (b) 터너 그래프

Fig. 1. Regular (6, 2, 3) LDPC code (a) Parity-check matrix and (b) Tanner graph.

matrix of a regular (n, γ, ρ) LDPC represents the connection between check nodes and various nodes, respectively. The Tanner graph has been introduced to represent LDPC codes. Tanner graph is a bipartite graph with variable nodes on one side and constraint or check nodes on the other side. In the graph, each variable node corresponds to a received symbol, each check node corresponds to a set of parity check equations, and each edge corresponds to a non-zero entry in the parity-check matrix. Fig. 1 shows an example of a regular (6, 2, 3) LDPC code. The following shows the steps of the sum-product algorithm based on the Tanner graph.

A. Initialization

In case of binary code of AWGN channel, the sum-product algorithm can be executed more effectively in log domain. It uses probability results by defining LLR (Log Likelihood Ratio). Each variable node(n) is allocated to a priority probability LLR $L(P_n)$. If it has the same probability with input, the probability of $x_n = 0, x_n = 1$ is the same, so $L(P_n)$ is described below.

$$L(P_n) = \frac{1}{\sigma^2} y_n, \quad \sigma^2 = \frac{1}{2R(E_b/N_o)} \quad (1)$$

At each element (m, n) of a parity-check matrix, $L(q_{n \rightarrow m}), L(r_{m \rightarrow n})$ are initialized as follows.

$$L(q_{n \rightarrow m}) = L(P_n), \quad L(r_{m \rightarrow n}) = 0 \quad (2)$$

B. Check node to Variable node

Each check node (m) gathers the results of all information $L(q_{n \rightarrow m})$ update n based on all information of other bits connected to the check node m .

$$L(r_{m \rightarrow n}) = \phi^{-1} \left[\sum_{n' \in L(m) \setminus n} \phi \{ L(q_{n' \rightarrow m}) \} \right] \quad (3)$$

$$\phi^{-1}(x) = 2 \tanh^{-1} \{ \exp(-x) \}$$

$$\phi(x) = -\log \{ \tanh(x/2) \}$$

C. Variable node to Check node

Each variable node (n) transfers the probability to the all connected check nodes. Initial value and operated results are computed by adding all of them.

$$L(q_{n \rightarrow m}) = L(P_n) + \sum_{m' \in M(n) \setminus n} L(r_{m' \rightarrow n}) \quad (4)$$

D. Decision

Decoder can get total posterior probability for a variable node (n) by adding the information from all the check nodes connected the variable node (n).

$$L(q_{n \rightarrow m}) = L(P_n) + \sum_{m' \in M(n)} L(r_{m' \rightarrow n}) \quad (5)$$

E. Syndrome check

If the results of parity-check matrix and XOR operation are zeros which means decoding process is completed, the decoder terminates decoding the current block and then, starts to decode the next block. Otherwise, it means that decoding process is not completed, it should repeat decoding process of the current block.

2. Decoder Architectures

One of most common measures to compare LDPC decoders is the level of parallelism. There are three kinds of architectures: serial, fully parallel, and partially parallel architecture. Serial architectures that traverse variable nodes and check nodes consecutively, would result in the lowest throughput, hence this solution is also out of scope for an efficient implementation for DVB-S2. In a fully parallel implementation, all variable and check-node calculations, are directly realized in hardware. All units are interconnected via many wires, leading to congestion in the layout. A fully parallel implementation of the DVB-S2 decoder is impractical since it limits the block length to 1024~2048 bits to avoid the bottleneck in the layout of the interconnection wire for data routing. A partially parallel architecture makes a compromise between a

serial architecture and a fully parallel architecture. It has the advantages of smaller chip size, memory efficiency and high processing speed.

III. Proposed LDPC Decoding Architecture

1. Benes Network

We need a switch block that can change the interconnection between variable nodes and check nodes to implement a LDPC decoder supporting variable parity-check matrices. The SE block in Fig. 2(a) is CROSS when control signal is '1', otherwise SE block is BAR. If the interconnection network is implemented by a general crossbar switch in Fig. 2(b), the complexity of the interconnection network would be increased as N^2 where N is the number of inputs to the interconnection network. If Benes network depicted in Fig. 2(c) is used, the complexity is $N/2 \times \{2(\log_2 N) - 1\}$, which means that it can decrease the area of the circuit. Although the Banyan network depicted in Fig. 2(d) reduces the complexity further to $N/2 \times (\log_2 N)$, the Banyan network cannot be used because it requires more complex control schemes due to its asymmetry.

Switching network is more complex than the fixed wire interconnection and causes more delay. To

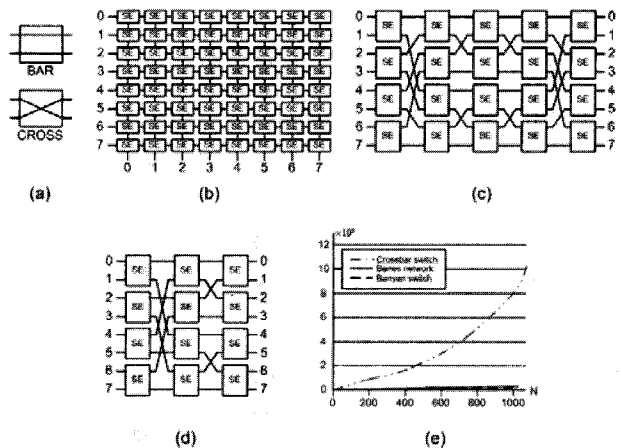


그림 2. 8x8 스위치 (a) SE 블록, (b) Crossbar 스위치, (c) Benes 네트워크, (d) Banyan 네트워크, (e) 복잡도 비교

Fig. 2. 8x8 Switch (a) SE block, (b) Crossbar switch, (c) Benes network, (d) Banyan network, (e) Comparison of complexity.

minimize the wiring complexity we use broadcasting method. Furthermore, we reduce the number of the levels in Benes network by reducing the number of input messages.

2. Broadcasting

When the length of interconnection becomes longer, it increases the delay time and lowers the transmission speed of data. To solve this problem, we organize decoder as in Fig 3. The following equations are the changed sum-product algorithm equations for the broadcasting method^[4, 6].

$$L(r_m) = \sum_{n' \in L(m)} \phi \{L(q_{n' \rightarrow m})\} \quad (6)$$

$$L(r_{m \rightarrow n}) = L(r_m) - \phi \{L(q_{n \rightarrow m})\} \quad (7)$$

$$L(q_n) = L(P_n) + \sum_{n' \in M(n)} \phi^{-1} \{L(r_{m' \rightarrow n})\} \quad (8)$$

$$L(q_{n \rightarrow m}) = L(q_n) - \phi^{-1} \{L(r_{m \rightarrow n})\} \quad (9)$$

Fig. 4(a) shows a check node performs the operations corresponding to equations (7) and (8).

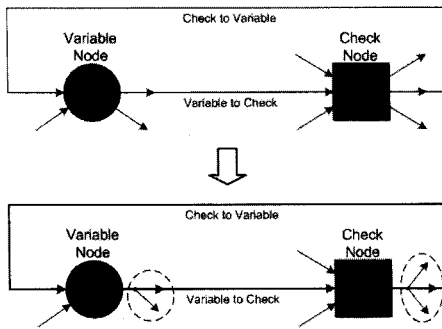


그림 3. Broadcasting 방법
Fig. 3. Broadcasting method.

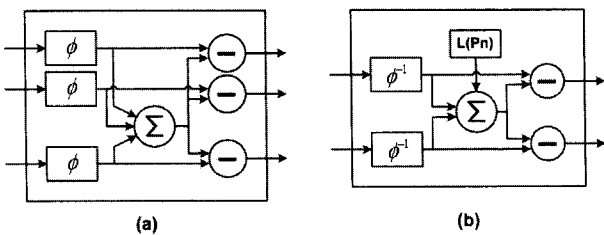


그림 4. 기본 연산 단위 (a) 체크노드, (b) 변수노드
Fig. 4. Processing unit (a) Check node and (b) variable node.

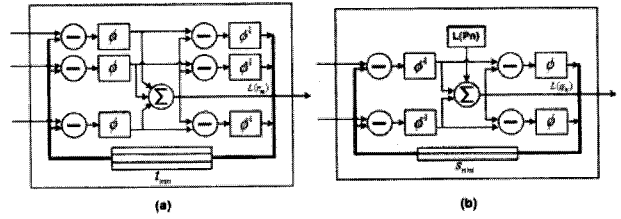


그림 5. Broadcasting 방법을 적용한 기본 연산 단위 (a) 체크노드, (b) 변수노드
Fig. 5. Processing unit changed for the broadcasting method (a) Check node and (b) Variable node.

Equation (9) and (10) are performed in the variable node of Fig. 4(b).

Equations from (11) to (14) perform check node to variable node calculations. Similarly, equations from (15) to (18) perform variable node to check node calculations. By using the equations from (11) to (18), only $L(q_n)$ and $L(r_m)$ are transmitted and hence, the interconnection can be reduce. However, the broadcasting method increases the size of lookup tables and registers.

$$L(q_{n \rightarrow m}) = L(q_n) - t_{mn} \quad (10)$$

$$L(r_m) = \sum_{n' \in L(m)} \phi \{L(q_{n' \rightarrow m})\} \quad (11)$$

$$L(r_{m \rightarrow n}) = L(r_m) - \phi \{L(q_{n' \rightarrow m})\} \quad (12)$$

$$t_{mn} = \phi^{-1} \{L(r_{m \rightarrow n})\} \quad (13)$$

$$L(r_{m \rightarrow n}) = L(r_m) - s_{nm} \quad (14)$$

$$L(q_n) = L(P_n) + \sum_{m' \in M(n)} \phi^{-1} \{L(r_{m' \rightarrow n})\} \quad (15)$$

$$L(q_{n \rightarrow m}) = L(q_n) - L(r_{m \rightarrow n}) \quad (16)$$

$$s_{nm} = \phi \{L(q_{n \rightarrow m})\} \quad (17)$$

We exploit partially parallel architecture to reduce the area of the LDPC decoder. By using the demultiplexer and multiplexer at the front and back of the switch block to time-share the variable and the check nodes, we can reduce the number of the variable nodes and the check nodes. Due to the reduced number of the variable and the check nodes,

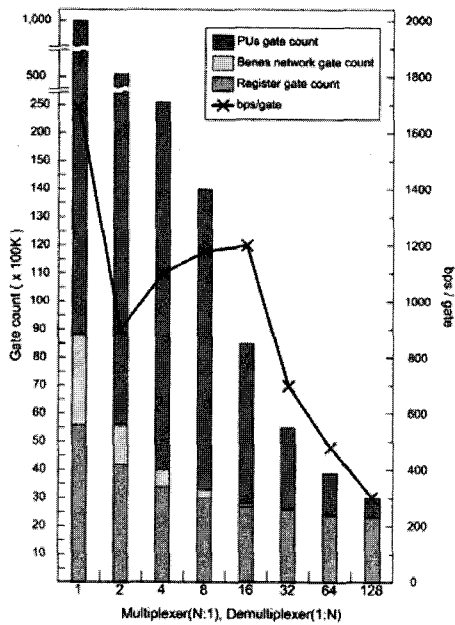


그림 6. 다중화기와 역 다중화기의 비율에 따른 하드웨어의 비율과 처리속도
 Fig. 6. Hardware cost versus processing multiplexer and demultiplexer.

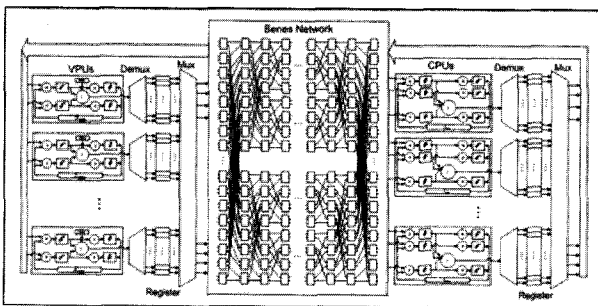


그림 7. 제안한 LDPC 복호기의 최상위 블록 다이어그램
 Fig. 7. Top-level block diagram of the proposed LDPC decoder.

the number of the levels in Benes network, the total chip area, and the delay of wire are decreased. It can also support various code rates and parity-check matrices by using input message select signals. Hardware cost varies according to the multiplexer and demultiplexer used as shown in Fig. 6. In the case of a fully parallel architecture which is the left most case in Fig. 6, the bit rate is 3.41Gbps, but the gate count is almost 10^9 gates. However, when the decoder exploits multiplexers with the large number of inputs in order to reduce the chip area, the bit rate becomes low. In the proposed design we calculate the bit rate per gate to determine the number of

multiplexer inputs. We use multiplexers and demultiplexers with 16 inputs and 16 outputs, respectively since Fig. 6 shows that the bit rate per gate is the most high when 16-to-1 multiplexers and 1-to-16 demultiplexers are used.

Fig. 7 shows the top-level block diagram of the proposed LDPC decoder.

3. Scheduling

One iteration of the decoding consists of check node operation, result transmission, and variable node operation. The result transmission can be overlapped both the check node operation and the variable node operation as shown in Fig. 8. By inserting pipeline registers that contains the results of the variable nodes and the check nodes, we can implement two-stage pipeline.

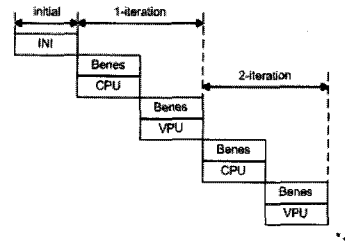


그림 8. 스케줄링
 Fig. 8. Scheduling.

IV. Experimental Results

We designed a LDPC decoder based on the proposed partially parallel architecture using a 0.18um

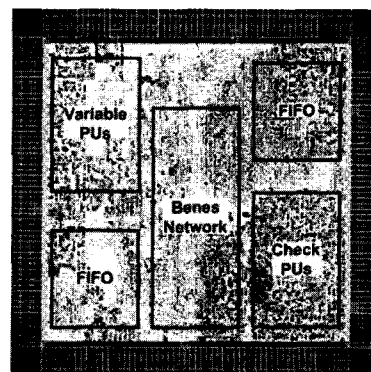


그림 9. 제안된 LDPC 복호기의 레이아웃
 Fig. 9. Layout of the proposed LDPC decoder.

표 1. LDPC 복호기의 성능 비교

Table 1. Comparison of LDPC decoders.

No.	[7]	[8]	[9]	Proposed
Area (μm^2)	49.5	22.689	4.2	16.26
Throughput (Mbps)	90 (41-77iter.)	255 (30iter.)	90 (30iter.)	193.2 (30iter.)
Operating Freq.(MHz)	200	270	300	200
Technology (nm)	130	130	90	180

6-Metal CMOS process. Fig. 9 shows the layout of the proposed LDPC decoder. Table 1 compares the proposed decoder with the previous works. The area of the proposed decoder is much smaller than those of [7] and [8] considering the technology used. This is because the proposed decoder uses less interconnection wire and memory. In the case of throughput the proposed decoder shows better performance than [7] and [9].

V. Conclusion

In this paper we propose a DVB-S2 LDPC decoder architecture. The proposed architecture exploits Benes to support multiple code rates and various parity-check matrices. Benes network is symmetric and nonblocking network and the data transmission through Benes network can be easily overlapped with the variable node and the check node operation. To reduce the interconnection complexity between the variable nodes and the check nodes we use the broadcasting method. The proposed decoder implemented with a 0.18 μm 6-Metal CMOS process shows the performance of 193.2Mbps (@200MHz, 30iteration). The proposed LDPC decoder occupies smaller chip area than the previous works.

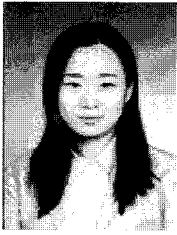
Reference

- [1] R. G. Gallager, "Low Density Parity Check Codes," IRE Trans Information Theory, vol. IT-8, no. 1, pp. 21-28, 1962.
- [2] D. J. C. Mackay, "Good Error-Correcting Codes

Based on Very Sparse Matrices," IEEE Trans. Information Theory, Vol. 45, no. 2, pp. 399-431, Mar. 1999.

- [3] F. Kienle, T. Brack and N. When, "A synthesizable IP core for DVB-S2 LDPC code decoding," In Proceedings of Design, Automation and Test in Europe, vol. 3, pp. 100-105, Mar. 2005.
- [4] A. Darabiha, A. C. carusome and F. R. Kshischang, "Multi-Gbit/sec Low Density Parity Check Decoders with Reduced Interconnect Complexity," IEEE International Symposium Circuit and Systems, vol. 5, pp. 5194-5197, May 2005.
- [5] G. Malema and M. Liebelt, "Interconnection Network for Structured Low-Density Parity-Check Decoders," IEEE Asia-Pacific Conference on Communication, pp. 537-540, Oct. 2005.
- [6] S. H. Kang and I. C. Park, "Loosely coupled Memory-Based Decoding Architecture for Low Density Parity Check Codes," IEEE Transaction on Circuits and System, vol. 53, no. 5, May 2006.
- [7] P. Urard et, al. "A 135Mb/s DVB-S2 compliant codec based on 64,800bits LDPC and BCH codes," IEEE Solid-state Circuits Conference, vol. 1, pp. 446-609, Feb. 2005.
- [8] F. Kienle, T. Brack and N. When, "A synthesizable IP fore for DVB-S2 LDPC code decoding," IEEE conference on Design Automation and Test in Europe, vol. 3, pp. 100-105, Mar. 2005.
- [9] J. Dielissen, A Hekstra and V. Berg, "Low cost LDPC decoder for DVB-S2," IEEE Conference on Design Automation and Test in Europe, vol. 2, pp. 06-10, Mar. 2006.

저 자 소 개



류 혜 진(학생회원)
2006년 8월 전북대학교 전자정보
공학부 전기전자공학과
학사 졸업.
2006년 8월~현재 전북대학교
전자정보공학부
전자공학과 석사 재학중

<주관심분야 : 통신, 신호처리, SoC 설계>



이 종 열(정회원)
1993년 한국과학기술원 전자전산
학과 졸업 (B.S.).
1996년 한국과학기술원 전자전산
학과 졸업 (M.S.).
2002년 한국과학기술원 전자전산
학과 박사 (Ph.D.).

2002년 9월~2003년 9월 하이닉스 반도체 선임
연구원

2003년 10월~2004년 2월 한국과학기술원 BK21
초빙교수

2004년 3월~2006년 3월 전북대학교 전자정보공
학부 전임강사

2006년 4월~현재 전북대학교 전자정보공학부
조교수

<주관심분야: SoC 설계, 내장형 프로세서 설계,
내장형 소프트웨어 최적화>