

두 개의 직렬 Barrel-Rotator를 이용한 QC-LDPC 복호기용 저면적 Multi-Size Circular Shifter

강형주*

Low-Complexity Multi-Size Circular Shifter for QC-LDPC Decoder Based on Two Serial Barrel-Rotators

Hyeong-Ju Kang*

School of Computer Science and Engineering, Korea University of Technology and Education, Cheonan,
Chungbuk 31253, Korea

요 약

Low-density parity-check(LDPC) 코드는 우수한 에러 정정 능력으로 인해 점점 많은 통신 표준에서 채택되고 있으며 그 중 구현이 용이한 quasi-cyclic LDPC(QC-LDPC)가 많이 사용되고 있다. QC-LDPC 복호기에서는 데이터들을 rotation할 수 있는 cyclic-shifter가 필요하며, 이 cyclic-shifter는 다양한 크기의 rotation을 수행할 수 있어야 한다. 이러한 cyclic-shifter를 multi-size circular shifter(MSCS)라고 부르며, 이 논문에서는 MSCS를 적은 면적으로 구현한 구조를 제안한다. 기존의 직렬로 배치된 barrel-rotator 구조에서 rotation의 성질을 이용하여 필요 없는 멀티플렉서를 가려내고 이들을 제거함으로써 저면적을 구현하였다. 실험 결과 면적을 약 12% 줄일 수 있었다.

ABSTRACT

The low-density parity-check(LDPC) code has been adopted in many communication standards due to its error correcting performance, and the quasi-cyclic LDPC(QC-LDPC) is widely used because of implementation easiness. In the QC-LDPC decoder, a cyclic-shifter is required to rotate data in various sizes. This kind of cyclic-shifters are called multi-size circular shifter(MSCS), and this paper proposes a low-complexity structure for MSCS. In the conventional serially-placed two barrel-rotators, the unnecessary multiplexers are revealed and removed, leading to low-complexity. The experimental results show that the area is reduced by about 12%.

키워드 : LDPC 복호기, QC-LDPC 복호기, cyclic-shifter, multi-size circular shifter

Key word : LDPC decoder, QC-LDPC decoder, cyclic-shifter, multi-size circular shifter

Received 02 July 2015, Revised 29 July 2015, Accepted 06 August 2015

* Corresponding Author Hyeong-Ju Kang(E-mail:hjkang@koreatech.ac.kr, Tel:+82-41-560-1420)

School of Computer Science and Engineering, Korea University of Technology and Education, Cheonan, Chungbuk 31253, Korea

Open Access <http://dx.doi.org/10.6109/jkiice.2015.19.8.1839>

print ISSN: 2234-4772 online ISSN: 2288-4165

©This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.
Copyright © The Korea Institute of Information and Communication Engineering.

I. 서 론

Low-density parity-check (LDPC) 코드는 우수한 여러 정정 능력과 병렬화가 가능한 복호 알고리즘으로 인해 최근 많은 통신 표준에 채택되고 있는 에러 정정 코드이다. LDPC 코드에는 여러 하위 부류가 있으며 그 중에서 많이 사용되는 것이 quasi-cyclic LDPC(QC-LDPC) 코드이다[1].

QC-LDPC 코드는 parity-check 행렬이 여러 개의 작은 부행렬로 이루어져 있으며, 각각의 부행렬은 identity 행렬의 cyclic shift이다. 이러한 QC-LDPC 코드의 복호기 설계에 있어서 부행렬에 따라 데이터를 rotation시키는 cyclic-shifter가 필수적이다. 일반적인 cyclic-shifter는 rotation 크기가 정해져 있어서 barrel-rotator와 같은 구조로 만들 수 있으나, QC-LDPC 복호기에서 요구되는 cyclic-shifter는 rotation 크기를 바꿀 수 있어야 하므로 일반적인 barrel-rotator 구조로는 만들기 어려운 면이 있다. 이러한 cyclic-shifter는 multi-size cyclic shifter (MSCS)라 불리며 이에 대해 많은 연구가 이루어져 왔다[2]. 대표적인 구조로는 두 개의 barrel-rotator를 이용하는 구조가 있으며, 두 barrel-rotator를 논문 [3]과 같이 병렬로 배치하거나 논문 [4]와 같이 직렬로 배치하여 이용한다. 다른 구조로는 Benes 네트워크를 이용하는 방법이 있다[5]. Benes 네트워크는 N개의 입력을 permutation하여 출력하는 회로이므로 다양한 크기의 cyclic-shift를 실행할 수 있다.

이러한 기존 구조들 중에 두 개의 barrel-rotator를 직렬로 배치하는 방법은 병렬로 배치하는 방법에 비해 rotation 크기의 특성을 이용하여 회로의 크기를 줄일 수 있는 장점이 있다. 직렬로 배치된 barrel-rotator 중 앞쪽의 barrel-rotator는 일반적인 barrel-rotator를 사용하고, 뒤쪽의 barrel-rotator는 요구되는 rotation 크기에 따라 최적화가 이루어진다. 그리고 두 barrel-rotator의 결과 중에서 원하는 부분을 선택하여 최종 결과를 만들어 내는 멀티플렉서들이 마지막에 배치된다.

이 논문에서는 이러한 cyclic-shifter에 있어서 직렬로 배치된 barrel-rotator 회로의 크기를 줄일 수 있는 방법을 제시한다. 직렬로 배치된 barrel-rotator 중 뒤쪽의 barrel-rotator의 경우, 그 결과가 모두 뒷단의 멀티플렉서에 의해 선택되지는 않는다. 일부는 어느 경우에도 필요하지 않는 것들이 있으며, 이 결과들에 대한 회로

를 제거함으로써 회로의 크기를 더 줄일 수 있다.

이 논문은 다음과 같이 구성되어 있다. 2장에서는 MSCS의 기본 동작에 대해 살펴 본 뒤 기존 구조들에 대해 설명한다. 그리고 3장에서는 MSCS의 동작에 대해 기술한 뒤, 기존의 MSCS 구조들을 설명한다. 4장에서 개선된 구조를 제안하고, 5장에서 기존 구조와 비교한 뒤, 6장에서 결론을 맺는다.

II. QC-LDPC 복호기의 multi-size cyclic shifter

이 장에서는 QC-LDPC 코드의 복호기에서 사용되는 multi-size cyclic-shifter와 그 구조에 대해 설명한다.

2.1. Multi-size cyclic shifter (MSCS)

QC-LDPC 코드의 복호기에서 사용되는 multi-size cyclic shifter는, 각각이 w-bit인 N개의 입력 데이터 ($a[0], a[1], \dots, a[N-1]$)와 제어값 p와 z에 대해 다음과 같이 N개의 데이터($b[0], b[1], \dots, b[N-1]$)를 출력하는 회로이다.

$$b[i] = \begin{cases} a[i+p] & \text{if } 0 \leq i < z-p \\ a[i+p-z] & \text{if } z-p \leq i < z \end{cases} \quad (1)$$

출력 데이터 중 $i \geq z$ 인 $b[i]$ 는 사용되지 않는 데이터이므로 무엇이 출력되어도 상관없다.

N과 z는 사용되는 QC-LDPC 코드에 따라 결정되는 값으로써 표준에 따라 다르다. 예를 들어, IEEE 802.16e WiMAX에서 사용되는 QC-LDPC 코드의 경우 $N=96$ 이고 $z = 4k$ ($6 \leq k \leq 24$)이다.

2.2. 기존의 MSCS 구조

MSCS의 구조는 크게 barrel-rotator를 이용하는 구조와 Benes 네트워크를 이용하는 구조로 나눌 수 있다. Barrel-rotator를 이용하는 구조는 다시 병렬적으로 배치하는 구조와 직렬적으로 배치하는 구조가 있다. 병렬적으로 배치하는 구조의 경우 한 barrel-rotator는 원래 방향대로 p만큼 rotate하고 다른 하나는 그 반대 방향으로 z-p만큼 rotate한다. 각각을 left rotator와 right rotator라고 한다면, 각각의 출력 $l[i]$ 와 $r[i]$ 는 다음과 같다.

$$l[i] = \begin{cases} a[i+p] & \text{for } 0 \leq i < N-p \\ a[i+p-N] & \text{for } N-p \leq i < N \end{cases} \quad (2)$$

$$r[i] = \begin{cases} a[i-z+p+N] & \text{for } 0 \leq i < z-p \\ a[i-z+p] & \text{for } z-p \leq i < N \end{cases} \quad (3)$$

최종 출력은 다음과 같이 선택한다.

$$b[i] = \begin{cases} l[i] & \text{for } 0 \leq i < z-p \\ r[i] & \text{for } z-p \leq i < z \end{cases} \quad (4)$$

이러한 과정을 도식화하여 설명한 것이 그림 1과 그림 2이다.

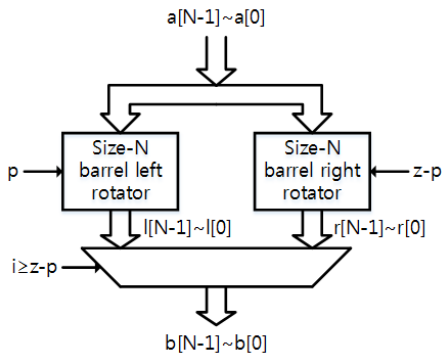


그림 1. 병렬로 두 개의 barrel rotator를 배치한 MSCS 구조
Fig. 1 A MSCS structure to place two barrel rotators in a parallel way

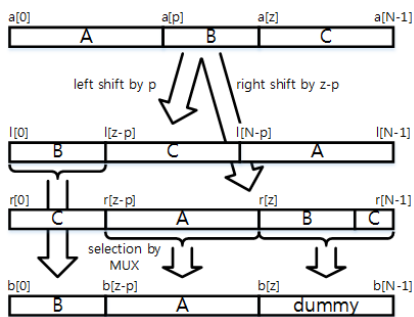


그림 2. 그림 1 구조의 동작
Fig. 2 The operation of the structure in Fig. 1

Barrel-rotator를 직렬로 배치하는 구조는 크기가 N인 barrel-rotator로 입력 데이터를 우선 p만큼 rotate시킨 뒤, 그 결과를 N-z만큼 한 번 더 rotate시킨다. 첫 번째 barrel-rotator의 결과를 c[i], 두 번째 barrel-rotator의 결과를 d[i]라고 한다면 다음과 같이 표현할 수 있다.

$$c[i] = \begin{cases} a[i+p] & \text{for } 0 \leq i < N-p \\ a[i+p-N] & \text{for } N-p \leq i < N \end{cases} \quad (5)$$

$$d[i] = \begin{cases} c[i+N-z] & \text{for } 0 \leq i < z \\ c[i-z] & \text{for } z \leq i < N \end{cases} \quad (6)$$

이 때 d[i]를 a[i]에 대해 다시 기술하면 다음과 같이 정리할 수 있다.

$$d[i] = \begin{cases} a[i+p-z+N] & \text{for } 0 \leq i < z-p \\ a[i+p-z] & \text{for } z-p \leq i < N \end{cases} \quad (7)$$

이 두 결과, c[i]와 d[i] 중에 멀티플렉서를 통해서 다음과 같이 선택하여 최종 결과 b[i]를 출력한다.

$$b[i] = \begin{cases} c[i] & \text{for } 0 \leq i < z-p \\ d[i] & \text{for } z-p \leq i < z \end{cases} \quad (8)$$

이러한 과정을 도식화하여 설명한 것이 그림 3과 그림 4이다.

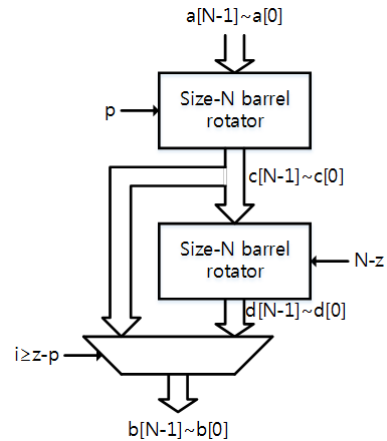


그림 3. 직렬로 두 개의 barrel rotator를 배치한 MSCS 구조
Fig. 3 A MSCS structure to place two barrel rotators in a serial way

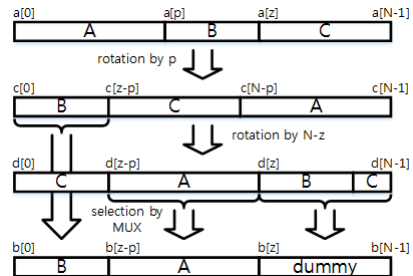


그림 4. 그림 3 구조의 동작
Fig. 4 The operation of the structure in Fig. 3

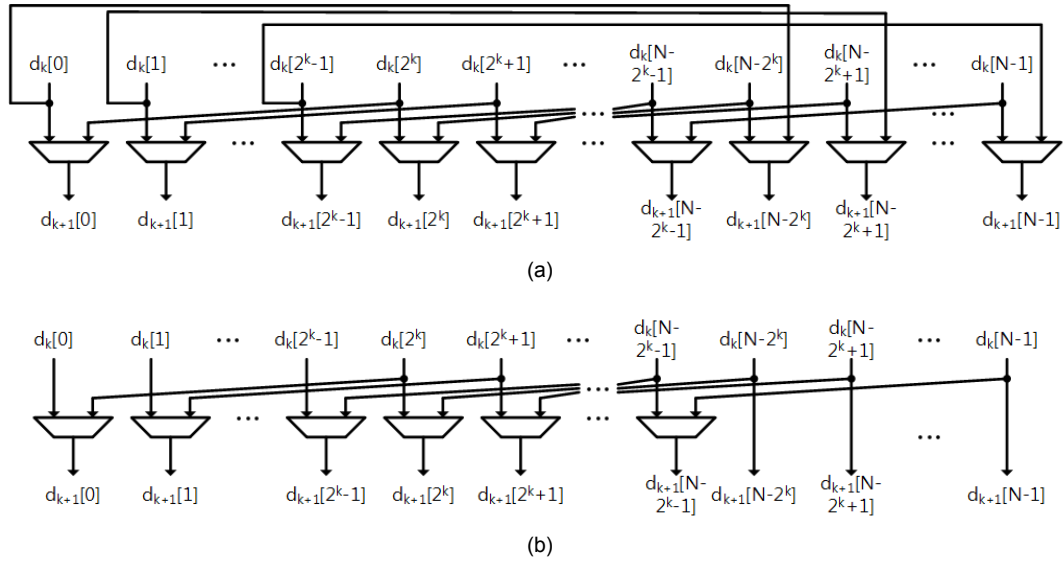


그림 5. 직렬로 배치된 두 번째 barrel rotator에서 (a) 기존의 k번째 단과 (b) 제안하는 k번째 단
 Fig. 5 (a) The conventional k-th stage and (b) the proposed k-th stage in the serially placed second barrel rotator

직렬로 배치한 구조의 경우, rotation 크기의 특성을 이용하여 회로의 크기를 줄일 수 있다. 예를 들어, IEEE 802.16e에서 사용되는 QC-LDPC의 경우, $N=96$ 이고 z 는 $[24,96]$ 범위의 4의 배수들이다. 그러므로 두 번째 barrel-rotator의 입력인 $N-z$ 는 항상 4의 배수이다. 이를 이용하면 두 번째 barrel-rotator는 멀티플렉서 단의 수를 두 단 줄일 수 있다.

III. 제안하는 multi-size cyclic shifter 구조

이 논문에서 제안하는 MSCS 구조는 직렬로 배치한 barrel-rotator를 기본으로 한다. 식(8)을 보면 $b[i]$ 는 $0 \leq i < z$ 인 i 에 대해서만 선택한다. 그러므로 $i > z$ 인 $d[i]$ 는 최종 결과와 상관없는 출력들이다. $i \geq z$ 인 $d[i]$ 들이 불필요한 값들이라면 이를 이용하여 두 번째 barrel-rotator의 크기를 줄일 수 있다.

이것을 barrel-rotator의 각 단계별로 적용하면 다음과 같다. Barrel-rotator가 총 m 단으로 이루어져 있고(즉, $2^{m-1} < N \leq 2^m$), $d_{k+1}[i]$ 를 k 번째($0 \leq k < m$) 단을 거친 데이터라고 하자. 그러면 다음과 같이 barrel-rotator의 동작을 기술할 수 있다. 우선 입력은 다음과

같다.

$$d_0[i] = c[i] \text{ for } 0 \leq i < N \quad (9)$$

만일 두 번째 barrel-rotator의 입력인 $y = N-z$ 를 2진수로 표현했을 때 2^k 자리의 수, y_k 가 0이면

$$d_{k+1}[i] = d_k[i] \text{ for } 0 \leq i < N \quad (10)$$

이고 y_k 가 1이면

$$d_{k+1}[i] = \begin{cases} d_k[i+2^k] & \text{for } 0 \leq i < N-2^k \\ d_k[i+2^k-N] & \text{for } N-2^k \leq i < N \end{cases} \quad (11)$$

이 된다. 마지막으로 출력은 다음과 같이 정의된다.

$$d[i] = d_m[i] \text{ for } 0 \leq i < N \quad (12)$$

이제 $d_{k+1}[i]$ 들 중 필요 없는 부분을 구별하기 위해 y_k 가 1이라고 가정하고, $k+1$ 번째 단 이후에 rotate하는 양을 x_k 라고 하자. 그러면 $y \geq x_k + 2^k$ 와 $z \leq N - x_k - 2^k$ 이 성립한다. 그리고 $d_{k+1}[i]$ 와 $d[i]$ 사이에는 다음과 같은 관계를 적용할 수 있다.

$$d[i] = \begin{cases} d_{k+1}[i+x_k] & \text{for } 0 \leq i < N-x_k \\ d_{k+1}[i+x_k-N] & \text{for } N-x_k \leq i < N \end{cases} \quad (13)$$

이를 바꿔 적으면 다음과 같이 된다.

$$d_{k+1}[i] = \begin{cases} d[i - x_k + N] & \text{for } 0 \leq i < x_k \\ d[i - x_k] & \text{for } x_k \leq i < N \end{cases} \quad (14)$$

그런데 앞에서 설명했듯이 $i \geq z$ 인 $d[i]$ 들은 필요 없는 데이터들이어서 그 값이 무엇이 되든 상관없다. 이때 $z \leq N - x_k - 2^k$ 의 관계를 이용하면 $i \geq N - x_k - 2^k$ 인 $d[i]$ 들은 필요 없는 데이터임이 보장된다. 이것을 식(14)에 대입하면 $i \geq N - 2^k$ 인 $d_{k+1}[i]$ 들은 나중에 필요가 없게 되는 데이터들이다.

이렇게 필요 없는 데이터들은 정확히 선택할 필요가 없으므로 이를 이용하면 멀티플렉서를 줄일 수 있다. 식(11)에서 아래 부분의 식은 $i \geq N - 2^k$ 인 경우에 대한 것이므로 아래와 같이 수정을 해도 최종 결과에는 영향을 미치지 않을 것이다.

$$d_{k+1}[i] = \begin{cases} d_k[i + 2^k] & \text{for } 0 \leq i < N - 2^k \\ d_k[i] & \text{for } N - 2^k \leq i < N \end{cases} \quad (15)$$

이렇게 되면 $i \geq N - 2^k$ 에 대해서는 y_k 가 0일 때나 1일 때나 $d_{k+1}[i]$ 의 값이 같게 되고, 그러면 $i \geq N - 2^k$ 범위의 $d_{k+1}[i]$ 에 대해서는 멀티플렉서가 필요 없게 된다. 이렇게 필요 없게 되는 멀티플렉서의 개수는 k 번째 단에 대해 2^k 개이고, 전체에 대해 생각하면 $\sum_{k=0}^{m-1} 2^k = 2^m - 1$ 개의 멀티플렉서를 줄일 수 있다.

그림 5의 (a)는 두 번째 barrel rotator의 k 번째 단을 그린 그림이다. 이 회로에서 $i \geq N - 2^k$ 범위의 $d_{k+1}[i]$ 에 대해서 멀티플렉서가 필요 없으므로 그림 5의 (b)과 같이 멀티플렉서들을 없앨 수 있다.

IV. 실험 결과

제안하는 구조와 기존의 구조를 비교하기 위해 두 구조를 RTL 수준에서 구현한 뒤 합성하여 비교하였다. 합성 라이브러리는 $0.18\mu\text{m}$ 공정을 이용하였으며 Cadence사의 RTL Compiler로 합성하였다. 입력 데이터의 데이터 폭은 8bit로 설계하였고 802.16e WiMAX 표준을 위한 MSCS를 구현하였다.

합성 결과는 표 1에 제시하였다. 표의 두 번째 열이 논문 [4]에서 제안된 구조를 합성했을 때의 면적이며 세 번째 열이 이 논문에서 제안하는 구조를 적용했을 때의 면적이다. 면적은 μm^2 단위로 제시하였다. 표를 보면 이 논문에서 제안하는 구조를 적용했을 때 지연시간도 약간 줄어들면서 약 12.4% 정도 면적이 줄어들었음을 확인할 수 있다.

표 1. 그림 3 구조의 기존 방식과 제안하는 방식의 비교
Table. 1 Comparison between the conventional method and the proposed method for the fig. 3 structure

	Conventional	Proposed
Area (μm^2)	589530.24	516337.920
Area ratio	1	0.876
Delay time (ps)	3124	3095

V. 결론

이 논문에서는 QC-LDPC 코드의 복호기에서 사용되는 MSCS를 위한 새로운 구조를 제안하였다. 이 구조에서는 기존의 직렬 barrel-rotator 구조를 개선하여, 필요 없는 멀티플렉서를 제거함으로써 면적을 줄일 수 있다. 본 논문에서는 802.16e WiMAX에 대해서만 비교하였으나 QC-LDPC를 사용하는 다른 통신 표준의 복호기에 대해서도 사용될 수 있다.

REFERENCES

- [1] R. M. Tanner, D. Sridhara, A. Sridharan, T. E. Fuja, and D. J. Costello, Jr., "LDPC block and convolutional codes based on circulant matrices," *IEEE Transactions on Information Theory*, vol. 50, no. 12, pp. 2966-2984, Dec. 2004.
- [2] M. Rovini, G. Gentile, and L. Fanucci, "Multi-size circular shifting networks for decoders of structured LDPC codes," *Electronics Letters*, vol. 43, no. 17, pp. 938-940, Aug. 2007.
- [3] X. Chen, S. Lin, and V. Akella, "QSN - A simple circular-shift network for reconfigurable quasi-cyclic LDPC decoders," *IEEE Transactions on Circuits and Systems - II: Express Briefs*, vol. 57, no. 10, pp. 782-786, Oct. 2010.

- [4] B. Xiang, D. Bao, S. Huang, and X. Zeng, "An 847–955 Mb/s 342–397 mW dual-path fully-overlapped QC-LDPC decoder for WiMAX system in 0.13 μm CMOS," *IEEE Journal of Solid-State Circuits*, vol. 46, no. 6, pp. 1416-1432, Jun. 2011.
- [5] D. Oh and K. K. Parhi, "Low-complexity switch network for reconfigurable LDPC decoders," *IEEE Transactions on Very Large Scale Integration(VLSI) Systems*, vol. 18, no. 1, pp. 85-94, Jan. 2010.



강형주(Hyeong-Ju Kang)

1998년 한국과학기술원 전기및전자공학과 학사
2000년 한국과학기술원 전기및전자공학과 석사
2005년 한국과학기술원 전자전산학과 박사
2005년~2006년 (주)매그나칩반도체 선임연구원
2006년~2009년 (주)지씨티리씨치 선임연구원
2009년~현재 한국기술교육대학교 컴퓨터공학부 전임강사/조교수
※ 관심분야 : VLSI설계 및 CAD, 마이크로프로세서 설계, 통신 모뎀 설계