



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2017-0103729
(43) 공개일자 2017년09월13일

(51) 국제특허분류(Int. Cl.)
H03M 13/27 (2006.01) H03M 13/11 (2006.01)
H04L 1/00 (2006.01)
(52) CPC특허분류
H03M 13/2703 (2013.01)
H03M 13/1102 (2013.01)
(21) 출원번호 10-2017-0111423(분할)
(22) 출원일자 2017년08월31일
심사청구일자 없음
(62) 원출원 특허 10-2015-0000672
원출원일자 2015년01월05일
심사청구일자 2017년04월28일
(30) 우선권주장
62/001,168 2014년05월21일 미국(US)

(71) 출원인
삼성전자주식회사
경기도 수원시 영통구 삼성로 129 (매탄동)
(72) 발명자
정홍실
경기도 수원시 영통구 도청로17번길 23, 5301동
1801호(이의동, 자연앤자이)
김경중
서울특별시 구로구 디지털로 316, 403호(구로동,
한라오피스텔)
(뒷면에 계속)
(74) 대리인
정홍식, 김태현

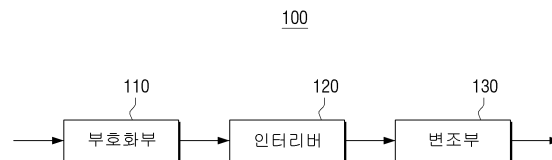
전체 청구항 수 : 총 2 항

(54) 발명의 명칭 송신 장치 및 그의 인터리빙 방법

(57) 요약

송신 장치가 개시된다. 본 송신 장치는 LDPC 부호화를 수행하여 LDPC 부호어를 생성하는 부호화부, LDPC 부호어를 인터리빙하는 인터리버 및 인터리빙된 LDPC 부호어를 변조 심볼에 맵핑하는 변조부를 포함하며, 변조부는 LDPC 부호어를 구성하는 복수의 비트 그룹 중 기설정된 비트 그룹에 포함된 비트를 변조 심볼 내의 기설정된 비트에 맵핑할 수 있다.

대표도 - 도1



(52) CPC특허분류

H04L 1/0071 (2013.01)

(72) 발명자

명세호

서울특별시 강남구 테헤란로52길 16, 102동 2101
호(역삼동, 테헤란IPARK)

다니엘 앤솔레구이

영국 스테인즈 티더블유18 4큐이 미들섹스 사우스
스트리트커뮤니케이션즈 하우스

벨카셈 모우호우체

영국 스테인즈 티더블유18 4큐이 미들섹스 사우스
스트리트 커뮤니케이션즈 하우스

명세서

청구범위

청구항 1

인터리빙 방법에 있어서,

코드워드를 복수의 비트 그룹으로 구분하는 단계;

상기 복수의 비트 그룹을 인터리빙하여 인터리빙된 코드워드를 제공하는 단계;

상기 인터리빙된 코드워드의 비트들을 패러렐 스트림(parallel streams)으로 디멀티플렉싱하여 데이터 셀들을 생성하는 단계;

상기 데이터 셀들을 256-QAM(quadrature amplitude modulation)에 대한 성상점들에 맵핑하는 단계; 및

상기 성상점들에 기초한 신호를 전송하는 단계;를 포함하고,

상기 패리티 비트들은, 입력 비트들을 5/15의 코드 레이트 및 16200의 코드 길이에 따른 LDPC(low density parity check) 코드에 기초하여 인코딩함에 의해 생성되고,

상기 복수의 비트 그룹을 인터리빙하는 단계는, 하기의 관계식에 기초하여 상기 복수의 비트 그룹을 인터리빙하는, 인터리빙 방법:

$$Y_j = X_{\pi(j)} \text{ for } (0 \leq j < N_{\text{group}}),$$

여기에서, X_j 는 상기 복수의 비트 그룹 중 j 번째 비트 그룹이고, Y_j 는 상기 인터리빙된 j 번째 비트 그룹이고, N_{group} 은 상기 복수의 비트 그룹의 개수이고, $\pi(j)$ 는 상기 인터리빙을 위한 퍼뮤테이션 오더이고,

상기 $\pi(j)$ 는 하기의 표와 같이 정의된다.

Code Rate	Order of interleaving $\pi(j) \ (0 \leq j < 45)$																							
	j	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
		23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	
5/15	$\pi(j)$	4	23	3	6	18	5	0	2	7	26	21	27	39	42	38	31	1	34	20	37	40	24	43
		25	33	9	22	36	30	35	11	10	17	32	13	12	41	15	14	19	16	8	44	29	28	

청구항 2

제1항에 있어서,

상기 복수의 비트 그룹 각각은, 360 개의 비트들을 포함하는 것을 특징으로 하는 송신 장치.

발명의 설명

기술 분야

[0001] 본 발명은 송신 장치 및 그의 인터리빙 방법에 관한 것으로, 더욱 상세하게는 데이터를 처리하여 전송하는 송신 장치 및 그의 인터리빙 방법에 관한 것이다.

배경 기술

[0002] 21세기 정보화 사회에서 방송 통신 서비스는 본격적인 디지털화, 다채널화, 광대역화, 고품질화의 시대를 맞이하고 있다. 특히 최근에 고화질 디지털 TV 및 PMP, 휴대방송 기기 보급이 확대됨에 따라 디지털 방송 서비스도 다양한 수신방식 지원에 대한 요구가 증대되고 있다.

[0003] 이러한 요구에 따라 표준 그룹에서는 다양한 표준을 제정하여, 사용자의 니즈를 만족시킬 수 있는 다양한 서비

스를 제공하고 있는 실정에서, 보다 우수한 복호화 및 수신 성능을 통해 보다 나은 서비스를 제공하기 위한 방안의 모색이 요청된다.

발명의 내용

해결하려는 과제

[0004] 본 발명은 상술한 필요성에 따른 것으로, 본 발명의 목적은 LDPC 부호어를 구성하는 복수의 비트 그룹 중 기설정된 비트 그룹에 포함된 비트를 변조 심볼 내의 기설정된 비트에 맵핑시켜 전송할 수 있는 송신 장치 및 그의 인터리빙 방법을 제공함에 있다.

과제의 해결 수단

[0005] 이상과 같은 목적을 달성하기 위한 본 발명의 일 실시 예에 따른 송신 장치는 LDPC 부호화를 수행하여 LDPC 부호어를 생성하는 부호화부, 상기 LDPC 부호어를 인터리빙하는 인터리버 및 상기 인터리빙된 LDPC 부호어를 변조 심볼에 맵핑하는 변조부를 포함하며, 상기 변조부는 상기 LDPC 부호어를 구성하는 복수의 비트 그룹 중 기설정된 비트 그룹에 포함된 비트를 상기 변조 심볼 내의 기설정된 비트에 맵핑한다.

[0006] 여기에서, 상기 복수의 비트 그룹 각각은 M 개의 비트로 구성되며, M 은 N_{ldpc} 와 K_{ldpc} 의 공약수이며, $Q_{ldpc}=(N_{ldpc}-K_{ldpc})/M$ 이 성립하도록 결정될 수 있다. 이 경우, Q_{ldpc} 는 상기 패리티 검사 행렬을 구성하는 정보어 부분 행렬의 열 그룹 내에서 열들에 대한 시클릭 쉬프트 파라미터 값, N_{ldpc} 는 상기 LDPC 부호어의 길이, K_{ldpc} 는 상기 LDPC 부호어를 구성하는 정보어 비트들의 길이이다.

[0007] 한편, 상기 인터리버는 상기 LDPC 부호어를 구성하는 패리티 비트들을 인터리빙하는 패리티 인터리버, 상기 패리티 인터리빙된 LDPC 부호어를 상기 복수의 비트 그룹으로 구분하고, 상기 복수의 비트 그룹의 순서를 비트 그룹 단위로 재정렬하는 그룹 인터리버 및, 상기 순서가 재정렬된 복수의 비트 그룹을 인터리빙하는 블록 인터리버를 포함할 수 있다.

[0008] 여기에서, 상기 그룹 인터리버는 하기의 수학식 21에 기초하여 상기 복수의 비트 그룹의 순서를 비트 그룹 단위로 재정렬할 수 있다.

[0009] 이 경우, 수학식 21에서 $\pi(j)$ 는 LDPC 부호어의 길이, 변조 방식 및 부호율 중 적어도 하나에 기초하여 결정될 수 있다.

[0010] 그리고, 수학식 21에서 $\pi(j)$ 는 LDPC 부호어의 길이가 16200이고, 변조 방식이 256-QAM이고 부호율이 7/15인 경우, 하기의 표 16과 같이 정의될 수 있다.

[0011] 또한, 수학식 21에서 $\pi(j)$ 는 LDPC 부호어의 길이가 16200이고 변조 방식이 256-QAM이고 부호율이 9/15인 경우, 하기의 표 17과 같이 정의될 수 있다.

[0012] 그리고, 수학식 21에서 $\pi(j)$ 는 LDPC 부호어의 길이가 16200이고 변조 방식이 256-QAM이고 부호율이 13/15인 경우, 하기의 표 19와 같이 정의될 수 있다.

[0013] 한편, 상기 인터리버는 상기 LDPC 부호어를 상기 복수의 비트 그룹으로 구분하고, 상기 복수의 비트 그룹의 순서를 비트 그룹 단위로 재정렬하는 그룹 인터리버 및, 상기 순서가 재정렬된 복수의 비트 그룹을 인터리빙하는 블록 인터리버를 포함할 수 있다.

[0014] 여기에서, 상기 그룹 인터리버는 하기의 수학식 21에 기초하여 상기 복수의 비트 그룹의 순서를 비트 그룹 단위로 재정렬할 수 있다.

[0015] 이 경우, 수학식 21에서 $\pi(j)$ 는 LDPC 부호어의 길이, 변조 방식 및 부호율 중 적어도 하나에 기초하여 결정될 수 있다.

[0016] 또한, 수학식 21에서 $\pi(j)$ 는 DPC 부호어의 길이가 16200이고 변조 방식이 256-QAM이고 부호율이 5/15인 경우, 하기의 표 15와 같이 정의될 수 있다.

[0017] 한편, 상기 블록 인터리버는 상기 복수의 비트 그룹을 비트 그룹 단위로 복수의 열 각각에 열 방향으로 라이트하고, 상기 복수의 비트 그룹이 비트 그룹 단위로 라이트된 상기 복수의 열의 각 행을 행 방향으로 리드하여 인

터리빙을 수행할 수 있다.

- [0018] 여기에서, 상기 블록 인터리버는 상기 복수의 비트 그룹 중 상기 복수의 열 각각에 비트 그룹 단위로 라이트 가능한 적어도 일부의 비트 그룹을 상기 복수의 열 각각에 순차적으로 라이트한 후, 상기 복수의 열 각각에서 상기 적어도 일부의 비트 그룹이 비트 그룹 단위로 라이트 되고 남은 나머지 영역에 나머지 비트 그룹을 분할하여 라이트할 수 있다.
- [0019] 한편, 본 발명의 일 실시 예에 따른 송신 장치의 인터리빙 방법은 LDPC 부호화를 수행하여 LDPC 부호어를 생성하는 단계, 상기 LDPC 부호어를 인터리빙하는 단계 및 상기 인터리빙된 LDPC 부호어를 변조 심볼에 맵핑하는 단계를 포함하며, 상기 맵핑하는 단계는 상기 LDPC 부호어를 구성하는 복수의 비트 그룹 중 기설정된 비트 그룹에 포함된 비트를 상기 변조 심볼 내의 기설정된 비트에 맵핑할 수 있다.
- [0020] 여기에서, 상기 복수의 비트 그룹 각각은 M 개의 비트로 구성되며, M 은 N_{ldpc} 와 K_{ldpc} 의 공약수이며, $Q_{ldpc}=(N_{ldpc}-K_{ldpc})/M$ 이 성립하도록 결정될 수 있다. 이 경우, Q_{ldpc} 는 상기 패리티 검사 행렬을 구성하는 정보어 부분 행렬의 열 그룹 내에서 열들에 대한 시클릭 쉬프트 파라미터 값, N_{ldpc} 는 상기 LDPC 부호어의 길이, K_{ldpc} 는 상기 LDPC 부호어를 구성하는 정보어 비트들의 길이이다.
- [0021] 한편, 상기 인터리빙하는 단계는 상기 LDPC 부호어를 구성하는 패리티 비트들을 인터리빙하는 단계, 상기 패리티 인터리빙된 LDPC 부호어를 상기 복수의 비트 그룹으로 구분하고, 상기 복수의 비트 그룹의 순서를 비트 그룹 단위로 재정렬하는 단계 및, 상기 순서가 재정렬된 복수의 비트 그룹을 인터리빙하는 단계를 포함할 수 있다.
- [0022] 여기에서, 상기 복수의 비트 그룹의 순서를 비트 그룹 단위로 재정렬하는 단계는 하기의 수학적 식 21에 기초하여 상기 복수의 비트 그룹의 순서를 비트 그룹 단위로 재정렬할 수 있다.
- [0023] 또한, 수학적 식 21에서 $\pi(j)$ 는 LDPC 부호어의 길이, 변조 방식 및 부호율 중 적어도 하나에 기초하여 결정될 수 있다.
- [0024] 그리고, 수학적 식 21에서 $\pi(j)$ 는 LDPC 부호어의 길이가 16200이고 변조 방식이 256-QAM이고 부호율이 7/15인 경우, 하기의 표 16과 같이 정의될 수 있다.
- [0025] 또한, 수학적 식 21에서 $\pi(j)$ 는 LDPC 부호어의 길이가 16200이고 변조 방식이 256-QAM이고 부호율이 9/15인 경우, 하기의 표 17과 같이 정의될 수 있다.
- [0026] 그리고, 수학적 식 21에서 $\pi(j)$ 는 LDPC 부호어의 길이가 16200이고 변조 방식이 256-QAM이고 부호율이 13/15인 경우, 하기의 표 19와 같이 정의될 수 있다.
- [0027] 한편, 상기 인터리빙하는 단계는 상기 LDPC 부호어를 상기 복수의 비트 그룹으로 구분하고, 상기 복수의 비트 그룹의 순서를 비트 그룹 단위로 재정렬하는 단계 및, 상기 순서가 재정렬된 복수의 비트 그룹을 인터리빙하는 단계를 포함할 수 있다.
- [0028] 여기에서, 상기 복수의 비트 그룹의 순서를 비트 그룹 단위로 재정렬하는 단계는 하기의 수학적 식 21에 기초하여 상기 복수의 비트 그룹의 순서를 비트 그룹 단위로 재정렬할 수 있다.
- [0029] 또한, 수학적 식 21에서 $\pi(j)$ 는 LDPC 부호어의 길이, 변조 방식 및 부호율 중 적어도 하나에 기초하여 결정될 수 있다.
- [0030] 그리고, 수학적 식 21에서 $\pi(j)$ 는 LDPC 부호어의 길이가 16200이고 변조 방식이 256-QAM이고 부호율이 5/15인 경우, 하기의 표 15와 같이 정의될 수 있다.
- [0031] 한편, 상기 복수의 비트 그룹을 인터리빙하는 단계는 상기 복수의 비트 그룹을 비트 그룹 단위로 복수의 열 각각에 열 방향으로 라이트하고, 상기 복수의 비트 그룹이 비트 그룹 단위로 라이트된 상기 복수의 열의 각 행을 행 방향으로 리드하여 인터리빙을 수행할 수 있다.
- [0032] 여기에서, 상기 복수의 비트 그룹을 인터리빙하는 단계는 상기 복수의 비트 그룹 중 상기 복수의 열 각각에 비트 그룹 단위로 라이트 가능한 적어도 일부의 비트 그룹을 상기 복수의 열 각각에 순차적으로 라이트한 후, 상기 복수의 열 각각에서 상기 적어도 일부의 비트 그룹이 비트 그룹 단위로 라이트 되고 남은 나머지 영역에 나머지 비트 그룹을 분할하여 라이트할 수 있다.

발명의 효과

[0033] 이러한 본 발명의 다양한 실시 예에 따르면, 보다 우수한 복호화 및 수신 성능을 제공할 수 있게 된다.

도면의 간단한 설명

[0034] 도 1은 본 발명의 일 실시 예에 따른 송신 장치의 구성을 설명하기 위한 블록도,
 도 2 내지 도 4는 본 발명의 다양한 실시 예에 따른 패리티 검사 행렬의 구조를 설명하기 위한 도면들,
 도 5는 본 발명의 일 실시 예에 따른 인터리버의 구성을 설명하기 위한 블록도,
 도 6 내지 도 8은 본 발명의 일 실시 예에 따른 인터리빙 방법을 설명하기 위한 도면들,
 도 9 내지 도 14는 본 발명의 일 실시 예에 따른 블록 인터리버의 인터리빙 동작을 설명하기 위한 도면들,
 도 15는 본 발명의 일 실시 예에 따른 디멀티플렉서의 동작을 설명하기 위한 도면,
 도 16은 본 발명의 일 실시 예에 따른 수신 장치의 구성을 설명하기 위한 블록도,
 도 17은 본 발명의 일 실시 예에 따른 디인터리버의 구성을 설명하기 위한 블록도,
 도 18은 본 발명의 일 실시 예에 따른 블록 디인터리버의 디인터리빙 동작을 설명하기 위한 도면, 그리고
 도 19는 본 발명의 일 실시 예에 따른 인터리빙 방법을 설명하기 위한 흐름도이다.

발명을 실시하기 위한 구체적인 내용

[0035] 이하에서는 첨부된 도면을 참조하여 본 발명을 더욱 상세하게 설명한다.

[0036] 도 1은 본 발명의 일 실시 예에 따른 송신 장치의 구성을 설명하기 위한 블록도이다. 도 1에 따르면, 송신 장치(100)는 부호화부(110), 인터리버(120) 및 변조부(130)(또는, '성상도 맵퍼'라 할 수 있다)를 포함한다.

[0037] 부호화부(110)는 패리티 검사 행렬(Parity Check Matrix, PCM)에 기초하여 LDPC(Low Density Parity Check) 부호화를 수행하여 LDPC 부호어를 생성한다. 이를 위해, 부호화부(110)는 LPDC 부호화를 수행하는 LDPC 인코더(미도시)를 포함할 수 있다.

[0038] 구체적으로, 부호화부(110)는 입력되는 비트들을 정보어 비트들로 LDPC 부호화를 수행하여, 정보어 비트들과 패리티 비트들(즉, LDPC 패리티 비트들)로 구성된 LDPC 부호어를 생성할 수 있다. 이 경우, LPDC 부호는 시스템메틱 코드(systematic code)라는 점에서, 정보어 비트들이 LDPC 부호어에 그대로 포함될 수 있다.

[0039] 여기에서, LDPC 부호어는 정보어 비트들과 패리티 비트들로 구성된다. 예를 들어, LDPC 부호어는 N_{ldpc} 개의 비트로 구성되며, K_{ldpc} 개의 비트로 이루어진 정보어 비트들과 $N_{parity}=N_{ldpc}-K_{ldpc}$ 개의 비트로 이루어진 패리티 비트들을 포함할 수 있다.

[0040] 이 경우, 부호화부(110)는 패리티 검사 행렬에 기초하여 LDPC 부호화를 수행하여 LDPC 부호어를 생성할 수 있다. 즉, LDPC 부호화를 수행하는 과정은 $H \cdot C^T=0$ 을 만족하도록 LDPC 부호어를 생성하는 과정이라는 점에서, 부호화부(110)는 LDPC 부호화 시 패리티 검사 행렬을 이용할 수 있다. 여기에서, H는 패리티 검사 행렬을 나타내고, C는 LDPC 부호어를 나타낸다.

[0041] 이를 위해, 송신 장치(100)는 별도의 메모리를 구비하여 다양한 형태의 패리티 검사 행렬을 기저장하고 있을 수 있다.

[0042] 예를 들어, 송신 장치(100)는 DVB-C2(Digital Video Broadcasting-Cable version 2), DVB-S2(Digital Video Broadcasting-Satellite-Second Generation), DVB-T2(Digital Video Broadcasting-Second Generation Terrestria) 등의 규격에서 정의된 패리티 검사 행렬을 기저장하거나, 또는 현재 표준 제정 중인 복미 디지털 방송 표준 시스템 ATSC(Advanced Television Systems Committee) 3.0 규격에서 정의된 패리티 검사 행렬을 기저장하고 있을 수 있다. 하지만, 이는 일 예일 뿐이며, 송신 장치(100)는 이 외에도 다양한 형태의 패리티 검사 행렬을 기저장하고 있을 수 있다.

[0043] 이하에서는 첨부된 도면을 참조하여, 본 발명의 다양한 실시 예들에 따른 패리티 검사 행렬의 구조에 대해 설명하도록 한다. 이하에서의 패리티 검사 행렬에서 1을 제외한 부분의 원소는 0이다.

- [0044] 일 예로, 본 발명의 일 실시 예에 따른 패리티 검사 행렬은 도 2와 같은 구조를 가질 수 있다.
- [0045] 도 2를 참조하면, 패리티 검사 행렬(200)은 정보어 비트들에 대응되는 부분 행렬인 정보어 부분 행렬(210)과 패리티 비트들에 대응되는 부분 행렬인 패리티 부분 행렬(220)로 구성된다.
- [0046] 정보어 부분 행렬(210)은 K_{ldpc} 개의 열(column)을 포함하고, 패리티 부분 행렬(220)은 $N_{parity}=N_{ldpc}-K_{ldpc}$ 개의 열을 포함한다. 한편, 패리티 검사 행렬(200)의 행(row)의 개수는 패리티 부분 행렬(220)의 열의 개수 $N_{parity}=N_{ldpc}-K_{ldpc}$ 와 동일하다.
- [0047] 또한, 패리티 검사 행렬(200)에서 N_{ldpc} 는 LDPC 부호어의 길이, K_{ldpc} 는 정보어비트들의 길이, $N_{parity}=N_{ldpc}-K_{ldpc}$ 는 패리티 비트들의 길이를 나타낸다. 여기에서, LDPC 부호어, 정보어 비트들 및 패리티 비트들의 길이는 LDPC 부호어, 정보어 비트들 및 패리티 비트들 각각에 포함되는 비트들의 개수를 의미한다.
- [0048] 이하에서는 정보어 부분 행렬(210)과 패리티 부분 행렬(220)의 구조에 대해 살펴보도록 한다.
- [0049] 정보어 부분 행렬(210)은 K_{ldpc} 개의 열(즉, 0 번째 열부터 $K_{ldpc}-1$ 번째 열)을 포함하는 행렬로, 다음과 같은 규칙을 따른다.
- [0050] 첫째, 정보어 부분 행렬(210)을 구성하는 K_{ldpc} 개의 열들은 M 개씩 동일한 그룹에 속하며, 총 K_{ldpc}/M 개의 열 그룹(column group)들로 구분된다. 동일한 열 그룹 내에 속한 열들은 서로 Q_{ldpc} 만큼 시클릭 쉬프트(cyclic shift)된 관계를 가진다. 즉, Q_{ldpc} 는 패리티 검사 행렬을 구성하는 정보어 부분 행렬의 열 그룹 내에서 열들에 대한 시클릭 쉬프트 파라미터 값으로 볼 수 있다.
- [0051] 여기에서, M 은 정보어 부분 행렬(210)에서 열의 패턴이 반복되는 간격이고(일 예로, $M=360$)이고, Q_{ldpc} 는 정보어 부분 행렬(210)에서 각 열이 시클릭 쉬프트되는 크기이다. M 은 N_{ldpc} 와 K_{ldpc} 의 공약수(common divisor)이며, $Q_{ldpc}=(N_{ldpc}-K_{ldpc})/M$ 이 성립하도록 결정된다. 여기에서, M 및 Q_{ldpc} 는 정수이고, K_{ldpc}/M 도 정수가 된다. 한편, M 및 Q_{ldpc} 는 LDPC 부호어의 길이와 부호율(code rate, CR)에 따라 다양한 값을 가질 수 있다.
- [0052] 예를 들어, $M=360$ 이고 LDPC 부호어의 길이 N_{ldpc} 가 64800인 경우 Q_{ldpc} 는 하기의 표 1과 같이 정의되고, $M=360$ 이고 LDPC 부호어의 길이 N_{ldpc} 가 16200인 경우 Q_{ldpc} 는 하기의 표 2와 같이 정의될 수 있다.

표 1

부호율	N_{ldpc}	M	Q_{ldpc}
5/15	64800	360	120
6/15	64800	360	108
7/15	64800	360	96
8/15	64800	360	84
9/15	64800	360	72
10/15	64800	360	60
11/15	64800	360	48
12/15	64800	360	36
13/15	64800	360	24

[0053]

표 2

부호율	N_{ldpc}	M	Q_{ldpc}
5/15	16200	360	30
6/15	16200	360	27
7/15	16200	360	24
8/15	16200	360	21
9/15	16200	360	18
10/15	16200	360	15
11/15	16200	360	12
12/15	16200	360	9
13/15	16200	360	6

[0054]

[0055]

둘째, i 번째($i=0,1,\dots,K_{ldpc}/M-1$) 열 그룹의 0 번째 열의 차수(degree)(여기에서, 차수는 열에 존재하는 1 값의 개수로, 동일한 열 그룹에 속하는 모든 열들의 차수는 동일하다)를 D_i 라 하고, i 번째 열 그룹의 0 번째 열에서

1이 있는 각 행의 위치(또는, 인덱스)를 $R_{i,0}^{(0)}, R_{i,0}^{(1)}, \dots, R_{i,0}^{(D_i-1)}$ 이라 하면, i 번째 열 그룹 내의 j 번째 열에서 k 번째 1이 위치한 행의 인덱스 $R_{i,j}^{(k)}$ 는 하기의 수학적 식 1과 같이 결정된다.

수학적 식 1

$$R_{i,j}^{(k)} = R_{i,(j-1)}^{(k)} + Q_{ldpc} \bmod (N_{ldpc} - K_{ldpc})$$

[0056]

[0057]

여기에서, $k=0,1,2,\dots,D_i-1$, $i=0,1,\dots,K_{ldpc}/M-1$, $j=1,2,\dots,M-1$ 이다.

[0058]

한편, 수학적 식 1은 하기의 수학적 식 2와 같이 동일하게 표현될 수 있다.

수학적 식 2

$$R_{i,j}^{(k)} = \left\{ R_{i,0}^{(k)} + (j \bmod M) \times Q_{ldpc} \right\} \bmod (N_{ldpc} - K_{ldpc})$$

[0059]

[0060]

여기에서, $k=0,1,2,\dots,D_i-1$, $i=0,1,\dots,K_{ldpc}/M-1$, $j=1,2,\dots,M-1$ 이다. 여기서, $j=1,2,\dots,M-1$ 이기 때문에 수학적 식 2의 $(j \bmod M)$ 은 j 로 볼 수 있다.

[0061]

이들 수학적 식에서, $R_{i,j}^{(k)}$ 는 i 번째 열 그룹 내의 j 번째 열에서 k 번째 1이 위치한 행의 인덱스, N_{ldpc} 는 LDPC 부호어의 길이, K_{ldpc} 는 정보어 비트들의 길이, D_i 는 i 번째 열 그룹에 속하는 열들의 차수, M 은 하나의 열 그룹에 속하는 열의 개수, Q_{ldpc} 는 각 열이 시클릭 쉬프트되는 크기를 의미한다.

[0062]

결국, 이들 수학적 식을 참조하면 $R_{i,j}^{(k)}$ 값만을 알면 i 번째 열 그룹 내의 j 번째 열에서 k 번째 1이 있는 행의 인덱스 $R_{i,j}^{(k)}$ 를 알 수 있게 된다. 그러므로, 각각의 열 그룹 내의 0 번째 열에서 k 번째 1이 있는 행의 인덱스 값을 저장하면, 도 2의 구조를 갖는 패리티 검사 행렬(200)(즉, 패리티 검사 행렬(200)의 정보어 부분 행렬

(210))에서 1이 있는 열과 행의 위치가 파악될 수 있다.

[0063] 상술한 규칙들에 따르면, i 번째 열 그룹에 속하는 열들의 차수는 모두 D_i 로 동일하다. 따라서, 상술한 규칙들에 따라 패리티 검사 행렬에 대한 정보를 저장하고 있는 LDPC 부호는 다음과 같이 간략하게 표현될 수 있다.

[0064] 예를 들어, N_{ldpc} 가 30, K_{ldpc} 가 15, Q_{ldpc} 가 3인 경우, 3 개의 열 그룹의 0 번째 열에서 1이 위치한 행의 위치 정보는 하기 수학적 식 3과 같은 수열들로 표현될 수 있으며, 이는 '무게-1 위치 수열(weight-1 position sequence)'이라 지칭될 수 있다.

수학적 식 3

$$R_{1,0}^{(1)} = 1, R_{1,0}^{(2)} = 2, R_{1,0}^{(3)} = 8, R_{1,0}^{(4)} = 10,$$

$$R_{2,0}^{(1)} = 0, R_{2,0}^{(2)} = 9, R_{2,0}^{(3)} = 13,$$

$$R_{3,0}^{(1)} = 0, R_{3,0}^{(2)} = 14.$$

[0065]

[0066] 여기에서, $R_{i,j}^{(k)}$ 는 i 번째 열 그룹 내의 j 번째 열에서 k 번째 1이 있는 행의 인덱스를 의미한다.

[0067] 각 열 그룹의 0 번째 열에서 1이 위치한 행의 인덱스를 나타내는 수학적 식 3과 같은 무게-1 위치 수열들은 하기의 표 3과 같이 보다 간략하게 표현될 수 있다.

표 3

1	2	8	10
0	9	13	
0	14		

[0068]

[0069] 표 3은 패리티 검사 행렬에서 1 값을 가지는 원소의 위치를 나타낸 것으로서, i 번째 무게-1 위치 수열은 i 번째 열 그룹에 속한 0 번째 열에서 1이 있는 행의 인덱스들로 표현된다.

[0070] 상술한 내용에 기초하여 본 발명의 일 실시 예에 다른 패리티 검사 행렬의 정보어 부분 행렬(210)은 하기의 표 4 내지 표 12에 의해 정의될 수 있다.

[0071] 구체적으로, 표 4 내지 표 12는 정보어 부분 행렬(210)의 i 번째 열 그룹의 0 번째 열에서 1이 위치한 행의 인덱스들을 나타낸다. 즉, 정보어 부분 행렬(210)은 각각 M 개의 열을 포함하는 복수의 열 그룹으로 구성되며, 복수의 열 그룹 각각의 0 번째 열에서 1의 위치는 표 4 내지 표 12에 의해 정의될 수 있다.

[0072] 여기에서, i 번째 열 그룹의 0 번째 열에서 1이 위치한 행의 인덱스들은 “addresses of parity bit accumulators”를 의미한다. 한편, “addresses of parity bit accumulators”는 DVB-C2/S2/T2 등의 규격 또는 현재 표준 제정 중인 ATSC 3.0 규격에서 정의된 바와 동일한 의미를 가진다는 점에서 구체적인 설명은 생략하도록 한다.

[0073] 일 예로, LDPC 부호어의 길이 N_{ldpc} 가 16200, 부호율이 5/15, M 이 360인 경우, 정보어 부분 행렬(210)의 i 번째 열 그룹의 0 번째 열에서 1이 위치한 행의 인덱스들은 하기의 표 4와 같다.

표 4

i	i번째 열 그룹의 0번째 열에서 10이 위치한 행의 인덱스
0	245 449 491 980 1064 1194 1277 1671 2026 3186 4399 4900 5283 5413 5558 6570 7492 7768 7837 7984 8306 8483 8685 9357 9642 10045 10179 10261 10338 10412
1	1318 1584 1682 1860 1954 2000 2062 3387 3441 3879 3931 4240 4302 4446 4603 5117 5588 5675 5793 5955 6097 6221 6449 6616 7218 7394 9535 9896 10009 10763
2	105 472 785 911 1168 1450 2550 2851 3277 3624 4128 4460 4572 4669 4783 5102 5133 5199 5905 6647 7028 7086 7703 8121 8217 9149 9304 9476 9736 9884
3	1217 5338 5737 8334
4	855 994 2979 9443
5	7506 7811 9212 9982
6	848 3313 3380 3990
7	2095 4113 4620 9946
8	1488 2396 6130 7483
9	1002 2241 7067 10418
10	2008 3199 7215 7502
11	1161 7705 8194 8534
12	2316 4803 8649 9359
13	125 1880 3177
14	1141 8033 9072

[0074]

[0075] 다른 예로, LDPC 부호어의 길이 N_{ldpc} 가 16200, 부호율이 7/15, M이 360인 경우, 정보어 부분 행렬(210)의 i 번째 열 그룹의 0 번째 열에서 1이 위치한 행의 인덱스들은 하기의 표 5 또는 표 6과 같다.

표 5

i	i번째 열 그룹의 0번째 열에서 10이 위치한 행의 인덱스
0	553 742 9011327 1544 2179 2519 3131 3280 3603 3789 37924253 5340 5934 5962 6004 6698 7793 8001 8058 8126 8276 8559
1	503 590 598 1185 1266 1336 1806 2473 3021 3356 3490 3680 3936 4501 4659 5891 6132 6340 6602 7447 8007 8045 80598249
2	795 831 9471330 1502 2041 2328 2513 2814 2829 4048 4802 6044 6109 6461 6777 6800 7099 7126 8095 8428 8519 8556 8610
3	601 787 8991757 2259 2518 2783 2816 2823 2949 3396 43304494 4684 4700 4837 4881 4975 5130 5464 65546912 7094 8297
4	4229 5628 7917 7992
5	1506 3374 4174 5547
6	4275 5650 8208 8533
7	1504 1747 3433 6345
8	3659 6955 7575 7852
9	607 3002 4913 6453
10	3533 6860 7895 8048
11	4094 6366 8314
12	2206 4513 5411
13	32 3882 5149
14	389 3121 4626
15	1308 4419 6520
16	2092 2373 6849
17	1815 3679 7152
18	3582 3979 6948
19	1049 2135 3754
20	2276 4442 6591

[0076]

표 6

i	i번째 열 그룹의 0번째 열에서 1이 위치한 행의 인덱스
0	432 655 893 942 1285 1427 1738 2199 2441 2565 2932 3201 4144 4419 4678 4963 5423 5922 6433 6564 6656 7478 7514 7892
1	220 453 690 826 1116 1425 1488 1901 3119 3182 3568 3800 3953 4071 4782 5038 5555 6836 6871 7131 7609 7850 8317 8443
2	300 454 497 930 1757 2145 2314 2372 2467 2819 3191 3256 3699 3984 4538 4965 5461 5742 5912 6135 6649 7636 8078 8455
3	24 65 565 609 990 1319 1394 1465 1918 1976 2463 2987 3330 3677 4195 4240 4947 5372 6453 6950 7066 8412 8500 8599
4	1373 4668 5324 7777
5	189 3930 5766 6877
6	3 2961 4207 5747
7	1108 4768 6743 7106
8	1282 2274 2750 6204
9	2279 2587 2737 6344
10	2889 3164 7275 8040
11	133 2734 5081 8386
12	437 3203 7121
13	4280 7128 8490
14	619 4563 6206
15	2799 6814 6991
16	244 4212 5925
17	1719 7657 8554
18	53 1895 6685
19	584 5420 6856
20	2958 5834 8103

[0077]

[0078] 다른 예로, LDPC 부호어의 길이 N_{ldpc} 가 16200, 부호율이 9/15, M이 360인 경우, 정보어 부분 행렬(210)의 i 번째 열 그룹의 0 번째 열에서 1이 위치한 행의 인덱스들은 하기의 표 7 또는 표 8과 같다.

표 7

i	i번째 열 그룹의 0번째 열에서 1이 위치한 행의 인덱스
0	212 255 540 967 1033 1517 1538 3124 3408 3800 4373 4864 4905 5163 5177 6186
1	275 660 1351 2211 28763063 3433 4088 4273 4544 4618 4632 5548 6101 6111 6136
2	279 335 494865 1662 1681 3414 3775 4252 45955272 5471 5796 5907 5986 6008
3	345 352 3094 3188 42974338 4490 4865 5303 6477
4	222 681 1218 3169 3850 4878 4954 5666 6001 6237
5	172 512 1536 1559 21792227 3334 4049 6464
6	716 934 1694 2890 3276 3608 4332 4468 5945
7	1133 1593 1825 2571 3017 4251 5221 5639 5845
8	1076 1222 6465
9	159 5064 6078
10	374 4073 5357
11	2833 5526 5845
12	1594 3639 5419
13	1028 1392 4239
14	115 622 2175
15	300 1748 6245
16	2724 3276 5349
17	1433 6117 6448
18	485 663 4955
19	711 1132 4315
20	177 3266 4339
21	1171 4841 4982
22	33 1584 3692
23	2820 3485 4249
24	1716 2428 3125
25	250 2275 6338
26	108 1719 4961

[0079]

표 8

i	i번째 열 그룹의 0번째 열에서 1이 위치한 행의 인덱스
0	350 462 1291 1383 1821 2235 2493 3328 3353 3772 3872 3923 4259 4426 4542 4972 5347 6217 6246 6332 6386
1	177 869 1214 1253 1398 1482 1737 2014 2161 2331 3108 3297 3438 4388 4430 4456 4522 4783 5273 6037 6395
2	347 501 658 966 1622 1659 1934 2117 2527 3168 3231 3379 3427 3739 4218 4497 4894 5000 5167 5728 5975
3	319 398 599 1143 1796 3198 3521 3886 4139 4453 4556 4636 4688 4753 4986 5199 5224 5496 5698 5724 6123
4	162 257 304 524 945 1695 1855 2527 2780 2902 2958 3439 3484 4224 4769 4928 5156 5303 5971 6358 6477
5	807 1695 2941 4276
6	2652 2857 4660 6358
7	329 2100 2412 3632
8	1151 1231 3872 4869
9	1561 3565 5138 5303
10	407 794 1455
11	3438 5683 5749
12	1504 1985 3563
13	440 5021 6321
14	194 3645 5923
15	1217 1462 6422
16	1212 4715 5973
17	4098 5100 5642
18	5512 5857 6226
19	2583 5506 5933
20	784 1801 4890
21	4734 4779 4875
22	938 5081 5377
23	127 4125 4704
24	1244 2178 3352
25	3659 6350 6465
26	1686 3464 4336

[0080]

[0081]

다른 예로, LDPC 부호어의 길이 N_{ldpc} 가 16200, 부호율이 11/15, M이 360인 경우, 정보어 부분 행렬(210)의 i 번째 열 그룹의 0 번째 열에서 1이 위치한 행의 인덱스들은 하기의 표 9 또는 표 10과 같다.

표 9

i	i번째 열 그룹의 0번째 열에서 1이 위치한 행의 인덱스
0	49 719 784 794 968 2382 2685 2873 2974 2995 3540 4179
1	272 281 374 1279 2034 2067 2112 3429 3613 3815 3838 4216
2	206 714 820 1800 1925 2147 2168 2769 2806 3253 3415 4311
3	62 159 166 605 1496 1711 2652 3016 3347 3517 3654 4113
4	363 733 1118 2062 2613 2736 3143 3427 3664 4100 4157 4314
5	57 142 436 983 1364 2105 2113 3074 3639 3835 4164 4242
6	870 921 950 1212 1861 2128 2707 2993 3730 3968 3983 4227
7	185 2684 3263
8	2035 2123 2913
9	883 2221 3521
10	1344 1773 4132
11	438 3178 3650
12	543 756 1639
13	1057 2337 2898
14	171 3298 3929
15	1626 2960 3503
16	484 3050 3323
17	2283 2336 4189
18	2732 4132 4318
19	225 2335 3497
20	600 2246 2658
21	1240 2790 3020
22	301 1097 3539
23	1222 1267 2594
24	1364 2004 3603
25	1142 1185 2147
26	564 1505 2086
27	697 991 2908
28	1467 2073 3462
29	2574 2818 3637
30	748 2577 2772
31	1151 1419 4129
32	164 1238 3401

[0082]

표 10

i	i번째 열 그룹의 0번째 열에서 1이 위치한 행의 인덱스
0	108 297 703 742 1345 1443 1495 1628 1812 2341 2559 2669 2810 2877 3442 3690 3755 3904 4264
1	180 211 477 788 824 1090 1272 1578 1685 1948 2050 2195 2233 2546 2757 2946 3147 3299 3544
2	627 741 1135 1157 1226 1333 1378 1427 1454 1696 1757 1772 2099 2208 2592 3354 3580 4066 4242
3	9 795 959 989 1006 1032 1135 1209 1382 1484 1703 1855 1985 2043 2629 2845 3136 3450 3742
4	230 413 801 829 1108 1170 1291 1759 1793 1827 1976 2000 2423 2466 2917 3010 3600 3782 4143
5	56 142 236 381 1050 1141 1372 1627 1985 2247 2340 3023 3434 3519 3957 4013 4142 4164 4279
6	298 1211 2548 3643
7	73 1070 1614 1748
8	1439 2141 3614
9	284 1564 2629
10	607 660 855
11	1195 2037 2753
12	49 1198 2562
13	296 1145 3540
14	1516 2315 2382
15	154 722 4016
16	759 2375 3825
17	162 194 1749
18	2335 2422 2632
19	6 1172 2583
20	726 1325 1428
21	985 2708 2769
22	255 2801 3181
23	2979 3720 4090
24	208 1428 4094
25	199 3743 3757
26	1229 2059 4282
27	458 1100 1387
28	1199 2481 3284
29	1161 1467 4060
30	959 3014 4144
31	2666 3960 4125
32	2809 3834 4318

[0083]

[0084]

다른 예로, LDPC 부호어의 길이 N_{ldpc} 가 16200, 부호율이 13/15, M이 360인 경우, 정보어 부분 행렬(210)의 i 번째 열 그룹의 0 번째 열에서 1이 위치한 행의 인덱스들은 하기의 표 11 또는 표 12와 같다.

표 11

i	i번째 열 그룹의 0번째 열에서 1이 위치한 행의 인덱스
0	71 334 645 779 786 1124 1131 1267 1379 1554 1766 1798 1939
1	6 183 364 506 512 922 972 981 1039 1121 1537 1840 2111
2	6 71 153 204 253 268 781 799 873 1118 1194 1661 2036
3	6 247 353 581 921 940 1108 1146 1208 1268 1511 1527 1671
4	6 37 466 548 747 1142 1203 1271 1512 1516 1837 1904 2125
5	6 171 863 953 1025 1244 1378 1396 1723 1783 1816 1914 2121
6	1268 1360 1647 1769
7	6 458 1231 1414
8	183 535 1244 1277
9	107 360 498 1456
10	6 2007 2059 2120
11	1480 1523 1670 1927
12	139 573 711 1790
13	6 1541 1889 2023
14	6 374 957 1174
15	287 423 872 1285
16	6 1809 1918
17	65 818 1396
18	590 766 2107
19	192 814 1843
20	775 1163 1256
21	42 735 1415
22	334 1008 2055
23	109 596 1785
24	406 534 1852
25	684 719 1543
26	401 465 1040
27	112 392 621
28	82 897 1950
29	887 1962 2125
30	793 1088 2159
31	723 919 1139
32	610 839 1302
33	218 1080 1816
34	627 1646 1749
35	496 1165 1741
36	916 1055 1662
37	182 722 945
38	5 595 1674

[0085]

표 12

i	i번째 열 그룹의 0번째 열에서 1이 위치한 행의 인덱스
0	37 144 161 199 220 496 510 589 731 808 834 965 1249 1264 1311 1377 1460 1520 1598 1707 1958 2055 2099 2154
1	20 27 165 462 546 583 742 796 1095 1110 1129 1145 1169 1190 1254 1363 1383 1463 1718 1835 1870 1879 2108 2128
2	288 362 463 505 638 691 745 861 1006 1083 1124 1175 1247 1275 1337 1353 1378 1506 1588 1632 1720 1868 1980 2135
3	405 464 478 511 566 574 641 766 785 802 836 996 1128 1239 1247 1449 1491 1537 1616 1643 1668 1950 1975 2149
4	86 192 245 357 363 374 700 713 852 903 992 1174 1245 1277 1342 1369 1381 1417 1463 1712 1900 1962 2053 2118
5	101 327 378 550
6	186 723 1318 1550
7	118 277 504 1835
8	199 407 1776 1965
9	387 1253 1328 1975
10	62 144 1163 2017
11	100 475 572 2136
12	431 865 1568 2055
13	283 640 981 1172
14	220 1038 1903 2147
15	483 1318 1358 2118
16	92 961 1709 1810
17	112 403 1485 2042
18	431 1110 1130 1365
19	587 1005 1206 1588
20	704 1113 1943
21	375 1487 2100
22	1507 1950 2110
23	962 1613 2038
24	554 1295 1501
25	488 784 1446
26	871 1935 1964
27	54 1475 1504
28	1579 1617 2074
29	1856 1967 2131
30	330 1582 2107
31	40 1056 1809
32	1310 1353 1410
33	232 554 1939
34	168 641 1099
35	333 437 1556
36	153 622 745
37	719 931 1188
38	237 638 1607

[0086]

[0087]

한편, 상술한 예에서는 LDPC 부호어의 길이가 16200이고 부호율이 5/15, 7/15, 9/15, 11/15, 13/15인 경우만을 설명하였으나, 이는 일 예에 불과하고 LDPC 부호어의 길이가 64800인 경우이거나 다른 부호율을 갖는 경우에도 정보어 부분 행렬(210)에서 1의 위치가 다양하게 정의될 수 있다.

[0088]

한편, 상술한 표 4 내지 표 12에서 각 i 번째 열 그룹에 대응되는 수열 내의 숫자들의 순서가 바뀌어도 동일한 부호의 패리티 검사 행렬이라는 점에서, 표 4 내지 표 12에서 각 i 번째 열 그룹에 대응되는 수열 내의 순서가 바뀐 경우도 본 발명에서 고려하는 부호의 한 가지 일 예가 될 수 있다.

[0089]

또한, 표 4 내지 표 12에서 각 열 그룹에 대응되는 수열들의 나열 순서가 바뀌어도 부호의 그래프 상의 사이클 특성 및 차수 분포 등의 대수적 특성이 바뀌지 않기 때문에, 표 4 내지 표 12에 나타난 수열들의 나열 순서가 바뀐 경우도 한 가지 일 예가 될 수 있다.

[0090]

또한, 표 4 내지 표 12에서 임의의 열 그룹에 대응되는 수열들에 대해 모두 동일하게 Q_{ldpc} 의 배수를 더한 결과 또한 부호의 그래프 상의 사이클 특성이나 차수 분포 등의 대수적 특성이 바뀌지 않기 때문에, 표 4 내지 표 12에 나타난 수열들에 대해 모두 동일하게 Q_{ldpc} 의 배수를 더한 결과도 한 가지 일 예가 될 수 있다. 여기에서 주의해야 할 점은 주어진 수열에 Q_{ldpc} 배수만큼 더했을 경우 그 값이 $(N_{ldpc}-K_{ldpc})$ 이상의 값이 나올 경우에는 그 값에 $(N_{ldpc}-K_{ldpc})$ 에 대한 모듈로(modulo) 연산을 적용한 값으로 바꾸어 적용해야 한다는 것이다.

[0091]

한편, 표 4 내지 표 12와 같이 정보어 부분 행렬(210)의 i 번째 열 그룹의 0 번째 열에서 1이 존재하는 행의 위치가 정의되면, 이를 Q_{ldpc} 만큼 시클릭 쉬프트하여 각 열 그룹의 다른 열에서 1이 존재하는 행의 위치가 정의될 수 있다.

[0092]

예를 들어, 표 4의 경우, 정보어 부분 행렬(210)의 0 번째 열 그룹의 0 번째 열의 경우, 245 번째 행, 449 번째 행, 491 번째 행,...에 1이 존재한다.

- [0093] 이 경우, $Q_{ldpc}=(N_{ldpc}-K_{ldpc})/M=(16200-5400)/360=30$ 이므로, 0 번째 열 그룹의 1 번째 열에서 1이 위치한 행의 인덱스는 $275(=245+30)$, $479(=449+30)$, $521(=491+30), \dots$ 이고, 0 번째 열 그룹의 2 번째 열에서 1이 위치한 행의 인덱스는 $305(=275+30)$, $509(=479+30)$, $551(=521+30), \dots$ 이 될 수 있다.
- [0094] 이와 같은 방식에 의해, 각 열 그룹의 모든 행에서 1이 위치한 행의 인덱스가 정의될 수 있다.
- [0095] 한편, 도 2와 같은 패리티 검사 행렬(200)에서 패리티 부분 행렬(220)은 다음과 같이 정의될 수 있다.
- [0096] 패리티 부분 행렬(220)은 $N_{ldpc}-K_{ldpc}$ 개의 열(즉, K_{ldpc} 번째 열부터 $N_{ldpc}-1$ 번째 열)을 포함하는 부분 행렬로, 이중 대각(dual diagonal 또는 staircase) 구조를 갖는다. 따라서, 패리티 부분 행렬(220)에 포함되는 열 중에서 마지막 열(즉, $N_{ldpc}-1$ 번째 열)을 제외한 나머지 열들의 차수는 모두 2이며, 마지막 열의 차수는 1이 된다.
- [0097] 결국, 패리티 검사 행렬(200)에서 정보어 부분 행렬(210)은 표 4 내지 표 12에 의해 정의되며 패리티 부분 행렬(220)은 이중 대각 구조를 가질 수 있다.
- [0098] 한편, 도 2에 도시된 패리티 검사 행렬(200)의 열과 행을 하기의 수학식 4 및 수학식 5에 기초하여 퍼뮤테이션(permutation)하면, 도 2에 도시된 패리티 검사 행렬(200)은 도 3에 도시된 패리티 검사 행렬(300)의 형태로 나타내어질 수 있다.

수학식 4

$$Q_{ldpc} \cdot i + j \Rightarrow M \cdot j + i \quad (0 \leq i < M, 0 \leq j < Q_{ldpc})$$

수학식 5

$$K_{ldpc} + Q_{ldpc} \cdot k + l \Rightarrow K_{ldpc} + M \cdot l + k \quad (0 \leq k < M, 0 \leq l < Q_{ldpc})$$

- [0101] 수학식 4 및 수학식 5에 기초하여 퍼뮤테이션을 수행하는 방법은 다음과 같다. 여기에서, 로우 퍼뮤테이션과 컬럼 퍼뮤테이션은 동일한 원리가 적용된다는 점에서, 이하에서는 로우 퍼뮤테이션을 일 예로 설명하도록 한다.
- [0102] 로우 퍼뮤테이션의 경우, X 번째 행에 대해 $X=Q_{ldpc} \times i + j$ 를 만족하는 i, j를 산출하고, 산출된 i, j를 $M \times j + i$ 에 대입하여 X 번째 행이 퍼뮤테이션되는 행을 산출하게 된다. 예를 들어, 7 번째 행의 경우, $7=2 \times i + j$ 를 만족하는 i, j는 각각 3, 1이 되므로, 7 번째 행은 $10 \times 1 + 3 = 13$ 번째 행으로 퍼뮤테이션된다.
- [0103] 이와 같은 방식으로 로우 퍼뮤테이션 및 컬럼 퍼뮤테이션을 수행하면, 도 2의 패리티 검사 행렬은 도 3과 같이 나타낼 수 있게 된다.
- [0104] 도 3을 참조하면, 패리티 검사 행렬(300)은 패리티 검사 행렬(300)을 다수의 부분 블록(partial block)들로 분할하고, 부분 블록들 각각에 $M \times M$ 사이즈의 준 순환(quasi-cyclic) 행렬을 대응시키는 형태를 갖는다.
- [0105] 이에 따라, 도 3과 같은 구조를 갖는 패리티 검사 행렬(300)은 $M \times M$ 사이즈의 행렬 단위로 구성된다. 즉, 패리티 검사 행렬(300)은 다수의 부분 블록에 $M \times M$ 사이즈를 갖는 부분 행렬이 나열되어 구성된다.
- [0106] 이와 같이, 패리티 검사 행렬(300)은 $M \times M$ 사이즈의 준 순환 행렬 단위로 구성되므로, M 개의 열들을 열 블록(column-block), M 개의 행들을 행 블록(row-block)이라 명명할 수 있다. 이에 따라, 본 발명에서 사용하는 도 3과 같은 구조를 갖는 패리티 검사 행렬(300)은 $N_{qc_column}=N_{ldpc}/M$ 개의 열 블록과 $N_{qc_row}=N_{parity}/M$ 개의 행 블록으로 구성되는 것으로 볼 수 있다.
- [0107] 이하에서는, $M \times M$ 사이즈를 갖는 부분 행렬에 대하여 설명하도록 한다.
- [0108] 첫째, 0 번째 행 블록의 $(N_{qc_column}-1)$ 번째 열 블록 A(330)은 하기의 수학식 6의 형태를 갖는다.

수학식 6

$$A = \begin{bmatrix} 0 & 0 & \dots & 0 & 0 \\ 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & 0 \end{bmatrix}$$

[0109]

[0110] 이와 같이, A(330)은 $M \times M$ 행렬로, 0 번째 행과 $(M-1)$ 번째 열의 값들은 모두 '0'이고, $0 \leq i \leq (M-2)$ 에 대하여 i 번째 열의 $(i+1)$ 번째 행은 '1'이며 그 외의 모든 값들은 '0'이다.

[0111] 둘째, 패리티 부분 행렬(320)에서 $0 \leq i \leq (N_{ldpc} - K_{ldpc})/M - 1$ 에 대하여 $(K_{ldpc}/M + i)$ 번째 열 블록의 i 번째 행 블록은 단위 행렬 $I_{M \times M}(340)$ 로 구성된다. 또한, $0 \leq i \leq (N_{ldpc} - K_{ldpc})/M - 2$ 에 대하여 $(K_{ldpc}/M + i)$ 번째 열 블록의 $(i+1)$ 번째 행 블록은 단위 행렬 $I_{M \times M}(340)$ 로 구성된다.

[0112] 셋째, 정보어 부분 행렬(310)을 구성하는 블록(350)은 순환 행렬 P 가 시클릭 쉬프트된 형태인 $P^{a_{ij}}$ 또는, 순환 행렬 P 가 시클릭 쉬프트된 행렬 $P^{a_{ij}}$ 이 합해진 형태(또는, 중첩된 형태)가 될 수 있다.

[0113] 일 예로, 순환 행렬 P 의 위첨자 a_{ij} 가 1일 때(즉, P^1), 블록 $P(350)$ 의 형태는 하기의 수학식 7과 같이 나타낼 수 있다.

수학식 7

$$P = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \\ 1 & 0 & 0 & \dots & 0 \end{bmatrix}$$

[0114]

[0115] 순환 행렬 P 는 $M \times M$ 사이즈를 갖는 정사각 행렬로서, 순환 행렬 P 는 M 개의 행들 각각의 무게가 1이고, M 개의 열들 각각의 무게 역시 1인 행렬을 나타낸다. 그리고, 순환 행렬 P 는 위첨자 a_{ij} 가 0일 때 즉, P^0 는 단위 행렬 $I_{M \times M}$ 를 나타낸다. 그리고, 표기상의 편의를 위하여, 위첨자 a_{ij} 가 ∞ 일 때 즉, P^∞ 는 영(zero) 행렬을 정의한다.

[0116] 한편, 도 3에서 패리티 검사 행렬(300)의 i 번째 행 블록과 j 번째 열 블록이 교차하는 지점에 존재하는 부분 행렬은 $P^{a_{ij}}$ 가 될 수 있다. 따라서, i 와 j 는 정보어 부분에 해당하는 부분 블록들의 행 블록과 열 블록의 개수를 나타낸다. 따라서, 패리티 검사 행렬(300)은 전체 열의 개수가 $N_{ldpc} = M \times N_{qc_column}$ 이고, 전체 행의 개수가

$N_{\text{parity}}=M \times N_{\text{qc_row}}$ 가 된다. 즉, 패리티 검사 행렬(300)은 $N_{\text{qc_column}}$ 개의 "열 블록"과 $N_{\text{qc_row}}$ 개의 "행 블록"으로 구성된다.

[0117] 이하에서는 도 2와 같은 패리티 검사 행렬(200)에 기초하여 LDPC 부호화를 수행하는 방법에 대해 설명하도록 한다. 한편, 설명의 편의를 위해 패리티 검사 행렬(200)이 표 4와 같이 정의되는 경우를 일 예로 LDPC 부호화 과정을 개략적으로 설명하도록 한다.

[0118] 먼저, 길이가 K_{ldpc} 인 정보어 비트들을 $[i_0, i_1, i_2, \dots, i_{K_{\text{ldpc}}-1}]$ 라 하고, 길이가 $N_{\text{ldpc}}-K_{\text{ldpc}}$ 인 패리티 비트들을 $[p_0, p_1, p_2, \dots, p_{N_{\text{ldpc}}-K_{\text{ldpc}}-1}]$ 라 할 때, 하기와 같은 과정에 의해 LDPC 부호화가 수행될 수 있다.

[0119] 단계 1) 패리티 비트들을 '0'으로 초기화한다. 즉, $p_0=p_1=p_2=\dots=p_{N_{\text{ldpc}}-K_{\text{ldpc}}-1}=0$

[0120] 단계 2) 표 4의 첫 번째 행(즉, $i=0$ 인 행)에서 정의되는 패리티 비트의 어드레스를 패리티 비트의 인덱스로 갖는 패리티 비트에 0 번째 정보어 비트 i_0 를 누적(accumulate)한다. 이는 아래의 수학적 식 8과 같이 표현될 수 있다.

수학적 식 8

$$\begin{array}{ll} P_{245} = P_{245} \oplus i_0 & P_{6570} = P_{6570} \oplus i_0 \\ P_{449} = P_{449} \oplus i_0 & P_{7492} = P_{7492} \oplus i_0 \\ P_{491} = P_{491} \oplus i_0 & P_{7768} = P_{7768} \oplus i_0 \\ P_{980} = P_{980} \oplus i_0 & P_{7837} = P_{7837} \oplus i_0 \\ P_{1064} = P_{1064} \oplus i_0 & P_{7984} = P_{7984} \oplus i_0 \\ P_{1194} = P_{1194} \oplus i_0 & P_{8306} = P_{8306} \oplus i_0 \\ P_{1277} = P_{1277} \oplus i_0 & P_{8483} = P_{8483} \oplus i_0 \\ P_{1671} = P_{1671} \oplus i_0 & P_{8685} = P_{8685} \oplus i_0 \\ P_{2026} = P_{2026} \oplus i_0 & P_{9357} = P_{9357} \oplus i_0 \\ P_{3186} = P_{3186} \oplus i_0 & P_{9642} = P_{9642} \oplus i_0 \\ P_{4399} = P_{4399} \oplus i_0 & P_{10045} = P_{10045} \oplus i_0 \\ P_{4900} = P_{4900} \oplus i_0 & P_{10179} = P_{10179} \oplus i_0 \\ P_{5283} = P_{5283} \oplus i_0 & P_{10261} = P_{10261} \oplus i_0 \\ P_{5413} = P_{5413} \oplus i_0 & P_{10338} = P_{10338} \oplus i_0 \\ P_{5558} = P_{5558} \oplus i_0 & P_{10412} = P_{10412} \oplus i_0 \end{array}$$

[0121]

[0122] 여기에서, i_0 는 0 번째 정보어 비트, p_i 는 i 번째 패리티 비트, $\oplus$ 는 바이너리 연산을 의미한다. 바이너리 연산에 의하면, $1 \oplus 1$ 은 0, $1 \oplus 0$ 은 1, $0 \oplus 1$ 은 1, $0 \oplus 0$ 은 0이다.

[0123] 단계 3) 나머지 359 개의 정보어 비트들 $i_m(m=1,2,\dots,359)$ 을 패리티 비트에 누적한다. 여기에서, 나머지 정보어 비트들은 i_0 와 동일한 열 그룹에 속하는 정보어 비트들일 수 있다. 이때, 패리티 비트의 어드레스는 하기의 수학적 식 9에 기초하여 결정될 수 있다.

수학식 9

$$(x + (m \bmod 360) \times Q_{ldpc}) \bmod (N_{ldpc} - K_{ldpc})$$

[0124]

[0125]

여기에서, x 는 정보어 비트 i_0 에 대응되는 패리티 비트 누적기(parity bit accumulator)의 어드레스이고, Q_{ldpc} 는 정보어에 대응되는 부분 행렬에서 각 열이 시클릭 쉬프트되는 크기로, 표 4의 경우 30이 될 수 있다. 그리고, $m=1, 2, \dots, 359$ 이기 때문에 수학식 9의 $(m \bmod 360)$ 은 m 으로 볼 수 있다.

[0126]

결국, 수학식 9에 기초하여 산출된 패리티 비트의 어드레스를 인덱스로 하는 패리티 비트 각각에 정보어 비트들 $i_m(m=1, 2, \dots, 359)$ 각각을 누적하며, 일 예로, 정보어 비트 i_1 에 대해 하기의 수학식 10과 같은 연산이 수행될 수 있다.

수학식 10

$$\begin{aligned} P_{275} &= P_{275} \oplus i_1 & P_{6600} &= P_{6600} \oplus i_1 \\ P_{479} &= P_{479} \oplus i_1 & P_{7522} &= P_{7522} \oplus i_1 \\ P_{521} &= P_{521} \oplus i_1 & P_{7798} &= P_{7798} \oplus i_1 \\ P_{1010} &= P_{1010} \oplus i_1 & P_{7867} &= P_{7867} \oplus i_1 \\ P_{1094} &= P_{1094} \oplus i_1 & P_{8014} &= P_{8014} \oplus i_1 \\ P_{1224} &= P_{1224} \oplus i_1 & P_{8336} &= P_{8336} \oplus i_1 \\ P_{1307} &= P_{1307} \oplus i_1 & P_{8513} &= P_{8513} \oplus i_1 \\ P_{1701} &= P_{1701} \oplus i_1 & P_{8715} &= P_{8715} \oplus i_1 \\ P_{2056} &= P_{2056} \oplus i_1 & P_{9387} &= P_{9387} \oplus i_1 \\ P_{3216} &= P_{3216} \oplus i_1 & P_{9672} &= P_{9672} \oplus i_1 \\ P_{4429} &= P_{4429} \oplus i_1 & P_{10075} &= P_{10072} \oplus i_1 \\ P_{4930} &= P_{4930} \oplus i_1 & P_{10209} &= P_{10209} \oplus i_1 \\ P_{5313} &= P_{5313} \oplus i_1 & P_{10291} &= P_{10291} \oplus i_1 \\ P_{5443} &= P_{5443} \oplus i_1 & P_{10368} &= P_{10368} \oplus i_1 \\ P_{5588} &= P_{5588} \oplus i_1 & P_{10442} &= P_{10442} \oplus i_1 \end{aligned}$$

[0127]

[0128]

여기에서, i_1 는 1 번째 정보어 비트, p_i 는 i 번째 패리티 비트, $\oplus$ 는 바이너리 연산을 의미한다. 바이너리 연산에 의하면, $1 \oplus 1$ 은 0, $1 \oplus 0$ 은 1, $0 \oplus 1$ 은 1, $0 \oplus 0$ 은 0이다.

[0129]

단계 4) 표 4의 두 번째 행(즉, $i=1$ 인 행)에서 정의되는 패리티 비트의 어드레스를 패리티 비트의 인덱스로 갖는 패리티 비트에 360 번째 정보어 비트 i_{360} 를 누적한다.

[0130]

단계 5) 정보어 비트 i_{360} 과 동일한 그룹에 속하는 나머지 359 개의 정보어 비트들을 패리티 비트에 누적한다. 이때, 패리티 비트의 어드레스는 수학식 9에 기초하여 결정될 수 있다. 다만, 이 경우, x 는 정보어 비트 i_{360} 에 대응되는 패리티 비트 누적기의 어드레스가 된다.

[0131]

단계 6) 상술한 단계 4 및 단계 5와 같은 과정을 표 4의 모든 열 그룹에 대해 반복한다.

[0132]

단계 7) 결국, 하기와 같은 수학식 11에 기초하여 패리티 비트 p_i 를 산출하게 된다. 이때, i 는 1로 초기화된다.

수학식 11

$$p_i = p_i \oplus p_{i-1} \quad i = 1, 2, \dots, N_{ldpc} - K_{ldpc} - 1$$

수학식 11에서, p_i 는 i 번째 패리티 비트, N_{ldpc} 는 LDPC 부호어의 길이, K_{ldpc} 는 LDPC 부호어 중 정보어의 길이, $\oplus$ 는 바이너리 연산을 의미한다.

결국, 부호화부(110)는 상술한 방식에 따라 패리티 비트들을 산출할 수 있게 된다.

다른 예로, 본 발명의 일 실시 예에 따른 패리티 검사 행렬은 도 4와 같은 구조를 가질 수 있다.

도 4를 참조하면, 패리티 검사 행렬(400)은 5 개의 행렬(A, B, C, Z, D)로 구성될 수 있으며, 이하에서는 패리티 검사 행렬(400)의 구조에 대해 설명하기 위해 각 행렬의 구조에 대해 설명하도록 한다.

먼저, 도 4와 같은 패리티 검사 행렬(400)과 관련된 파라미터 값들인 M_1 , M_2 , Q_1 , Q_2 는 LDPC 부호어의 길이 및 부호율에 따라 하기의 표 13과 같이 정의될 수 있다.

표 13

Rate	Length	Sizes			
		M_1	M_2	Q_1	Q_2
1/15	16200	2520	12600	7	35
	64800	1080	59400	3	165
2/15	16200	3240	10800	9	30
	64800	1800	54360	5	151
3/15	16200	1080	11880	3	33
	64800	1800	50040	5	139
4/15	16200	1080	10800	3	30
	64800	1800	45720	5	127
5/15	16200	720	10080	2	28
	64800	1440	41760	4	116
6/15	16200	1080	8640	3	24
	64800	1080	37800	3	105

한편, 행렬 A는 K 개의 열과 g 개의 행으로 구성되며, 행렬 C는 K+g 개의 열과 N-K-g 개의 행으로 구성된다. 여기에서, K는 정보어 비트들의 길이이고, N은 LDPC 부호어의 길이이다.

그리고, 행렬 A와 행렬 C에서 i 번째 열 그룹의 0 번째 열에서 1이 위치한 행의 인덱스들은 LDPC 부호어의 길이 및 부호율에 따라 하기의 표 14에 기초하여 정의될 수 있다. 이 경우에도, 행렬 A와 행렬 C 각각에서 열의 패턴이 반복되는 간격 즉, 동일한 그룹에 속하는 열의 개수는 360이 될 수 있다.

일 예로, LDPC 부호어의 길이 N이 16200, 부호율이 5/15인 경우, 행렬 A와 행렬 C의 i 번째 열 그룹의 0 번째 열에서 1이 위치한 행의 인덱스들은 하기의 표 14와 같다.

표 14

i	i번째 열 그룹의 0번째 열에서 1이 위치한 행의 인덱스
0	69 244 706 5145 5994 6066 6763 6815 8509
1	257 541 618 3933 6188 7048 7484 8424 9104
2	69 500 536 1494 1669 7075 7553 8202 10305
3	11 189 340 2103 3199 6775 7471 7918 10530
4	333 400 434 1806 3264 5693 8534 9274 10344
5	111 129 260 3562 3676 3680 3809 5169 7308 8280
6	100 303 342 3133 3952 4226 4713 5053 5717 9931
7	83 87 374 828 2460 4943 6311 8657 9272 9571
8	114 166 325 2680 4698 7703 7886 8791 9978 10684
9	281 542 549 1671 3178 3955 7153 7432 9052 10219
10	202 271 608 3860 4173 4203 5169 6871 8113 9757
11	16 359 419 3333 4198 4737 6170 7987 9573 10095
12	235 244 584 4640 5007 5563 6029 6816 7678 9968
13	123 449 646 2460 3845 4161 6610 7245 7686 8651
14	136 231 468 835 2622 3292 5158 5294 5584 9926
15	3085 4683 8191 9027 9922 9928 10550
16	2462 3185 3976 4091 8089 8772 9342

[0143]

[0144]

한편, 상술한 예에서는 LDPC 부호어의 길이가 16200이고 부호율이 5/15인 경우만을 설명하였으나 이는 일 예에 불과하고, LDPC 부호어의 길이가 64800인 경우이거나 다른 부호율을 갖는 경우에도 행렬 A와 행렬 C의 i 번째 열 그룹의 0 번째 열에서 1이 위치한 행의 인덱스들은 다양하게 정의될 수 있다.

[0145]

이하에서는, 표 14를 일 예로, 행렬 A와 행렬 C에서 1이 존재하는 행의 위치에 대해 구체적으로 설명하도록 한다.

[0146]

표 14에서 LDPC 부호어의 길이 N이 16200이고 부호율이 5/15이므로, 표 12를 참조할 때, 표 14에 의해 정의되는 패리티 검사 행렬(400)에서 $M_1=720$, $M_2=10080$, $Q_1=2$, $Q_2=28$ 이 될 수 있다.

[0147]

여기에서, Q_1 은 행렬 A에서 동일한 열 그룹 내에 속한 열들이 시클릭 쉬프트되는 크기이고, Q_2 는 행렬 C에서 동일한 열 그룹 내에 속한 열들이 시클릭 쉬프트되는 크기이다.

[0148]

그리고, $Q_1=M_1/L$, $Q_2=M_2/L$, $M_1=g$, $M_2=N-K-g$ 이고, L은 행렬 A, C 각각에서 열의 패턴이 반복되는 간격으로 일 예로, 360이 될 수 있다.

[0149]

한편, 행렬 A, C 각각에서 1이 위치하는 행의 인덱스는 M_1 값에 기초하여 결정될 수 있다.

[0150]

예를 들어, 표 14의 경우 $M_1=720$ 이라는 점에서, 행렬 A에서 i 번째 열 그룹의 0 번째 열에서 1이 존재하는 행의 위치는 표 14의 인덱스 값들 중에서 720 보다 작은 값들에 기초하여 결정될 수 있으며, 행렬 C에서 i 번째 열 그룹의 0 번째 열에서 1이 존재하는 행의 위치는 표 14의 인덱스 값들 중에서 720 이상인 값들에 기초하여 결정될 수 있다.

[0151]

구체적으로, 표 14에서 0 번째 열 그룹에 대응되는 수열은 "69, 244, 706, 5145, 5994, 6066, 6763, 6815, 8509"이다. 따라서, 행렬 A의 0 번째 열 그룹의 0 번째 열의 경우, 69 번째 행, 244 번째 행, 706 번째 행에 각각 1이 위치할 수 있고, 행렬 C의 0 번째 열 그룹의 0 번째 열의 경우 5145 번째 행, 5994 번째 행, 6066 번째 행, 6763 번째 행, 6815 번째 행, 8509 번째 행에 각각 1이 위치할 수 있다.

[0152]

한편, 행렬 A의 경우 각 열 그룹의 0 번째 열에서 1의 위치가 정의되면 이를 Q_1 만큼 시클릭 쉬프트하여 각 열 그룹의 다른 열에서 1이 존재하는 행의 위치가 정의될 수 있고, 행렬 C의 경우 각 열 그룹의 0 번째 열에서 1의 위치가 정의되면 이를 Q_2 만큼 시클릭 쉬프트하여 각 열 그룹의 다른 열에서 1이 존재하는 행의 위치가 정의될 수 있다.

[0153]

상술한 예에서, 행렬 A의 0 번째 열 그룹의 0 번째 열의 경우, 69 번째 행, 244 번째 행, 706 번째 행에 1이 존재한다. 이 경우, $Q_1=2$ 이므로, 0 번째 열 그룹의 1 번째 열에서 1이 위치한 행의 인덱스는 $71(=69+2)$, $246(=244+2)$, $708(=706+2)$ 이고, 0 번째 열 그룹의 2 번째 열에서 1이 위치한 행의 인덱스는 $73(=71+2)$, $248(=246+2)$, $710(=708+2)$ 이 될 수 있다.

[0154]

한편, 행렬 C의 0 번째 열 그룹의 0 번째 열의 경우, 5145 번째 행, 5994 번째 행, 6066 번째 행, 6763 번째 행, 6815 번째 행, 8509 번째 행에 1이 존재한다. 이 경우, $Q_2=28$ 이므로, 0 번째 열 그룹의 1 번째 열에서 1이 위치한 행의 인덱스는 $5173(=5145+28)$, $6022(=5994+28)$, $6094(=6066+28)$, $6791(=6763+28)$, $6843(=6815+28)$,

8537(=8509+28)이고, 0 번째 열 그룹의 2 번째 열에서 1이 위치한 행의 인덱스는 5201(=5173+28), 6050(=6022+28), 6122(=6094+28), 6819(=6791+28), 6871(=6843+28), 8565(=8537+28)가 될 수 있다.

[0155] 이와 같은 방식에 따라 행렬 A 및 행렬 C의 모든 열 그룹에서 1이 존재하는 행의 위치가 정의될 수 있다.

[0156] 한편, 행렬 B는 이중 대각 구조를 가지며, 행렬 D는 대각 구조(즉, 행렬 D는 항등 행렬(identity matrix)이 된다)를 가지며, 행렬 Z는 영(zero) 행렬이 될 수 있다.

[0157] 결국, 상술한 바와 같은 구조를 갖는 행렬 A, B, C, D, Z에 의해 도 4와 같은 패리티 검사 행렬(400)의 구조가 정의될 수 있게 된다.

[0158] 이하에서는 도 4와 같은 패리티 검사 행렬(400)에 기초하여 LDPC 부호화를 수행하는 방법에 대해 설명하도록 한다. 한편, 설명의 편의를 위해 패리티 검사 행렬(400)이 표 14와 같이 정의되는 경우를 일 예로 LDPC 부호화 과정을 개략적으로 설명하도록 한다.

[0159] 예를 들어, 정보어 블록 $S=(s_0, s_1, \dots, s_{K-1})$ 을 LDPC 부호화하는 경우, 패리티 비트

$P=(p_0, p_1, \dots, P_{M_1+M_2-1})$ 를 포함하는 LDPC 부호어 $\Lambda=(\lambda_0, \lambda_1, \dots, \lambda_{N-1})=(s_0, s_1, \dots, s_{K-1},$

$p_0, p_1, \dots, P_{M_1+M_2-1})$ 가 생성될 수 있다.

[0160] 여기에서, M_1 및 M_2 각각은 이중 대각 구조를 갖는 행렬 B 및 대각 구조를 갖는 행렬 C 각각의 사이즈를 나타내며, $M_1=g$, $M_2=N-K-g$ 가 될 수 있다.

[0161] 한편, 패리티 비트를 산출하는 과정은 다음과 같이 나타낼 수 있다. 이하에서도 설명의 편의를 위해, 패리티 검사 행렬(400)이 표 14와 같이 정의되는 경우를 일 예로 설명하도록 한다.

[0162] 단계 1) $\lambda_i=s_i$ ($i=0, 1, \dots, K-1$), $p_j=0$ ($j=0, 1, \dots, M_1+M_2-1$)로 초기화한다.

[0163] 단계 2) 표 14의 첫 번째 행(즉, $i=0$ 인 행)에서 정의되는 패리티 비트의 어드레스에 0 번째 정보어 비트 λ_0 을 누적한다. 이는 아래의 수학적 식 12와 같이 표현될 수 있다.

수학적 식 12

$$\begin{aligned} P_{69} &= P_{69} \oplus \lambda_0 & P_{6066} &= P_{6066} \oplus \lambda_0 \\ P_{244} &= P_{244} \oplus \lambda_0 & P_{6763} &= P_{6763} \oplus \lambda_0 \\ P_{706} &= P_{706} \oplus \lambda_0 & P_{6815} &= P_{6815} \oplus \lambda_0 \\ P_{5145} &= P_{5145} \oplus \lambda_0 & P_{8509} &= P_{8509} \oplus \lambda_0 \\ P_{5994} &= P_{5994} \oplus \lambda_0 \end{aligned}$$

[0164]

[0165] 단계 3) 다음 $L-1$ 개의 정보어 비트 $\lambda_m(m=1, 2, \dots, L-1)$ 에 대해, λ_m 를 하기와 같은 수학적 식 13에 기초하여 산출되는 패리티 비트 어드레스에 누적한다.

수학적 식 13

$$\begin{aligned} &(\chi + m \times Q_1) \bmod M_1 \text{ (if } \chi < M_1) \\ &M_1 + \{(\chi - M_1 + m \times Q_2) \bmod M_2\} \text{ (if } \chi \geq M_1) \end{aligned}$$

[0166]

[0167] 여기에서, x 는 0 번째 정보어 비트 λ_0 에 대응되는 패리티 비트 누적기의 어드레스이다.

[0168] 그리고, $Q_1=M_1/L$, $Q_2=M_2/L$ 이다. 또한, 표 14에서는 LDPC 부호어의 길이 N 이 16200이고, 부호율이 5/15이므로, 표 13을 참조하면 $M_1=720$, $M_2=10080$, $Q_1=2$, $Q_2=28$, $L=360$ 이 될 수 있다.

[0169] 이에 따라, 1 번째 정보어 비트 λ_1 에 대해 하기의 수학식 14와 같은 연산이 수행될 수 있다.

수학식 14

$$\begin{aligned} P_{71} &= P_{71} \oplus \lambda_1 & P_{6094} &= P_{6094} \oplus \lambda_1 \\ P_{246} &= P_{246} \oplus \lambda_1 & P_{6791} &= P_{6791} \oplus \lambda_1 \\ P_{708} &= P_{708} \oplus \lambda_1 & P_{6843} &= P_{6843} \oplus \lambda_1 \\ P_{5173} &= P_{5173} \oplus \lambda_1 & P_{8537} &= P_{8537} \oplus \lambda_1 \\ P_{6022} &= P_{6022} \oplus \lambda_1 \end{aligned}$$

[0170]

[0171] 단계 4) L 번째 정보어 비트 λ_L 에 대해 표 14의 두 번째 행(즉, $i=1$ 인 행)과 같은 패리티 비트의 어드레스가 주어진다. 이 점에서, 상술한 방식과 유사하게, 이후의 $L-1$ 개 정보어 비트 λ_m ($m=L+1, L+2, \dots, 2L-1$)에 대한 패리티 비트의 어드레스를 수학식 13에 기초하여 산출한다. 이 경우, x 는 정보어 비트 λ_L 에 대응되는 패리티 비트 누적기의 어드레스로, 표 14의 두 번째 행에 기초하여 얻어질 수 있다.

[0172] 단계 5) 각 그룹의 L 개의 새로운 정보어 비트들에 대해, 표 14의 새로운 행들을 패리티 비트 누적기의 어드레스로 하여 상술한 과정을 반복한다.

[0173] 단계 6) 부호어 비트 λ_0 부터 λ_{K-1} 까지 상술한 과정이 반복된 이후, $i=1$ 부터 순차적으로 하기의 수학식 15에 대한 값을 산출한다.

수학식 15

$$P_i = P_i \oplus P_{i-1} \quad (i=1, 2, \dots, M_1-1)$$

[0174]

[0175] 단계 7) 이중 대각 구조를 갖는 행렬 B 에 대응되는 패리티 비트 λ_K 부터 λ_{K+M_1-1} 까지를 하기의 수학식 16에 기초하여 산출한다.

수학식 16

$$\lambda_{K+L \times t+s} = p_{Q_1 \times s+t} \quad (0 \leq s < L, 0 \leq t < Q_1)$$

[0176]

[0177] 단계 8) 각 그룹의 L 개의 새로운 부호어 비트 λ_K 부터 λ_{K+M_1-1} 까지에 대한 패리티 비트 누적기의 어드레스는 표 14 및 수학식 13에 기초하여 산출한다.

[0178] 단계 9) 부호어 비트 λ_K 부터 λ_{K+M_1-1} 까지 적용된 이후, 대각 구조를 갖는 행렬 C에 대응되는 패리티 비트 λ_{K+M_1} 부터 $\lambda_{K+M_1+M_2-1}$ 까지를 하기의 수학식 17에 기초하여 산출한다.

수학식 17

$$\lambda_{K+M_1+L \times t+s} = p_{M_1+Q_2 \times s+t} \quad (0 \leq s < L, 0 \leq t < Q_2)$$

[0179]

[0180] 결국, 이와 같은 방식에 따라 패리티 비트들을 산출할 수 있게 된다.

[0181] 도 1로 돌아가서, 부호화부(110)는 3/15, 4/15, 5/15, 6/15, 7/15, 8/15, 9/15, 10/15, 11/15, 12/15, 13/15 등과 같은 다양한 부호율을 이용하여 LDPC 부호화를 수행할 수 있다. 그리고, 부호화부(110)는 정보어 비트들의 길이 및 부호율에 기초하여 16200, 64800 등과 같은 다양한 길이를 갖는 LDPC 부호어를 생성할 수 있다.

[0182] 이 경우, 부호화부(110)는 패리티 검사 행렬을 이용하여 LDPC 부호화를 수행할 수 있으며, 패리티 검사 행렬의 구체적인 구조에 대해서는 도 2 내지 도 4와 함께 상술한 바 있다.

[0183] 또한, 부호화부(110)는 LDPC 부호화뿐만 아니라, BCH(Bose, Chaudhuri, Hocquenghem) 부호화를 수행할 수도 있다. 이를 위해, 부호화부(110)는 BCH 부호화를 수행하는 BCH 인코더(미도시)를 더 포함할 수 있다.

[0184] 이 경우, 부호화부(110)는 BCH 부호화 및 LDPC 부호화 순으로 부호화를 수행할 수 있다. 구체적으로, 부호화부(110)는 입력되는 비트들에 BCH 부호화를 수행하여 BCH 패리티 비트를 부가하고, BCH 패리티 비트가 부가된 비트들을 정보어 비트들로 LDPC 부호화를 수행하여, LDPC 부호어를 생성할 수도 있다.

[0185] 인터리버(120)는 LDPC 부호어를 인터리빙한다. 즉, 인터리버(120)는 LDPC 부호어를 부호화부(110)로부터 전달받아 다양한 인터리빙 룰에 기초하여 LDPC 부호어를 인터리빙할 수 있다.

[0186] 특히, 인터리버(120)는 LDPC 부호어를 구성하는 복수의 비트 그룹(또는, 복수의 그룹 또는 복수의 블록) 중 기 설정된 비트 그룹에 포함된 비트가 변조 심볼 내의 기설정된 비트에 맵핑되도록, LDPC 부호어를 인터리빙할 수 있다. 이에 따라, 변조부(130)는 LDPC 부호어를 구성하는 복수의 비트 그룹 중 기설정된 비트 그룹에 포함된 비트를 변조 심볼 내의 기설정된 비트에 맵핑할 수 있다.

[0187] 이를 위해, 도 5와 같이, 인터리버(120)는 패리티 인터리버(121), 그룹 인터리버(또는, 그룹-와이즈(group-wise) 인터리버, 122), 그룹 트위스트 인터리버(123) 및 블록 인터리버(124)를 포함할 수 있다.

[0188] 패리티 인터리버(121)는 LDPC 부호어를 구성하는 패리티 비트들을 인터리빙한다.

[0189] 구체적으로, 패리티 인터리버(121)는 도 2와 같은 구조를 갖는 패리티 검사 행렬(200)에 기초하여 LDPC 부호어가 생성된 경우, 하기의 수학식 18을 이용하여 LDPC 부호어 중에서 패리티 비트들만을 인터리빙할 수 있다.

수학식 18

$$u_i = c_i \text{ for } 0 \leq i < K_{ldpc}$$

$$u_{K_{ldpc}+M \cdot t+s} = c_{K_{ldpc}+Q_{ldpc} \cdot s+t} \text{ for } 0 \leq s < M, 0 \leq t < Q_{ldpc}$$

[0190]

[0191] 여기에서, M은 정보어 부분 행렬(210)에서 열의 패턴이 반복되는 간격 즉, 열 그룹에 포함된 열의 개수(일 예로, M=360)이고, Q_{ldpc} 는 정보어 부분 행렬(210)에서 각 열이 시클릭 쉬프트되는 크기이다. 즉, 패리티 인터리

버(121)는 LDPC 부호어 $c=(c_0, c_1, \dots, c_{N_{ldpc}-1})$ 에 대해 패리티 인터리빙을 수행하여

$U=(u_0, u_1, \dots, u_{N_{ldpc}-1})$ 를 출력할 수 있다.

- [0192] 이와 같은 방법으로 패리티 인터리빙된 LDPC 부호어는 일정한 수의 연속된 비트들이 유사한 복호화 특성(예를 들어, 사이클 분포, 열의 차수 등)을 갖도록 구성될 수 있다.
- [0193] 예를 들어, LDPC 부호어는 연속된 M 개의 비트 단위로 동일한 특성을 가질 수 있다. 여기에서, M은 정보어 부분 행렬(210)에서 열의 패턴이 반복되는 간격으로, 일 예로 360이 될 수 있다.
- [0194] 구체적으로, LDPC 부호어 비트들과 패리티 검사 행렬의 곱은 '0'이 되어야 한다. 이는 i가 0부터 $N_{ldpc}-1$ 까지의 i 번째 LDPC 부호어 비트 $c_i(i=0,1,\dots, N_{ldpc}-1)$ 와 i 번째 패리티 검사 행렬의 열의 곱들의 합이 '0' 벡터가 되어야 한다는 것을 의미한다. 따라서, i 번째 LDPC 부호어 비트는 패리티 검사 행렬의 i 번째 열에 대응되는 것으로 볼 수 있다.
- [0195] 한편, 도 2와 같은 패리티 검사 행렬(200)의 경우, 정보어 부분 행렬(210)은 각각 M 개의 열 씩 동일한 그룹에 속하며, 열 그룹 단위로 동일한 특성을 갖는다(가령, 동일한 열 그룹 내의 열들은 동일한 차수 분포와 동일한 사이클 특성을 갖는다).
- [0196] 이 경우, 정보어 비트들에서 연속된 M 개의 비트들은 정보어 부분 행렬(210)의 동일한 열 그룹에 대응되므로, 정보어 비트들은 동일한 부호어 특성을 갖는 연속된 M 개의 비트들로 구성될 수 있다. 한편, LDPC 부호어의 패리티 비트들이 패리티 인터리버(121)에 의해 인터리빙되면, LDPC 부호어의 패리티 비트들도 동일한 부호어 특성을 갖는 연속된 M 개의 비트들로 구성될 수 있다.
- [0197] 다만, 도 3과 같은 패리티 검사 행렬(300) 및 도 4와 같은 패리티 검사 행렬(400)에 기초하여 부호화된 LDPC 부호어에 대해서는 패리티 인터리빙이 수행되지 않을 수 있으며, 이 경우, 패리티 인터리버(121)는 생략 가능하다.
- [0198] 그룹 인터리버(122)는 패리티 인터리빙된 LDPC 부호어를 복수의 비트 그룹으로 구분하고, 복수의 비트 그룹의 순서를 비트 그룹 단위(bits group wise)로 재정렬할 수 있다. 즉, 그룹 인터리버(122)는 복수의 비트 그룹을 비트 그룹 단위로 인터리빙할 수 있다.
- [0199] 한편, 경우에 따라 패리티 인터리버(121)가 생략되는 경우, 그룹 인터리버(122)는 LDPC 부호어를 복수의 비트 그룹으로 구분하고, 복수의 비트 그룹의 순서를 비트 그룹 단위로 재정렬할 수 있다.
- [0200] 이를 위해, 그룹 인터리버(122)는 하기의 수학식 19 또는 수학식 20을 이용하여 패리티 인터리빙된 LDPC 부호어를 복수의 비트 그룹으로 구분한다.

수학식 19

$$X_j = \left\{ u_k \mid j = \left\lfloor \frac{k}{360} \right\rfloor, 0 \leq k < N_{ldpc} \right\} \text{ for } 0 \leq j < N_{group}$$

[0201]

수학식 20

$$X_j = \left\{ u_k \mid 360 \times j \leq k < 360 \times (j+1), 0 \leq k < N_{ldpc} \right\} \text{ for } 0 \leq j < N_{group}$$

[0202]

- [0203] 이들 수학식에서, N_{group} 은 비트 그룹의 전체 개수, X_j 는 j 번째 비트 그룹, u_k 는 그룹 인터리버(122)로 입력되는

k 번째 LDPC 부호어 비트를 나타낸다. 그리고, $\left\lfloor \frac{k}{360} \right\rfloor$ 는 $k/360$ 이하의 가장 큰 정수를 나타낸다.

- [0204] 한편, 이들 수학식에서 360은 정보어 부분 행렬에서 열의 패턴이 반복되는 간격인 M의 일 예를 나타낸다는 점에서, 이들 수학식에서 360은 M으로 변경 가능하다.
- [0205] 한편, 복수의 비트 그룹으로 구분된 LDPC 부호어는 도 6과 같이 나타낼 수 있다.
- [0206] 도 6을 참조하면, LDPC 부호어는 복수의 비트 그룹으로 구분되며, 각 비트 그룹은 연속된 M 개의 비트로 구성되는 것을 알 수 있다. 여기에서, M이 360인 경우, 복수의 비트 그룹 각각은 360 개의 비트로 구성될 수 있다. 이에 따라, 각 비트 그룹은 패리티 검사 행렬의 각 열 그룹에 대응되는 비트들로 구성될 수 있다.
- [0207] 구체적으로, LDPC 부호어는 M 개의 비트씩 구분되어지므로, K_{ldpc} 개의 정보어 비트들은 (K_{ldpc}/M) 개의 비트 그룹으로 구분되고 $N_{ldpc}-K_{ldpc}$ 개의 패리티 비트들은 $(N_{ldpc}-K_{ldpc})/M$ 개의 비트 그룹으로 구분된다. 이에 따라, LDPC 부호어는 총 (N_{ldpc}/M) 개의 비트 그룹으로 구분될 수 있다.
- [0208] 예를 들어, $M=360$ 이고 LDPC 부호어의 길이 N_{ldpc} 가 16200인 경우 LDPC 부호어를 구성하는 비트 그룹의 개수 N_{group} 은 $45(=16200/360)$ 가 될 수 있고, $M=360$ 이고 LDPC 부호어의 길이 N_{ldpc} 가 64800인 경우 LDPC 부호어를 구성하는 비트 그룹의 개수 N_{group} 은 $180(=64800/360)$ 가 될 수 있다.
- [0209] 이와 같이, 그룹 인터리버(122)가 LDPC 부호어를 연속된 M 개의 비트씩 동일한 비트 그룹으로 구분하는 것은, 상술한 바와 같이 LDPC 부호어가 연속된 M 개의 비트 단위로 동일한 부호어 특성을 갖기 때문이다. 이에 따라, LDPC 부호어를 연속된 M 개의 비트 단위로 구분하는 경우, 동일한 부호어 특성을 갖는 비트들이 동일한 비트 그룹에 포함될 수 있다.
- [0210] 한편, 상술한 예에서는 각 비트 그룹을 구성하는 비트의 개수가 M인 것으로 설명하였으나 이는 일 예에 불과하며, 각 비트 그룹을 구성하는 비트의 개수는 다양하게 변경 가능하다.
- [0211] 일 예로, 각 비트 그룹을 구성하는 비트의 개수는 M의 약수가 될 수 있다. 즉, 각 비트 그룹을 구성하는 비트의 개수는 패리티 검사 행렬의 정보어 부분 행렬의 열 그룹을 구성하는 열의 개수의 약수가 될 수 있다. 이 경우, 각 비트 그룹은 M의 약수 개의 비트로 구성될 수 있다. 예를 들어, 정보어 부분 행렬의 열 그룹을 구성하는 열의 개수가 360인 경우 즉, $M=360$ 인 경우, 그룹 인터리버(122)는 각 비트 그룹을 구성하는 비트의 개수가 360의 약수 중 어느 하나가 되도록, LDPC 부호어를 복수의 비트 그룹으로 구분할 수 있다.
- [0212] 다만, 이하에서는 설명의 편의를 위해, 비트 그룹을 구성하는 비트의 개수가 M인 경우에 대해서만 설명하도록 한다.
- [0213] 이후, 그룹 인터리버(122)는 LDPC 부호어를 비트 그룹 단위로 인터리빙한다. 구체적으로, 그룹 인터리버(122)는 LDPC 부호어를 복수의 비트 그룹으로 그룹핑하고, 복수의 비트 그룹을 비트 그룹 단위로 재정렬할 수 있다. 즉, 그룹 인터리버(122)는 LDPC 부호어를 구성하는 복수의 비트 그룹의 위치를 서로 변경하여 LDPC 부호어를 구성하는 복수의 비트 그룹의 순서를 그룹 단위로 재정렬할 수 있다.
- [0214] 여기에서, 그룹 인터리버(122)는 복수의 비트 그룹 중 동일한 변조 심볼에 맵핑되는 비트들을 포함하는 비트 그룹이 기설정된 간격만큼 이격 배치되도록 복수의 비트 그룹의 순서를 비트 그룹 단위로 재정렬할 수 있다.
- [0215] 이 경우, 그룹 인터리버(122)는 블록 인터리버(124)를 구성하는 행 및 열의 개수, LDPC 부호어를 구성하는 비트 그룹의 개수 및 각 비트 그룹에 포함된 비트 수 등을 고려하여 동일한 변조 심볼에 맵핑되는 비트들을 포함하는 비트 그룹들이 일정한 간격만큼 이격 배치되도록 복수의 비트 그룹의 순서를 비트 그룹 단위로 재정렬할 수 있다.
- [0216] 이를 위해, 그룹 인터리버(122)는 하기의 수학식 21을 이용하여 복수의 그룹의 순서를 비트 그룹 단위로 재정렬할 수 있다.

수학식 21

$$Y_j = X_{\pi(j)} \quad (0 \leq j < N_{group})$$

[0217]

- [0218] 여기에서, X_j 는 그룹 인터리빙 전의 j 번째 비트 그룹을 나타내고, Y_j 는 그룹 인터리빙 후의 j 번째 비트 그룹을 나타낸다. 그리고, $\pi(j)$ 는 인터리빙 순서를 나타내는 파라미터로, LDPC 부호어의 길이, 변조 방식 및 부호율 중 적어도 하나에 의해 결정될 수 있다.
- [0219] 따라서, $X_{\pi(j)}$ 는 그룹 인터리빙 전 $\pi(j)$ 번째 비트 그룹을 나타내며, 수학적 식 21은 인터리빙 전 $\pi(j)$ 번째 비트 그룹이 인터리빙 후 j 번째 비트 그룹으로 인터리빙되는 것을 의미하게 된다.
- [0220] 한편, 본 발명의 일 실시 예에 따른 $\pi(j)$ 의 구체적인 일 예는 하기의 표 15 내지 표 27과 같이 정의될 수 있다.
- [0221] 이 경우, $\pi(j)$ 은 LDPC 부호어의 길이 및 부호율에 따라 정의되며, 패리티 검사 행렬 또한, LDPC 부호어의 길이 및 부호율에 따라 정의된다. 따라서, LDPC 부호어의 길이 및 부호율에 따라 특정 패리티 검사 행렬에 기초하여 LDPC 부호화가 수행된 경우, 해당 LDPC 부호어의 길이 및 부호율을 만족하는 $\pi(j)$ 에 기초하여 LDPC 부호어가 비트 그룹 단위로 인터리빙될 수 있다.
- [0222] 예를 들어, 부호화부(110)가 16200의 길이를 갖는 LDPC 부호어를 생성하기 위해 5/15의 부호율로 LDPC 부호화를 수행한 경우, 그룹 인터리버(122)는 하기의 표 15 내지 표 27 중에서 LDPC 부호어의 길이가 16200이고 부호율이 5/15에서 정의되는 $\pi(j)$ 를 이용하여 인터리빙을 수행할 수 있다.
- [0223] 일 예로, LDPC 부호어의 길이가 16200이고 부호율이 5/15이고, 변조 방식이 256-QAM(quadrature amplitude modulation)인 경우, $\pi(j)$ 는 하기의 표 15와 같이 정의될 수 있다. 특히, 표 15의 경우, 표 14에 의해 정의되는 패리티 검사 행렬에 기초하여 LDPC 부호화를 수행한 경우에 적용될 수 있다.

표 15

	Order of bits group to be block interleaved $\pi(j)$ ($0 \leq j < 45$)																						
j-th block of Group-wise interleaver output	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	
$\pi(j)$ -th block of Group-wise interleaver input	4	23	3	6	18	5	0	2	7	26	21	27	39	42	38	31	1	34	20	37	40	24	43
	25	33	9	22	36	30	35	11	10	17	32	13	12	41	15	14	19	16	8	44	29	28	

[0224]

- [0225] 표 15의 경우, 수학적 식 21은 $Y_0=X_{\pi(0)}=X_4$, $Y_1=X_{\pi(1)}=X_{23}$, $Y_2=X_{\pi(2)}=X_3$, ..., $Y_{43}=X_{\pi(43)}=X_{29}$, $Y_{44}=X_{\pi(44)}=X_{28}$ 과 같이 나타낼 수 있다. 이에 따라, 그룹 인터리버(122)는 4 번째 비트 그룹을 0 번째로, 23 번째 비트 그룹을 1 번째로, 3 번째 비트 그룹을 2 번째로, ..., 29 번째 비트 그룹을 43 번째로, 28 번째 비트 그룹을 44 번째로 순서를 변경하여 복수의 비트 그룹의 순서를 비트 그룹 단위로 재정렬할 수 있다.
- [0226] 다른 예로, LDPC 부호어의 길이가 16200이고 부호율이 7/15이고, 변조 방식이 256-QAM인 경우, $\pi(j)$ 는 하기의 표 16과 같이 정의될 수 있다. 특히, 표 16의 경우, 표 5에 의해 정의되는 패리티 검사 행렬에 기초하여 LDPC 부호화를 수행한 경우에 적용될 수 있다.

표 16

	Order of bits group to be block interleaved $\pi(j)$ ($0 \leq j < 45$)																						
j-th block of Group-wise interleaver output	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	
$\pi(j)$ -th block of Group-wise interleaver input	13	16	4	12	44	15	8	14	0	3	30	20	35	21	10	6	19	17	26	39	7	24	9
	27	5	37	23	32	40	31	38	42	34	25	36	2	22	43	33	28	1	18	11	41	29	

[0227]

- [0228] 표 16의 경우, 수학적 식 21은 $Y_0=X_{\pi(0)}=X_{13}$, $Y_1=X_{\pi(1)}=X_{16}$, $Y_2=X_{\pi(2)}=X_4$, ..., $Y_{43}=X_{\pi(43)}=X_{41}$, $Y_{44}=X_{\pi(44)}=X_{29}$ 와 같이 나타낼 수 있다. 이에 따라, 그룹 인터리버(122)는 13 번째 비트 그룹을 0 번째로, 16 번째 비트 그룹을 1 번째로, 4 번째 비트 그룹을 2 번째로, ..., 41 번째 비트 그룹을 43 번째로, 29 번째 비트 그룹을 44 번째로 순서를 변경하여 복수의 비트 그룹의 순서를 비트 그룹 단위로 재정렬할 수 있다.

[0229] 다른 예로, LDPC 부호어의 길이가 16200이고 부호율이 9/15이고, 변조 방식이 256-QAM인 경우, $\pi(j)$ 는 하기의 표 17과 같이 정의될 수 있다. 특히, 표 17의 경우, 표 7에 의해 정의되는 패리티 검사 행렬에 기초하여 LDPC 부호화를 수행한 경우에 적용될 수 있다.

표 17

	Order of bits group to be block interleaved $\pi(j) (0 \leq j < 45)$																						
j-th block of Group-wise interleaver output	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
$\pi(j)$ -th block of Group-wise interleaver input	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	
	5	7	9	22	10	12	3	43	6	4	24	13	14	11	15	18	19	17	16	41	25	26	20
	23	21	33	31	28	39	36	30	37	27	32	34	35	29	2	42	0	1	8	40	38	44	

[0230]

[0231] 표 17의 경우, 수학적 식 21은 $Y_0=X_{\pi(0)}=X_5$, $Y_1=X_{\pi(1)}=X_7$, $Y_2=X_{\pi(2)}=X_9$, ..., $Y_{43}=X_{\pi(43)}=X_{38}$, $Y_{44}=X_{\pi(44)}=X_{44}$ 와 같이 나타낼 수 있다. 이에 따라, 그룹 인터리버(122)는 5 번째 비트 그룹을 0 번째로, 7 번째 비트 그룹을 1 번째로, 9 번째 비트 그룹을 2 번째로, ..., 38 번째 비트 그룹을 43 번째로, 44 번째 비트 그룹을 44 번째로 순서를 변경하여 복수의 비트 그룹의 순서를 비트 그룹 단위로 재정렬할 수 있다.

[0232] 다른 예로, LDPC 부호어의 길이가 16200이고 부호율이 11/15이고, 변조 방식이 256-QAM인 경우, $\pi(j)$ 는 하기의 표 18과 같이 정의될 수 있다. 특히, 표 18의 경우, 표 9에 의해 정의되는 패리티 검사 행렬에 기초하여 LDPC 부호화를 수행한 경우에 적용될 수 있다.

표 18

	Order of bits group to be block interleaved $\pi(j) (0 \leq j < 45)$																						
j-th block of Group-wise interleaver output	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
$\pi(j)$ -th block of Group-wise interleaver input	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	
	25	13	4	5	31	20	2	8	10	22	17	24	19	23	28	18	29	27	26	9	16	21	7
	11	14	44	34	33	12	35	43	6	42	41	3	1	38	40	39	37	0	30	32	15	36	

[0233]

[0234] 표 18의 경우, 수학적 식 21은 $Y_0=X_{\pi(0)}=X_{25}$, $Y_1=X_{\pi(1)}=X_{13}$, $Y_2=X_{\pi(2)}=X_4$, ..., $Y_{43}=X_{\pi(43)}=X_{15}$, $Y_{44}=X_{\pi(44)}=X_{36}$ 과 같이 나타낼 수 있다. 이에 따라, 그룹 인터리버(122)는 25 번째 비트 그룹을 0 번째로, 13 번째 비트 그룹을 1 번째로, 4 번째 비트 그룹을 2 번째로, ..., 15 번째 비트 그룹을 43 번째로, 36 번째 비트 그룹을 44 번째로 순서를 변경하여 복수의 그룹의 순서를 비트 그룹 단위로 재정렬할 수 있다.

[0235] 다른 예로, LDPC 부호어의 길이가 16200이고 부호율이 13/15이고, 변조 방식이 256-QAM인 경우, $\pi(j)$ 는 하기의 표 19와 같이 정의될 수 있다. 특히, 표 19의 경우, 표 11에 의해 정의되는 패리티 검사 행렬에 기초하여 LDPC 부호화를 수행한 경우에 적용될 수 있다.

표 19

	Order of bits group to be block interleaved $\pi(j) (0 \leq j < 45)$																						
j-th block of Group-wise interleaver output	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
$\pi(j)$ -th block of Group-wise interleaver input	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	
	9	13	10	7	11	6	1	14	12	8	21	15	4	36	25	30	24	28	29	20	27	5	18
	17	22	33	0	16	23	31	42	3	40	39	41	43	37	44	26	2	19	38	32	35	34	

[0236]

[0237] 표 19의 경우, 수학적 식 21은 $Y_0=X_{\pi(0)}=X_9$, $Y_1=X_{\pi(1)}=X_{13}$, $Y_2=X_{\pi(2)}=X_{10}$, ..., $Y_{43}=X_{\pi(43)}=X_{35}$, $Y_{44}=X_{\pi(44)}=X_{34}$ 와 같이 나타낼 수 있다. 이에 따라, 그룹 인터리버(122)는 9 번째 비트 그룹을 0 번째로, 13 번째 비트 그룹을 1 번째로, 10 번째 비트 그룹을 2 번째로, ..., 35 번째 비트 그룹을 43 번째로, 34 번째 비트 그룹을 44 번째로 순서를 변경하여 복수의 비트 그룹의 순서를 비트 그룹 단위로 재정렬할 수 있다.

[0238] 다른 예로, LDPC 부호어의 길이가 16200이고 부호율이 5/15이고, 변조 방식이 256-QAM인 경우, $\pi(j)$ 는 하기의 표 20과 같이 정의될 수 있다. 특히, 표 20의 경우, 표 4에 의해 정의되는 패리티 검사 행렬에 기초하여 LDPC 부호화를 수행한 경우에 적용될 수 있다.

표 20

	Order of bits group to be block interleaved $\pi(j) (0 \leq j < 45)$																						
j-th block of Group-wise interleaver output	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	
$\pi(j)$ -th block of Group-wise interleaver input	8	9	0	7	4	10	12	14	1	13	16	11	3	6	42	28	35	21	32	20	29	39	22
	37	17	18	25	34	24	43	30	27	33	23	15	44	19	36	41	2	5	26	38	31	40	

[0239]

[0240] 표 20의 경우, 수학적 식 21은 $Y_0=X_{\pi(0)}=X_8$, $Y_1=X_{\pi(1)}=X_9$, $Y_2=X_{\pi(2)}=X_0$, ..., $Y_{43}=X_{\pi(43)}=X_{31}$, $Y_{44}=X_{\pi(44)}=X_{40}$ 과 같이 나타낼 수 있다. 이에 따라, 그룹 인터리버(122)는 8 번째 비트 그룹을 0 번째로, 9 번째 비트 그룹을 1 번째로, 0 번째 비트 그룹을 2 번째로, ..., 31 번째 비트 그룹을 43 번째로, 40 번째 비트 그룹을 44 번째로 순서를 변경하여 복수의 비트 그룹의 순서를 비트 그룹 단위로 재정렬할 수 있다.

[0241] 다른 예로, LDPC 부호어의 길이가 16200이고 부호율이 7/15이고, 변조 방식이 256-QAM인 경우, $\pi(j)$ 는 하기의 표 21과 같이 정의될 수 있다. 특히, 표 21의 경우, 표 6에 의해 정의되는 패리티 검사 행렬에 기초하여 LDPC 부호화를 수행한 경우에 적용될 수 있다.

표 21

	Order of bits group to be block interleaved $\pi(j) (0 \leq j < 45)$																						
j-th block of Group-wise interleaver output	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	
$\pi(j)$ -th block of Group-wise interleaver input	9	8	4	0	14	1	28	18	17	20	11	13	5	15	10	16	33	41	38	21	7	32	6
	24	36	31	37	43	22	26	27	35	44	25	34	29	23	30	3	39	2	12	19	42	40	

[0242]

[0243] 표 21의 경우, 수학적 식 21은 $Y_0=X_{\pi(0)}=X_9$, $Y_1=X_{\pi(1)}=X_8$, $Y_2=X_{\pi(2)}=X_4$, ..., $Y_{43}=X_{\pi(43)}=X_{42}$, $Y_{44}=X_{\pi(44)}=X_{40}$ 과 같이 나타낼 수 있다. 이에 따라, 그룹 인터리버(122)는 9 번째 비트 그룹을 0 번째로, 8 번째 비트 그룹을 1 번째로, 4 번째 비트 그룹을 2 번째로, ..., 42 번째 비트 그룹을 43 번째로, 40 번째 비트 그룹을 44 번째로 순서를 변경하여 복수의 비트 그룹의 순서를 비트 그룹 단위로 재정렬할 수 있다.

[0244] 다른 예로, LDPC 부호어의 길이가 16200이고 부호율이 9/15이고, 변조 방식이 256-QAM인 경우, $\pi(j)$ 는 하기의 표 22와 같이 정의될 수 있다. 특히, 표 22의 경우, 표 8에 의해 정의되는 패리티 검사 행렬에 기초하여 LDPC 부호화를 수행한 경우에 적용될 수 있다.

표 22

	Order of bits group to be block interleaved $\pi(j) (0 \leq j < 45)$																						
j-th block of Group-wise interleaver output	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	
$\pi(j)$ -th block of Group-wise interleaver input	14	4	9	8	0	18	28	20	1	17	13	5	11	15	10	21	41	16	38	33	24	7	6
	32	36	37	31	22	26	43	44	34	27	35	25	30	23	3	29	39	2	12	19	42	40	

[0245]

[0246] 표 22의 경우, 수학적 식 21은 $Y_0=X_{\pi(0)}=X_{14}$, $Y_1=X_{\pi(1)}=X_4$, $Y_2=X_{\pi(2)}=X_9$, ..., $Y_{43}=X_{\pi(43)}=X_{42}$, $Y_{44}=X_{\pi(44)}=X_{40}$ 과 같이 나타낼 수 있다. 이에 따라, 그룹 인터리버(122)는 14 번째 비트 그룹을 0 번째로, 4 번째 비트 그룹을 1 번째로, 9 번째 비트 그룹을 2 번째로, ..., 42 번째 비트 그룹을 43 번째로, 40 번째 비트 그룹을 44 번째로 순서를 변경

하여 복수의 비트 그룹의 순서를 비트 그룹 단위로 재정렬할 수 있다.

[0247] 다른 예로, LDPC 부호어의 길이가 16200이고 부호율이 11/15이고, 변조 방식이 256-QAM인 경우, $\pi(j)$ 는 하기의 표 23과 같이 정의될 수 있다. 특히, 표 23의 경우, 표 10에 의해 정의되는 패리티 검사 행렬에 기초하여 LDPC 부호화를 수행한 경우에 적용될 수 있다.

표 23

	Order of bits group to be block interleaved $\pi(j) \ (0 \leq j < 45)$																							
j-th block of Group-wise interleaver output	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	
	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44		
	10	28	30	8	9	14	16	15	0	13	27	18	22	17	20	5	29	25	41	26	3	2	34	
$\pi(j)$ -th block of Group-wise interleaver input	6	4	38	40	35	7	24	43	19	33	23	39	11	36	42	44	37	1	12	32	31	21		

[0248]

[0249] 표 23의 경우, 수학적 식 21은 $Y_0=X_{\pi(0)}=X_{10}$, $Y_1=X_{\pi(1)}=X_{28}$, $Y_2=X_{\pi(2)}=X_{30}$, ..., $Y_{43}=X_{\pi(43)}=X_{31}$, $Y_{44}=X_{\pi(44)}=X_{21}$ 과 같이 나타낼 수 있다. 이에 따라, 그룹 인터리버(122)는 10 번째 비트 그룹을 0 번째로, 28 번째 비트 그룹을 1 번째로, 30 번째 비트 그룹을 2 번째로, ..., 31 번째 비트 그룹을 43 번째로, 21 번째 비트 그룹을 44 번째로 순서를 변경하여 복수의 비트 그룹의 순서를 비트 그룹 단위로 재정렬할 수 있다.

[0250] 다른 예로, LDPC 부호어의 길이가 16200이고 부호율이 13/15이고, 변조 방식이 256-QAM인 경우, $\pi(j)$ 는 하기의 표 24와 같이 정의될 수 있다. 특히, 표 24의 경우, 표 12에 의해 정의되는 패리티 검사 행렬에 기초하여 LDPC 부호화를 수행한 경우에 적용될 수 있다.

표 24

	Order of bits group to be block interleaved $\pi(j) (0 \leq j < 45)$																							
j-th block of Group-wise interleaver output	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	
	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44		
$\pi(j)$ -th block of Group-wise interleaver input	21	19	7	8	6	11	15	9	14	12	18	13	23	16	17	34	20	32	27	1	2	0	10	
	3	4	35	25	31	30	28	40	39	44	42	41	22	26	29	43	24	5	36	37	38	33		

[0251]

[0252] 표 24의 경우, 수학적 식 21은 $Y_0=X_{\pi(0)}=X_{21}$, $Y_1=X_{\pi(1)}=X_{19}$, $Y_2=X_{\pi(2)}=X_7$, ..., $Y_{43}=X_{\pi(43)}=X_{38}$, $Y_{44}=X_{\pi(44)}=X_{33}$ 과 같이 나타낼 수 있다. 이에 따라, 그룹 인터리버(122)는 21 번째 비트 그룹을 0 번째로, 19 번째 비트 그룹을 1 번째로, 7 번째 비트 그룹을 2 번째로, ..., 38 번째 비트 그룹을 43 번째로, 33 번째 비트 그룹을 44 번째로 순서를 변경하여 복수의 비트 그룹의 순서를 비트 그룹 단위로 재정렬할 수 있다.

[0253] 다른 예로, LDPC 부호어의 길이가 16200이고 부호율이 11/15이고, 변조 방식이 256-QAM인 경우, $\pi(j)$ 는 하기의 표 25와 같이 정의될 수 있다. 특히, 표 25의 경우, 표 9에 의해 정의되는 패리티 검사 행렬에 기초하여 LDPC 부호화를 수행한 경우에 적용될 수 있다.

표 25

	Order of bits group to be block interleaved $\pi(j) (0 \leq j < 45)$																						
j-th block of Group-wise interleaver output	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	
$\pi(j)$ -th block of Group-wise interleaver input	20	16	5	0	31	1	2	26	10	23	13	24	34	22	28	30	29	25	8	9	27	21	12
	18	14	7	17	41	33	35	44	39	42	3	15	4	38	40	6	37	32	11	19	43	36	

[0254]

[0255] 표 25의 경우, 수학적 식 21은 $Y_0=X_{\pi(0)}=X_{20}$, $Y_1=X_{\pi(1)}=X_{16}$, $Y_2=X_{\pi(2)}=X_5$, ..., $Y_{43}=X_{\pi(43)}=X_{43}$, $Y_{44}=X_{\pi(44)}=X_{36}$ 과 같이 나타낼 수 있다. 이에 따라, 그룹 인터리버(122)는 20 번째 비트 그룹을 0 번째로, 16 번째 비트 그룹을 1 번째로,

5 번째 비트 그룹을 2 번째로, ..., 43 번째 비트 그룹을 43 번째로, 36 번째 비트 그룹을 44 번째로 순서를 변경하여 복수의 비트 그룹의 순서를 비트 그룹 단위로 재정렬할 수 있다.

[0256] 다른 예로, LDPC 부호어의 길이가 16200이고 부호율이 9/15이고, 변조 방식이 256-QAM인 경우, $\pi(j)$ 는 하기의 표 26과 같이 정의될 수 있다. 특히, 표 26의 경우, 표 8에 의해 정의되는 패리티 검사 행렬에 기초하여 LDPC 부호화를 수행한 경우에 적용될 수 있다.

표 26

	Order of bits group to be block interleaved $\pi(j)$ ($0 \leq j < 45$)																							
j-th block of Group-wise interleaver output	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	
$\pi(j)$ -th block of Group-wise interleaver input	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44		
	8	4	0	1	20	19	10	12	5	22	21	15	26	17	16	11	28	6	42	13	25	33	18	
	9	14	31	43	44	23	36	34	27	2	38	37	35	40	30	29	3	24	32	7	41	39		

[0257]

[0258] 표 26의 경우, 수학적 식 21은 $Y_0=X_{\pi(0)}=X_8$, $Y_1=X_{\pi(1)}=X_4$, $Y_2=X_{\pi(2)}=X_0$, ..., $Y_{43}=X_{\pi(43)}=X_{41}$, $Y_{44}=X_{\pi(44)}=X_{39}$ 와 같이 나타낼 수 있다. 이에 따라, 그룹 인터리버(122)는 8 번째 비트 그룹을 0 번째로, 4 번째 비트 그룹을 1 번째로, 0 번째 비트 그룹을 2 번째로, ..., 41 번째 비트 그룹을 43 번째로, 39 번째 비트 그룹을 44 번째로 순서를 변경하여 복수의 비트 그룹의 순서를 비트 그룹 단위로 재정렬할 수 있다.

[0259] 다른 예로, LDPC 부호어의 길이가 16200이고 부호율이 11/15이고, 변조 방식이 256-QAM인 경우, $\pi(j)$ 는 하기의 표 27과 같이 정의될 수 있다. 특히, 표 27의 경우, 표 10에 의해 정의되는 패리티 검사 행렬에 기초하여 LDPC 부호화를 수행한 경우에 적용될 수 있다.

표 27

	Order of bits group to be block interleaved $\pi(j)$ ($0 \leq j < 45$)																						
j-th block of Group-wise interleaver output	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
$\pi(j)$ -th block of Group-wise interleaver input	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	
	28	30	10	32	9	1	24	15	0	2	12	27	14	17	22	29	41	25	11	26	4	21	6
	13	3	35	18	16	7	40	23	19	39	33	43	42	5	44	36	37	8	20	38	31	34	

[0260]

[0261] 표 27의 경우, 수학적 식 21은 $Y_0=X_{\pi(0)}=X_{28}$, $Y_1=X_{\pi(1)}=X_{30}$, $Y_2=X_{\pi(2)}=X_{10}$, ..., $Y_{43}=X_{\pi(43)}=X_{31}$, $Y_{44}=X_{\pi(44)}=X_{34}$ 와 같이 나타낼 수 있다. 이에 따라, 그룹 인터리버(122)는 28 번째 비트 그룹을 0 번째로, 30 번째 비트 그룹을 1 번째로, 10 번째 비트 그룹을 2 번째로, ..., 31 번째 비트 그룹을 43 번째로, 34 번째 비트 그룹을 44 번째로 순서를 변경하여 복수의 비트 그룹의 순서를 비트 그룹 단위로 재정렬할 수 있다.

[0262] 한편, 상술한 예에서는 LDPC 부호어의 길이가 16200이고 부호율이 5/15, 7/15, 9/15, 11/15, 13/15인 경우만을 설명하였으나 이는 일 예에 불과하고, LDPC 부호어의 길이가 64800인 경우이거나 다른 부호율을 갖는 경우에도 인터리빙 패턴은 다양하게 정의될 수 있다.

[0263] 이와 같이, 그룹 인터리버(12)는 수학적 식 21 및 표 15 내지 표 27을 이용하여 복수의 비트 그룹의 순서를 비트 그룹 단위로 재정렬할 수 있다.

[0264] 한편, 표 15 내지 표 27에서 "j-th block of Group-wise Interleaver output"는 인터리빙 후 그룹 인터리버(122)에서 j 번째로 출력되는 비트 그룹을 나타내고, " $\pi(j)$ -th block of Group-wise Interleaver input"은 그룹 인터리버(122)에 $\pi(j)$ 번째로 입력되는 비트 그룹을 나타낸다.

[0265] 또한, LDPC 부호어를 구성하는 그룹들은 그룹 인터리버(122)에 의해 비트 그룹 단위로 순서가 재정렬된 후, 후술할 블록 인터리버(124)에 의해 블록 인터리빙된다는 점에서, 표 15 내지 표 27에서 $\pi(j)$ 와 관련하여 "Order of bits group to be block interleaved"와 같이 기재하였다.

[0266] 이러한 방식에 따라, 그룹 인터리빙된 LDPC 부호어는 도 7과 같다. 도 7에 도시된 LDPC 부호어를 도 6에 도시된

그룹 인터리빙되기 전의 LDPC 부호어와 비교하면, LDPC 부호어를 구성하는 복수의 비트 그룹의 순서가 재정렬된 것을 알 수 있다.

[0267] 즉, 도 6 및 도 7과 같이 LDPC 부호어는 그룹 인터리빙되기 전에 비트 그룹 X_0 , 비트 그룹 $X_1, \dots$, 비트 그룹

$X_{N_{group}-1}$ 순으로 배치되었다가, 그룹 인터리빙되어 비트 그룹 Y_0 , 비트 그룹 $Y_1, \dots$, 비트 그룹 $Y_{N_{group}-1}$ 순으로 배치될 수 있다. 이 경우, 그룹 인터리빙에 의해 각 비트 그룹들이 배치되는 순서는 표 15 내지 표 27에 기초하여 결정될 수 있다.

[0268] 그룹 트위스트 인터리버(123)는 동일한 비트 그룹 내의 비트들을 인터리빙한다. 즉, 그룹 트위스트 인터리버(123)는 동일한 비트 그룹 내에 존재하는 비트들의 순서를 변경하여 동일한 비트 그룹 내의 비트들의 순서를 재정렬할 수 있다.

[0269] 이 경우, 그룹 트위스트 인터리버(123)는 동일한 비트 그룹 내의 비트들을 일정한 개수의 비트만큼씩 시클릭 쉬프트하여, 동일한 비트 그룹 내의 비트들의 순서를 재정렬할 수 있다.

[0270] 예를 들어, 도 8과 같이, 그룹 트위스트 인터리버(123)는 비트 그룹 Y_1 에 포함된 비트들을 우측 방향으로 1 비트만큼 시클릭 쉬프트시킬 수 있다. 이 경우, 도 8과 같이 비트 그룹 Y_1 에서 0 번째, 1 번째, 2 번째, $\dots$, 358 번째, 359 번째에 각각 위치하던 비트들은 1 비트만큼 우측으로 시클릭 쉬프트되어, 시클릭 쉬프트되기 전의 359 번째 위치하던 비트가 비트 그룹 Y_1 내에서 가장 앞쪽에 위치하게 되고 시클릭 쉬프트되기 전의 0 번째, 1 번째, 2 번째, $\dots$, 358 번째에 각각 위치하던 비트들은 차례로 우측으로 1 비트만큼 쉬프트되어 위치하게 된다.

[0271] 또한, 그룹 트위스트 인터리버(123)는 각 비트 그룹별로 서로 다른 개수의 비트만큼을 시클릭 쉬프트하여, 각 그룹 내의 비트들의 순서를 재정렬할 수도 있다.

[0272] 예를 들어, 그룹 트위스트 인터리버(123)는 비트 그룹 Y_1 에 포함된 비트들을 우측 방향으로 1 비트만큼 시클릭 쉬프트하고, 비트 그룹 Y_2 에 포함된 비트들을 우측 방향으로 3 비트만큼 시클릭 쉬프트할 수 있다.

[0273] 다만, 상술한 그룹 트위스트 인터리버(123)는 경우에 따라 생략될 수도 있다.

[0274] 또한, 상술한 예에서 그룹 트위스트 인터리버(123)가 그룹 인터리버(122) 이후에 배치되는 것으로 설명하였으나 이 역시 일 예에 불과하다. 즉, 그룹 트위스트 인터리버(123)는 비트 그룹 내에서 해당 비트 그룹을 구성하는 비트들의 순서를 변경할 뿐 비트 그룹 자체의 순서를 변경하는 것은 아니라는 점에서 그룹 인터리버(122) 전에 배치될 수도 있다.

[0275] 블록 인터리버(124)는 순서가 재정렬된 복수의 비트 그룹을 인터리빙한다. 구체적으로, 블록 인터리버(124)는 그룹 인터리버(122)에 의해 비트 그룹의 순서가 재정렬된 복수의 비트 그룹을 비트 그룹 단위로 인터리빙할 수 있다. 여기에서, 블록 인터리버(124)는 각각 복수의 행(row)을 포함하는 복수의 열(column)로 구성되며, 변조 방식에 따라 결정되는 변조 차수에 기초하여 재정렬된 복수의 비트 그룹을 구분하여 인터리빙할 수 있다.

[0276] 이 경우, 블록 인터리버(124)는 그룹 인터리버(122)에 의해 비트 그룹의 순서가 재정렬된 복수의 비트 그룹을 비트 그룹 단위로 인터리빙할 수 있으며, 구체적으로, 블록 인터리버(124)는 제1 파트(part 1) 및 제2 파트(part 2)를 이용하여 재정렬된 복수의 비트 그룹을 변조 차수에 따라 구분하여 인터리빙할 수 있다.

[0277] 구체적으로, 블록 인터리버(124)는 복수의 열 각각을 제1 파트 및 제2 파트로 구분하고, 복수의 비트 그룹을 제1 파트를 구성하는 복수의 열에 비트 그룹 단위로 순차적으로 라이트(write)하고, 나머지 비트 그룹을 구성하는 비트들을 복수의 열의 개수에 기초하여 각각 기설정된 비트 수로 구성되는 서브 비트 그룹(sub bit group)으로 분할하고 분할된 서브 비트 그룹을 제2 파트를 구성하는 복수의 열에 순차적으로 라이트하여 인터리빙을 수행할 수 있다.

[0278] 여기에서, 비트 그룹 단위로 인터리빙되는 그룹의 개수는 블록 인터리버(124)를 구성하는 행 및 열의 개수, 비트 그룹의 개수 및 각 비트 그룹에 포함된 비트 수 중 적어도 하나에 따라 결정될 수 있다. 즉, 블록 인터리버(124)는 블록 인터리버(124)를 구성하는 행 및 열의 개수, 비트 그룹의 개수 및 각 비트 그룹에 포함된 비트 수 중 적어도 하나를 고려하여 복수의 비트 그룹 중 비트 그룹 단위로 인터리빙되는 비트 그룹을 결정하고, 해당 비트 그룹을 비트 그룹 단위로 인터리빙하고, 나머지 비트 그룹을 구성하는 비트들을 모다 서브 비트 그룹으로

분할하여 인터리빙할 수 있다. 예를 들어, 블록 인터리버(124)는 제1 파트를 이용하여 복수의 비트 그룹 중 적어도 일부를 비트 그룹 단위로 인터리빙하고, 제2 파트를 이용하여 나머지 비트 그룹을 분할하여 인터리빙할 수 있다.

- [0279] 한편, 비트 그룹 단위로 인터리빙된다는 것은 동일한 비트 그룹에 포함된 비트들은 동일한 열에 라이트되는 것을 의미한다. 즉, 블록 인터리버(124)는 비트 그룹 단위로 인터리빙되는 비트 그룹의 경우 동일한 비트 그룹에 포함된 비트들을 분할하지 않고 동일한 열에 라이트하고, 비트 그룹 단위로 인터리빙되지 않는 비트 그룹의 경우 해당 그 비트 룩에 포함된 비트들을 분할하여 서로 다른 열에 라이트하여 인터리빙을 수행할 수 있다.
- [0280] 이에 따라, 제1 파트를 구성하는 행의 개수는 비트 그룹 하나에 포함된 비트 수(가령, 360)의 배수가 되고, 제2 파트를 구성하는 행의 개수는 그룹 하나에 포함된 비트 수보다 작을 수 있다.
- [0281] 또한, 제1 파트에 의해 인터리빙되는 모든 비트 그룹은 동일한 비트 그룹에 포함된 비트들이 제1 파트의 동일한 열에 라이트되어 인터리빙되고, 제2 파트에 의해 인터리빙되는 적어도 하나의 비트 그룹은 제2 파트를 구성하는 적어도 두 개의 열에 분할되어 라이트될 수 있다.
- [0282] 이러한 인터리빙 방식에 대한 구체적인 내용은 후술하기로 한다.
- [0283] 한편, 그룹 트위스트 인터리버(123)에서 수행되는 인터리빙은 비트 그룹 내에서 비트들의 순서를 변경하는 것일 뿐 인터리빙에 의해 비트 그룹 자체의 순서가 변경되지 않는다. 따라서, 블록 인터리버(124)에서 블록 인터리빙되는 비트 그룹의 순서 즉, 블록 인터리버(124)에 입력되는 비트 그룹의 순서는 그룹 인터리버(122)에 의해 결정될 수 있다. 예를 들어, 블록 인터리버(124)에 의해 블록 인터리빙되는 비트 그룹의 순서는 표 15 내지 표 27에서 정의되는 $\pi(j)$ 에 의해 결정될 수 있다.
- [0284] 상술한 바와 같이, 블록 인터리버(124)는 각각 복수의 행으로 이루어진 복수의 열을 이용하여 순서가 재정렬된 복수의 비트 그룹을 비트 그룹 단위로 인터리빙할 수 있다.
- [0285] 이 경우, 블록 인터리버(124)는 복수의 열을 적어도 두 개의 파트로 구분하여 LDPC 부호어를 인터리빙할 수 있다. 예를 들어, 블록 인터리버(124)는 복수의 열 각각을 제1 파트 및 제2 파트로 구분하여 LDPC 부호어를 구성하는 복수의 비트 그룹을 인터리빙할 수 있다.
- [0286] 이 경우, 블록 인터리버(124)는 LDPC 부호어를 구성하는 비트 그룹의 개수가 블록 인터리버(124)를 구성하는 열의 개수의 정수 배가 되는지 여부에 따라 복수의 열 각각을 N 개(N은 2 이상의 정수)의 파트로 구분하여 인터리빙을 수행할 수 있다.
- [0287] 먼저, 블록 인터리버(124)는 LDPC 부호어를 구성하는 비트 그룹의 개수가 블록 인터리버(124)를 구성하는 열의 개수의 정수 배가 되는 경우, 복수의 열 각각을 파트를 구분하지 않고 LDPC 부호어를 구성하는 복수의 비트 그룹을 비트 그룹 단위로 인터리빙할 수 있다.
- [0288] 구체적으로, 블록 인터리버(124)는 LDPC 부호어를 구성하는 복수의 비트 그룹을 비트 그룹 단위로 열 각각에 열 방향으로 라이트하고, 복수의 비트 그룹이 비트 그룹 단위로 라이트된 복수의 열의 각 행을 행 방향으로 리드(read)하여 인터리빙을 수행할 수 있다.
- [0289] 이 경우, 블록 인터리버(124)는 LDPC 부호어를 구성하는 비트 그룹의 개수를 블록 인터리버(124)를 구성하는 열의 개수로 나눈 몫(quotient)만큼의 비트 그룹에 포함된 비트들을 복수의 열 각각에 열 방향으로 순차적으로 라이트하고, 비트들이 라이트된 복수의 열의 각 행을 행 방향으로 리드하여 인터리빙을 수행할 수 있다.
- [0290] 이하에서는 설명의 편의를 위해, 그룹 인터리버(122)에서 인터리빙된 후 j 번째에 위치하는 비트 그룹을 비트 그룹 Y_j 라 하도록 한다.
- [0291] 예를 들어, 블록 인터리버(124)가 각각 R_i 개의 행을 포함하는 C 개의 열로 구성되는 경우를 가정한다. 그리고, LDPC 부호어가 N_{group} 개의 비트 그룹으로 구성되고, 비트 그룹의 개수인 N_{group} 이 C의 배수가 되는 경우를 가정한다.
- [0292] 이 경우, LDPC 부호어를 구성하는 비트 그룹의 개수 N_{group} 를 블록 인터리버(124)를 구성하는 열의 개수 C로 나눈 몫이 $A(=N_{\text{group}}/C)$ 인 경우(A는 0보다 큰 정수), 블록 인터리버(124)는 각 열에 $A(=N_{\text{group}}/C)$ 개의 비트 그룹씩을 순차적으로 열 방향으로 라이트하고, 각 열에 라이트된 비트들을 행 방향으로 리드하여 인터리빙을 수행할 수

있다.

- [0293] 예를 들어, 도 9와 같이, 블록 인터리버(124)는 제1 컬럼의 1 번째 행부터 R_1 번째 행에 비트 그룹(Y_0), 비트 그룹(Y_1), ..., 비트 그룹(Y_{A-1}) 각각에 포함된 비트들을 라이트하고, 제2 컬럼의 1 번째 행부터 R_1 번째 행에 비트 그룹(Y_A), 비트 그룹(Y_{A+1}), ..., 비트 그룹(Y_{2A-1}) 각각에 포함된 비트들을 라이트하고, ..., 제C 컬럼의 1 번째 행부터 R_1 번째 행에 비트 그룹(Y_{CA-A}), 비트 그룹(Y_{CA-A+1}), ..., 비트 그룹(Y_{CA-1}) 각각에 포함된 비트들을 라이트하고, 복수의 열의 각 행에 라이트된 비트들을 행 방향으로 리드할 수 있다.
- [0294] 이에 따라, 블록 인터리버(124)는 LDPC 부호어를 구성하는 모든 비트 그룹들을 비트 그룹 단위로 인터리빙하게 된다.
- [0295] 다만, 블록 인터리버(124)는 LDPC 부호어를 구성하는 비트 그룹의 개수가 블록 인터리버(124)를 구성하는 열의 개수의 정수 배가 되지 않는 경우, 복수의 열 각각을 두 개의 파트로 구분하여 LDPC 부호어를 구성하는 복수의 비트 그룹 중 일부를 비트 그룹 단위로 인터리빙하고, 나머지 비트 그룹을 구성하는 비트들을 모아 서브 비트 그룹으로 분할하여 인터리빙할 수 있다. 이 경우, 나머지 비트 그룹에 포함된 비트들 즉, LDPC 부호어를 구성하는 그룹의 개수를 열의 개수로 나눈 경우의 그 나머지(remainder)만큼의 비트 그룹에 포함된 비트들은 비트 그룹 단위로 인터리빙되는 것이 아니라, 열의 개수에 따라 각 열에 분할되어 인터리빙될 수 있다.
- [0296] 구체적으로, 블록 인터리버(124)는 복수의 열 각각을 두 개의 파트로 구분하여 LDPC 부호어를 인터리빙할 수 있다.
- [0297] 이 경우, 블록 인터리버(124)는 블록 인터리버(124)를 구성하는 열의 개수, LDPC 부호어를 구성하는 비트 그룹의 개수 및 복수의 비트 그룹 각각을 구성하는 비트 수에 기초하여 복수의 열을 제1 파트 및 제2 파트로 구분할 수 있다.
- [0298] 여기에서, 복수의 비트 그룹 각각은 360 개의 비트로 구성될 수 있다. 그리고, LDPC 부호어를 구성하는 비트 그룹의 개수는 LDPC 부호어의 길이 및 각 비트 그룹에 포함된 비트 수에 따라 결정된다. 예를 들어, 길이가 16200 인 LDPC 부호어를 각 비트 그룹이 360 개의 비트로 구성되도록 구분하면 LDPC 부호어는 45 개의 비트 그룹으로 구분되고, 길이가 64800인 LDPC 부호어를 각 비트 그룹이 360 개의 비트로 구성되도록 구분하면 LDPC 부호어는 180 개의 비트 그룹으로 구분될 수 있다. 또한, 블록 인터리버(124)를 구성하는 열의 개수는 변조 방식에 따라 결정될 수 있으며, 이와 관련된 구체적인 예는 후술하기로 한다.
- [0299] 이에 따라, 제1 파트 및 제2 파트 각각을 구성하는 행의 개수는 블록 인터리버(124)를 구성하는 열의 개수, LDPC 부호어를 구성하는 비트 그룹의 개수 및 복수의 비트 그룹 각각을 구성하는 비트 수에 기초하여 결정될 수 있다.
- [0300] 구체적으로, 제1 파트는 복수의 열 각각에서, 블록 인터리버(124)를 구성하는 열의 개수, LDPC 부호어를 구성하는 비트 그룹의 개수 및 각 비트 그룹을 구성하는 비트 수에 따라 LDPC 부호어를 구성하는 복수의 비트 그룹 중 복수의 열 각각에 비트 그룹 단위로 라이트 가능한 적어도 하나의 비트 그룹에 포함된 비트 수만큼의 행으로 구성될 수 있다.
- [0301] 그리고, 제2 파트는 복수의 열 각각에서, 복수의 열 각각을 구성하는 행에서 복수의 열 각각에 비트 그룹 단위로 라이트 가능한 적어도 일부의 비트 그룹에 포함된 비트 수만큼의 행을 제외한 행으로 구성될 수 있다. 구체적으로, 제2 파트의 행의 개수는 제1 파트에 대응되는 비트 그룹을 제외한 모든 비트 그룹에 포함된 비트의 수를 블록 인터리버(124)를 구성하는 열의 개수로 나눈 몫과 동일한 값을 가질 수 있다. 즉, 제2 파트의 행의 개수는 LDPC 부호어를 구성하는 비트 그룹 중 제1 파트에 라이트되고 남은 비트 그룹들에 포함된 비트 수를 열의 개수로 나눈 몫과 동일한 값을 가질 수 있다.
- [0302] 한편, 블록 인터리버(124)는 복수의 열 각각을, 각 열에 비트 그룹 단위로 라이트 가능한 비트 그룹에 포함된 비트 수만큼의 행을 포함하는 제1 파트와 그 외의 나머지 행을 포함하는 제2 파트로 복수의 열 각각을 구분할 수 있다.
- [0303] 이에 따라, 제1 파트는 비트 그룹에 포함된 비트 수 즉, M의 정수 배만큼의 행으로 구성될 수 있다. 다만, 상술한 바와 같이, 각 비트 그룹을 구성하는 부호어 비트의 개수는 M의 약수가 될 수 있다는 점에서, 제1 파트는 각 비트 그룹을 구성하는 비트의 개수의 정수 배만큼의 행으로 구성될 수도 있다.
- [0304] 이 경우, 블록 인터리버(124)는 제1 파트 및 제2 파트에서 LDPC 부호어를 동일한 방식으로 라이트 및 리드하여

인터리빙을 수행할 수 있다.

- [0305] 구체적으로, 블록 인터리버(124)는 LDPC 부호어를 제1 파트 및 제2 파트 각각을 구성하는 복수의 열에 열 방향으로 라이트하고, LDPC 부호어가 라이트된 제1 파트 및 제2 파트 각각을 구성하는 복수의 열을 행 방향으로 리드하여 인터리빙을 수행할 수 있다.
- [0306] 즉, 블록 인터리버(124)는 복수의 열 각각에서 비트 그룹 단위로 라이트 가능한 적어도 일부의 비트 그룹에 포함된 비트들을 제1 파트를 구성하는 복수의 열 각각에 순차적으로 라이트하고, 복수의 비트 그룹에서 적어도 일부의 비트 그룹을 제외한 나머지 비트 그룹에 포함된 비트들을 분할하여 제2 파트를 구성하는 복수의 열 각각에 열 방향으로 라이트하고, 제1 파트 및 제2 파트 각각을 구성하는 복수의 열 각각에 라이트된 비트들을 행 방향으로 리드하여 인터리빙을 수행할 수 있다.
- [0307] 이 경우, 블록 인터리버(124)는 복수의 그룹에서 적어도 일부의 비트 그룹을 제외한 나머지 비트 그룹을 블록 인터리버(124)를 구성하는 열의 개수에 기초하여 분할하여 인터리빙을 수행할 수 있다.
- [0308] 구체적으로, 블록 인터리버(124)는 나머지 비트 그룹에 포함된 비트들을 복수의 열의 개수로 분할하고, 분할된 비트들 각각을 제2 파트를 구성하는 복수의 열 각각에 열 방향으로 라이트하고, 분할된 비트들이 라이트된 제2 파트를 구성하는 복수의 열을 행 방향으로 리드하여 인터리빙을 수행할 수 있다.
- [0309] 즉, 블록 인터리버(124)는 LDPC 부호어를 구성하는 복수의 비트 그룹 중 제1 파트에 라이트되고 남은 비트 그룹 즉, LDPC 부호어를 구성하는 비트 그룹의 개수를 열의 개수로 나눈 경우의 그 나머지(remainder)만큼의 비트 그룹에 포함된 비트들을 열의 개수로 분할하고, 분할된 비트들을 제2 파트의 각 열에 순차적으로 열 방향으로 라이트할 수 있다.
- [0310] 예를 들어, 블록 인터리버(124)가 각각 R_1 개의 행을 포함하는 C 개의 열로 구성되는 경우를 가정한다. 그리고, LDPC 부호어가 N_{group} 개의 비트 그룹으로 구성되고, 비트 그룹의 개수인 N_{group} 이 C 의 배수가 되지 않으며, $A \times C + 1 = N_{\text{group}}$ 인 경우를 가정한다(A 는 0보다 큰 정수). 즉, LDPC 부호어를 구성하는 비트 그룹의 개수를 열의 개수로 나눈 경우, 몫이 A 이고 나머지가 1인 경우를 가정한다.
- [0311] 이 경우, 블록 인터리버(124)는 도 10 및 도 11과 같이 각 열이 R_1 개의 행을 포함하는 제1 파트와 R_2 개의 행을 포함하는 제2 파트로 구분할 수 있다. 이 경우, R_1 은 각 열에 비트 그룹 단위로 라이트 가능한 비트 그룹에 포함된 비트 수만큼이 될 수 있으며, R_2 는 각 열을 구성하는 행의 개수에서 R_1 을 제외한 값이 될 수 있다.
- [0312] 즉, 상술한 예에서 각 열에 비트 그룹 단위로 라이트 가능한 그룹의 수는 A 개이고, 각 열의 제1 파트는 A 개의 그룹에 포함된 비트 수 즉, $A \times M$ 개만큼의 행으로 구성될 수 있다.
- [0313] 이 경우, 블록 인터리버(124)는 각 열에 그룹 단위로 라이트 가능한 그룹 즉, A 개의 그룹에 포함된 비트들을 열 방향으로 각 열의 제1 파트에 라이트한다.
- [0314] 즉, 블록 인터리버(124)는 도 10 및 도 11과 같이 제1 컬럼의 제1 파트를 구성하는 1 번째 행부터 R_1 번째 행에 비트 그룹(Y_0), 비트 그룹(Y_1), ..., 비트 그룹(Y_{A-1}) 각각에 포함된 비트들을 라이트하고, 제2 컬럼의 제1 파트를 구성하는 1 번째 행부터 R_1 번째 행에 비트 그룹(Y_A), 비트 그룹(Y_{A+1}), ..., 비트 그룹(Y_{2A-1}) 각각에 포함된 비트들을 라이트하고, ..., 제C 컬럼의 제1 파트를 구성하는 1 번째 행부터 R_1 번째 행에 비트 그룹(Y_{CA-A}), 비트 그룹(Y_{CA-A+1}), ..., 비트 그룹(Y_{CA-1}) 각각에 포함된 비트들을 라이트한다.
- [0315] 이와 같이, 블록 인터리버 (124)는 각 열에 비트 그룹 단위로 라이트 가능한 비트 그룹에 포함된 비트들을 각 열의 제1 파트에 비트 그룹 단위로 라이트한다.
- [0316] 즉, 상술한 예에서, 비트 그룹(Y_0), 비트 그룹(Y_1), ..., 비트 그룹(Y_{A-1}) 각각에 포함된 비트들은 분할되지 않고 모두 제1 컬럼에 라이트되고, 비트 그룹(Y_A), 비트 그룹(Y_{A+1}), ..., 비트 그룹(Y_{2A-1}) 각각에 포함된 비트들은 분할되지 않고 모두 제2 컬럼에 라이트되고, ..., 비트 그룹(Y_{CA-A}), 비트 그룹(Y_{CA-A+1}), ..., 비트 그룹(Y_{CA-1}) 각각에 포함된 비트들은 분할되지 않고 모두 제C 컬럼에 라이트될 수 있다. 이와 같이, 제1 파트에 의해 인터리빙되는 모든 비트 그룹은 동일한 비트 그룹에 포함된 비트들이 제1 파트의 동일한 열에 라이트되는 것으로 볼 수 있다.

- [0317] 이후, 블록 인터리버(124)는 복수의 비트 그룹 중에서 각 열의 제1 파트에 라이트된 그룹을 제외한 나머지 비트 그룹에 포함된 비트들을 분할하여 각 열의 제2 파트에 열 방향으로 라이트할 수 있다. 이때, 블록 인터리버(124)는 동일한 개수의 비트가 각 열의 제2 파트에 라이트되도록, 각 열의 제1 파트에 라이트된 그룹을 제외한 나머지 비트 그룹에 포함된 비트들을 열의 개수로 분할하고, 분할된 각 비트를 제2 파트의 각 열에 열 방향으로 라이트할 수 있다.
- [0318] 상술한 예에서 $A \times C + 1 = N_{\text{group}}$ 를 만족하므로, LDPC 부호어를 구성하는 비트 그룹을 순차적으로 제1 파트에 라이트할 때, LDPC 부호어의 마지막 비트 그룹인 비트 그룹($Y_{N_{\text{group}}-1}$)가 제1 파트에 라이트되지 못하고 남게 된다. 이에 따라, 블록 인터리버(124)는 도 10과 같이 비트 그룹($Y_{N_{\text{group}}-1}$)에 포함된 비트들을 C 개로 분할하고, 분할된 각 비트들(즉, 마지막 비트 그룹($Y_{N_{\text{group}}-1}$)에 포함된 비트들 C로 나눈 몫만큼의 비트들)을 각 열의 제2 파트에 순차적으로 라이트할 수 있다.
- [0319] 여기에서, 열의 개수에 기초하여 분할된 비트들 각각을 서브 비트 그룹이라 명명할 수 있으며, 이 경우, 서브 비트 그룹 각각이 제2 파트의 각 열에 라이트되는 것으로 볼 수 있다. 즉, 비트 그룹에 포함된 비트들이 분할되어 서브 비트 그룹을 형성할 수 있다.
- [0320] 즉, 블록 인터리버(124)는 제1 컬럼의 제2 파트를 구성하는 1 번째 행부터 R_2 번째 행까지 비트를 라이트하고, 제2 컬럼의 제2 파트를 구성하는 1 번째 행부터 R_2 번째 행까지 비트를 라이트하고, ..., 제C 컬럼의 제2 파트를 구성하는 1 번째 행에서 R_2 번째 행까지 비트를 라이트할 수 있다. 이때, 블록 인터리버(124)는 도 10과 같이 비트들을 각 열의 제2 파트에 열 방향으로 라이트할 수 있다.
- [0321] 즉, 제2 파트에서는 비트 그룹을 구성하는 비트들이 동일한 열에 라이트되지 않고, 복수의 열에 라이트될 수 있다. 즉, 상술한 예에서, 마지막 비트 그룹($Y_{N_{\text{group}}-1}$)은 M 개의 비트들로 구성되므로, 마지막 비트 그룹($Y_{N_{\text{group}}-1}$)에 포함된 비트들은 M/C 개씩 분할되어 각 컬럼에 라이트될 수 있다. 즉, 마지막 비트 그룹($Y_{N_{\text{group}}-1}$)에 포함된 비트들은 M/C 개씩 분할되고, 분할된 M/C 개씩 서브 비트 그룹을 형성하며, 서브 비트 그룹 각각이 제2 파트의 각 열에 라이트될 수 있다.
- [0322] 이에 따라, 제2 파트에 의해 인터리빙되는 적어도 하나의 비트 그룹은 적어도 하나의 비트 그룹에 포함된 비트들이 제2 파트를 구성하는 적어도 두 개의 열에 분할되어 라이트되는 것으로 볼 수 있다.
- [0323] 한편, 상술한 예에서 블록 인터리버(124)는 제2 파트에 열 방향으로 비트들을 라이트하는 것으로 설명하였으나 이는 일 예에 불과하다. 즉, 블록 인터리버(124)는 제2 파트의 복수의 열에 비트들을 행 방향으로 라이트할 수도 있다. 이 경우, 블록 인터리버(124)는 제1 파트에 대해서는 상술한 방식과 동일한 방식으로 비트들을 라이트할 수 있다.
- [0324] 구체적으로, 도 11을 참조하면 블록 인터리버(124)는 제 1 컬럼에서 제2 파트를 구성하는 1 번째 행부터 제C 컬럼에서 제2 파트를 구성하는 1 번째 행까지 비트들을 라이트하고, 제1 컬럼에서 제2 파트를 구성하는 2 번째 행부터 제C 컬럼에서 제2 파트를 구성하는 2 번째 행까지 비트들을 라이트하고, ..., 제1 컬럼에서 제2 파트를 구성하는 R_2 번째 행부터 제C 컬럼에서 제2 파트를 구성하는 R_2 번째 행까지 비트들을 라이트할 수 있다.
- [0325] 한편, 블록 인터리버(124)는 각 파트의 각 행에 라이트된 비트들을 행 방향으로 순차적으로 리드하게 된다. 즉, 블록 인터리버(124)는 도 10 및 도 11과 같이 복수의 열의 제1 파트의 각 행에 라이트된 비트들을 행 방향으로 순차적으로 리드하고, 복수의 열의 제2 파트의 각 행에 라이트된 비트들을 행 방향으로 순차적으로 리드할 수 있다.
- [0326] 이에 따라, 블록 인터리버(124)는 LDPC 부호어를 구성하는 복수의 비트 그룹 중 일부를 비트 그룹 단위로 인터

리빙하고, 나머지 일부는 분할하여 인터리빙을 수행할 수 있다. 즉, 블록 인터리버(124)는 복수의 비트 그룹 중 기설정된 개수의 비트 그룹을 구성하는 LDPC 부호어를 제1 파트를 구성하는 복수의 열에 비트 그룹 단위로 라이트하고, 나머지 비트 그룹을 구성하는 비트들을 모아 제2 파트를 구성하는 열 각각에 분할하여 라이트하고, 제1 및 제2 파트를 구성하는 복수의 열을 행 방향으로 리드하여 인터리빙을 수행할 수 있다.

[0327] 이와 같이, 블록 인터리버(124)는 도 9 내지 도 11에서 설명한 방식을 이용하여 복수의 비트 그룹을 인터리빙할 수 있다.

[0328] 특히, 도 10의 경우 제1 파트에 속하지 않는 비트 그룹에 포함된 비트들은 제2 파트에 열 방향으로 라이트되고 행 방향으로 리드된다는 점에서, 제1 파트에 속하지 않는 비트 그룹에 포함된 비트들의 순서가 재정될 수 있다. 이와 같이, 제1 파트에 속하지 않는 비트 그룹에 포함된 비트들은 인터리빙된다는 점에서 인터리빙이 수행되지 않았을 경우에 비해, BER(bit error rate)/FER(frame error rate) 성능이 향상될 수 있다.

[0329] 다만, 도 11과 같이 제1 파트에 속하지 않는 비트 그룹은 인터리빙되지 않을 수도 있다. 즉, 도 11과 같이 블록 인터리버(124)는 제1 파트에 속하지 않는 비트 그룹에 포함된 비트들을 제2 파트에 행 방향으로 라이트 및 리드한다는 점에서, 제1 파트에 속하지 않는 비트 그룹에 포함된 비트들은 순서가 변경되지 않고 순차적으로 변조부(130)로 출력될 수 있다. 이 경우, 제1 파트에 속하지 않은 비트 그룹에 포함된 비트들은 순차적으로 출력되어 변조 심볼에 맵핑될 수 있다.

[0330] 한편, 도 10 및 도 11에서는 복수의 비트 그룹 중 마지막 하나의 비트 그룹이 제2 파트에 라이트되는 것으로 설명하였으나 이는 일 예에 불과하며, 제2 파트에 라이트되는 그룹의 개수는 LDPC 부호어를 구성하는 그룹의 전체 개수, 열 및 행의 개수, 전송 안테나 개수 등에 따라 다양하게 변경될 수 있음은 물론이다.

[0331] 한편, 블록 인터리버(124)는 하기의 표 28 및 표 29과 같은 구조를 가질 수 있다.

표 28

	$N_{ldpc} = 64800$					
	QPSK	16 QAM	64 QAM	256 QAM	1024 QAM	4096 QAM
C	2	4	6	8	10	12
R_1	32400	16200	10800	7920	6480	5400
R_2	0	0	0	180	0	0

[0332]

표 29

	$N_{ldpc} = 16200$					
	QPSK	16 QAM	64 QAM	256 QAM	1024 QAM	4096 QAM
C	2	4	6	8	10	12
R_1	7920	3960	2520	1800	1440	1080
R_2	180	90	180	225	180	270

[0333]

[0334] 여기에서, C(또는, N_c)는 블록 인터리버(124)를 구성하는 열의 개수이고, R_1 은 각 열에서 제1 파트를 구성하는 행의 개수, R_2 는 각 열에서 제2 파트를 구성하는 행의 개수이다.

[0335] 표 28 및 표 29를 참조하면, 열의 개수는 변조 방식에 따른 변조 차수와 동일한 값을 가지며, 복수의 열 각각은 LDPC 부호어를 구성하는 비트 수를 복수의 열의 개수로 나눈 값만큼의 행으로 구성되는 것을 알 수 있다.

[0336] 예를 들어, LDPC 부호어의 길이 $N_{ldpc}=16200$ 이고, 256-QAM 방식으로 변조를 수행하는 경우, 256-QAM의 경우 변조 차수는 8이므로 블록 인터리버(124)는 8 개의 열로 구성되며, 각 열은 $R_1+R_2=2025(=16200/8)$ 개의 행으로 구성됨을 알 수 있다.

[0337] 한편, 표 28 및 표 29를 참조하면, LDPC 부호어를 구성하는 비트 그룹의 개수가 열의 개수의 정수 배인 경우,

블록 인터리버(124)는 각 열을 구분하지 않고 인터리빙을 수행한다는 점에서 R_1 이 각 열을 구성하는 행의 개수가 되고, $R_2=0$ 이 된다. 또한, LDPC 부호어를 구성하는 그룹의 개수가 열의 개수의 정수 배가 되지 않는 경우, 블록 인터리버(124)는 각 열을 R_1 개의 행으로 구성되는 제1 파트와 R_2 개의 행으로 구성되는 제2 파트로 구분하여 인터리빙을 수행한다.

[0338] 한편, 표 28 및 표 29와 같이, 블록 인터리버(124)의 열의 개수가 변조 심볼을 구성하는 비트의 수와 동일한 경우, 동일한 비트 그룹에 포함된 비트들은 변조 심볼에서 하나의 비트에 맵핑될 수 있다.

[0339] 예를 들어, $N_{ldpc}=16200$ 이고, 변조 방식이 256-QAM인 경우, 블록 인터리버(124)는 각각 2025 개의 행을 포함하는 8 개의 열로 구성될 수 있다. 이 경우, 복수의 비트 그룹 각각에 포함된 비트들은 8 개의 열에 라이트되고, 각 열에서 동일한 행에 라이트되었던 비트들이 순차적으로 출력된다. 이때, 변조 방식이 256-QAM인 경우 8 비트가 하나의 변조 심볼을 구성한다는 점에서, 동일한 비트 그룹에 포함된 비트들 즉, 하나의 열에서 출력되는 비트들은 변조 심볼에서 하나의 비트에 맵핑될 수 있다. 예를 들어, 제1 컬럼에 라이트되는 그룹에 포함된 비트들은 각 변조 심볼의 첫 번째 비트에 맵핑될 수 있다.

[0340] 한편, 표 28 및 표 29를 참조하면, 블록 인터리버(124)의 전체 행(row)의 개수 즉, R_1+R_2 는 N_{ldpc}/C 임을 알 수 있다.

[0341] 그리고, 제1 파트의 행의 개수인 R_1 은 각 그룹에 포함된 비트들의 개수인 M (예를 들어, $M=360$)의 정수 배로 $\lfloor N_{group}/C \rfloor \times M$ 이고, 제2 파트의 행의 개수인 R_2 는 $N_{ldpc}/C - R_1$ 이 될 수 있다. 여기에서, $\lfloor N_{group}/C \rfloor$ 는 N_{group}/C 이하의 가장 큰 정수를 나타낸다. 이와 같이, R_1 은 각 비트 그룹에 포함된 비트의 개수인 M 의 정수 배가 된다는 점에서, R_1 에는 비트 그룹 단위의 비트들이 라이트될 수 있다.

[0342] 또한, 표 28 및 표 29를 참조하면, LDPC 부호어를 구성하는 비트 그룹의 개수가 열의 개수의 배수가 아닌 경우, 블록 인터리버(124)는 각 열을 2 개의 파트로 구분하여 인터리빙을 수행하는 것을 알 수 있다.

[0343] 구체적으로, LDPC 부호어의 길이를 열의 개수로 나눈 값이 각 열에 포함된 전체 행의 개수가 된다. 이때, LDPC 부호어를 구성하는 비트 그룹의 개수가 열의 개수의 배수인 경우, 각 열은 2 개의 파트로 구분되지 않는다. 다만, LDPC 부호어를 구성하는 비트 그룹의 개수가 열의 개수의 배수가 되지 않는 경우, 각 열은 2 개의 파트로 구분될 수 있다.

[0344] 예를 들어, 표 28과 같이 블록 인터리버(124)의 열의 개수가 변조 심볼을 구성하는 비트의 개수와 동일하고, LDPC 부호어가 64800 개의 비트로 구성되는 경우를 가정한다. 이때, LDPC 부호어를 구성하는 각 비트 그룹은 360 개의 비트로 구성되며, LDPC 부호어는 $64800/360=180$ 개의 비트 그룹으로 구성된다.

[0345] 한편, 변조 방식이 16-QAM인 경우, 블록 인터리버(124)는 4 개의 열로 구성되며, 각 열은 $64800/4=16200$ 개의 행으로 구성될 수 있다.

[0346] 이때, LDPC 부호어를 구성하는 비트 그룹의 개수를 열의 개수로 나눈 값은 $180/4=45$ 가 되므로, 각 열을 2 개의 파트로 구분하지 않아도 각 열에 비트 그룹 단위로 비트들이 라이트될 수 있다. 즉, 각 열에는 LDPC 부호어를 구성하는 비트 그룹의 개수를 열의 개수로 나눈 몫인 45 개의 비트 그룹에 포함된 비트들 즉, $45 \times 360=16200$ 개의 비트들이 라이트될 수 있다.

[0347] 다만, 변조 방식이 256-QAM인 경우, 블록 인터리버(124)는 8 개의 열로 구성되며, 각 열은 $64800/8=8100$ 개의 행으로 구성될 수 있다.

[0348] 이때, LDPC 부호어를 구성하는 비트 그룹의 개수를 열의 개수로 나눈 값은 $180/8=22.5$ 가 되므로, LDPC 부호어를 구성하는 비트 그룹의 개수는 열의 개수의 정수 배가 되지 않는다. 이에 따라, 블록 인터리버(124)는 비트 그룹 단위로 인터리빙을 수행하기 위해, 8 개의 열 각각을 2 개의 파트로 구분하게 된다.

[0349] 이때, 각 열의 제1 파트에는 비트 그룹 단위로 비트들이 라이트되어야 하므로, 각 열의 제1 파트에 비트 그룹 단위로 라이트될 수 있는 비트 그룹의 개수는 LDPC 부호어를 구성하는 그룹의 개수를 열의 개수로 나눈 몫인 22 개가 되며, 이에 따라, 각 열의 제1 파트는 $22 \times 360=7920$ 개의 행으로 구성될 수 있다. 이에 따라, 각 열의 제1 파트에는 22 개의 비트 그룹에 포함된 7920 개의 비트들이 라이트될 수 있다.

- [0350] 한편, 각 열의 제2 파트는 각 열의 전체 행에서 제1 파트를 구성하는 행을 제외한 행으로 구성된다. 따라서, 각 열의 제2 파트는 $8100-7920=180$ 개의 행으로 구성될 수 있다.
- [0351] 이때, 각 열의 제2 파트에는 제1 파트에 라이트되지 못한 나머지 비트 그룹에 포함되는 비트들이 분할되어 라이트될 수 있다.
- [0352] 구체적으로, 제1 파트에는 $22 \times 8=176$ 개의 비트 그룹이 라이트되므로, 제2 파트에 라이트되는 그룹의 개수는 $180-176=4$ 개이다(가령, LDPC 부호어를 구성하는 비트 그룹(Y_0), 비트 그룹(Y_1), 비트 그룹(Y_2), ..., 비트 그룹(Y_{178}), 비트 그룹(Y_{179}) 중 비트 그룹(Y_{176}), 비트 그룹(Y_{177}), 비트 그룹(Y_{178}), 비트 그룹(Y_{179})가 될 수 있다).
- [0353] 이에 따라, 블록 인터리버(124)는 LDPC 부호어를 구성하는 그룹 중 제1 파트에 라이트되고 남은 4 개의 비트 그룹을 각 열의 제2 파트에 순차적으로 라이트할 수 있다.
- [0354] 즉, 블록 인터리버(124)는 비트 그룹(Y_{176})에 포함된 360 개의 비트 중 180 개의 비트를 제1 열의 제2 파트의 1 번째 행부터 180 번째 행까지 열 방향으로 라이트하고, 남은 180 개의 비트를 제2 열의 제2 파트의 1 번째 행부터 180 번째 행까지 열 방향으로 라이트할 수 있다. 그리고, 블록 인터리버(124)는 비트 그룹(Y_{177})에 포함된 360 개의 비트 중 180 개의 비트를 제3 열의 제2 파트의 1 번째 행부터 180 번째 행까지 열 방향으로 라이트하고, 남은 180 개의 비트를 제4 열의 제2 파트의 1 번째 행부터 180 번째 행까지 열 방향으로 라이트할 수 있다. 그리고, 블록 인터리버(124)는 비트 그룹(Y_{178})에 포함된 360 개의 비트 중 180 개의 비트를 제5 열의 제2 파트의 1 번째 행부터 180 번째 행까지 열 방향으로 라이트하고, 남은 180 개의 비트를 제6 열의 제2 파트의 1 번째 행부터 180 번째 행까지 열 방향으로 라이트할 수 있다. 그리고, 블록 인터리버(124)는 비트 그룹(Y_{179})에 포함된 360 개의 비트 중 180 개의 비트를 제7 열의 제2 파트의 1 번째 행부터 180 번째 행까지 열 방향으로 라이트하고, 남은 180 개의 비트를 제8 열의 제2 파트의 1 번째 행부터 180 번째 행까지 열 방향으로 라이트할 수 있다.
- [0355] 이에 따라, 제1 파트에 라이트되고 남은 비트 그룹에 포함된 비트들은 제2 파트에서 동일한 열에 라이트되지 않고, 복수의 열에 나누어져 라이트될 수 있다.
- [0356] 이하에서는 도 12를 참조하여 도 5의 블록 인터리버(124)의 구체적인 일 예에 대해 보다 구체적으로 설명한다.
- [0357] 그룹 인터리빙된 LDPC 부호어 ($v_0, v_1, \dots, v_{N_{ldpc}-1}$)는 $V=\{Y_0, Y_1, \dots, Y_{N_{group}-1}\}$ 과 같이 Y_j 가 연속적으로 배치될 수 있다.
- [0358] 그룹 인터리빙 이후, LDPC 부호어는 도 12와 같은 블록 인터리버(124)에 의해 인터리빙될 수 있다. 이 경우, 블록 인터리버(124)는 블록 인터리버(124)를 구성하는 열의 개수 및 비트 그룹의 비트 수에 기초하여 복수의 열을 제1 파트와 제2 파트로 구분할 수 있다. 이 경우, 제1 파트에서는 비트 그룹을 구성하는 비트들이 동일한 열에 라이트되고, 제2 파트에서는 비트 그룹을 구성하는 비트들이 복수의 열에 라이트될 수 있다.
- [0359] 구체적으로, 입력 비트 v_i 는 제1 파트부터 시작하여 제2 파트까지 순차적(serially)으로 열 방향(column wise)으로 라이트되고, 제1 파트부터 제2 파트까지 순차적으로 행 방향(row wise)으로 리드된다. 이에 따라, 제1 파트에서 동일한 비트 그룹에 포함된 비트들은 각 변조 심볼에서 하나의 비트에 맵핑될 수 있다.
- [0360] 이 경우, 변조 방식 및 LDPC 부호어의 길이(즉, code length)에 따른 블록 인터리버(124)의 제1 파트와 제2 파트의 열의 개수와 행의 개수는 하기의 표 30과 같을 수 있다. 여기에서, 블록 인터리버(124)의 열의 개수는 변조 심볼을 구성하는 비트의 수와 같을 수 있다. 그리고, 제1 파트의 행의 개수인 N_{r1} 과 제2 파트의 행의 개수인 N_{r2} 의 합은 N_{ldpc}/N_c 와 동일하다(여기에서, N_c 는 열의 개수). 그리고, $N_{r1}(=\lfloor N_{group}/N_c \rfloor \times 360)$ 은 360의 배수이므로, 복수의 비트 그룹이 제1 파트에 라이트될 수 있다.

표 30

Modulation	Rows in Part 1 N_{r1}		Rows in Part 2 N_{r2}		Columns N_c
	$N_{ldpc} = 64800$	$N_{ldpc} = 16200$	$N_{ldpc} = 64800$	$N_{ldpc} = 16200$	
QPSK	32400	7920	0	180	2
16-QAM	16200	3960	0	90	4
64-QAM	10800	2520	0	180	6
256-QAM	7920	1800	180	225	8
1024-QAM	6480	1440	0	180	10
4096-QAM	5400	1080	0	270	12

[0361]

[0362]

[0363]

이하에서는 블록 인터리버(124)의 동작에 대해 보다 구체적으로 설명하도록 한다.

구체적으로, 도 12와 같이 입력 비트 $v_i (0 \leq i < N_c \times N_{r1})$ 는 블록 인터리버(124)의 제1 파트의 c_i 열의 r_i 행에 라

이트된다. 여기에서, c_i 와 r_i 는 각각 $c_i = \left\lfloor \frac{i}{N_{r1}} \right\rfloor$, $r_i = (i \bmod N_{r1})$ 와 같다.

[0364]

그리고, 입력 비트 $v_i (N_c \times N_{r1} \leq i < N_{ldpc})$ 는 블록 인터리버(124)의 제2 파트의 c_i 열의 r_i 행에 라이트된다. 여기

에서, c_i 와 r_i 는 각각 $c_i = \left\lfloor \frac{(i - N_c \times N_{r1})}{N_{r2}} \right\rfloor$, $r_i = N_{r1} + \{(i - N_c \times N_{r1}) \bmod N_{r2}\}$ 와 같다.

[0365]

한편, 출력 비트 $q_j (0 \leq j < N_{ldpc})$ 는 r_j 행의 c_j 열에서 리드된다. 여기에서, c_j 와 r_j 는 각각 $r_j = \left\lfloor \frac{j}{N_c} \right\rfloor$, $c_j = (j \bmod N_c)$ 와 같다.

[0366]

예를 들어, LDPC 부호어의 길이 N_{ldpc} 가 64800이고, 변조 방식이 256-QAM인 경우, 블록 인터리버(124)에서 출력되는 비트의 순서는

$(q_0, q_1, q_2, \dots, q_{63357}, q_{63358}, q_{63359}, q_{63360}, q_{63361}, \dots, q_{64799}) = (v_0, v_{7920}, v_{15840}, \dots, v_{47519}, v_{55439}, v_{63359}, v_{63360}, v_{63540}, \dots, v_{64799})$

와 같을 수 있다. 여기에서, 우측 항에서의 인덱스를 모든 8 개의 열에 대해 보다 자세히 나타내면 0, 7920, 15840, 23760, 31680, 39600, 47520, 55440, 1, 7921, 15841, 23761, 31681, 39601, 47521, 55441, $\dots$, 7919, 15839, 23759, 31679, 39599, 47519, 55439, 63359, 63360, 63540, 63720, 63900, 64080, 64260, 64440, 64620, $\dots$, 63539, 63719, 63899, 64079, 64259, 64439, 64619, 64799와 같다.

[0367]

이하에서는 구체적인 예를 들어 블록 인터리버(124)의 인터리빙 동작을 설명하도록 한다.

[0368]

블록 인터리버(124)는 복수의 비트 그룹을 비트 그룹 단위로 열 각각에 열 방향으로 라이트하고, 복수의 비트 그룹이 비트 그룹 단위로 라이트된 복수의 열의 각 행을 행 방향으로 리드하여 인터리빙을 수행할 수 있다.

[0369]

이 경우, 블록 인터리버(124)를 구성하는 열의 개수는 변조 방식에 따라 달라질 수 있으며, 행의 개수는 LDPC 부호어의 길이/열의 개수가 될 수 있다. 예를 들어, 변조 방식이 256-QAM인 경우 블록 인터리버(124)는 8 개의 열로 구성될 수 있다. 이때, LDPC 부호어의 길이 N_{ldpc} 가 16200인 경우 행의 개수는 $16200/8=2025$ 가 될 수 있다.

[0370]

이하에서는 블록 인터리버(124)가 복수의 비트 그룹을 비트 그룹 단위로 인터리빙하는 방법에 대해 구체적으로 설명하도록 한다.

[0371]

블록 인터리버(124)는 LDPC 부호어를 구성하는 비트 그룹의 개수가 열의 개수의 정수 배인 경우, 비트 그룹의 개수를 열의 개수로 나눈 값만큼의 비트 그룹을 각 열에 순차적으로 비트 그룹 단위로 라이트하여 인터리빙을 수행할 수 있다.

- [0372] 예를 들어, 변조 방식이 256-QAM이고 LDPC 부호어의 길이 N_{ldpc} 가 16200인 경우, 블록 인터리버(124)는 각각 2025 개의 행을 포함하는 8 개의 열로 구성될 수 있다. 여기에서, LDPC 부호어의 길이 N_{ldpc} 가 16200인 경우 LDPC 부호어는 $16200/360=45$ 개의 비트 그룹으로 구분되므로, 변조 방식이 256-QAM인 경우 LDPC 부호어를 구성하는 비트 그룹의 개수(=45)는 열의 개수(=8)의 정수 배가 되지 않게 된다. 즉, LDPC 부호어를 구성하는 비트 그룹의 개수를 열의 개수로 나눈 경우 나머지가 발생하게 된다.
- [0373] 이와 같이, LDPC 부호어를 구성하는 비트 그룹의 개수가 블록 인터리버(124)를 구성하는 열의 개수의 정수 배가 되지 않는 경우, 블록 인터리버(124)는 각 열을 N 개(N 은 2 이상의 정수)의 파트로 구분하여 인터리빙을 수행할 수 있다.
- [0374] 구체적으로, 블록 인터리버(124)는 각 열에 비트 그룹 단위로 라이트 가능한 비트 그룹에 포함된 비트 수만큼의 행을 포함하는 파트(즉, 제1 파트)와 그 외의 나머지 행을 포함하는 파트(즉, 제2 파트)로 각 열을 구분하고, 구분된 파트 각각을 이용하여 인터리빙을 수행할 수 있다.
- [0375] 여기에서, 비트 그룹 단위로 라이트 가능한 비트 그룹에 포함된 비트 수만큼의 행을 포함하는 파트 즉, 제1 파트는 M 의 정수 배만큼의 행으로 구성될 수 있다. 즉, 변조 방식이 256-QAM인 경우 블록 인터리버(124)의 각 열은 2025 개의 행으로 구성된다는 점에서, 블록 인터리버(124)의 각 열은 $1800(=360 \times 5)$ 개의 행을 포함하는 제1 파트와 $225(=2025-1800)$ 개의 행을 포함하는 제2 파트로 구성될 수 있다.
- [0376] 이 경우, 블록 인터리버(124)는 복수의 비트 그룹 중에서 복수의 열 각각에 비트 그룹 단위로 라이트 가능한 적어도 일부의 비트 그룹을 복수의 열 각각에 순차적으로 라이트한 후, 복수의 열 각각에서 적어도 일부의 비트 그룹이 비트 그룹 단위로 라이트되고 남은 나머지 영역에 나머지 비트 그룹을 분할하여 라이트할 수 있다. 즉, 블록 인터리버(124)는 라이트 가능한 적어도 일부의 비트 그룹에 포함된 비트들을 비트 그룹 단위로 각 열의 제1 파트에 라이트하고, 나머지 비트 그룹에 포함된 비트들을 분할하여 각 열의 제2 파트에 라이트할 수 있다.
- [0377] 예를 들어, 변조 방식이 256-QAM인 경우, 도 13 및 도 14와 같이 블록 인터리버(124)는 8 개의 열을 포함하며, 각 열은 1800 개의 행을 포함하는 제1 파트와 225 개의 행을 포함하는 제2 파트로 구분될 수 있다.
- [0378] 이 경우, 블록 인터리버(124)는 각 열에 그룹 단위로 라이트 가능한 비트 그룹에 포함된 비트들을 열 방향으로 각 열의 제1 파트에 라이트 한다.
- [0379] 즉, 블록 인터리버(124)는 도 13 및 도 14와 같이 제1 열의 제1 파트를 구성하는 1 번째 행부터 1800 번째 행에 비트 그룹(Y_0), 비트 그룹(Y_1), ..., 비트 그룹(Y_4) 각각에 포함된 비트들을 라이트하고, 제2 열의 제1 파트를 구성하는 1 번째 행부터 1800 번째 행에 비트 그룹(Y_5), 비트 그룹(Y_6), ..., 비트 그룹(Y_9) 각각에 포함된 비트들을 라이트하고, ..., 제8 열의 제1 파트를 구성하는 1 번째 행부터 1800 번째 행에 비트 그룹(Y_{35}), 비트 그룹(Y_{36}), ..., 비트 그룹(Y_{39}) 각각에 포함된 비트들을 라이트할 수 있다.
- [0380] 이와 같이, 블록 인터리버(124)는 각 열에 그룹 단위로 라이트 가능한 비트 그룹에 포함된 비트들을 각 열의 제1 파트에 비트 그룹 단위로 라이트한다.
- [0381] 이후, 블록 인터리버(124)는 복수의 그룹 중에서 각 열의 제1 파트에 라이트된 그룹을 제외한 나머지 비트 그룹에 포함된 비트들을 분할하여 각 열의 제2 파트에 열 방향으로 라이트할 수 있다. 이때, 블록 인터리버(124)는 동일한 개수의 비트가 각 열의 제2 파트에 라이트되도록, 각 열의 제1 파트에 라이트된 비트 그룹을 제외한 나머지 비트 그룹에 포함된 비트들을 열의 개수로 분할하고, 분할된 각 비트를 제2 파트의 각 열에 열 방향으로 라이트할 수 있다.
- [0382] 예를 들어, 도 13과 같이, 블록 인터리버(124)는 LDPC 부호어를 구성하는 비트 그룹 중 제1 파트에 라이트되고 남은 비트 그룹인 비트 그룹(Y_{40}), 비트 그룹(Y_{41}), 비트 그룹(Y_{42}), 비트 그룹(Y_{43}), 비트 그룹(Y_{44})을 각 열의 제2 파트에 순차적으로 라이트할 수 있다.
- [0383] 즉, 블록 인터리버(124)는 비트 그룹(Y_{40})에 포함된 360 개의 비트 중 225 개의 비트를 제1 열의 제2 파트의 1 번째 행부터 225 번째 행까지 열 방향으로 라이트하고, 남은 135 개의 비트를 제2 열의 제2 파트의 1 번째 행부터 135 번째 행까지 열 방향으로 라이트할 수 있다. 그리고, 블록 인터리버(124)는 비트 그룹(Y_{41})에 포함된 360 개의 비트 중 90 개의 비트를 제2 열의 제2 파트의 136 번째 행부터 225 번째 행까지 열 방향으로 라이트하고, 남은 270 개의 비트 중 225 개의 비트를 제3 열의 제2 파트의 1 번째 행부터 225 번째 행까지 열 방향으로 라이

트하고 45 개의 비트를 제4 열의 제2 파트의 1 번째 행부터 45 번째 행까지 열 방향으로 라이트할 수 있다. 그리고, 블록 인터리버(124)는 비트 그룹(Y_{42})에 포함된 360 개의 비트 중 180 개의 비트를 제4 열의 제2 파트의 46 번째 행부터 225 번째 행까지 열 방향으로 라이트하고, 남은 180 개의 비트를 제5 열의 제2 파트의 1 번째 행부터 180 번째 행까지 열 방향으로 라이트할 수 있다. 그리고, 블록 인터리버(124)는 비트 그룹(Y_{43})에 포함된 360 개의 비트 중 45 개의 비트를 제5 열의 제2 파트의 181 번째 행부터 225 번째 행까지 열 방향으로 라이트하고, 남은 315 개의 비트 중 225 개의 비트를 제6 열의 제2 파트의 1 번째 행부터 225 번째 행까지 열 방향으로 라이트하고 90 개의 비트를 제7 열의 제2 파트의 1 번째 행부터 90 번째 행까지 열 방향으로 라이트할 수 있다. 그리고, 블록 인터리버(124)는 비트 그룹(Y_{44})에 포함된 360 개의 비트 중 135 개의 비트를 제7 열의 제2 파트의 91 번째 행부터 225 번째 행까지 열 방향으로 라이트하고, 남은 225 개의 비트를 제8 열의 제2 파트의 1 번째 행부터 225 번째 행까지 열 방향으로 라이트할 수 있다.

- [0384] 이에 따라, 제1 파트에 라이트되고 남은 비트 그룹에 포함된 비트들은 제2 파트에서 동일한 열에 라이트되지 않고, 복수의 열에 나누어져 라이트될 수 있다.
- [0385] 한편, 상술한 예에서 블록 인터리버(124)는 제2 파트에 열 방향으로 비트들을 라이트하는 것으로 설명하였으나 이는 일 예에 불과하다. 즉, 블록 인터리버(124)는 제2 파트의 복수의 열에 비트들을 행 방향으로 라이트할 수도 있다. 다만, 블록 인터리버(124)는 제1 파트에 대해서는 상술한 방식과 동일한 방식으로 비트들을 라이트할 수 있다.
- [0386] 구체적으로, 도 14를 참조하면 블록 인터리버(124)는 제1 열의 제2 파트의 1 번째 행부터 제8 열의 제2 파트의 1 번째 행까지 비트들을 라이트하고, 제1 열의 제2 파트의 2 번째 행부터 제6 열의 제2 파트의 2 번째 행까지 비트들을 라이트하고, ..., 제1 열의 제2 파트의 180 번째 행부터 제6 열의 제2 파트의 180 번째 행까지 비트들을 라이트할 수 있다.
- [0387] 이에 따라, 비트 그룹(Y_{40})에 포함된 비트들은 제1 열의 제2 파트의 1 번째 행부터 제8 열의 제2 파트의 45 번째 행까지 순차적으로 라이트될 수 있고, 비트 그룹(Y_{41})에 포함된 비트들은 제1 열의 제2 파트의 46 번째 행부터 제8 열의 제2 파트의 90 번째 행까지 순차적으로 라이트될 수 있고, 비트 그룹(Y_{42})에 포함된 비트들은 제1 열의 제2 파트의 91 번째 행부터 제8 열의 제2 파트의 135 번째 행까지 순차적으로 라이트될 수 있고, 비트 그룹(Y_{43})에 포함된 비트들은 제1 열의 제2 파트의 136 번째 행부터 제8 열의 제2 파트의 180 번째 행까지 순차적으로 라이트될 수 있고, 비트 그룹(Y_{44})에 포함된 비트들은 제1 열의 제2 파트의 181 번째 행부터 제8 열의 제2 파트의 225 번째 행까지 순차적으로 라이트될 수 있다.
- [0388] 한편, 블록 인터리버(124)는 각 파트의 각 행에 라이트된 비트들을 행 방향으로 순차적으로 리드하게 된다. 즉, 블록 인터리버(124)는 도 13 및 도 14와 같이 복수의 열의 제1 파트의 각 행에 라이트된 비트들을 행 방향으로 순차적으로 리드하고, 복수의 열의 제2 파트의 각 행에 라이트된 비트들을 행 방향으로 순차적으로 리드할 수 있다.
- [0389] 이와 같이, 블록 인터리버(124)는 도 9 내지 도 14에서 설명한 방식을 이용하여 LDPC 부호어를 구성하는 복수의 비트 그룹을 인터리빙할 수 있다.
- [0390] 변조부(130)는 인터리빙된 LDPC 부호어를 변조 심볼에 맵핑한다. 구체적으로, 변조부(130)는 인터리빙된 LDPC 부호어를 디멀티플렉싱하고, 디멀티플렉싱된 LDPC 부호어를 변조하여 성상도에 맵핑할 수 있다.
- [0391] 이 경우, 변조부(130)는 복수의 비트 그룹 각각에 포함된 비트들을 이용하여 변조 심볼을 생성할 수 있다.
- [0392] 즉, 상술한 바와 같이, 블록 인터리버(124)의 각 열에는 서로 다른 비트 그룹에 포함된 비트들이 라이트되고, 블록 인터리버(124)는 각 열에 라이트된 비트들을 행 방향으로 리드하게 된다. 이 경우, 변조부(130)는 블록 인터리버(124)의 각 열에서 리드된 비트들을 변조 심볼의 각 비트에 맵핑시켜 변조 심볼을 생성하게 된다. 이에 따라, 변조 심볼의 각 비트는 서로 다른 그룹에 포함된 비트가 될 수 있다.
- [0393] 예를 들어, 변조 심볼이 C 개의 비트로 구성된 경우를 가정한다. 이 경우, 블록 인터리버(124)의 C 개의 열의 각 행에서 리드된 비트들이 변조 심볼의 각 비트에 맵핑될 수 있다는 점에서, 결국, C 개의 비트로 구성된 변조 심볼의 각 비트들은 C 개의 서로 다른 그룹에 포함된 비트들이 될 수 있다.

- [0394] 이하에서는 이를 보다 구체적으로 설명하도록 한다.
- [0395] 먼저, 변조부(130)는 인터리빙된 LDPC 부호어를 디멀티플렉싱할 수 있다. 이를 위해, 변조부(130)는 인터리빙된 LDPC 부호어를 디멀티플렉싱하기 위한 디멀티플렉서(미도시)를 포함할 수 있다.
- [0396] 디멀티플렉서(미도시)는 인터리빙된 LDPC 부호어를 디멀티플렉싱한다. 구체적으로, 디멀티플렉서(미도시)는 인터리빙된 LDPC 부호어에 대해 시리얼-투-패러럴(serial-to-parallel) 변환을 수행하여, 인터리빙된 LDPC 부호어를 일정한 개수의 비트를 갖는 셀(cell)(또는, 데이터 셀(data dcell))로 디멀티플렉싱할 수 있다.
- [0397] 예를 들어, 도 15와 같이 디멀티플렉서(미도시)는 인터리버(120)에서 출력되는 LDPC 부호어 $Q=(q_0, q_1, q_2, \dots)$ 를 입력받고, 입력된 LDPC 부호어 비트들을 순차적으로 복수의 서브 스트림 각각에 출력하여 입력된 LDPC 부호어 비트들을 셀로 변환하여 출력할 수 있다.
- [0398] 이 경우, 복수의 서브 스트림 각각에서 동일한 인덱스를 갖는 비트들이 동일한 셀을 구성할 수 있다. 이에 따라, 각 셀들은 $(y_{0,0}, y_{1,0}, \dots, y_{n_{MOD}-1,0})=(q_0, q_1, \dots, q_{n_{MOD}-1})$, $(y_{0,1}, y_{1,1}, \dots, y_{n_{MOD}-1,1})=(q_{n_{MOD}}, q_{n_{MOD}+1}, \dots, q_{2 \times n_{MOD}-1}), \dots$ 와 같이 구성될 수 있다.
- [0399] 한편, 서브 스트림의 개수 $N_{substreams}$ 는 변조 심볼을 구성하는 비트의 개수 n_{MOD} 와 동일하다. 이에 따라, 각 셀을 구성하는 비트의 개수는 변조 심볼을 구성하는 비트의 개수(즉, 변조 차수)와 동일할 수 있다.
- [0400] 예를 들어, 변조 방식이 256-QAM인 경우 변조 심볼을 구성하는 비트의 개수 $n_{MOD}=8$ 이므로, 서브 스트림의 개수 $N_{substreams}=8$ 가 될 수 있고, 각 셀들은 $(y_{0,0}, y_{1,0}, y_{2,0}, y_{3,0}, y_{4,0}, y_{5,0}, y_{6,0}, y_{7,0})=(q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_7)$, $(y_{0,1}, y_{1,1}, y_{2,1}, y_{3,1}, y_{4,1}, y_{5,1}, y_{6,1}, y_{7,1})=(q_8, q_9, q_{10}, q_{11}, q_{12}, q_{13}, q_{14}, q_{15})$, $(y_{0,2}, y_{1,2}, y_{2,2}, y_{3,2}, y_{4,2}, y_{5,2}, y_{6,2}, y_{7,2})=(q_{16}, q_{17}, q_{18}, q_{19}, q_{20}, q_{21}, q_{22}, q_{23}), \dots$ 와 같이 구성될 수 있다.
- [0401] 한편, 변조부(130)는 디멀티플렉싱된 LDPC 부호어를 변조 심볼에 맵핑할 수 있다.
- [0402] 구체적으로, 변조부(130)는 디멀티플렉서(미도시)에서 출력되는 비트들(즉, 셀들)을 256-QAM 등과 같은 다양한 변조 방식을 이용하여 변조할 수 있다. 예를 들어, 변조부(130)는 변조 방식이 QPSK, 16-QAM, 64-QAM, 256-QAM, 1024-QAM, 4096-QAM인 경우 변조 심볼을 구성하는 비트의 개수 n_{MOD} (즉, 변조 차수)는 각각 2, 4, 6, 8, 10, 12 개가 될 수 있다.
- [0403] 이 경우, 디멀티플렉서(미도시)에서 출력되는 각 셀은 변조 심볼을 구성하는 개수만큼의 비트로 이루어진다는 점에서, 변조부(130)는 디멀티플렉서(미도시)에서 출력되는 셀 각각을 순차적으로 성상점(constellation point)에 맵핑하여 변조 심볼을 생성할 수 있다. 여기에서, 변조 심볼은 성상도(constellation)에서의 성상점에 대응된다.
- [0404] 다만, 경우에 따라, 상술한 디멀티플렉서(미도시)는 생략될 수 있다. 이 경우, 변조부(130)는 인터리빙된 비트들을 순차적으로 일정한 개수만큼 구분하고, 이를 성상점에 맵핑하여 변조 심볼을 생성할 수 있다. 이 경우, 변조부(130)는 인터리빙된 비트들을 변조 방식에 따라 n_{MOD} 개 비트씩 순차적으로 성상점에 맵핑하여 변조 심볼을 생성할 수 있다.
- [0405] 한편, 변조부(130)는 non-uniform constellation(NUC) 방식을 이용하여 디멀티플렉서(미도시)에서 출력되는 셀들을 성상점에 맵핑하여 변조를 수행할 수 있다.
- [0406] non uniform constellation 방식에서는 1 사분면의 성상점을 정의하면 다른 3 개의 사분면에 대한 성상점은 하기와 같은 방법에 의해 결정될 수 있다. 예를 들어, 1 사분면에 대해 정의한 성상점들의 집합을 X라고 할 때 2 사분면의 경우에는 $-\text{conj}(X)$, 3 사분면의 경우에는 $\text{conj}(X)$, 4사분면의 경우는 $-(X)$ 가 된다.
- [0407] 즉, 1 사분면을 정의하면 다른 사분면은 하기와 같이 표현될 수 있다.
- [0408] 1 Quarter(1 사분면) = X
- [0409] 2 Quarter(2 사분면) = $-\text{conj}(X)$
- [0410] 3 Quarter(3 사분면) = $\text{conj}(X)$

- [0411] 4 Quarter(4 사분면) = -X
- [0412] 구체적으로, non-uniform M-QAM을 사용한다고 할 경우, M개의 성상점을 $z=\{z_0, z_1, \dots, z_{M-1}\}$ 으로 정의할 수 있다. 이때, 1 사분면에 존재하는 성상점을 $\{x_0, x_1, x_2, \dots, x_{M/4-1}\}$ 로 정의할 때, z는 하기와 같이 정의할 수 있다.
- [0413] z_0 부터 $z_{M/4-1}$ = x_0 부터 $x_{M/4}$
- [0414] $z_{M/4}$ 부터 $z_{2 \times M/4-1}$ = $-\text{conj}(x_0$ 부터 $x_{M/4})$
- [0415] $z_{2 \times M/4}$ 부터 $z_{3 \times M/4-1}$ = $\text{conj}(x_0$ 부터 $x_{M/4})$
- [0416] $z_{3 \times M/4}$ 부터 $z_{4 \times M/4-1}$ = $-(x_0$ 부터 $x_{M/4})$
- [0417] 따라서, 변조부(130)는 디멀티플렉서(미도시)에서 출력되는 출력 비트인 $[y_0, \dots, y_{m-1}]$ 을 $L = \sum_{i=0}^{m-1} (y_i x 2^{m-1})$ 을 인 텍스로 하는 z_L 로 매핑하여, 출력 비트들을 non-uniform constellation 방식으로 성상점에 맵핑할 수 있다.
- [0418] 상술한 non-uniform constellation 방식에 따라 정의되는 성상도의 일 예는 부호율이 5/15, 7/15, 9/15, 11/15, 13/15인 경우 하기의 표 31과 같이 나타낼 수 있다.

표 31

	CR 5/15	CR 7/15	CR 9/15	CR 13/15
Label (int.)	Constellation	Constellation	Constellation	Constellation
0	0.1524 + 0.3087i	0.1170 + 0.3003i	0.0899 + 0.1337i	1.2412 + 1.0688i
1	0.1525 + 0.3087i	0.1171 + 0.3003i	0.0910 + 0.1377i	1.2668 + 0.8034i
2	0.1513 + 0.3043i	0.1204 + 0.3233i	0.0873 + 0.3862i	0.9860 + 1.1758i
3	0.1513 + 0.3043i	0.1204 + 0.3233i	0.0883 + 0.3873i	1.0365 + 0.9065i
4	0.1682 + 0.3004i	0.1454 + 0.2877i	0.1115 + 0.1442i	1.2111 + 0.5135i
5	0.1682 + 0.3005i	0.1453 + 0.2877i	0.1135 + 0.1472i	1.4187 + 0.6066i
6	0.1663 + 0.2964i	0.1566 + 0.3074i	0.2067 + 0.3591i	1.0103 + 0.4879i
7	0.1663 + 0.2964i	0.1565 + 0.3074i	0.1975 + 0.3621i	1.0380 + 0.6906i
8	0.1964 + 0.6584i	0.1427 + 0.6856i	0.1048 + 0.7533i	0.6963 + 1.3442i
9	0.1965 + 0.6583i	0.1562 + 0.6826i	0.1770 + 0.7412i	0.7089 + 1.1122i
10	0.1967 + 0.6652i	0.1422 + 0.6584i	0.1022 + 0.5904i	0.1256 + 1.4745i
11	0.1968 + 0.6652i	0.1529 + 0.6560i	0.1191 + 0.5890i	0.8331 + 0.9455i
12	0.3371 + 0.5987i	0.3840 + 0.5856i	0.4264 + 0.6230i	0.6615 + 0.6012i
13	0.3370 + 0.5987i	0.3723 + 0.5931i	0.3650 + 0.6689i	0.6894 + 0.7594i
14	0.3414 + 0.6039i	0.3651 + 0.5660i	0.3254 + 0.5153i	0.8373 + 0.5633i
15	0.3413 + 0.6039i	0.3559 + 0.5718i	0.2959 + 0.5302i	0.8552 + 0.7410i
16	0.3087 + 0.1524i	0.3003 + 0.1170i	0.3256 + 0.0768i	1.2666 + 0.1027i
17	0.3087 + 0.1525i	0.3003 + 0.1171i	0.3266 + 0.0870i	1.4915 + 0.1198i
18	0.3043 + 0.1513i	0.3233 + 0.1204i	0.4721 + 0.0994i	1.0766 + 0.0945i
19	0.3043 + 0.1513i	0.3233 + 0.1204i	0.4721 + 0.1206i	0.9007 + 0.0848i
20	0.3004 + 0.1682i	0.2877 + 0.1454i	0.2927 + 0.1267i	1.2454 + 0.3064i
21	0.3005 + 0.1682i	0.2877 + 0.1453i	0.2947 + 0.1296i	1.4646 + 0.3600i
22	0.2964 + 0.1663i	0.3074 + 0.1566i	0.3823 + 0.2592i	1.0570 + 0.2995i
23	0.2964 + 0.1663i	0.3074 + 0.1565i	0.3944 + 0.2521i	0.9140 + 0.2530i
24	0.6584 + 0.1964i	0.6856 + 0.1427i	0.7755 + 0.1118i	0.5461 + 0.0679i
25	0.6583 + 0.1965i	0.6826 + 0.1562i	0.7513 + 0.2154i	0.5681 + 0.1947i
26	0.6652 + 0.1967i	0.6584 + 0.1422i	0.6591 + 0.1033i	0.6874 + 0.0537i
27	0.6652 + 0.1968i	0.6560 + 0.1529i	0.6446 + 0.1737i	0.7375 + 0.1492i
28	0.5987 + 0.3371i	0.5856 + 0.3840i	0.5906 + 0.4930i	0.6290 + 0.4553i
29	0.5987 + 0.3370i	0.5931 + 0.3723i	0.6538 + 0.4155i	0.6007 + 0.3177i
30	0.6039 + 0.3414i	0.5660 + 0.3651i	0.4981 + 0.3921i	0.7885 + 0.4231i
31	0.6039 + 0.3413i	0.5718 + 0.3559i	0.5373 + 0.3586i	0.7627 + 0.2849i
32	0.3183 + 1.5992i	0.1683 + 1.7041i	0.1630 + 1.6621i	0.0816 + 1.1632i
33	0.3186 + 1.5991i	0.4972 + 1.6386i	0.4720 + 1.5898i	0.0830 + 0.9813i
34	0.2756 + 1.3848i	0.1495 + 1.3560i	0.1268 + 1.3488i	0.2528 + 1.2315i
35	0.2759 + 1.3847i	0.3814 + 1.3099i	0.3752 + 1.2961i	0.2502 + 1.0100i
36	0.9060 + 1.3557i	1.0862 + 1.3238i	1.0398 + 1.2991i	0.0732 + 0.6827i
37	0.9058 + 1.3559i	0.8074 + 1.5101i	0.7733 + 1.4772i	0.0811 + 0.8293i
38	0.7846 + 1.1739i	0.8534 + 1.0644i	0.8380 + 1.0552i	0.2159 + 0.6673i

[0419]

39	0.7843 + 1.1741i	0.6568 + 1.1958i	0.6242 + 1.2081i	0.2359 + 0.8283i
40	0.2257 + 0.9956i	0.1552 + 0.9481i	0.1103 + 0.9397i	0.4302 + 1.4458i
41	0.2259 + 0.9956i	0.2200 + 0.9352i	0.2415 + 0.9155i	0.5852 + 0.9680i
42	0.2276 + 1.0326i	0.1577 + 1.0449i	0.1118 + 1.1163i	0.4528 + 1.2074i
43	0.2278 + 1.0326i	0.2548 + 1.0255i	0.3079 + 1.0866i	0.4167 + 1.0099i
44	0.5446 + 0.8635i	0.5609 + 0.7800i	0.5647 + 0.7638i	0.5035 + 0.6307i
45	0.5445 + 0.8636i	0.5060 + 0.8167i	0.4385 + 0.8433i	0.5359 + 0.7954i
46	0.5694 + 0.8910i	0.6276 + 0.8501i	0.6846 + 0.8841i	0.3580 + 0.6532i
47	0.5692 + 0.8911i	0.5452 + 0.9052i	0.5165 + 1.0034i	0.3841 + 0.8207i
48	1.5992 + 0.3183i	1.7041 + 0.1683i	1.6489 + 0.1630i	0.0576 + 0.0745i
49	1.5991 + 0.3186i	1.6386 + 0.4972i	1.5848 + 0.4983i	0.0581 + 0.2241i
50	1.3848 + 0.2756i	1.3560 + 0.1495i	1.3437 + 0.1389i	0.1720 + 0.0742i
51	1.3847 + 0.2759i	1.3099 + 0.3814i	1.2850 + 0.4025i	0.1753 + 0.2222i
52	1.3557 + 0.9060i	1.3238 + 1.0862i	1.2728 + 1.0661i	0.0652 + 0.5269i
53	1.3559 + 0.9058i	1.5101 + 0.8074i	1.4509 + 0.7925i	0.0611 + 0.3767i
54	1.1739 + 0.7846i	1.0644 + 0.8534i	1.0249 + 0.8794i	0.1972 + 0.5178i
55	1.1741 + 0.7843i	1.1958 + 0.6568i	1.1758 + 0.6545i	0.1836 + 0.3695i
56	0.9956 + 0.2257i	0.9481 + 0.1552i	0.9629 + 0.1113i	0.4145 + 0.0709i
57	0.9956 + 0.2259i	0.9352 + 0.2200i	0.9226 + 0.2849i	0.4266 + 0.2100i
58	1.0326 + 0.2276i	1.0449 + 0.1577i	1.1062 + 0.1118i	0.2912 + 0.0730i
59	1.0326 + 0.2278i	1.0255 + 0.2548i	1.0674 + 0.3393i	0.2982 + 0.2177i
60	0.8635 + 0.5446i	0.7800 + 0.5609i	0.7234 + 0.6223i	0.4766 + 0.4821i
61	0.8636 + 0.5445i	0.8167 + 0.5060i	0.8211 + 0.4860i	0.4497 + 0.3448i
62	0.8910 + 0.5694i	0.8501 + 0.6276i	0.8457 + 0.7260i	0.3334 + 0.5025i
63	0.8911 + 0.5692i	0.9052 + 0.5452i	0.9640 + 0.5518i	0.3125 + 0.3601i
64	-0.1524 + 0.3087i	-0.1170 + 0.3003i	-0.0899 + 0.1337i	-1.2412 + 1.0688i
65	-0.1525 + 0.3087i	-0.1171 + 0.3003i	-0.0910 + 0.1377i	-1.2668 + 0.8034i
66	-0.1513 + 0.3043i	-0.1204 + 0.3233i	-0.0873 + 0.3862i	-0.9860 + 1.1758i
67	-0.1513 + 0.3043i	-0.1204 + 0.3233i	-0.0883 + 0.3873i	-1.0365 + 0.9065i
68	-0.1682 + 0.3004i	-0.1454 + 0.2877i	-0.1115 + 0.1442i	-1.2111 + 0.5135i
69	-0.1682 + 0.3005i	-0.1453 + 0.2877i	-0.1135 + 0.1472i	-1.4187 + 0.6066i
70	-0.1663 + 0.2964i	-0.1566 + 0.3074i	-0.2067 + 0.3591i	-1.0103 + 0.4879i
71	-0.1663 + 0.2964i	-0.1565 + 0.3074i	-0.1975 + 0.3621i	-1.0380 + 0.6906i
72	-0.1964 + 0.6584i	-0.1427 + 0.6856i	-0.1048 + 0.7533i	-0.6963 + 1.3442i
73	-0.1965 + 0.6583i	-0.1562 + 0.6826i	-0.1770 + 0.7412i	-0.7089 + 1.1122i
74	-0.1967 + 0.6652i	-0.1422 + 0.6584i	-0.1022 + 0.5904i	-0.1256 + 1.4745i
75	-0.1968 + 0.6652i	-0.1529 + 0.6560i	-0.1191 + 0.5890i	-0.8331 + 0.9455i
76	-0.3371 + 0.5987i	-0.3840 + 0.5856i	-0.4264 + 0.6230i	-0.6615 + 0.6012i
77	-0.3370 + 0.5987i	-0.3723 + 0.5931i	-0.3650 + 0.6689i	-0.6894 + 0.7594i
78	-0.3414 + 0.6039i	-0.3651 + 0.5660i	-0.3254 + 0.5153i	-0.8373 + 0.5633i
79	-0.3413 + 0.6039i	-0.3559 + 0.5718i	-0.2959 + 0.5302i	-0.8552 + 0.7410i
80	-0.3087 + 0.1524i	-0.3003 + 0.1170i	-0.3256 + 0.0768i	-1.2666 + 0.1027i
81	-0.3087 + 0.1525i	-0.3003 + 0.1171i	-0.3266 + 0.0870i	-1.4915 + 0.1198i
82	-0.3043 + 0.1513i	-0.3233 + 0.1204i	-0.4721 + 0.0994i	-1.0766 + 0.0945i

[0420]

83	-0.3043 + 0.1513i	-0.3233 + 0.1204i	-0.4721 + 0.1206i	-0.9007 + 0.0848i
84	-0.3004 + 0.1682i	-0.2877 + 0.1454i	-0.2927 + 0.1267i	-1.2454 + 0.3064i
85	-0.3005 + 0.1682i	-0.2877 + 0.1453i	-0.2947 + 0.1296i	-1.4646 + 0.3600i
86	-0.2964 + 0.1663i	-0.3074 + 0.1566i	-0.3823 + 0.2592i	-1.0570 + 0.2995i
87	-0.2964 + 0.1663i	-0.3074 + 0.1565i	-0.3944 + 0.2521i	-0.9140 + 0.2530i
88	-0.6584 + 0.1964i	-0.6856 + 0.1427i	-0.7755 + 0.1118i	-0.5461 + 0.0679i
89	-0.6583 + 0.1965i	-0.6826 + 0.1562i	-0.7513 + 0.2154i	-0.5681 + 0.1947i
90	-0.6652 + 0.1967i	-0.6584 + 0.1422i	-0.6591 + 0.1033i	-0.6874 + 0.0537i
91	-0.6652 + 0.1968i	-0.6560 + 0.1529i	-0.6446 + 0.1737i	-0.7375 + 0.1492i
92	-0.5987 + 0.3371i	-0.5856 + 0.3840i	-0.5906 + 0.4930i	-0.6290 + 0.4553i
93	-0.5987 + 0.3370i	-0.5931 + 0.3723i	-0.6538 + 0.4155i	-0.6007 + 0.3177i
94	-0.6039 + 0.3414i	-0.5660 + 0.3651i	-0.4981 + 0.3921i	-0.7885 + 0.4231i
95	-0.6039 + 0.3413i	-0.5718 + 0.3559i	-0.5373 + 0.3586i	-0.7627 + 0.2849i
96	-0.3183 + 1.5992i	-0.1683 + 1.7041i	-0.1630 + 1.6621i	-0.0816 + 1.1632i
97	-0.3186 + 1.5991i	-0.4972 + 1.6386i	-0.4720 + 1.5898i	-0.0830 + 0.9813i
98	-0.2756 + 1.3848i	-0.1495 + 1.3560i	-0.1268 + 1.3488i	-0.2528 + 1.2315i
99	-0.2759 + 1.3847i	-0.3814 + 1.3099i	-0.3752 + 1.2961i	-0.2502 + 1.0100i
100	-0.9060 + 1.3557i	-1.0862 + 1.3238i	-1.0398 + 1.2991i	-0.0732 + 0.6827i
101	-0.9058 + 1.3559i	-0.8074 + 1.5101i	-0.7733 + 1.4772i	-0.0811 + 0.8293i
102	-0.7846 + 1.1739i	-0.8534 + 1.0644i	-0.8380 + 1.0552i	-0.2159 + 0.6673i
103	-0.7843 + 1.1741i	-0.6568 + 1.1958i	-0.6242 + 1.2081i	-0.2359 + 0.8283i
104	-0.2257 + 0.9956i	-0.1552 + 0.9481i	-0.1103 + 0.9397i	-0.4302 + 1.4458i
105	-0.2259 + 0.9956i	-0.2200 + 0.9352i	-0.2415 + 0.9155i	-0.5852 + 0.9680i
106	-0.2276 + 1.0326i	-0.1577 + 1.0449i	-0.1118 + 1.1163i	-0.4528 + 1.2074i
107	-0.2278 + 1.0326i	-0.2548 + 1.0255i	-0.3079 + 1.0866i	-0.4167 + 1.0099i
108	-0.5446 + 0.8635i	-0.5609 + 0.7800i	-0.5647 + 0.7638i	-0.5035 + 0.6307i
109	-0.5445 + 0.8636i	-0.5060 + 0.8167i	-0.4385 + 0.8433i	-0.5359 + 0.7954i
110	-0.5694 + 0.8910i	-0.6276 + 0.8501i	-0.6846 + 0.8841i	-0.3580 + 0.6532i
111	-0.5692 + 0.8911i	-0.5452 + 0.9052i	-0.5165 + 1.0034i	-0.3841 + 0.8207i
112	-1.5992 + 0.3183i	-1.7041 + 0.1683i	-1.6489 + 0.1630i	-0.0576 + 0.0745i
113	-1.5991 + 0.3186i	-1.6386 + 0.4972i	-1.5848 + 0.4983i	-0.0581 + 0.2241i
114	-1.3848 + 0.2756i	-1.3560 + 0.1495i	-1.3437 + 0.1389i	-0.1720 + 0.0742i
115	-1.3847 + 0.2759i	-1.3099 + 0.3814i	-1.2850 + 0.4025i	-0.1753 + 0.2222i
116	-1.3557 + 0.9060i	-1.3238 + 1.0862i	-1.2728 + 1.0661i	-0.0652 + 0.5269i
117	-1.3559 + 0.9058i	-1.5101 + 0.8074i	-1.4509 + 0.7925i	-0.0611 + 0.3767i
118	-1.1739 + 0.7846i	-1.0644 + 0.8534i	-1.0249 + 0.8794i	-0.1972 + 0.5178i
119	-1.1741 + 0.7843i	-1.1958 + 0.6568i	-1.1758 + 0.6545i	-0.1836 + 0.3695i
120	-0.9956 + 0.2257i	-0.9481 + 0.1552i	-0.9629 + 0.1113i	-0.4145 + 0.0709i
121	-0.9956 + 0.2259i	-0.9352 + 0.2200i	-0.9226 + 0.2849i	-0.4266 + 0.2100i
122	-1.0326 + 0.2276i	-1.0449 + 0.1577i	-1.1062 + 0.1118i	-0.2912 + 0.0730i
123	-1.0326 + 0.2278i	-1.0255 + 0.2548i	-1.0674 + 0.3393i	-0.2982 + 0.2177i
124	-0.8635 + 0.5446i	-0.7800 + 0.5609i	-0.7234 + 0.6223i	-0.4766 + 0.4821i
125	-0.8636 + 0.5445i	-0.8167 + 0.5060i	-0.8211 + 0.4860i	-0.4497 + 0.3448i
126	-0.8910 + 0.5694i	-0.8501 + 0.6276i	-0.8457 + 0.7260i	-0.3334 + 0.5025i

[0421]

127	-0.8911 + 0.5692i	-0.9052 + 0.5452i	-0.9640 + 0.5518i	-0.3125 + 0.3601i
128	0.1524 - 0.3087i	0.1170 - 0.3003i	0.0899 - 0.1337i	1.2412 - 1.0688i
129	0.1525 - 0.3087i	0.1171 - 0.3003i	0.0910 - 0.1377i	1.2668 - 0.8034i
130	0.1513 - 0.3043i	0.1204 - 0.3233i	0.0873 - 0.3862i	0.9860 - 1.1758i
131	0.1513 - 0.3043i	0.1204 - 0.3233i	0.0883 - 0.3873i	1.0365 - 0.9065i
132	0.1682 - 0.3004i	0.1454 - 0.2877i	0.1115 - 0.1442i	1.2111 - 0.5135i
133	0.1682 - 0.3005i	0.1453 - 0.2877i	0.1135 - 0.1472i	1.4187 - 0.6066i
134	0.1663 - 0.2964i	0.1566 - 0.3074i	0.2067 - 0.3591i	1.0103 - 0.4879i
135	0.1663 - 0.2964i	0.1565 - 0.3074i	0.1975 - 0.3621i	1.0380 - 0.6906i
136	0.1964 - 0.6584i	0.1427 - 0.6856i	0.1048 - 0.7533i	0.6963 - 1.3442i
137	0.1965 - 0.6583i	0.1562 - 0.6826i	0.1770 - 0.7412i	0.7089 - 1.1122i
138	0.1967 - 0.6652i	0.1422 - 0.6584i	0.1022 - 0.5904i	0.1256 - 1.4745i
139	0.1968 - 0.6652i	0.1529 - 0.6560i	0.1191 - 0.5890i	0.8331 - 0.9455i
140	0.3371 - 0.5987i	0.3840 - 0.5856i	0.4264 - 0.6230i	0.6615 - 0.6012i
141	0.3370 - 0.5987i	0.3723 - 0.5931i	0.3650 - 0.6689i	0.6894 - 0.7594i
142	0.3414 - 0.6039i	0.3651 - 0.5660i	0.3254 - 0.5153i	0.8373 - 0.5633i
143	0.3413 - 0.6039i	0.3559 - 0.5718i	0.2959 - 0.5302i	0.8552 - 0.7410i
144	0.3087 - 0.1524i	0.3003 - 0.1170i	0.3256 - 0.0768i	1.2666 - 0.1027i
145	0.3087 - 0.1525i	0.3003 - 0.1171i	0.3266 - 0.0870i	1.4915 - 0.1198i
146	0.3043 - 0.1513i	0.3233 - 0.1204i	0.4721 - 0.0994i	1.0766 - 0.0945i
147	0.3043 - 0.1513i	0.3233 - 0.1204i	0.4721 - 0.1206i	0.9007 - 0.0848i
148	0.3004 - 0.1682i	0.2877 - 0.1454i	0.2927 - 0.1267i	1.2454 - 0.3064i
149	0.3005 - 0.1682i	0.2877 - 0.1453i	0.2947 - 0.1296i	1.4646 - 0.3600i
150	0.2964 - 0.1663i	0.3074 - 0.1566i	0.3823 - 0.2592i	1.0570 - 0.2995i
151	0.2964 - 0.1663i	0.3074 - 0.1565i	0.3944 - 0.2521i	0.9140 - 0.2530i
152	0.6584 - 0.1964i	0.6856 - 0.1427i	0.7755 - 0.1118i	0.5461 - 0.0679i
153	0.6583 - 0.1965i	0.6826 - 0.1562i	0.7513 - 0.2154i	0.5681 - 0.1947i
154	0.6652 - 0.1967i	0.6584 - 0.1422i	0.6591 - 0.1033i	0.6874 - 0.0537i
155	0.6652 - 0.1968i	0.6560 - 0.1529i	0.6446 - 0.1737i	0.7375 - 0.1492i
156	0.5987 - 0.3371i	0.5856 - 0.3840i	0.5906 - 0.4930i	0.6290 - 0.4553i
157	0.5987 - 0.3370i	0.5931 - 0.3723i	0.6538 - 0.4155i	0.6007 - 0.3177i
158	0.6039 - 0.3414i	0.5660 - 0.3651i	0.4981 - 0.3921i	0.7885 - 0.4231i
159	0.6039 - 0.3413i	0.5718 - 0.3559i	0.5373 - 0.3586i	0.7627 - 0.2849i
160	0.3183 - 1.5992i	0.1683 - 1.7041i	0.1630 - 1.6621i	0.0816 - 1.1632i
161	0.3186 - 1.5991i	0.4972 - 1.6386i	0.4720 - 1.5898i	0.0830 - 0.9813i
162	0.2756 - 1.3848i	0.1495 - 1.3560i	0.1268 - 1.3488i	0.2528 - 1.2315i
163	0.2759 - 1.3847i	0.3814 - 1.3099i	0.3752 - 1.2961i	0.2502 - 1.0100i
164	0.9060 - 1.3557i	1.0862 - 1.3238i	1.0398 - 1.2991i	0.0732 - 0.6827i
165	0.9058 - 1.3559i	0.8074 - 1.5101i	0.7733 - 1.4772i	0.0811 - 0.8293i
166	0.7846 - 1.1739i	0.8534 - 1.0644i	0.8380 - 1.0552i	0.2159 - 0.6673i
167	0.7843 - 1.1741i	0.6568 - 1.1958i	0.6242 - 1.2081i	0.2359 - 0.8283i
168	0.2257 - 0.9956i	0.1552 - 0.9481i	0.1103 - 0.9397i	0.4302 - 1.4458i
169	0.2259 - 0.9956i	0.2200 - 0.9352i	0.2415 - 0.9155i	0.5852 - 0.9680i
170	0.2276 - 1.0326i	0.1577 - 1.0449i	0.1118 - 1.1163i	0.4528 - 1.2074i

[0422]

171	0.2278 - 1.0326i	0.2548 - 1.0255i	0.3079 - 1.0866i	0.4167 - 1.0099i
172	0.5446 - 0.8635i	0.5609 - 0.7800i	0.5647 - 0.7638i	0.5035 - 0.6307i
173	0.5445 - 0.8636i	0.5060 - 0.8167i	0.4385 - 0.8433i	0.5359 - 0.7954i
174	0.5694 - 0.8910i	0.6276 - 0.8501i	0.6846 - 0.8841i	0.3580 - 0.6532i
175	0.5692 - 0.8911i	0.5452 - 0.9052i	0.5165 - 1.0034i	0.3841 - 0.8207i
176	1.5992 - 0.3183i	1.7041 - 0.1683i	1.6489 - 0.1630i	0.0576 - 0.0745i
177	1.5991 - 0.3186i	1.6386 - 0.4972i	1.5848 - 0.4983i	0.0581 - 0.2241i
178	1.3848 - 0.2756i	1.3560 - 0.1495i	1.3437 - 0.1389i	0.1720 - 0.0742i
179	1.3847 - 0.2759i	1.3099 - 0.3814i	1.2850 - 0.4025i	0.1753 - 0.2222i
180	1.3557 - 0.9060i	1.3238 - 1.0862i	1.2728 - 1.0661i	0.0652 - 0.5269i
181	1.3559 - 0.9058i	1.5101 - 0.8074i	1.4509 - 0.7925i	0.0611 - 0.3767i
182	1.1739 - 0.7846i	1.0644 - 0.8534i	1.0249 - 0.8794i	0.1972 - 0.5178i
183	1.1741 - 0.7843i	1.1958 - 0.6568i	1.1758 - 0.6545i	0.1836 - 0.3695i
184	0.9956 - 0.2257i	0.9481 - 0.1552i	0.9629 - 0.1113i	0.4145 - 0.0709i
185	0.9956 - 0.2259i	0.9352 - 0.2200i	0.9226 - 0.2849i	0.4266 - 0.2100i
186	1.0326 - 0.2276i	1.0449 - 0.1577i	1.1062 - 0.1118i	0.2912 - 0.0730i
187	1.0326 - 0.2278i	1.0255 - 0.2548i	1.0674 - 0.3393i	0.2982 - 0.2177i
188	0.8635 - 0.5446i	0.7800 - 0.5609i	0.7234 - 0.6223i	0.4766 - 0.4821i
189	0.8636 - 0.5445i	0.8167 - 0.5060i	0.8211 - 0.4860i	0.4497 - 0.3448i
190	0.8910 - 0.5694i	0.8501 - 0.6276i	0.8457 - 0.7260i	0.3334 - 0.5025i
191	0.8911 - 0.5692i	0.9052 - 0.5452i	0.9640 - 0.5518i	0.3125 - 0.3601i
192	-0.1524 - 0.3087i	-0.1170 - 0.3003i	-0.0899 - 0.1337i	-1.2412 - 1.0688i
193	-0.1525 - 0.3087i	-0.1171 - 0.3003i	-0.0910 - 0.1377i	-1.2668 - 0.8034i
194	-0.1513 - 0.3043i	-0.1204 - 0.3233i	-0.0873 - 0.3862i	-0.9860 - 1.1758i
195	-0.1513 - 0.3043i	-0.1204 - 0.3233i	-0.0883 - 0.3873i	-1.0365 - 0.9065i
196	-0.1682 - 0.3004i	-0.1454 - 0.2877i	-0.1115 - 0.1442i	-1.2111 - 0.5135i
197	-0.1682 - 0.3005i	-0.1453 - 0.2877i	-0.1135 - 0.1472i	-1.4187 - 0.6066i
198	-0.1663 - 0.2964i	-0.1566 - 0.3074i	-0.2067 - 0.3591i	-1.0103 - 0.4879i
199	-0.1663 - 0.2964i	-0.1565 - 0.3074i	-0.1975 - 0.3621i	-1.0380 - 0.6906i
200	-0.1964 - 0.6584i	-0.1427 - 0.6856i	-0.1048 - 0.7533i	-0.6963 - 1.3442i
201	-0.1965 - 0.6583i	-0.1562 - 0.6826i	-0.1770 - 0.7412i	-0.7089 - 1.1122i
202	-0.1967 - 0.6652i	-0.1422 - 0.6584i	-0.1022 - 0.5904i	-0.1256 - 1.4745i
203	-0.1968 - 0.6652i	-0.1529 - 0.6560i	-0.1191 - 0.5890i	-0.8331 - 0.9455i
204	-0.3371 - 0.5987i	-0.3840 - 0.5856i	-0.4264 - 0.6230i	-0.6615 - 0.6012i
205	-0.3370 - 0.5987i	-0.3723 - 0.5931i	-0.3650 - 0.6689i	-0.6894 - 0.7594i
206	-0.3414 - 0.6039i	-0.3651 - 0.5660i	-0.3254 - 0.5153i	-0.8373 - 0.5633i
207	-0.3413 - 0.6039i	-0.3559 - 0.5718i	-0.2959 - 0.5302i	-0.8552 - 0.7410i
208	-0.3087 - 0.1524i	-0.3003 - 0.1170i	-0.3256 - 0.0768i	-1.2666 - 0.1027i
209	-0.3087 - 0.1525i	-0.3003 - 0.1171i	-0.3266 - 0.0870i	-1.4915 - 0.1198i
210	-0.3043 - 0.1513i	-0.3233 - 0.1204i	-0.4721 - 0.0994i	-1.0766 - 0.0945i
211	-0.3043 - 0.1513i	-0.3233 - 0.1204i	-0.4721 - 0.1206i	-0.9007 - 0.0848i
212	-0.3004 - 0.1682i	-0.2877 - 0.1454i	-0.2927 - 0.1267i	-1.2454 - 0.3064i
213	-0.3005 - 0.1682i	-0.2877 - 0.1453i	-0.2947 - 0.1296i	-1.4646 - 0.3600i
214	-0.2964 - 0.1663i	-0.3074 - 0.1566i	-0.3823 - 0.2592i	-1.0570 - 0.2995i

[0423]

215	-0.2964 - 0.1663i	-0.3074 - 0.1565i	-0.3944 - 0.2521i	-0.9140 - 0.2530i
216	-0.6584 - 0.1964i	-0.6856 - 0.1427i	-0.7755 - 0.1118i	-0.5461 - 0.0679i
217	-0.6583 - 0.1965i	-0.6826 - 0.1562i	-0.7513 - 0.2154i	-0.5681 - 0.1947i
218	-0.6652 - 0.1967i	-0.6584 - 0.1422i	-0.6591 - 0.1033i	-0.6874 - 0.0537i
219	-0.6652 - 0.1968i	-0.6560 - 0.1529i	-0.6446 - 0.1737i	-0.7375 - 0.1492i
220	-0.5987 - 0.3371i	-0.5856 - 0.3840i	-0.5906 - 0.4930i	-0.6290 - 0.4553i
221	-0.5987 - 0.3370i	-0.5931 - 0.3723i	-0.6538 - 0.4155i	-0.6007 - 0.3177i
222	-0.6039 - 0.3414i	-0.5660 - 0.3651i	-0.4981 - 0.3921i	-0.7885 - 0.4231i
223	-0.6039 - 0.3413i	-0.5718 - 0.3559i	-0.5373 - 0.3586i	-0.7627 - 0.2849i
224	-0.3183 - 1.5992i	-0.1683 - 1.7041i	-0.1630 - 1.6621i	-0.0816 - 1.1632i
225	-0.3186 - 1.5991i	-0.4972 - 1.6386i	-0.4720 - 1.5898i	-0.0830 - 0.9813i
226	-0.2756 - 1.3848i	-0.1495 - 1.3560i	-0.1268 - 1.3488i	-0.2528 - 1.2315i
227	-0.2759 - 1.3847i	-0.3814 - 1.3099i	-0.3752 - 1.2961i	-0.2502 - 1.0100i
228	-0.9060 - 1.3557i	-1.0862 - 1.3238i	-1.0398 - 1.2991i	-0.0732 - 0.6827i
229	-0.9058 - 1.3559i	-0.8074 - 1.5101i	-0.7733 - 1.4772i	-0.0811 - 0.8293i
230	-0.7846 - 1.1739i	-0.8534 - 1.0644i	-0.8380 - 1.0552i	-0.2159 - 0.6673i
231	-0.7843 - 1.1741i	-0.6568 - 1.1958i	-0.6242 - 1.2081i	-0.2359 - 0.8283i
232	-0.2257 - 0.9956i	-0.1552 - 0.9481i	-0.1103 - 0.9397i	-0.4302 - 1.4458i
233	-0.2259 - 0.9956i	-0.2200 - 0.9352i	-0.2415 - 0.9155i	-0.5852 - 0.9680i
234	-0.2276 - 1.0326i	-0.1577 - 1.0449i	-0.1118 - 1.1163i	-0.4528 - 1.2074i
235	-0.2278 - 1.0326i	-0.2548 - 1.0255i	-0.3079 - 1.0866i	-0.4167 - 1.0099i
236	-0.5446 - 0.8635i	-0.5609 - 0.7800i	-0.5647 - 0.7638i	-0.5035 - 0.6307i
237	-0.5445 - 0.8636i	-0.5060 - 0.8167i	-0.4385 - 0.8433i	-0.5359 - 0.7954i
238	-0.5694 - 0.8910i	-0.6276 - 0.8501i	-0.6846 - 0.8841i	-0.3580 - 0.6532i
239	-0.5692 - 0.8911i	-0.5452 - 0.9052i	-0.5165 - 1.0034i	-0.3841 - 0.8207i
240	-1.5992 - 0.3183i	-1.7041 - 0.1683i	-1.6489 - 0.1630i	-0.0576 - 0.0745i
241	-1.5991 - 0.3186i	-1.6386 - 0.4972i	-1.5848 - 0.4983i	-0.0581 - 0.2241i
242	-1.3848 - 0.2756i	-1.3560 - 0.1495i	-1.3437 - 0.1389i	-0.1720 - 0.0742i
243	-1.3847 - 0.2759i	-1.3099 - 0.3814i	-1.2850 - 0.4025i	-0.1753 - 0.2222i
244	-1.3557 - 0.9060i	-1.3238 - 1.0862i	-1.2728 - 1.0661i	-0.0652 - 0.5269i
245	-1.3559 - 0.9058i	-1.5101 - 0.8074i	-1.4509 - 0.7925i	-0.0611 - 0.3767i
246	-1.1739 - 0.7846i	-1.0644 - 0.8534i	-1.0249 - 0.8794i	-0.1972 - 0.5178i
247	-1.1741 - 0.7843i	-1.1958 - 0.6568i	-1.1758 - 0.6545i	-0.1836 - 0.3695i
248	-0.9956 - 0.2257i	-0.9481 - 0.1552i	-0.9629 - 0.1113i	-0.4145 - 0.0709i
249	-0.9956 - 0.2259i	-0.9352 - 0.2200i	-0.9226 - 0.2849i	-0.4266 - 0.2100i
250	-1.0326 - 0.2276i	-1.0449 - 0.1577i	-1.1062 - 0.1118i	-0.2912 - 0.0730i
251	-1.0326 - 0.2278i	-1.0255 - 0.2548i	-1.0674 - 0.3393i	-0.2982 - 0.2177i
252	-0.8635 - 0.5446i	-0.7800 - 0.5609i	-0.7234 - 0.6223i	-0.4766 - 0.4821i
253	-0.8636 - 0.5445i	-0.8167 - 0.5060i	-0.8211 - 0.4860i	-0.4497 - 0.3448i
254	-0.8910 - 0.5694i	-0.8501 - 0.6276i	-0.8457 - 0.7260i	-0.3334 - 0.5025i
255	-0.8911 - 0.5692i	-0.9052 - 0.5452i	-0.9640 - 0.5518i	-0.3125 - 0.3601i

[0424]

[0425]

한편, 표 31에서는 non-uniform 256-QAM 방식에 따라 정의되는 정상도의 일 예는 나타내었으나 이는 일 예에 불과하다. 정상점은 non-uniform 256-QAM 방식에서 다양하게 정의될 수 있으며, 또한, 정상점은 non-uniform 16-QAM, non-uniform 64-QAM, non-uniform 1024-QAM, non-uniform 4096-QAM 등과 같은 다른 변조 방식에서도 다양하게 정의될 수 있다.

[0426]

한편, 본 발명에서 상술한 방식으로 인터리빙을 수행하는 이유는 다음과 같다.

[0427]

구체적으로, LDPC 부호어 비트들이 변조 심볼에 맵핑되는 경우, 이들은 변조 심볼 내에서 맵핑되는 위치에 따라 서로 다른 신뢰도(즉, 수신 성능 또는 수신 확률)를 가지게 된다. 한편, 패리티 검사 행렬의 구조에 따라 LDPC 부호어 비트들은 서로 다른 부호어 특성을 가질 수 있다. 즉, 패리티 검사 행렬의 열에 존재하는 1의 개수 즉, 열 차수에 따라 LDPC 부호어 비트들은 서로 다른 부호어 특성을 가질 수 있다.

[0428]

따라서, 본 발명에서 인터리버(120)는 LDPC 부호어 비트들의 부호어 특성 및 변조 심볼을 구성하는 비트들의 신뢰도를 모두 고려하여 특정한 부호어 특성을 갖는 LDPC 부호어 비트가 변조 심볼 내의 특정한 비트에 맵핑되도록 인터리빙을 수행하게 된다.

[0429]

예를 들어, 비트 그룹 X_0 내지 비트 그룹 X_{44} 로 구성된 LDPC 부호어를 수학식 21 및 표 16을 기반으로 그룹 인터리빙한 경우, 그룹 인터리버(122)는 비트 그룹 X_{13} , 비트 그룹 X_{16} , 비트 그룹 X_4 , ..., 비트 그룹 X_{41} , 비트 그룹 X_{29} 순으로 출력할 수 있다.

[0430]

이 경우, 블록 인터리버(124)를 구성하는 열의 개수는 8 개이며, 제1 파트의 행의 개수는 1800이며, 제2 파트의

행의 개수는 225이다.

- [0431] 이에 따라, LDPC 부호어를 구성하는 45 개의 비트 그룹 중 5 개의 비트 그룹 (X_{13} , X_{16} , X_4 , X_{12} , X_{44})이 블록 인터리버(124)의 첫 번째 열의 제1 파트에 입력되고, 5 개의 비트 그룹 (X_{15} , X_8 , X_{14} , X_0 , X_3)이 블록 인터리버(124)의 두 번째 열의 제1 파트에 입력되고, 5 개의 비트 그룹 (X_{30} , X_{20} , X_{35} , X_{21} , X_{10})이 블록 인터리버(124)의 세 번째 열의 제1 파트에 입력되고, 5 개의 그룹 (X_6 , X_{19} , X_{17} , X_{26} , X_{39})이 블록 인터리버(124)의 네 번째 열의 제1 파트에 입력되고, 5 개의 비트 그룹 (X_7 , X_{24} , X_9 , X_{27} , X_5)이 블록 인터리버(124)의 다섯 번째 열의 제1 파트에 입력되고, 5 개의 비트 그룹 (X_{37} , X_{23} , X_{32} , X_{40} , X_{31})이 블록 인터리버(124)의 여섯 번째 열의 제1 파트에 입력되고, 5 개의 비트 그룹 (X_{38} , X_{42} , X_{34} , X_{25} , X_{36})이 블록 인터리버(124)의 일곱 번째 열의 제1 파트에 입력되고, 5 개의 비트 그룹 (X_2 , X_{22} , X_{43} , X_{33} , X_{28})이 블록 인터리버(124)의 여덟 번째 열의 제1 파트에 입력될 수 있다.
- [0432] 그리고, 비트 그룹 X_{11} , 비트 그룹 X_{18} , 비트 그룹 X_{11} , 비트 그룹 X_{41} , 비트 그룹 X_{29} 는 블록 인터리버(124)의 제2 파트에 입력된다.
- [0433] 즉, 블록 인터리버(124)는 비트 그룹(X_1)에 포함된 360 개의 비트 중 225 개의 비트를 제1 열의 제2 파트의 1 번째 행부터 225 번째 행까지 열 방향으로 라이트하고, 남은 135 개의 비트를 제2 열의 제2 파트의 1 번째 행부터 135 번째 행까지 열 방향으로 라이트할 수 있다. 그리고, 블록 인터리버(124)는 비트 그룹(X_{18})에 포함된 360 개의 비트 중 90 개의 비트를 제2 열의 제2 파트의 136 번째 행부터 225 번째 행까지 열 방향으로 라이트하고, 남은 270 개의 비트 중 225 개의 비트를 제3 열의 제2 파트의 1 번째 행부터 225 번째 행까지 열 방향으로 라이트하고 45 개의 비트를 제4 열의 제2 파트의 1 번째 행부터 45 번째 행까지 열 방향으로 라이트할 수 있다. 그리고, 블록 인터리버(124)는 비트 그룹(X_{11})에 포함된 360 개의 비트 중 180 개의 비트를 제4 열의 제2 파트의 46 번째 행부터 225 번째 행까지 열 방향으로 라이트하고, 남은 180 개의 비트를 제5 열의 제2 파트의 1 번째 행부터 180 번째 행까지 열 방향으로 라이트할 수 있다. 그리고, 블록 인터리버(124)는 비트 그룹(X_{41})에 포함된 360 개의 비트 중 45 개의 비트를 제5 열의 제2 파트의 181 번째 행부터 225 번째 행까지 열 방향으로 라이트하고, 남은 315 개의 비트 중 225 개의 비트를 제6 열의 제2 파트의 1 번째 행부터 225 번째 행까지 열 방향으로 라이트하고 90 개의 비트를 제7 열의 제2 파트의 1 번째 행부터 90 번째 행까지 열 방향으로 라이트할 수 있다. 그리고, 블록 인터리버(124)는 비트 그룹(X_{29})에 포함된 360 개의 비트 중 135 개의 비트를 제7 열의 제2 파트의 91 번째 행부터 225 번째 행까지 열 방향으로 라이트하고, 남은 225 개의 비트를 제8 열의 제2 파트의 1 번째 행부터 225 번째 행까지 열 방향으로 라이트할 수 있다.
- [0434] 그리고, 블록 인터리버(124)는 각 열의 첫 번째 행부터 마지막 행에 입력된 비트들을 순차적으로 출력할 수 있으며, 블록 인터리버(124)에서 출력되는 비트들은 변조부(130)에 순차적으로 입력될 수 있다. 이 경우, 디멀티플렉서(미도시)는 생략되거나, 디멀티플렉서(미도시)는 입력되는 비트들의 순서를 변경함이 없이 순차적으로 출력할 수 있다. 이에 따라, 비트 그룹 X_{13} , 비트 그룹 X_{15} , 비트 그룹 X_{30} , 비트 그룹 X_6 , 비트 그룹 X_7 , 비트 그룹 X_{37} , 비트 그룹 X_{38} , 비트 그룹 X_2 각각에 포함된 비트가 변조 심볼을 구성할 수 있다.
- [0435] 이와 같이, 본 발명에서는 인터리빙을 통해 특정한 비트가 변조 심볼 내의 특정한 비트에 맵핑된다는 점에서, 수신 측에서 우수한 수신 성능 및 복호화 성능 모두를 달성할 수 있게 된다.
- [0436] 한편, 이하에서는 본 발명의 다양한 실시 예에서 그룹 인터리빙에 이용되는 파라미터인 $\pi(j)$ 를 결정하는 방법에 대해 설명하도록 한다. 먼저, 고려해야 할 criteria는 이하와 같다.
- [0437] criteria 1) 변조 방식과 부호율에 따라 다른 인터리빙 순서를 사용한다.
- [0438] criteria 2) LDPC 부호어의 각 비트 그룹별 성능 특성과 변조 신호를 구성하는 각 비트들의 성능 특성을 동시에 고려하도록 한다.
- [0439] 일 예로, LDPC 부호어의 경우 성능이 우수한 비트들이 leftmost bit들이며, 변조 심볼을 구성하는 비트 leftmost bit들이 성능이 우수할 수 있다. 즉, non-uniform 256-QAM을 구성하는 8 개의 비트들 y_0 , y_1 , y_2 , y_3 , y_4 , y_5 , y_6 , y_7 에 대하여 각 비트들의 수신 성능 $P(y_i)$ 의 상대적인 크기는 $P(y_0) \geq P(y_1) \geq P(y_2) \geq P(y_3) \geq P(y_4) \geq$

$P(y_5) \geq P(y_6) \geq P(y_7)$ 와 같다.

[0440] 그러므로, LDPC 부호어의 길이가 16200 이고 non-uniform 256-QAM(또는, 256-NUC)을 사용할 경우, 부호와 변조 방식의 특성을 동시에 고려하여, 45 개의 LDPC 비트 그룹을 non-uniform 256-QAM를 구성하는 8 개의 비트들 중 어떤 비트로 매핑할지를 결정하게 된다. 이 경우, Density evolution이라는 방법을 사용하여 예상 성능이 가장 우수하게 나오는 경우를 결정하도록 한다.

[0441] 즉, 45 개의 비트 그룹들을 8 개의 비트로 매핑할 수 있는 여러 가지 경우를 고려하고, 각각의 경우에 대하여 Density evolution 방법을 적용하여 이론적으로 예상할 수 있는 threshold 값을 구할 수 있다. 여기에서, threshold는 SNR 값으로 LDPC 부호를 전송할 경우 threshold 값 보다 높은 SNR영역에서는 error 확률이 '0'가 된다는 것을 의미한다. 그러므로, 여러 가지 매핑 가능한 경우 중에서 threshold 값이 작은 경우의 방법대로 LDPC 부호를 전송할 경우 우수한 성능을 보장할 수 있다. 다만, density evolution을 기반으로 인터리버(120)를 설계하는 것은 이론적인 접근 방법이다. 그러므로, density evolution이라는 이론적 접근 방법뿐만 아니라 실제 설계된 패리티 검사 행렬을 기반으로 한 부호 성능을 검증 및 사이클 분포를 기반으로 인터리버(120)를 설계하도록 한다

[0442] 한편, 45 개의 비트 그룹들을 8 개의 비트로 맵핑할 때, 패리티 검사 행렬에서 디그리가 같은 행에 연관된 비트 그룹들을 동일한 그룹으로 그룹핑하고, 그룹핑된 각 그룹 내에서 몇 개의 그룹을 8 개의 256-QAM 비트들 중 어떠한 비트들에 맵핑할지를 결정하도록 한다.

[0443] 예를 들어, LDPC 부호의 패리티 검사 행렬이 디그리가 16, 10, 3, 2인 행을 포함하며, 디그리가 16, 10, 3, 2인 행 각각에 연관된 비트 그룹들의 개수가 각각 3, 5, 19, 18인 경우를 가정하도록 한다.

[0444] 한편, non-uniform 256-QAM 방식의 경우 변조 심볼을 구성하는 각 비트들의 수신 성능 $P(y_i)$ 의 상대적인 크기는 $P(y_0) \geq P(y_1) \geq P(y_2) \geq P(y_3) \geq P(y_4) \geq P(y_5) \geq P(y_6) \geq P(y_7)$ 와 같다는 점에서, 변조 심볼을 구성하는 비트들 중 수신 성능에 가장 큰 영향을 주는 y_0, y_1 이며, 이에 따라, y_0, y_1 에 대해 어떠한 비트 그룹을 맵핑할 것인지를 먼저 결정하도록 한다.

[0445] 구체적으로, y_0 와 y_1 에 매핑되는 비트 그룹들에 대해서는 $P(y_0), P(y_1)$ 을 사용하고, 다른 비트들(즉, $y_2, y_3, y_4, y_5, y_6, y_7$)에 매핑되는 비트 그룹들에 대해서는 평균 확률을 사용하여, 비트 그룹들이 y_0, y_1 에 매핑되는 경우의 수를 하기와 같이 산출하도록 한다.

modulated bits	degree	mapping되는 경우의 수
y_0, y_1	16	${}_3C_{x1}$
	10	${}_5C_{w1}$
	3	${}_{19}C_{z1}$
	2	${}_{18}C_{l1}$
$y_2, y_3, y_4, y_5, y_6, y_7$	16	${}_3C_{(3-x1)}$
	10	${}_5C_{(5-w1)}$
	3	${}_{19}C_{(19-z1)}$
	2	${}_{18}C_{(18-l1)}$
sum		45

[0446]

[0447] 즉, y_0 와 y_1 에 매핑되는 비트 그룹들을 디그리가 16인 행과 연관된 비트 그룹들 중에서 x_1 개 선택하고, 디그리가 10인 행과 연관된 비트 그룹들 중에서 w_1 개 선택하고, 디그리가 3인 행과 연관된 비트 그룹들 중에서 z_1 개 선택하고, 디그리가 2인 행과 연관된 비트 그룹들 중에서 l_1 개 선택하는 경우, 그 경우의 수는 ${}_3C_{x1} + {}_5C_{w1} + {}_{19}C_{z1} + {}_{18}C_{l1}$ 이 될 수 있다.

[0448] 이에 따라, 나머지 비트들에 매핑되는 비트 그룹들에 대한 경우의 수는 ${}_3C_{(3-x1)} + {}_5C_{(5-w1)} + {}_{19}C_{(19-z1)} + {}_{18}C_{(18-l1)}$ 와 같이 될 수 있다.

[0449] 이후, 각 경우마다 Density evolution을 통해 성능을 추정한 후, 성능이 가장 우수할 것으로 예상되는 경우를 선택한다. 즉, Density evolution을 통해 가장 우수한 성능을 갖기 위해 디그리가 16, 10, 3, 2인 행과 연관된 비트 그룹들 각각에서 몇 개의 비트 그룹을 선택하여 y_0 와 y_1 에 맵핑하여야 하는지를 판단하여, x_1, w_1, z_1, l_1 을

결정하도록 한다.

[0450] 그리고, 결정된 x_1 , w_1 , z_1 , l_1 을 기초로, 다음으로 수신 성능에 영향을 주는 y_2 , y_3 에 대해 어떠한 비트 그룹을 맵핑할 것인지를 결정하도록 한다.

[0451] 이 경우, y_2 와 y_3 에 맵핑되는 비트 그룹들에 대해서는 $P(y_2)$, $P(y_3)$ 을 사용하고, 다른 비트들(즉, y_4 , y_5 , y_6 , y_7)에 맵핑되는 비트 그룹들에 대해서는 평균 확률을 사용하여, 비트 그룹들이 y_2 , y_3 에 맵핑되는 경우의 수를 하기와 같이 산출하도록 한다.

modulated bits	degree	mapping되는 경우의 수
y_0, y_1	16	x_1
	10	w_1
	3	z_1
	2	l_1
y_2, y_3	16	${}_{(3-x_1)}C_{x_2}$
	10	${}_{(5-w_1)}C_{w_2}$
	3	${}_{(19-z_1)}C_{z_2}$
	2	${}_{(18-l_1)}C_{l_2}$
y_4, y_5, y_6, y_7	16	${}_3C_{(3-x_1-x_2)}$
	10	${}_5C_{(5-w_1-w_2)}$
	3	${}_{19}C_{(19-z_1-z_2)}$
	2	${}_{18}C_{(18-l_1-l_2)}$
sum		45

[0452]

[0453] 즉, y_2 와 y_3 에 맵핑되는 비트 그룹들을 디그리가 16인 행과 연관된 비트 그룹들 중에서 x_2 개 선택하고, 디그리가 10인 행과 연관된 비트 그룹들 중에서 w_2 개 선택하고, 디그리가 3인 행과 연관된 비트 그룹들 중에서 z_2 개 선택하고, 디그리가 2인 행과 연관된 비트 그룹들 중에서 l_2 개 선택하는 경우, 그 경우의 수는 ${}_{(3-x_1)}C_{x_2} + {}_{(5-w_1)}C_{w_2} + {}_{(19-z_1)}C_{z_2} + {}_{(18-l_1)}C_{l_2}$ 이 될 수 있다.

[0454] 이에 따라, 나머지 비트들에 맵핑되는 비트 그룹들에 대한 경우의 수는 ${}_3C_{(3-x_1-x_2)} + {}_5C_{(5-w_1-w_2)} + {}_{19}C_{(19-z_1-z_2)} + {}_{18}C_{(18-l_1-l_2)}$ 와 같이 될 수 있다.

[0455] 이후, 각 경우마다 Density evolution을 통해 성능을 추정한 후, 성능이 가장 우수할 것으로 예상되는 경우를 선택한다. 즉, Density evolution을 통해 가장 우수한 성능을 갖기 위해 디그리가 16, 10, 3, 2인 행과 연관된 비트 그룹들 각각에서 몇 개의 비트 그룹을 선택하여 y_2 와 y_3 에 맵핑하는지를 판단하여, x_2 , w_2 , z_2 , l_2 을 결정하도록 한다.

[0456] 그리고, 결정된 x_2 , w_2 , z_2 , l_2 을 기초로, 상술한 방법을 통해 다음으로 수신 성능에 영향을 주는 y_4 , y_5 에 대해 몇 개의 비트 그룹을 맵핑할 것인지를 결정하여, 디그리가 16, 10, 3, 2인 행과 연관된 비트 그룹들 중에서 몇 개의 비트 그룹씩 변조 심볼을 구성하는 비트 각각에 맵핑되는지를 최종적으로 결정하게 된다.

[0457] 이에 따라, 각 디그리 값을 갖는 행과 연관된 비트 그룹 각각에서 몇 개의 비트 그룹이 어떠한 256-QAM 비트들에 맵핑되는 것이 가장 우수한 성능을 갖는 경우인지를 결정할 수 있으며, 이 경우를 만족하도록 LDPC 부호어의 특정 그룹을 변조 심볼 내의 특정 비트에 맵핑시킬 수 있는 인터리버(120)를 설계하도록 한다.

[0458] 결국, 이와 같은 방식에 따라 본 발명에서 제안하는 그룹 인터리빙 방법을 설계할 수 있게 된다.

[0459] 한편, 송신 장치(100)는 성상도에 맵핑된 신호를 변조하여 수신 장치(가령, 도 16의 1200)로 전송할 수 있다. 예를 들어, 송신 장치(100)는 OFDM(Orthogonal Frequency Division Multiplexing) 방식을 이용하여 성상도에 맵핑된 신호를 OFDM 프레임에 맵핑하고, 할당된 채널을 통해 수신 장치(1200)로 전송할 수 있다.

[0460] 도 16은 본 발명의 일 실시 예에 따른 수신 장치의 구성을 설명하기 위한 블록도이다. 도 16을 참조하면, 수신 장치(1200)는 복조부(1210), 멀티플렉서(1220), 디인터리버(1230) 및 복호화부(1240)를 포함한다.

[0461] 복조부(1210)는 송신 장치(100)에서 전송한 신호를 수신하여 복조한다. 구체적으로, 복조부(1210)는 수신된 신호를 복조하여 LDPC 부호어에 대응되는 값을 생성하고, 이를 멀티플렉서(1220)로 출력한다. 이 경우, 복조부

(1210)는 송신 장치(100)에서 사용된 변조 방식에 대응되도록 복조를 수행할 수 있다. 이를 위해, 송신 장치(100)는 변조 방식에 대한 정보를 수신 장치(1200)로 전송할 수 있으며, 또는, 송신 장치(100)는 수신 장치(1500)와의 사이에서 기정의된 변조 방식을 이용하여 변조를 수행할 수 있다.

[0462] 여기에서, LDPC 부호어에 대응되는 값은 수신된 신호에 대한 채널 값으로 표현될 수 있다. 채널 값을 결정하는 방법은 다양하게 존재할 수 있으며, 일 예로, LLR(Log Likelihood Ratio) 값을 결정하는 방법이 될 수 있다.

[0463] LLR 값은 송신 장치(100)에서 전송한 비트가 0일 확률과 1일 확률의 비율에 Log를 취한 값으로 나타낼 수 있다. 또는, LLR 값은 경판정(hard decision)에 따라 결정된 비트 값 자체가 될 수 있으며, 또한, LLR 값은 송신 장치(100)에서 전송한 비트가 0 또는 1일 확률이 속하는 구간에 따라 결정된 대표 값이 될 수도 있다.

[0464] 멀티플렉서(1220)는 복조부(1210)의 출력 값을 멀티플렉싱하여 디인터리버(1230)로 출력한다.

[0465] 구체적으로, 멀티플렉서(1220)는 송신 장치(100)에 구비된 디멀티플렉서(미도시)에 대응되는 구성요소로, 디멀티플렉서(미도시)에 대응되는 동작을 수행한다. 즉, 멀티플렉서(1220)는 디멀티플렉서(미도시)에서 수행된 동작을 역으로 수행하여, 복조부(1210)의 출력 값을 셀-투-비트(cell-to-bit) 변환하여 비트 단위의 LLR 값을 출력할 수 있다. 다만, 송신 장치(100)에서 디멀티플렉서(미도시)가 생략되는 경우, 수신 장치(1200)의 멀티플렉서(1220)는 생략될 수 있다.

[0466] 한편, 디멀티플렉싱 동작의 수행 여부에 대한 정보는 송신 장치(100)로부터 제공되거나, 송신 장치(100)와 수신 장치(1200) 사이에 기저장되어 있을 수 있다.

[0467] 디인터리버(1230)는 멀티플렉서(1220)의 출력 값을 디인터리빙하고, 이를 복호화부(1240)으로 출력한다.

[0468] 구체적으로, 디인터리버(1230)는 송신 장치(100)의 인터리버(120)에 대응되는 구성요소로, 인터리버(120)에 대응되는 동작을 수행한다. 즉, 디인터리버(1230)는 인터리버(120)에서 수행된 인터리빙 동작을 역으로 수행하여 LLR 값을 디인터리빙한다.

[0469] 이를 위해, 디인터리버(1230)는 도 17과 같이 블록 디인터리버(1231), 그룹 트위스트 디인터리버(1232), 그룹 디인터리버(1233) 및 패리티 디인터리버(1234)를 포함할 수 있다.

[0470] 블록 디인터리버(1231)는 멀티플렉서(1220)의 출력을 디인터리빙하고, 이를 그룹 트위스트 디인터리버(1232)로 출력한다.

[0471] 구체적으로, 블록 디인터리버(1231)는 송신 장치(100)에 구비된 블록 인터리버(124)에 대응되는 구성요소로, 블록 인터리버(124)에서 수행된 인터리빙 동작을 역으로 수행할 수 있다.

[0472] 즉, 블록 디인터리버(1231)는 복수의 열로 이루어진 적어도 하나의 행을 이용하여, 멀티플렉서(1220)에서 출력된 LLR 값을 각 행에 행 방향으로 라이트하고, LLR 값이 라이트된 복수의 행의 각 열을 열 방향으로 리드하여 디인터리빙을 수행할 수 있다.

[0473] 이 경우, 블록 인터리버(124)에서 열을 2 개의 파트로 구분하여 인터리빙을 수행한 경우, 블록 디인터리버(1231)는 행을 2 개의 파트로 구분하여 디인터리빙을 수행할 수 있다.

[0474] 또한, 블록 인터리버(124)에서 제1 파트에 속하지 않은 비트 그룹에 대해 행 방향으로 라이트 및 리드한 경우, 블록 디인터리버(1231)는 제1 파트에 속하지 않은 그룹에 대응되는 값은 행 방향으로 라이트 및 리드하여 디인터리빙을 수행할 수도 있다.

[0475] 이하에서는 도 18을 참조하여 블록 디인터리버(1231)에 대해 설명하도록 한다. 다만, 이는 일 예일 뿐이며, 블록 디인터리버(1231)는 다른 방법으로 구현될 수 있음은 물론이다.

[0476] 입력 LLR $v_i (0 \leq i < N_{ldpc})$ 는 블록 디인터리버(1231)의 r_i 행, c_i 열에 라이트된다. 여기에서, $c_i = (i \bmod N_c)$, $r_i = \left\lfloor \frac{i}{N_c} \right\rfloor$ 이다.

[0477] 한편, 출력 LLR $q_i (0 \leq i < N_c \times N_{r1})$ 는 블록 디인터리버(1231)의 제1 파트의 c_i 열, r_i 행으로부터 리드된다. 여기에서, $c_i = \left\lfloor \frac{i}{N_{r1}} \right\rfloor$, $r_i = (i \bmod N_{r1})$ 이다.

- [0478] 그리고, 출력 LLR $q_i(N_c \times N_{r1} \leq i < N_{ldpc})$ 는 블록 디인터리버(1231)의 제2 파트의 c_i 열, r_i 행으로부터 리드된다.
- $$c_i = \left\lfloor \frac{(i - N_c \times N_{r1})}{N_{r2}} \right\rfloor, \quad r_i = N_{r1} + \{(i - N_c \times N_{r1}) \bmod N_{r2}\}$$
- 여기에서, , 이다.
- [0479] 그룹 트위스트 디인터리버(1232)는 블록 디인터리버(1231)의 출력 값을 디인터리빙하고 이를 그룹 디인터리버(1233)으로 출력한다.
- [0480] 구체적으로, 그룹 트위스트 디인터리버(1232)는 송신 장치(100)에 구비된 그룹 트위스트 인터리버(123)에 대응되는 구성요소로, 그룹 트위스트 인터리버(123)에서 수행된 인터리빙 동작을 역으로 수행할 수 있다.
- [0481] 즉, 그룹 트위스트 디인터리버(1232)는 동일한 그룹 내에 존재하는 LLR 값의 순서를 변경하여 동일한 그룹 내의 LLR 값을 재정렬할 수 있다. 한편, 송신 장치(100)에서 그룹 트위스트 동작이 수행되지 않는 경우 그룹 트위스트 디인터리버(1232)는 생략될 수 있다.
- [0482] 그룹 디인터리버(1233, 혹은 그룹-와이즈 디인터리버)는 그룹 트위스트 디인터리버(1232)의 출력 값을 디인터리빙하고 이를 패리티 디인터리버(1234)로 출력한다.
- [0483] 구체적으로, 그룹 디인터리버(1233)는 송신 장치(100)에 구비된 그룹 인터리버(122)에 대응되는 구성요소로, 그룹 인터리버(122)에서 수행된 인터리빙 동작을 역으로 수행할 수 있다.
- [0484] 즉, 그룹 디인터리버(1233)는 복수의 비트 그룹의 순서를 비트 그룹 단위로 재정렬할 수 있다. 이 경우, 그룹 디인터리버(1233)는 LDPC 부호어의 길이, 변조 방식 및 부호율에 따라 표 15 내지 표 27과 같이 정의된 인터리빙 방식을 역으로 적용하여 복수의 비트 그룹의 순서를 비트 그룹 단위로 재정렬할 수 있다.
- [0485] 패리티 디인터리버(1234)는 그룹 디인터리버(1233)의 출력 값에 대해 패리티 디인터리빙을 수행하고, 이를 복호화부(1240)으로 출력한다.
- [0486] 구체적으로, 패리티 디인터리버(1234)는 송신 장치(100)에 구비된 패리티 인터리버(121)에 대응되는 구성요소로, 패리티 인터리버(121)에서 수행된 인터리빙 동작을 역으로 수행할 수 있다. 즉, 패리티 디인터리버(1234)는 그룹 디인터리버(1233)에서 출력되는 LLR 값들 중 패리티 비트들에 대응되는 LLR 값을 디인터리빙할 수 있다. 이 경우, 패리티 디인터리버(1234)는 수학적 18의 패리티 인터리빙 방식의 역으로 패리티 비트들에 대응되는 LLR 값을 디인터리빙할 수 있다.
- [0487] 다만, 패리티 디인터리버(1234)는 복호화부(1240)의 복호 방법 및 구현에 따라 생략될 수 있다.
- [0488] 한편, 도 16의 디인터리버(1230)는 도 17과 같이 3개 또는 4개의 구성요소로 구성될 수도 있지만, 구성요소들의 동작을 하나의 구성요소로 수행될 수도 있다. 예를 들어, 비트 그룹들 X_a, X_b, X_c, X_d 에 대하여 각각의 비트 그룹들에 속하는 하나의 비트들이 하나의 변조 심볼을 구성할 경우, 디인터리버(1230)에서는 수신된 하나의 변조 심볼을 기반으로 비트 그룹들에 대응되는 위치로 디인터리빙할 수 있다.
- [0489] 예를 들어, 부호율이 7/15이며 변조 방식이 256-QAM인 경우, 그룹 디인터리버(1233)는 표 16을 기초로 디인터리빙을 수행할 수 있다.
- [0490] 이 경우, 비트 그룹 $X_{13}, X_{15}, X_{30}, X_6, X_7, X_{37}, X_{38}, X_2$ 각각에 포함된 하나의 비트씩이 하나의 변조 심볼을 구성하게 된다. 비트 그룹 $X_{13}, X_{15}, X_{30}, X_6, X_7, X_{37}, X_{38}, X_2$ 에서 각각 한 비트씩이 하나의 변조 심볼을 구성하게 되므로, 디인터리버(1230)는 수신된 하나의 변조 심볼을 기반으로 비트 그룹 $X_{13}, X_{15}, X_{30}, X_6, X_7, X_{37}, X_{38}, X_2$ 에 대응되는 복호화 초기 값으로 매핑할 수 있다.
- [0491] 복호화부(1240)는 디인터리버(1230)의 출력 값을 이용하여 LDPC 복호화를 수행할 수 있다. 이를 위해, 복호화부(1240)는 LDPC 복호화를 수행하기 위한 LDPC 디코더(미도시)를 포함할 수 있다.
- [0492] 구체적으로, 복호화부(1240)는 송신 장치(100)의 부호화부(110)에 대응되는 구성요소로, 디인터리버(1230)에서 출력되는 LLR 값을 이용하여 LDPC 복호화를 수행하여 에러를 정정할 수 있다.
- [0493] 예를 들어, 복호화부(1240)는 합곱 알고리즘(sum-product algorithm)에 기반한 반복 복호 방식(iterative decoding)으로 LDPC 복호화를 수행할 수 있다. 여기에서, 합곱 알고리즘은 메시지 패싱 알고리즘(message passing algorithm)의 일종이며, 메시지 패싱 알고리즘이라 함은 bipartite 그래프 상에서 에지를 통해 메시지

들(가령, LLR 값)을 교환하고, 변수 노드들 혹은 검사 노드들로 입력되는 메시지들로부터 출력 메시지를 계산하여 업데이트하는 알고리즘을 나타낸다.

- [0494] 한편, 복호화부(1240)는 LDPC 복호화 시 패리티 검사 행렬을 이용할 수 있다. 이 경우, 복호화 시 이용되는 패리티 검사 행렬은 부호화부(110)에서 부호화 시 사용되는 패리티 검사 행렬과 동일한 구조를 가질 수 있으며 이에 대해서는 도 2 내지 도 4와 함께 상술한 바 있다.
- [0495] 한편, LDPC 복호화 시 이용되는 패리티 검사 행렬에 대한 정보 및 부호율 등에 대한 정보는 수신 장치(1200)에 기저장되어 있거나, 송신 장치(100)로부터 제공될 수 있다.
- [0496] 도 19는 본 발명의 일 실시 예에 따른 송신 장치의 인터리빙 방법을 설명하기 위한 흐름도이다.
- [0497] 먼저, 패리티 검사 행렬에 기초하여 LDPC 부호화를 수행하여 LDPC 부호어를 생성하고(S1410), LDPC 부호어를 인터리빙한다(S1420).
- [0498] 이후, 인터리빙된 LDPC 부호어를 변조 심볼에 맵핑한다(S1430). 이 경우, LDPC 부호어를 구성하는 복수의 비트 그룹 중 기설정된 비트 그룹에 포함된 비트를 변조 심볼 내의 기설정된 비트에 맵핑할 수 있다.
- [0499] 한편, 복수의 비트 그룹 각각은 복수의 비트 그룹 각각은 M 개의 비트로 구성되며, M 은 N_{ldpc} 와 K_{ldpc} 의 공약수이며, $Q_{ldpc}=(N_{ldpc}-K_{ldpc})/M$ 이 성립하도록 결정될 수 있다. 이 경우, Q_{ldpc} 는 상기 패리티 검사 행렬을 구성하는 정보어 부분 행렬의 열 그룹 내에서 열들에 대한 시클릭 쉬프트 파라미터 값, N_{ldpc} 는 상기 LDPC 부호어의 길이, K_{ldpc} 는 상기 LDPC 부호어를 구성하는 정보어 비트들의 길이이다.
- [0500] 또한, S1420 단계는 LDPC 부호어를 구성하는 패리티 비트들을 인터리빙하고, 패리티 인터리빙된 LDPC 부호어를 복수의 비트 그룹으로 구분하고 복수의 비트 그룹의 순서를 비트 그룹 단위로 재정렬하고, 순서가 재정렬된 복수의 비트 그룹을 인터리빙할 수 있다.
- [0501] 구체적으로, 수학적 식 21에 기초하여 복수의 비트 그룹의 순서를 비트 그룹 단위로 재정렬할 수 있다.
- [0502] 이 경우, 수학적 식 21에서 $\pi(j)$ 는 LDPC 부호어의 길이, 변조 방식 및 부호율 중 적어도 하나에 기초하여 결정될 수 있다.
- [0503] 일 예로, $\pi(j)$ 는 LDPC 부호어의 길이가 16200이고, 변조 방식이 256-QAM이고, 부호율이 7/15인 경우, 표 16과 같이 정의될 수 있다.
- [0504] 다른 예로, $\pi(j)$ 는 LDPC 부호어의 길이가 16200이고, 변조 방식이 256-QAM이고, 부호율이 9/15인 경우, 표 17과 같이 정의될 수 있다.
- [0505] 또 다른 예로, $\pi(j)$ 는 LDPC 부호어의 길이가 16200이고, 변조 방식이 256-QAM이고, 부호율이 13/15인 경우, 표 19와 같이 정의될 수 있다.
- [0506] 한편, S1420 단계는 LDPC 부호어를 복수의 비트 그룹으로 구분하고 복수의 비트 그룹의 순서를 비트 그룹 단위로 재정렬하고, 순서가 재정렬된 복수의 비트 그룹을 인터리빙할 수 있다.
- [0507] 구체적으로, 수학적 식 21에 기초하여 복수의 비트 그룹의 순서를 비트 그룹 단위로 재정렬할 수 있다.
- [0508] 이 경우, 수학적 식 21에서 $\pi(j)$ 는 LDPC 부호어의 길이, 변조 방식 및 부호율 중 적어도 하나에 기초하여 결정될 수 있다.
- [0509] 일 예로, $\pi(j)$ 는 LDPC 부호어의 길이가 16200이고 변조 방식이 256-QAM이고 부호율이 5/15인 경우, 표 15와 같이 정의될 수 있다.
- [0510] 다만, 이는 일 예일 뿐이며, $\pi(j)$ 는 상술한 표 15 내지 표 27과 같이 정의될 수 있다.
- [0511] 한편, 순서가 재정렬된 복수의 비트 그룹을 인터리빙하는 경우, 복수의 비트 그룹을 비트 그룹 단위로 복수의 열 각각에 열 방향으로 라이트하고, 복수의 비트 그룹이 비트 그룹 단위로 라이트된 복수의 열의 각 행을 행 방향으로 리드하여 인터리빙을 수행할 수 있다.
- [0512] 이 경우, 복수의 비트 그룹 중 복수의 열 각각에 비트 그룹 단위로 라이트 가능한 적어도 일부의 비트 그룹을 복수의 열 각각에 순차적으로 라이트한 후, 복수의 열 각각에서 적어도 일부의 비트 그룹이 비트 그룹 단위로 라이트 되고 남은 나머지 영역에 나머지 비트 그룹을 분할하여 라이트할 수 있다.

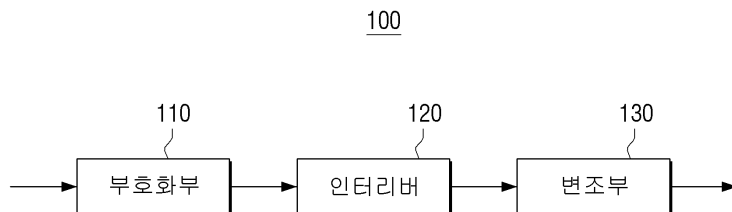
- [0513] 한편, 본 발명에 따른 인터리빙 방법을 순차적으로 수행하는 프로그램이 저장된 비일시적 판독 가능 매체(non-transitory computer readable medium)가 제공될 수 있다.
- [0514] 비일시적 판독 가능 매체란 레지스터, 캐쉬, 메모리 등과 같이 짧은 순간 동안 데이터를 저장하는 매체가 아니라 반영구적으로 데이터를 저장하며, 기기에 의해 판독(reading)이 가능한 매체를 의미한다. 구체적으로는, 다양한 어플리케이션 또는 프로그램들은 CD, DVD, 하드 디스크, 블루레이 디스크, USB, 메모리카드, ROM 등과 같은 비일시적 판독 가능 매체에 저장되어 제공될 수 있다.
- [0515] 또한, 송신 장치 및 수신 장치에 대해 도시한 상술한 블록도에서는 버스(bus)를 미도시하였으나, 각 장치에서 각 구성요소 간의 통신은 버스를 통해 이루어질 수도 있다. 또한, 각 장치에는 상술한 다양한 단계를 수행하는 CPU, 마이크로 프로세서 등과 같은 프로세서가 더 포함될 수도 있다.
- [0516] 또한, 이상에서는 본 발명의 바람직한 실시 예에 대하여 도시하고 설명하였지만, 본 발명은 상술한 특성의 실시 예에 한정되지 아니하며, 청구범위에서 청구하는 본 발명의 요지를 벗어남이 없이 당해 발명이 속하는 기술분야에서 통상의 지식을 가진자에 의해 다양한 변형실시가 가능한 것은 물론이고, 이러한 변형실시들은 본 발명의 기술적 사상이나 전망으로부터 개별적으로 이해되어져서는 안될 것이다.

부호의 설명

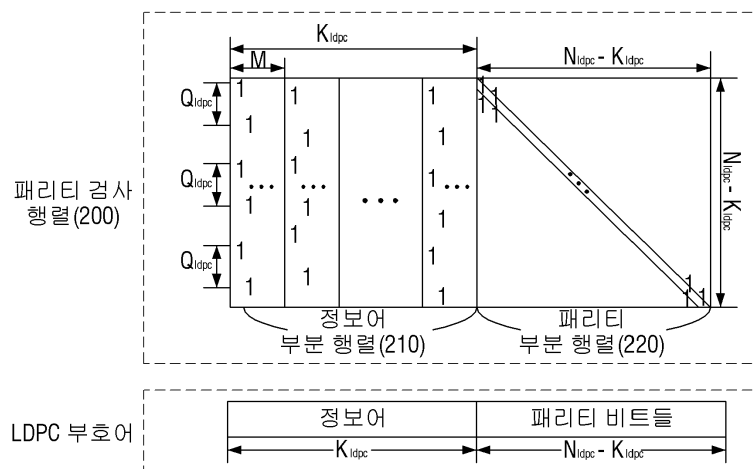
- [0517] 100 : 송신 장치 110 : 부호화부
120 : 인터리버 130 : 변조부

도면

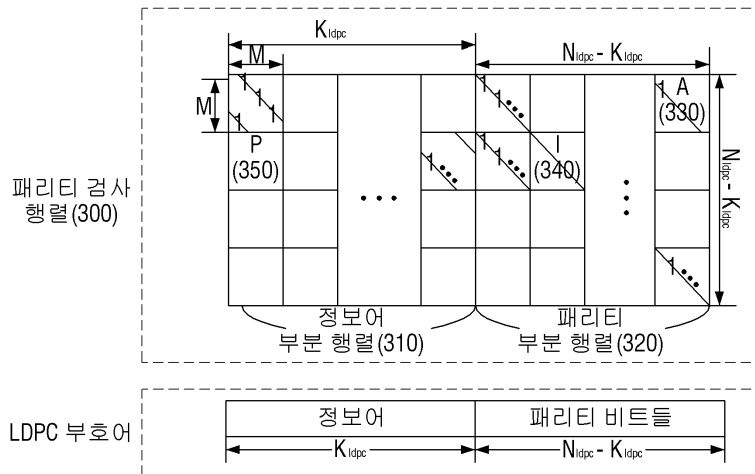
도면1



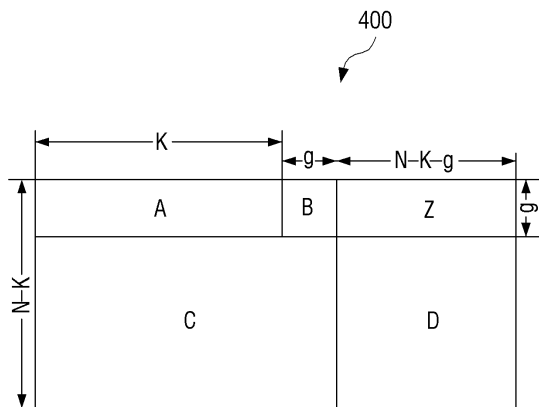
도면2



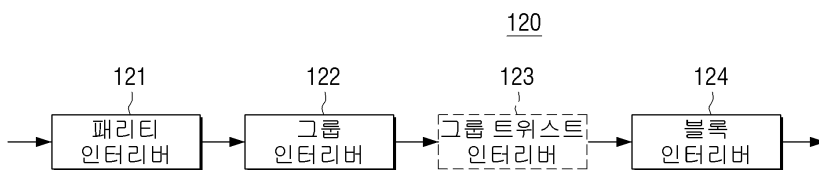
도면3



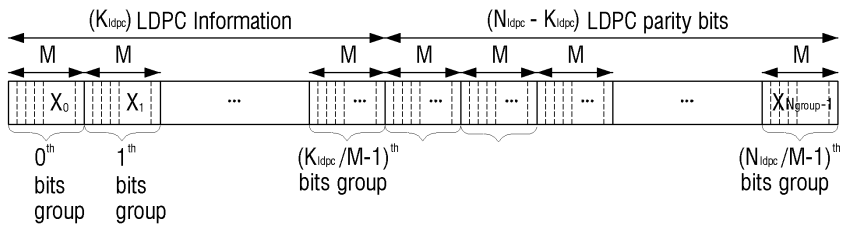
도면4



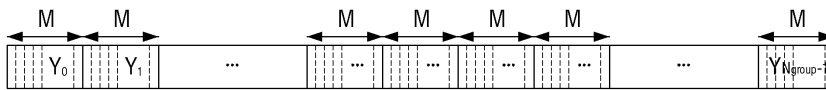
도면5



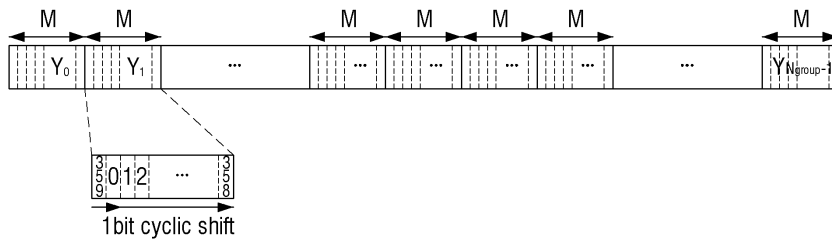
도면6



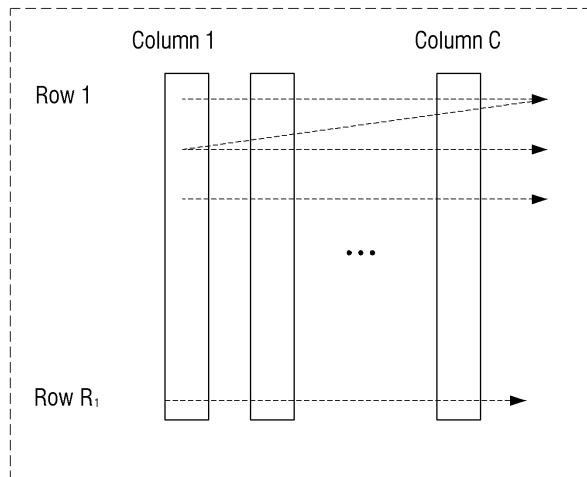
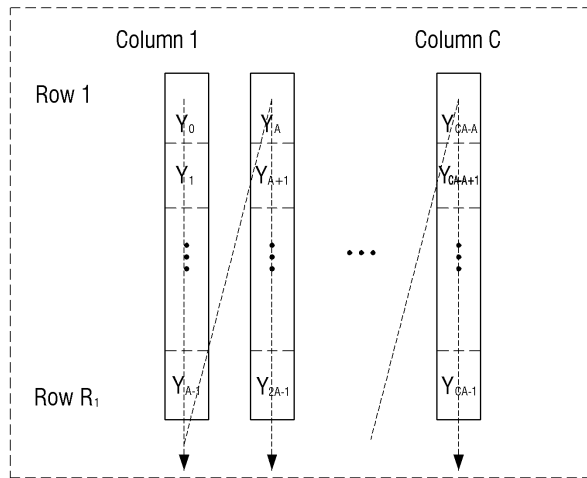
도면7



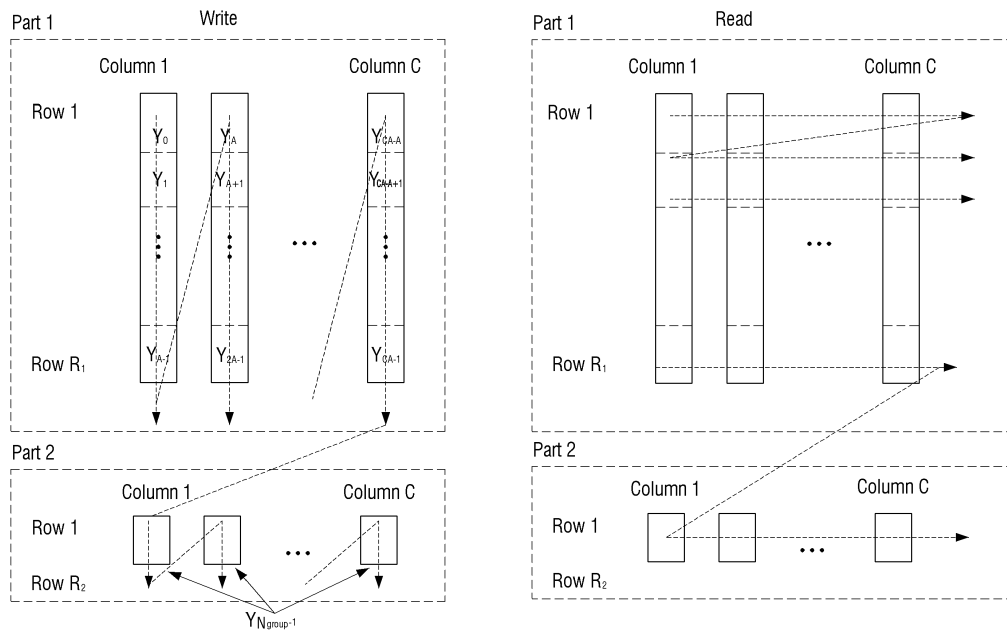
도면8



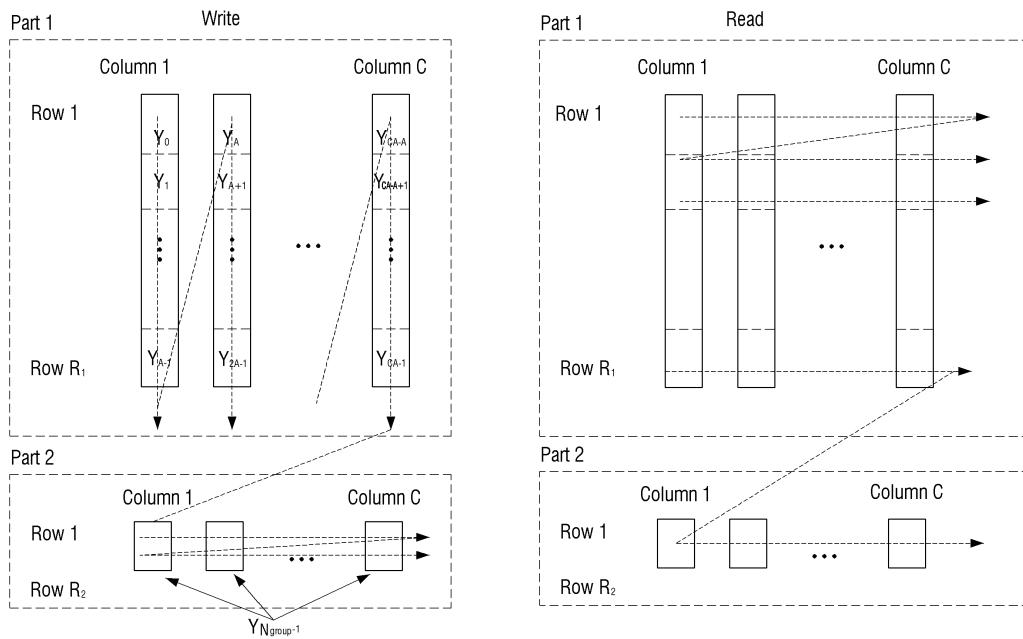
도면9



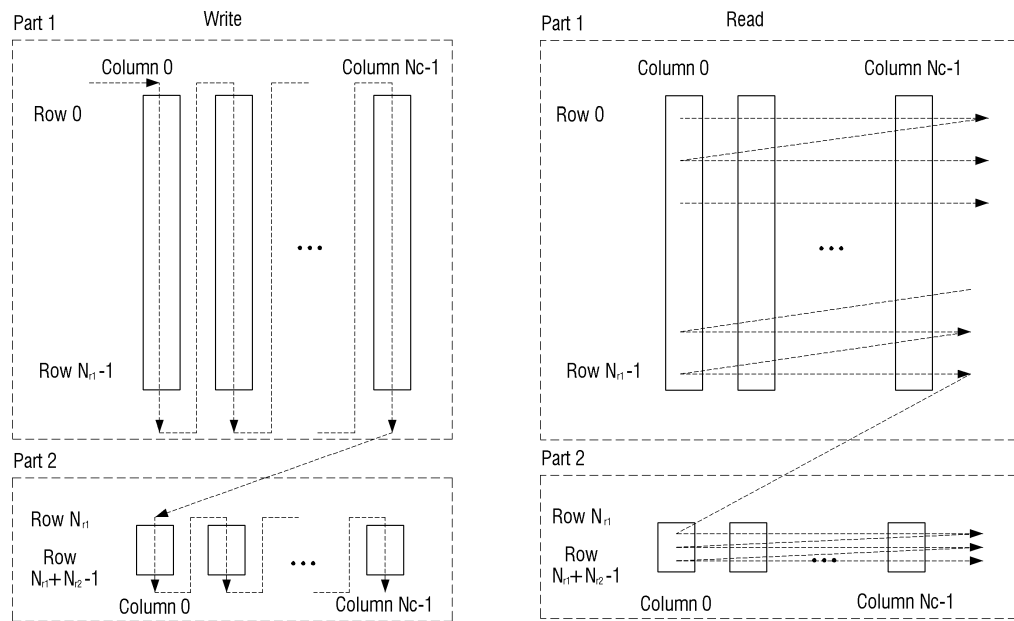
도면10



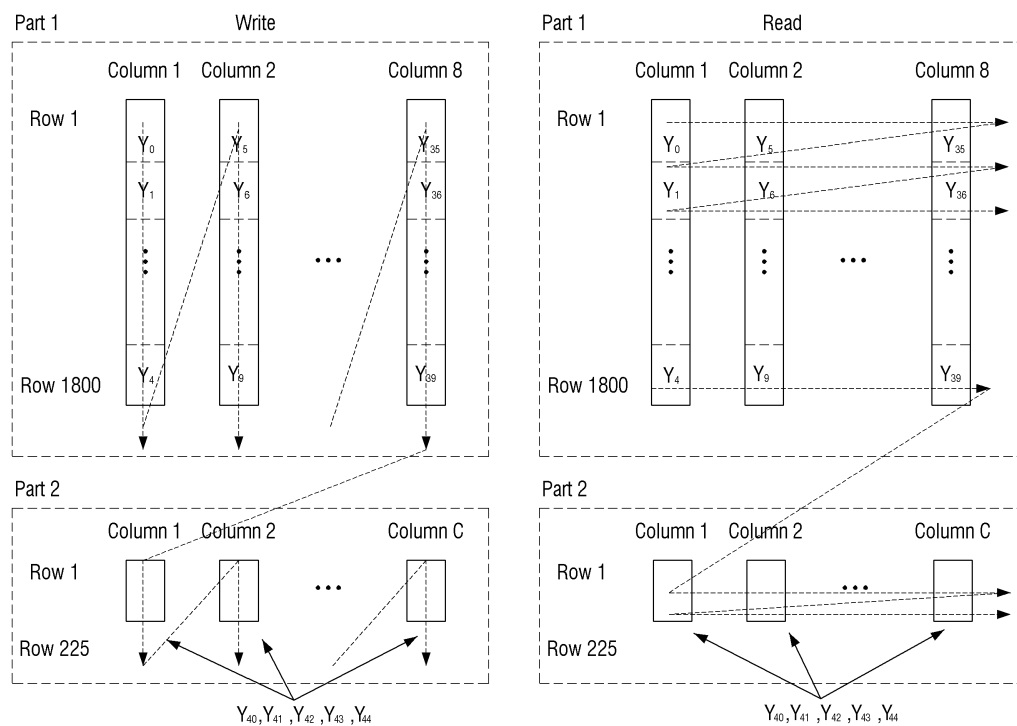
도면11



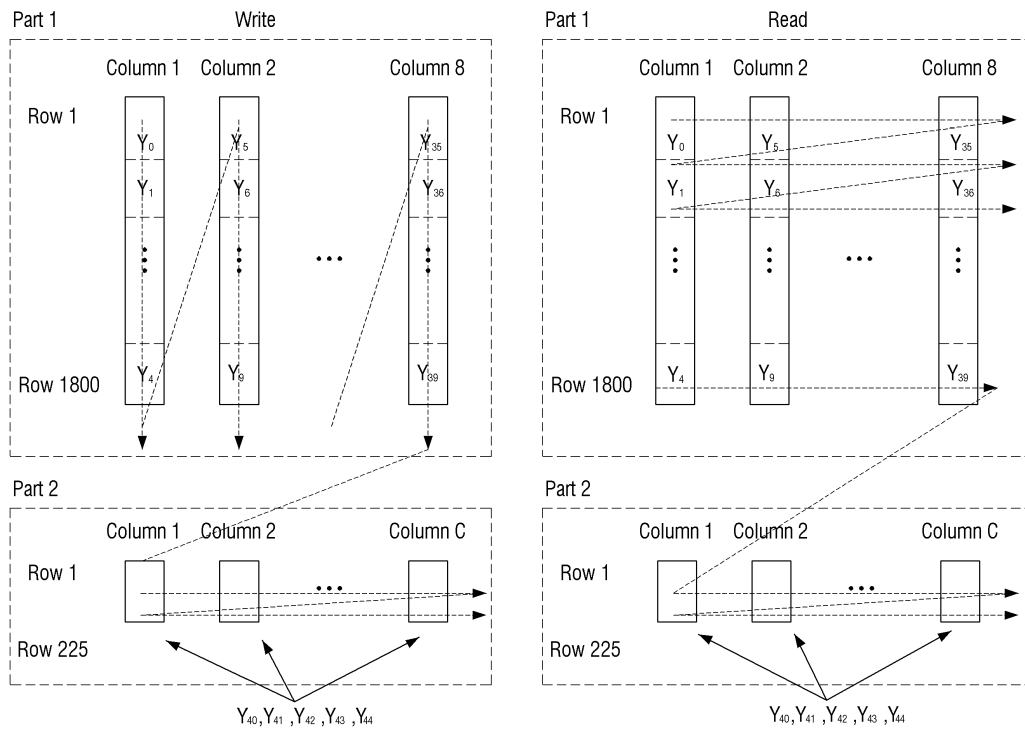
도면12



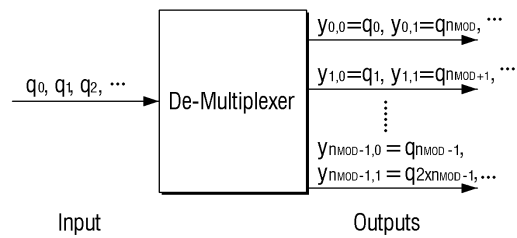
도면13



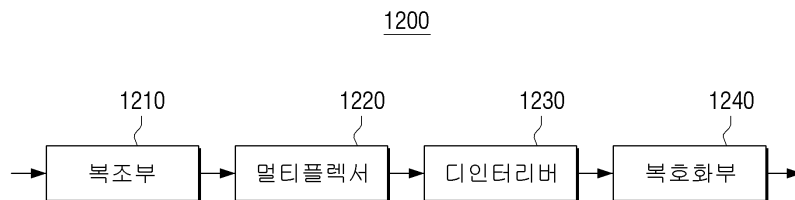
도면14



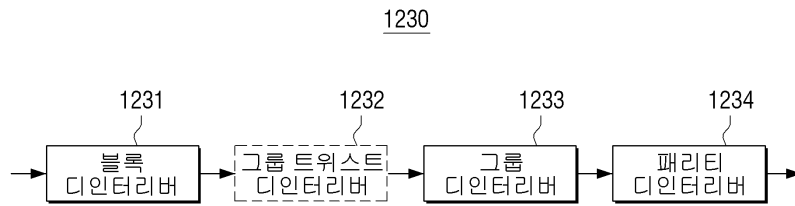
도면15



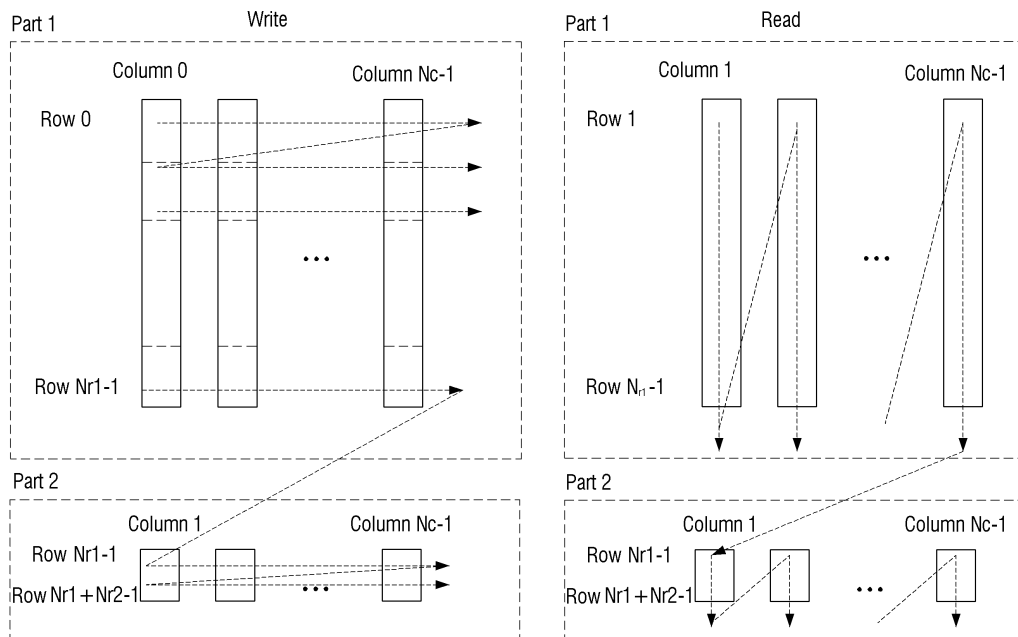
도면16



도면17



도면18



도면19

