

DC억압능력을 갖는 변조방식과 DC억압 성능비교에 관한 연구

심재성\*, 원용광\*, 서중업\*, 박인식\*, 신동호\*  
 (\* 삼성전자 중앙연구소 광메카Lab.)

A Study of Coding Scheme for DC-free RLL Sequence and a Comparison of its DC Suppression Performance

Shim Jae-Seong\*, Won Yong-Kwang\*, Seo Joong-Eon\*, Park In-Sik\*, Shin Dong-Ho\*  
 (\* Samsung Electronics Coporate R&D Center Opto-mechatronics Lab.)

**Abstract** - 광기록 저장매체의 기록밀도가 더욱 올라가면서 현재 콤팩트디스크나 DVD(Digital Versatile Disc)에 사용되고 있는  $d=2$ 인 코드보다는 검출성능이 좋은  $d=1$ 인 코드에 대한 관심이 점점 높아지고 있다. 광기록 저장매체에 사용할 수 있는 코드가 갖추어야 할 중요한 제한조건으로는 코드열의 전력밀도가 저주파에서 충분히 억압될 수 있어야 한다는 것이다. 광자기디스크나 하드디스크에서 사용된 적이 있는 (1,7)코드는 코드자체로서 DC억압능력을 갖고있지 않다. DC를 억압시키기 위해 보통 부가비트를 일정간격으로 삽입하거나 싱크를 이용하게 되는데 이와 같은 방법으로는 충분한 DC억압이 어렵거나 기록밀도를 올리는데 불리한 원인이 된다.

본 논문에서는 코드열 자체로서 DC억압능력을 갖고 있는 코드워드의 생성방식과 변조 알고리즘을 RLL(1,8)인 경우를 예로 제시하였다. 또한 본 논문에서 제시한 코드의 성능에 대한 검증으로 기존의 RLL(1,8)코드[1]와 본 논문에서 제시한 코드에 4%이하의 부가비트를 삽입하여 DC억압을 수행한 뒤 동일한 기록밀도에서 DC억압성능을 비교하였을 때 저주파 대역에서 15dB이상의 개선효과가 있었음을 확인하였다.

1. 서 론

일반적으로 광디스크에서 채용되고 있는 코드는 최소, 최대 구속장이 제한을 갖는 RLL(Run Length Limited) 코드이다. 여기서 최소 구속장을 결정하는 파라미터  $d$ 는 광디스크상의 마크(또는 피트)를 인식할 수 있는 검출성능의 정도(精度)와 코드의 기록밀도를 결정한다. 즉  $d$ 의 값이 커지면 코드의 기록밀도는 향상되지만 마크(또는 피트)를 예러 없이 인식하기 위해 高精度의 검출기가 요구되어진다. 현실적인 재생성능을 유지하면서 기록밀도를 최대로 할 수 있는  $d$ 의 값은 2로 보고되었고<sup>[2]</sup> 콤팩트디스크와 DVD(Digital Versatile Disc)에 사용되고 있는 EFM(Eight Fourteen Modulation)과 EFMPlus코드<sup>(3,4)</sup>도  $d=2$ 를 따르고 있다. 그러나 광기록 저장매체의 기록밀도가 더욱 올라가면서 기존의 검출기로는 만족할만한 신호의 검출성능을 달성하기가 어려워졌고 최근에는 이미 HDD등에 그 성능을 인정받아 널리 사용되고 있는 PRML(Partial Response Maximum Likelihood)기법의 사용을 검토하면서  $d=2$ 인 코드보다는 검출성능이 좋은  $d=1$ 인 코드에 대한 관심이 점점 높아지고 있다.

광기록 저장매체에 사용할 수 있는 코드가 갖추어야 할 또 하나의 특징으로는 코드열의 전력밀도가 저주파에서 충분히 억압할 수 있어야 한다. 디스크에 형성되어 있는 피트 또는 마크는 광검입 장치를 통해 읽혀지면서 피트열 또는 마크열로부터 원치 않는 저주파성분이 발생하는데 이 저주파대역은 서보에서는 주요한 제어 대역이 될 수 있다. 코드열에 저주파성분이 많이 포함되어 있다면 서보

계에서는 그만큼 마진확보라는 차원에서는 불리해지며 따라서 코드의 저주파억압능력은 코드의 성능을 평가하는데 중요한 요소이다.

본 논문에서는 코드열 자체로서 DC억압능력을 갖고 있는 코드워드생성방식과 변조 알고리즘을 RLL(1,8)인 경우를 예로 제시하고 또한 성능에 대한 검증으로 기존의 RLL(1,8)코드와 동일한 기록밀도에서 DC억압성능을 비교한다.

2. 본 론

2.1 코드워드 생성방식

본 연구에서 사용할 코드워드의 생성을 위해 최소, 최대 구속장  $d, k$ 를 각각 1과 8로 하였고 코드워드의 비트 수는 12로 하여 0부터  $2^{12}-1$ 로 생성되는 코드워드중  $(d,k)$ 를 위반하는 코드워드는 모두 버리고  $(d,k)$ 조건을 만족하는 코드워드는 끝 부분의 연속하는 0의 수가 1~3인 코드워드는 하나 더 중복시켜 생성하였다. 단 주코드 그룹의 MCG1에 속하면서 8이나 9로 시작하는 코드워드 이면서 끝 부분의 연속하는 0의 수가 1인 코드워드는 중복시키지 않았다. 표1은 이상의 조건으로 생성된 코드워드의 수를 나타낸다. 생성된 코드워드는 그가 갖는 특징에 따라 총 5개의 코드그룹으로 나뉘고 5개의 코드그룹은 각각 두개의 주코드그룹(MCG1, MCG2)과 중복코드의 복조를 위한 두개의 판단그룹(DCG1, DCG2)과 하나의 DSV코드그룹으로 나뉜다. 여기서 두개의 주코드그룹 MCG1과 MCG2에는 동일한 코드워드가 존재하지 않으며 두개의 판단그룹 DCG1, DCG2에도 마찬가지로 동일한 코드워드는 존재하지 않는다. 주코드그룹은 중복되지 않은 코드워드의 다음에 오는 코드워드가 속해있는 코드그룹이며 판단그룹은 중복된 코드워드의 다음에 오는 코드워드가 속해있는 코드그룹을 의미한다. DSV 코드그룹은 코드열의 DC억압을 위한 별도의 코드그룹이며 여기에 속한 70개의 코드워드는 모두 010로 시작하며 주코드그룹 MCG1에 속해있다. 주코드그룹과 판단그룹은 각각 256개의 코드워드를 가져야 한다. MCG1은 70개의 코드워드가 부족하고 MCG2에는 58개의 코드워드가 부족하다. 이 부족한 코드워드의 수는 모두 128개이며 이들 코드워드는 01로 시작하는 코드워드도 각각 보충한다. 따라서 DSV코드그룹을 제외하고 DCG2에 있는 코드워드중 한개의 코드워드를 제거하면 모든 코드그룹이 256개의 코드워드를 갖게 된다.

2.2 변조방식

소스워드에서 코드워드의 변조는 코드워드가 갖는 NCG(Next Code Group)와 코드열의 경로선택 알고리즘에 따라 이루어진다. 모든 코드워드는 그 코드워드 다음에 올 수 있는 다음 코드워드들의 코드그룹을 지정한다. 이를 NCG라 부르며 표2는 NCG의 판단 방법과

표1. 사용가능한 코드워드의 수

코드그룹	MCG1	MCG2	DCG1	DCG2	MCG1/MCG2	DSV코드그룹
코드워드의 특징	10으로 시작하는 코드워드	001~00000001으로 시작하는 코드워드	10, 00001 또는 000001로 시작하는 코드워드	01, 001 또는 0001로 시작하는 코드워드	01로 시작하는 코드워드	01로 시작하는 코드워드중 MCG1에 속할 코드워드
코드워드 수	186	198	256	257	128	70

표2. NCG의 판단 방법과 NCG가 지시하는 코드그룹

코드워드	NCG	다음 코드워드가 속해 있는 코드그룹
01로 끝나는 코드워드나 10으로 끝나면서 중복시키지 않은 코드워드	2	MCG2 또는 DSV 코드그룹*
10, 100 또는 1000으로 끝나는 중복된 코드워드중 첫번째 코드워드	3	DCG1
10, 100 또는 1000으로 끝나는 중복된 코드워드중 두번째 코드워드	4	DCG2
코드워드 끝부분의 연속된 0의 수가 4개 이상인 모든 코드워드	1	MCG1 또는 MCG2**

\*:70미만의 소스워드가 입력될 때 DSV코드그룹과 MCG2에 속해있는 코드워드 선택가능

\*\*:(d,k)조건을 위반하지 않는 조건에서 MCG2에 있는 코드워드도 선택가능

NCG가 지시하는 코드그룹을 나타낸다. 표2에서 알 수 있듯이 어떤 코드워드의 NCG가 1이나 2인 경우는 다음에 올 수 있는 코드워드가 두개일 가능성이 있는데 이렇게 두개의 코드워드가 올 경우가 DC억압제어가 가능한 경우이다.

두 개의 코드워드중 하나를 선택하는 방법은 feed-forward방식의 DC억압제어 알고리즘과 feed-back방식의 DC억압제어 알고리즘이 있다. Feed-back DC억압제어 알고리즘이 feed-forward제어 알고리즘보다 성능이 우수하다. 본 논문에서 사용한 코드열 선택 알고리즘은 feed-back DC억압제어 알고리즘 중에서 매 DC억압제어 시점에서 코드열의 누적 DSV(Digital Sum Value)가 최소가 되는 변조패스를 선택하지 않고 판단시점을 변조패스의 수에 따라서 지연해서 판단하는 multi-path DC억압제어 알고리즘이다. 그림 1은 코드열의 패스가 8인 경우의 multi-path DC억압제어방식이다. 그림 1에서 보면 multi-path DC억압제어 알고리즘은 매 DC억압제어가 가능한 순간마다 변조패스를 선택하지 않고 8패스에 모두 DC억압제어가 가능한 코드가 존재하는 경우에 한해서 누적 DSV가 최소가 되는 변조패스를 선택한다.

2.3 실험 결과

성능의 비교실험을 위해 본 연구에서는 비교대상으로서 종래의 RLL(1,8,2,3)코드를 선택하였다. 동일한 조건에서의 실험을 위해 0~4%의 부가비트를 본 논문에서 제시한 RLL(1,8,8,12)코드와 함께 추가하여 실험하였는데 부가비트가 포함될 때의 코드워드의 코드그룹은 DSV코드그룹은 사용하지 않았다. 즉 부가비트가 삽입될 때 NCG가 2인 경우 다음 코드워드가 속해 있는 코드그룹은 MCG2와 MCG1중에서 (d,k)를 위반하지 않는다는 조건에서 모두 사용 가능하다.

그림 2는 본 논문의 RLL(1,8,8,12)코드에 대한 multi-path DC억압제어방식의 효과를 보여주는 그림이다. 2-path를 사용할 때보다 4-path, 8-path를 사용할 때가 약1~2dB개선되었음을 알 수 있다. 그림 3은 부가비트를 사용하지 않았을 때의 본 논문의 RLL(1,8,8,12)코드의 psd

곡선으로 코드자체가 DC억압능력이 있음을 알 수 있다. 그림 4는 부가비트를 0~4%까지 삽입하였을 때 user bit rate에 정규화 된 10-3지점에서의 psd값을 종래의 RLL(1,8,2,3)코드와 비교한 그림이다. 전체적으로 본 논문에서 제시한 RLL(1,8,8,12)코드가 1~5dB 낮은 psd값을 나타내었고 특히 부가율이 낮을 수록 격차가 크게 나타났다.

3. 결 론

본 논문에서 제시한 RLL(1,8,8,12)코드는 코드열 자체가 부가비트없이도 DC억압능력이 있으며 0~4%의 부가비트를 동일하게 추가하였을 때 1~5dB정도 DC억압능력이 RLL(1,8,2,3)코드보다 우수함을 알 수 있었다. 코드의 에러전파율은 부가비트가 없었을 때 본 논문의 코드는 약 19~20%이고 RLL(1,8,2,3)코드는 18~19%정도로 1%정도 나쁘게 나타났으나 2%의 부가비트를 추가한 경우는 본 논문에서 제시한 RLL(1,8,8,12) 코드가 약 17%로 에러전파율이 떨어져 RLL(1,8,2,3)코드와 크게 차이가 없었음을 알 수 있었다.

(참 고 문 헌)

- [1] Josephus A. H. M. Kahlman, Kornelis A. Schouhamer Immink, "Device for encoding/decoding N-bit source words into corresponding M-bit channel words, and vice versa", United States Patent, Patent Number 5,477,222.
- [2] W.D. Huber, "Selection of modulation code parameters for maximum lineal density", *IEEE Trans. Magnet.*, vol. 16, no. 5, pp. 637-639, Sept. 1980.
- [3] DVD규격집 Part1 ver. 1.0 Phy-69 ~ Phy-79.
- [4] Kees A. Schouhamer Immink, "DVD의 기록용량을 높める 기록符號化方式EFMPPlus", *Nikkei Electronics*, no. 675, pp.161-165, Nov. 1996.

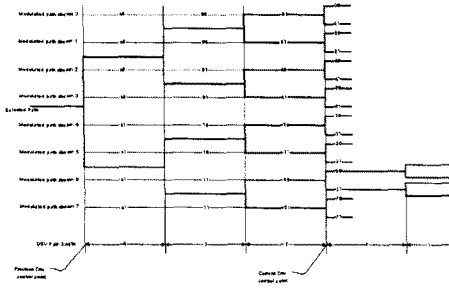


그림 1. Multi-path DC억압제어 방법에 따른 코드열 경로 선택

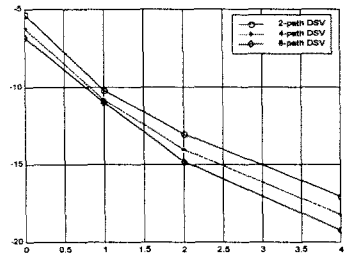


그림 2. Multi-path DC억압제어의 효과

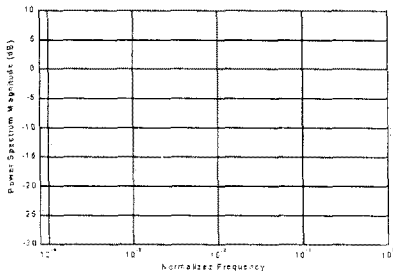


그림 3. 본 논문에서 제시한 RLL(1,8,8,12) 코드의 psd곡선(부가비트가 없을 때)

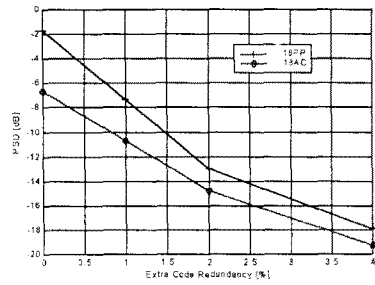


그림 4.  $10^{-3}$ 지점에서의 psd값의 비교  
(위:본 논문에서 제시한 RLL(1,8,8,12)코드  
아래:기존의 RLL(1,8,2,3)코드)