고효율의 멀티모드 데이터 변조방법

김 진 한, 심 재 성, 정 규 해
삼성전자 디지털미디어 연구소

The Coding Method with Multi-mode Technique

Jin-Han Kim, Jae-Seong Shim, Kiu-Hae Jung
Digital Media R&D Center, Samsung Electronics Co., LTD
kjhdm@samsung.com

Abstract

DC억압능력이 없거나 부족한 코드에 만족할만한 DC억압능력을 갖도록 하는 방법은 DC 제어 비트의 사용, Dual Code의 사용, Multimode Code의 사용 등이 있다. 어떤 방법이든 부가 비트가 사용되지만, 그 중에서 멀티모드 변조코드는 우수한 DC억압능력과 높은 코드효율을 갖고 있음에도 불구하고 복잡한 하드웨어와 높은 에러 전파율을 갖는 단점도 있다. 본 논문에서 제시하는 멀티모드 변조코드의 특징은 데이터열의 다중화를 위해 의사 스크램블 기법을, 다중화된 데이터열의 변조를 위해서는 DC-free RLL 변조코드를 사용한다. 의사 스크램블에 의한 데이터열의 다중화는 데이터를 복조할 때 에러전파 확률을 떨어뜨리는 효과가 있고 다중화된 데이터열의 변조를 위한 DC-free RLL 변조코드의 사용은 DC억압능력을 향상시키고 하드웨어가 훨씬 간단해진다.

1. 서론

데이터가 전송선을 통해서 전송되거나 기록 매체 상에 기록될 때, 이 데이터가 전송 또는 기록 전에 전송선 또는 기록 매체의 특성에 맞는 코드로 변조된다. 그런데, 변조에서 생긴 코드가 직류 성분을 포함하게 되면 디스크 드라이버의 서보 제어시 발생된 트랙킹 오류와 같은 다양한 오류 신호가 변화하기 쉬어지거나 지터(jitter)가 쉽게 발생된다. 그러므로, 데이터 전송과 저장을 위한 변조 코드의 매우 중요한 특성 중의 하나는 변조된 데이터열에 대한 DC(direct current)성분의 효율적인 억압 능력이다. DC억압능력이 없거나 부족한 코드에 만족할만한 DC 억압능력을 갖게 하는 방법은 DC 제어 비트의 사용[1], DC 제어용 보조 코드나 Dual Code와 같은 별도 코드변환표의 사용[2], Multimode Code의 사용[3],[4],[5] 등의 방법으로 Redundancy를 부가함으로써 코드워드 열의 DC 성분을 효율적으로 제어한다. 첫 번째 방법은 변조 전의 데이터열이나 변조 후의 코드 열에 DC 제어용 비트를 삽입하는 방식이고, 두 번째 방법은 DC제어를 위해 DC 제어용 보조 코드변환 표를 사용하는 방식으로서 보조 코드워드의 채널 비트수가 주 코드워드의 채널 비트 수보다 크거나 같다. 세 번째 방법은 스크램블 등의 방법을 이용하여 데이터 열을 다중화하여 변조한 다음 DC성분이 제일 작은 코드 열을 선택하는 방법으로 우수한 DC억압능력과 높은 코드효율을 갖고 있음에도 불구하고 복잡한 하드웨어와 높은 에러전파율을 갖는 단점도 있다.

이 논문에서는 코드효율을 증가시킨 멀티모드 코드 방식을 제시하며 기타 다른 코딩방식과 DC 억압성능 및 코드효율 등을 비교하였다.

2. Multimode Coding

본 논문에서는 데이터열의 다중화를 위해 의사

스크램블(Pseudo Scramble) 기법을 사용하며, 다중화된 데이터열의 변조를 위해 DC-free RLL code를 사용한다. 의사 스크램블에 의한 데이터열의 다중화는 데이터 복조시 에러전파를 감소시키고, 다중화된 데이터열의 변조를 위한 DC 억압능력이 있는 RLL 코드의 사용은 아예 DC 억압능력이 없는 코드를 사용할 경우보다 DC억압능력이 훨씬 우수하고 하드웨어가 간단해진다. 멀티모드 변조코드의 변조과정은 그림1과 같다.

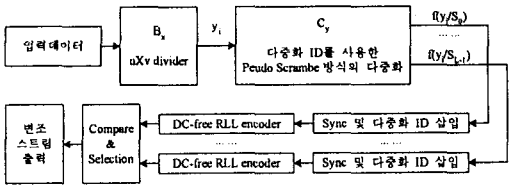

그림 1. 변조과정

입력 데이터 열을 $\underline{x}=(x_0, x_1, \ldots, x_{k-1})$로 표시하고(식1), 입력 데이터 열을 vXu(=k)로 나눈다(식2). vXu의 각각의 데이터 열에 a비트의 부가정보를 붙여서 $L=2^a$개의 데이터 열로 다중화한 다음 부가된 다중화정보 s에 따라 데이터 열을 랜덤 데이터로 변환하면, 하나의 데이터 열 y_i에 대해 2^a개로 다중화된 서로 다른 데이터가 만들어진다(식3, 4). 2^a개로 다중화된 랜덤 데이터 열에 대해 각각 Sync 및 다중화 ID를 부여하고 DC-free RLL변환을 실시하며, RLL변환된 2^a개의 스트림에 대해 DC성분이 가장 적은 데이터 열 하나를 선택한다. $f(\underline{y}_i/s)$는 다중화 정보 s를 이용하여 $\underline{y}_i$에 대해 의사 스크램블로 만든 결과를 의미한다(식4).

$$\underline{x} = (x_0, x_1, \ldots, x_{k-1}) : \text{input data stream} \qquad \text{식(1)}$$

$$\underline{B_x} = \begin{pmatrix} X_{0,0}, X_{0,1}, & \cdots, & X_{0,u-1} \\ X_{1,0}, X_{1,1}, & \cdots, & X_{1,u-1} \\ & \cdots & \\ X_{i,0}, X_{i,1}, \ldots, X_{i,j}, \ldots, & & X_{i,u-1} \\ & \cdots & \\ X_{v-1,0}, X_{v-1,1}, & \cdots, & X_{v-1,u-1} \end{pmatrix} \approx \begin{pmatrix} y_0 \\ y_1 \\ \cdots \\ y_i \\ \cdots \\ y_{v-1} \end{pmatrix} \qquad \text{식(2)}$$

$$\underline{C_y} = (\underline{C}_0, \underline{C}_1, \ldots, \underline{C}_i, \ldots, \underline{C}_{v-1}) \qquad \text{식(3)}$$

$$\underline{C_x} = \begin{pmatrix} S_0, & y^0_{i,0}, X_{i,1}, \ldots, X_{i,q-1}, y^0_{i,q}, X_{i,q+1}, \ldots, y^0_{i,2^a q}, \ldots X_{i,u-1} \\ S_1, & y^1_{i,0}, X_{i,1}, \ldots, X_{i,q-1}, y^1_{i,q}, X_{i,q+1}, \ldots, y^1_{i,2^a q}, \ldots X_{i,u-1} \\ & \cdots \\ S_{L-1}, & y^{L-1}_{i,0}, X_{i,1}, \ldots, X_{i,q-1}, y^{L-1}_{i,q}, X_{i,q+1}, \ldots, y^{L-1}_{i,2^a q}, \ldots X_{i,u-1} \end{pmatrix} = \begin{pmatrix} f(\underline{y}/S_0) \\ f(\underline{y}/S_1) \\ \cdots \\ f(\underline{y}/S_{L-1}) \end{pmatrix} \qquad \text{식(4)}$$

여기서 $x_{i,j}=x_{i\times u+j}$ u-1≠ q의 배수,

p=0, 1, .., r, r=(u-1)/q의 몫

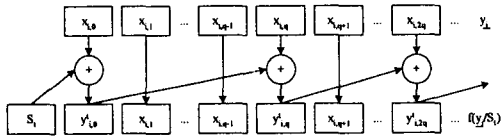

그림 2. 의사 스크램블

그림2의 방법으로 입력 데이터를 의사 스크램블로 변환할 경우에 XOR연산의 주기 q는 DC억압성능을 수용하는 수준에서 선택하면 된다. 그림3은 그림2의 방법으로 변환된 데이터를 재생할 경우의 에러전파 특징을 보여준다. 의사 스크램블로 만들어진 데이터를 RLL 변조한 후 광디스크와 같은 매체에 저장된 데이터 스트림을 재생할 때 데이터 스트림 $f(y_i/s_i)$에서 XOR연산된 $y^{*i}_{i,q}$에서 에러가 발생하면 복원된 $x^*_{i,q}$와 $x^*_{i,2q}$에 에러가 발생하여 에러가 전파된다. 그러나 XOR연산이 이루어지지 않은 $f(y_i/s_i)$에서는 에러가 발생하여도 에러전파는 발생하지 않는다. XOR연산의 주기 q가 커지면 에러전파확률은 떨어지나 DC억압성능도 저하되는 특징이 있으며 반대로 q가 작아지면 DC억압 성능에는 유리하나 에러전파확률이 증가한다.

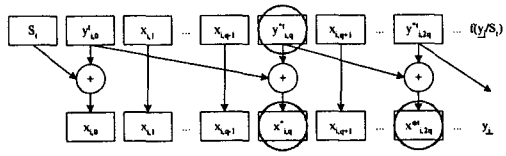

그림 3. 의사 스크램블의 에러 전파 특성

다중화 정보 S_i를 u 길이의 입력 데이터 열을 다중화하는 정보라고 할 때, 그림2는 S_i의 비트 수 a가 입력 데이터의 비트 수 m보다 작거나 같을 경우에 적용된다. a<m인 경우는 입력데이터의 일부

비트(LSB로부터 a비트 또는 MSB로부터 a비트 또는 m비트의 데이터 내에 임의의 a비트)를 이용해도 m비트 모두를 이용해서 의사 랜덤 데이터로 만드는 경우와 거의 동일한 성능을 보여 준다.

의사 스크램블 기법으로 다중화된 데이터 열을 변조하기 위해 사용하는 DC-free RLL 변조코드는 주 변조코드와 DC제어용 보조 변조코드로 이루어진다. 코드의 최소 런(Minimum Run Length Limit)을 d, 최대 런(Maximum Run Length Limit)을 k, 소스 데이터의 비트 수를 m, 변조후의 코드워드의 비트 수를 n이라 하고 코드워드의 LSB로부터 MSB방향으로 연속하는 0의 수를 EZ(End Zero), MSB로부터 LSB방향으로 연속하는 0의 수를 LZ(Lead Zero)라고 하면 일정한 범위 내에 있는 EZ의 코드워드를 LZ의 기준에 따라 분류할 수 있다. m=8인 소스 데이터의 변조를 위해서는 코드워드의 수가 최소 256개 이상이어야 한다. LZ의 기준에 따라 분류한 코드워드의 수를 조정하여 각각 256새의 코드워드를 갖는 주 코드 변환 표를 생성한다. DC제어용 보조 변조 코드는 주 변조 코드를 생성하는데 사용한 EZ 및 LZ의 범위를 변화시켜서 생성한 코드워드와 주 변조 코드 표에서 남는 코드워드를 합쳐 DC제어용 보조 코드 변환 표를 생성한다.

변조된 코드워드 열에서 그림4와 같이 코드워드 a와 코드워드 b가 연결되는 지점에서도 런 길이 (d, k)조건을 만족해야 한다. 또한 코드워드 b는 이전 코드워드 a의 EZ에 따라 그 코드워드 b가 속해 있는 코드그룹이 정해지게 되는데 주 코드 변환 표와 DC제어용 보조 코드 변환 표에서 코드워드수가 부족하여 다른 코드 변환 표에서 코드워드를 빌려온 코드그룹이 지정되는 경우에 (d,k)조건을 만족하지 못하는 경우가 발생한다. 표 1에 $d \le EZ_a+LZ_b \le k$를 위반하는 경우를 예시하였으며 이 경우에는 코드워드 a의 EZ가 바뀌게 되는데 이를 경계규칙이라 한다.

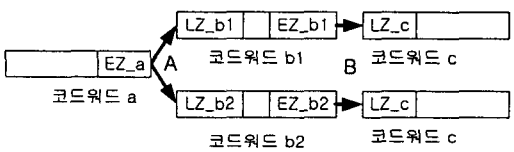

그림 4. 코드워드의 경계

표 1. 경계규칙의 예(d=1, k=7, m=8, n=12, EZ_a=0, LZ_b=0)

코드워드 a	코드워드 b	패리티 변화
xxxxxxxx0101 → xxxxxxxx0100	1010xxxxxxxx	있음
xxxxxxxx1001 → xxxxxxxx1010		없음
~		
xxx10000001 → xxx10000010		

주 코드 변환 표와 DC제어용 보조 코드 변환 표 내의 동일한 소스 데이터에 대한 한 쌍의 코드워드는 서로 패리티가 반대이며 동일한 다음 상태 값을 갖도록 배치하였다. 이는 DC제어가 가능한 시점에서 한 쌍의 코드워드열의 RDS가 서로 반대인 방향이 되어 반드시 DC제어가 가능하도록 하기 위한 고려이다. 그러나 본 논문에서 고안한 변조코드 규칙에는 경계규칙이 존재한다. 경계규칙에 따라 코드워드가 바뀌게 되고 코드워드열의 패리티가 변할 가능성이 있다. 이러한 변화가 DC제어가 가능한 시점에서 발생하면 한 쌍의 채널 코드워드열의 패리티를 서로 반대가 되도록 하는 기본 규칙에 어긋나게 된다. 따라서, 101xxxxxxxxx, xxxxxxxxx101, 101xxxxxx101 형태의 코드워드는 동일한 다음 상태 값에 동일한 소스 데이터가 되도록 배치하여야 한다. 위의 특징들과 표 2과 표 3의 EZ 및 LZ에 의해 코드워드를 분류하여 본 논문에 적용된 DC-free RLL(1,7) 변조코드를 생성하였다.

표 2. d=1, k=7, m=8, n=12, $0 \le EZ \le 5$인 주 코드워드의 분류

	LZ (EZ)	개수	추가	개수	삭제	개수	합계 (잉여분)
MCG1	$1 \le LZ \le 7$ ($0 \le EZ \le 5$)	210	1010xxxxxxxx ($0 \le EZ \le 5$)	51			261(5)
MCG2	$0 \le LZ \le 4$ ($0 \le EZ \le 5$)	316			1010xxxxxxxx ($0 \le EZ \le 5$)	51	265(9)
MCG3	$0 \le LZ \le 2$ ($0 \le EZ \le 5$)	264					264(8)

표 3. DC제어용 보조 코드워드(d=1, k=7, m=8, n=12)

	LZ (EZ)	개수	추가1	개수	추가2	개수	삭제	개수	합계
ACG1	$1 \le LZ \le 7$ ($6 \le EZ \le 7$)	8	1010xxxxxxxx ($6 \le EZ \le 7$)	2	MCG1의 잉여분	5			15
ACG2	$0 \le LZ \le 6$ ($6 \le EZ \le 7$)	12	$3 \le LZ \le 6$ ($0 \le EZ \le 5$)	21	MCG2의 잉여분	9	1010xxxxxxxx ($6 \le EZ \le 7$)	-2	40
ACG3	$0 \le LZ \le 3$ ($6 \le EZ \le 7$)	10	$LZ = 3$ ($0 \le EZ \le 5$)	33	MCG3의 잉여분	8			51

3. Simulation

본 논문에서 제안한 멀티모드 변조코드를 기존의 코드와 DC억압 성능 및 코드효율 측면에서 비교하였다.

그림5은 DC-free RLL(1,7) 변조코드의 DC 억압성능과 의사 스크램블을 이용한 a=2인 멀티모드 변조코드를 DC-free RLL(1,7) 변조코드에 적용한 경우(C-Code)의 DC 억압성능을 비교한 그림으로 멀티모드 변조코드의 DC 억압성능이 10^{-4}주파수에서 약 10dB정도 개선되었다.

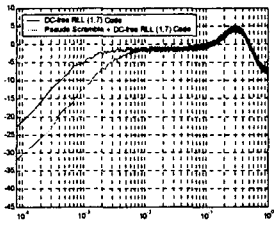

그림 5. 의사 스크램블 기법의 DC억압성능

그림6은 RLL(1,7) Parity Preserve 변조코드(A-Code)[1]와 스크램블 방식의 a=4인 멀티모드 코드에 DC억압 능력이 없는 RLL(1,7) 변조코드를 적용(B-Code)한 경우를 의사 스크램블 방식의 a=2인 멀티모드 변조코드에 DC-free RLL(1,7) 변조코드를 적용(C-Code)한 경우와 각각 DC억압 성능을 비교한 그림이다. DC억압 성능이 동일할 경우에 C-Code가 B-Code에 비해 다중화 정보 비트 수를 4비트에서 2비트로 줄일 수 있어서 하드웨어의 양을 50% 줄일 수 있다. 표4에서처럼 A-code와 B-code에 비해 C-code의 코드효율이 각각 1.3%, 0.5% 높다.

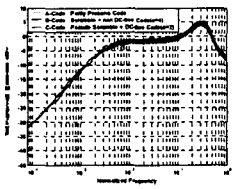

그림 6. RLL (1,7) DC 성능

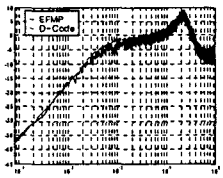

그림 7. RLL(2,10) DC 성능

그림7은 EFMP의 DC억압성능과 의사 스크램블을 이용한 a=4인 멀티모드 변조코드에 DC-free RLL (2,10) 변조코드를 적용(D-Code)한 경우의 DC억압성능을 비교한 그림이다. 표5에서 나타났듯이 동일한 DC억압성능을 가지면서도 코드효율이 높음을 알 수 있다.

표 4. RLL(1,7) 코드의 비교

Code type	R(m/n)	R/C(d,k)
A-code	0.6533	96.2%
B-code	0.6590	97.0%
C-code	0.6623	97.5%

표 5. RLL(2,10) 코드의 비교

Code type	R(m/n)	R/C(d,k)
EFMP	0.5	92.3%
D-Code	0.5270	97.3%

4. 결론

본 논문에서는 의사 스크램블 방식과 DC-free RLL 변조 코드를 이용한 멀티모드 변조코드를 제안하였다. 본 논문에서 제시한 멀티모드 변조코드와 다른 여러 코드를 동일한 DC억압 능력에 대해 하드웨어의 양 및 코드효율 측면에서 비교하였다. 시뮬레이션 결과에서 알 수 있듯이, 본 논문에서 제시한 멀티모드 변조코드가 Parity preserve 코드인 A-code나 EFMP 코드에 비해 코드효율이 높으며 종래의 멀티모드 변조코드 방식인 B-Code에 비해 코드효율과 하드웨어 양이 개선되었다.

참고 문헌

[1] K. A. S. Immink and J. Kahlman, WO 99/063671, Dec. 1999

[2] JS Shim, YK Won, and JW Ko, US Patent 6,281,815, Aug. 2001

[3] K.A.S. Immink, "Codes for Mass Data Storage Systems," *Shannon Foundation Publishers*, 1999

[4] Yoshihiro Hori, Hisashi Matsuyama, Akiomi Kunisa, Nobuo Itoh, Seiichiro Takahashi, Toshiaki Hioki, Kenji Asano, Noboru Mamiya, Yoshiharu Uchihara, Kenji Nakao, Satoshi Sumi, Kenji Torazawa, US Patent 6,198,710, Mar. 2001

[5] 심재성, 김진한, 정규해, 박인식, "에러전파를 줄인 멀티모드 방식의 고효율 데이터 변조방법," JCCI2003, April 2003