한국통신학회 1984년도 추계학술발표회 논문집 (1984. 11. 3)

순환 곱셈 코드를 이용한 인코딩 및 디코딩

84320

김 신 령　　　　강 창 언
연세대학교 공과대학 전자공학과

Encoding and Decoding using Cyclic Product Code

Sin Ryeong Kim　　　Chang Eon Kang
Dept. of Electronic Eng. Yonsei Univ.

* Abstract *

When the received sequence is not identical to the transmitted code word due to the channel noise effect,it is necessary to detect and correct errors.

In this paper, it is shown how to construct the encoder and the decoder using cyclic product codes. This system combines random and burst error correction and is easily decodable.

Performance has been obtained as expected.

1. 서 론

어러 통신 채널에서는 random 과 burst 에러를 동시에 정정할 수 있고 또 쉽게 효율적인 방법으로 디코딩될 수 있는 코드가 필요하게 되는데, 이러한 특성을 지닌 코드가 곱셈 코드 (Product Code)이다.

이 곱셈 코드는 1954년에 Elias 에 의하여 처음으로 소개 되었고 1965년에는 Burton 과 Weldon 이 실현을 쉽게하기 위하여 순환 곱셈 코드(Cyclic Product Code)를 정의하였으며 1968년에는 Abramson 에 의하여 가장 간단하고 Performance 가 좋은 Cascade Decoding 이 연구되어졌다.

본 논문에서는 순환 곱셈 코드를 error trapping 방식을 각각의 부(sub)코드의 decoder 에 적용하여 Cascade Decoding 으로 실현해 보았다.

2. 순환 곱셈 코드의 특성

통신 채널에서 Powerful 한 코드를 얻기위하여 두개 이상의 코드를 조합하게 되었고 이러한 코드를 수행하는 방법은 Code Array 를 제작해서 row (column)로 encoding하고 column(row)로 decoding 한다.

이때, 순환 코드의 실현이 간단하므로 새로운 순환코드인 순환 곱셈 코드를 정의한다.

(1) 순환 곱셈 코드의 구성

순환 곱셈 코드는 부 코드인 C_1 과 C_2 코드가 순환코드이고 코드 길이 n_1 과 n_2 가 relatively prime 한 곱셈 코드이다.

2차의 곱셈코드에 대하여 여기에서 다루기로 한다.

순환 곱셈 코드의 2차 Array 는 다음과 같다.

$$\begin{bmatrix} a_{00} & a_{01} & \cdots & a_{0(n_1-1)} \\ a_{10} & & \cdots & \vdots \\ \vdots & & & \\ a_{(n_1-1)0} & & \cdots & a_{(n_2-1)(n_1-1)} \end{bmatrix}$$

이때, code vector $\bar{b}_i$ 는 Chinese remainder theorem 에 의하여 나타내어 진다.

<u>Theorem)</u> $[a_{ij}]$ 가 순환 곱셈코드의 한 code word 이고 $i = 0,1, \ldots, n_1n_2-1$ 에 대하여 i 의 (Mod n_1)을 i_1 으로 그리고 i 의 (Mod n_2)를 i_2 라 한다.
이때, $n = n_1n_2$ 인 n 차의 code vector b_i 는 $a_{i_1i_2} = b_i$ 에 의해 나타난다.

(2) 생성 다항식

(n_1,k_1)코드의 생성 다항식의 $g_1(x)$이고 (n_2,k_2)코드의 생성 다항식이 $g_2(x)$라 하면 순환 곱셈코드의 생성 다항식 $g(x)$는 다음과 같다.

$$\begin{aligned} g(x) &= \mathrm{GCD}\left[x^{n_1n_2}-1,\ g_1(x^{bn_2})g_2(x^{an_1})\right] \\ &= \mathrm{LCM}\left[\mathrm{GCD}\{g_1(x^{bn_2}),\ x^{n_1n_2}-1\},\right. \\ &\qquad \left.\mathrm{GCD}\{g_2(x^{an_1}),\ x^{n_1n_2}-1\}\right]. \end{aligned}$$

윗 식에서 알 수 있듯이 순환 곱셈 코드의 생성 다항식은 부 코드의 생성 다항식 함수이다.

그리고 Interlacing 형태로 구성될 수 있음을 알 수 있다.

(3) 순환 곱셈코드의 에러 정정 능력

순환 곱셈의 에러 정정 능력은 C_1과 C_2 코드의 최소 거리가 d_1, d_2라 할때 순환 곱셈코드의 최소 거리가 $d = d_1 d_2$ 이므로 다음과 같이 쓸 수 있다.

C_1, C_2 코드의 random 에러 정정 능력이 $t_1 = (d_1-1)/2$ 이고 $t_2=(d_2-1)/2$ 이라면

$$t= \frac{(d-1)}{2} = \frac{(d_1 d_2-1)}{2} .$$

이제 burst 에러 정정 능력에 대하여 알아본다. 순환 곱셈 코드의 row 코드가 길이 n_1, random 에러 정정 능력 t_1을 가지고, column 코드는 n_2, t_2, B_2 를 가진다고 할 때 순환 곱셈 코드의 burst 에러 정정 B 는 다음과 같다.

$$B \geq n_1 t_2 + B_1$$
$$B \geq n_2 t_1 + B_2 .$$

예를 들면, 코드 Array 의 전송에서 $n_1 t_2+B_1$ 의 burst 에러가 발생했다면 우선 column 부터 decoding 한다.

그러므로 B_1 column 들만 에러를 가지게 되고, row code 가 B_1 의 burst 에러를 정정할 수 있으므로 이러한 B 이하의 에러가 정정된다.

(4) Cascade Decoder

순환 곱셈 코드의 Decoder의 복잡성은 C_1, C_2 코드에 의해 결정되므로 (n_1, k_1)과 (n_2, k_2)코드에 대한 Decoder 가 얼마나 복잡한 가에 의해서 정해지고 그것을 Block 으로 그려보면 다음과 같다.

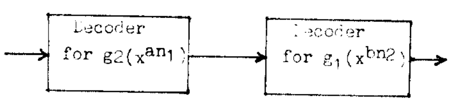

Fig.1 Cascade Decoder for Cyclic Product code

이 Cascade decoder 는 순환 곱셈 코드의 decoding 방식중 실현 가능하고, t 개 이상의 에러도 상당히 많은 양을 정정할 수 있으므로 performance 가 좋아서 사용한다.

3. Encoder 및 Decoder 설계

순환 곱셈 코드의 encoder 와 decoder 는 순환 코드의 interlacing 된 encoder, decoder 로 쉽게 실현할 수 있으므로 본 논문에서는 (3,1) 순환 코드와 (7,4)순환 Hamming 코드를 이용하여 간단한 순환 곱셈 코드를 실현하기로 한다.

다음의 상수로서 특성을 간단하게 나타낼 수 있다.

Code word ; $n= n_1 n_2$

정보 word ; $k= k_1 k_2$

최소 거리 ; $d= d_1 d_2$

이제 C_1 코드로서 (7,4) Hamming 코드, C_2 코드로 (3,1)순환 코드를 사용하여 각 특성을 구한다.

(1) 상수 : $n= 21$

$k=4$

$d= 9$; $t= 4$

(2) 생성 다항식

순환성을 갖기위한 조건인 relatively prime 을 적용하여 구한다.

$$\left.\begin{array}{l} an_1 + bn_2 = 1 \\ 7a + 3b = 1 \end{array}\right\} \quad a=1 , b=-2$$

$g_1(x)=1+x+x^3$, $g_2(x)=1+x+x^2$ 에서

$$g_1(x^{bn_2})=(1+x^{12}+x^{18})$$
$$g_2(x^{an_1})=(1+x^7+x^{14}) .$$

그러므로 순환 곱셈 코드의 g(x)는

$$g(x)=x^{17}+x^{15}+x^{14}+x^{10}+x^8+x^7+x^3+x+1 .$$

(3) Encoder

일반적인 순환 코드에 대한 encoder 방식을 조금만 변형시켜서 사용하면 된다.

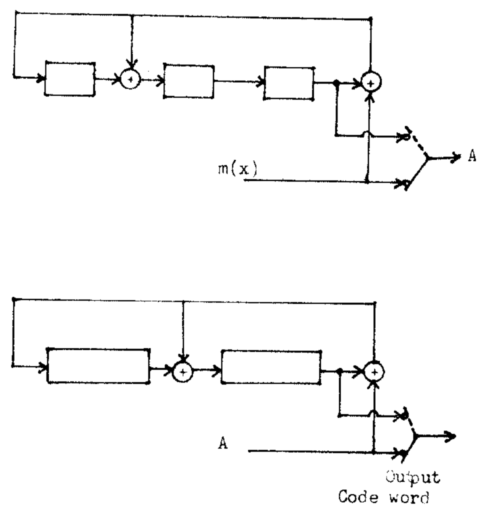

Fig.2 Encoder

(4) Decoder

각 부 코드의 decoder 를 error - trapping 방식을 이용하여 cascade decoding 하여 실현한다.

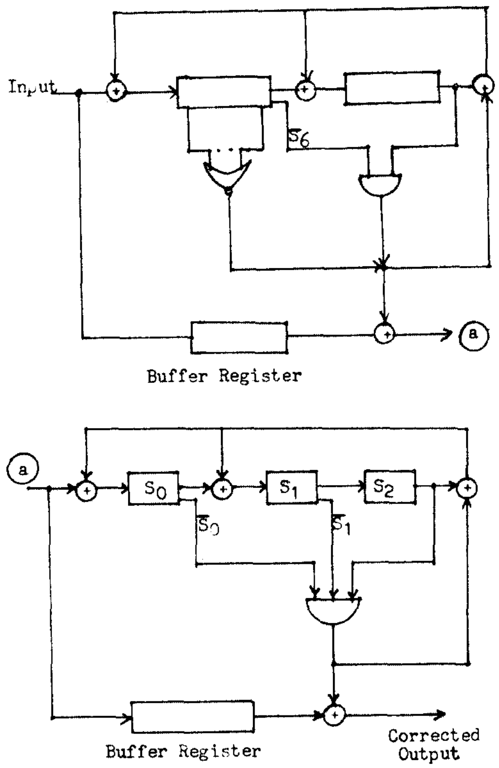

Fig. 3 Cascade Decoder

(5) 에러 정정 능력

random 에러 정정 능력을 t , burst 에러 정정 능력을 B 라 하면

$$t = \frac{d-1}{2} = 4$$

$$B \geq (n_1 t_2 + B_1) = 8.$$

4. 실험 및 결과 고찰

실험에 대한 계통도는

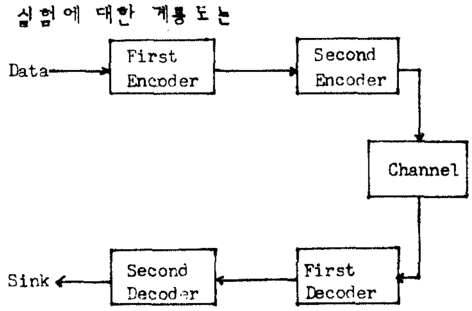

Fig.4 Block Diagram

위 회로에 대한 performance를 구해 보기로 한다.
여기서 channel 은 BSC 이라 하고 에러 확률을 p 라 한다. 이 때 에러가 서로 독립적으로 발생한다고 가정할 때 에러를 정정 못할 확률을 PM 을 구해본다.
첫번째 decoder 에서 에러를 정정 못할 확률은

$$n_1 P_e \leq \sum_{i=t+1}^{n_1} (i+1) \binom{n_1}{i} P^i (1-P)^{n_1-i}$$

이다. 두번째 decoder 에서 에러를 정정 못할 평균 확률은

$$PM \leq k_1 \sum_{i=t+1}^{n_2} \binom{n_2}{i} P_e^i (1-P_e)^{n_2-i}$$

이것을 Computer Simulation 을 통하여 구해보고 (21,4)코드의 t 가 4개라 할 때 PM 을 구하여 비교해 보면 다음 표와 같다.

DATA	PM1	PM2
.1	.0533395	.151965
.05	4.543E-03	.0188806
.04	1.97574E-03	8.87092E-03
.01	9.18944E-06	5.22305E-05
5E-03	5.90771E-07	3.494[illegible]
1E-03	9.66611E-10	5.90415E-09
8E-04	3.96364E-10	2.42492E-09
5E-04	6.05814E-11	3.71527E-10
1E-04	9.7146E-14	5.97687E-13

table 1. Performance

위의 결과 표에서 Cascade decoding 방식을 이용함으로써 performance 가 좋아짐을 알 수 있다.

5. 결 론

본 논문에서는 순환 곱셈 코드의 encoder 와 decoder 를 error trapping 방식을 이용하여 실제로 설계해 보았다.
이것으로 부터 interlacing 을 사용하여 encoder 와 decoder 에 대한 실험을 하였는데, 회로가 각 부 코드의 것을 그대로 이용하였으므로 간단함을 알 수 있고 효율이 좋은 decoding방식임을 알 수 있다.
이 순환 곱셈 코드는 random 과 burst 에러 정정이 가능 하므로 전화등에서 나타나는 이러한 종류의 에러 정정에 적합하다.

[참 고 문 헌]

1. Belakamp," Algebraic Coding Theory",
McGraw-Hill,1968.
2. Lin," An Introduction to Error Correcting
Codes", Prentice Hall,1970.
3. Peterson and Weldon," Error Correcting
Codes", MIT Press,1972.
4. John B.Fraleigh," A First Course in Abstract
Algebra", Addison-Wesley,1967.
5. Elias,P.," Error Free Coding", IRE Trans.Inf
Theory IT-4(1954):PP29-37.
6. Burton,H.O., and E.J.Weldon,Jr.," Cyclic
Product Codes", IEEE Trans.Inf.Theory
IT-11(1965):PP 433-439.
7. Abramson,N.M.," Cascade Decoding of Cyclic
Product Codes", IEEE Trans. on Comm.Tech.
COM-16(1968):PP 398-402.