

## 컴팩트 디스크를 위한 Reed Solomon 부호기/복호기 설계

김창훈\*, 박성모\*\*

전남대학교 정보통신협동과정\*, 전남대학교 컴퓨터공학과\*\*

Phone : 062 - 530 - 0798

### Design of Reed Solomon Encoder/Decoder for Compact Disks

Chang-Hoon Kim\*, Seong-Mo Park\*\*

\*Dept. of Interdisciplinary Program of information and telecommunication,  
Chonnam National Univ.,

\*\*Dept. of Computer Eng., Chonnam National Univ.

E-mail : chkim@ciscom.chonnam.ac.kr

#### Abstract

This paper describes design of a (32, 28) Reed Solomon decoder for optical compact disk with double error detecting and correcting capability. A variety of error correction codes(ECCs) have been used in magnetic recordings, and optical recordings. Among the various types of ECCs, Reed Solomon(RS) codes has emerged as one the most important ones. The most complex circuit in the RS decoder is the part for finding the error location numbers by solving error location polynomial, and the circuit has great influence on overall decoder complexity. We use RAM based architecture with Euclid's algorithm, Chien search algorithm and Forney algorithm. We have developed VHDL model and performed logic synthesis using the SYNOPSIS CAD tool. The total number of gate is about 11,000 gates.

#### I. 서론

최근 저장 방식이 디지털 방식으로 변환됨에 따라 막

본 논문은 한국과학재단 지정 전남대학교 고품질 전기 전자 부품 및 시스템 연구센터의 연구비 지원에 의하여 연구되었음

대한 양의 데이터, 오디오, 그리고 비디오 정보의 저장을 위한 광학디스크 시스템이 널리 사용되고 있다. 이의 대표적인 시스템을 보면, 오디오 신호를 광학식 디스크에 저장하기 위한 컴팩트 디스크 시스템(compact disc system), 오디오 정보를 자기 테이프에 저장하기 위한 DAT(digital audio tape) 시스템, 그리고 데이터를 광학식 디스크에 저장하는 CD-ROM (compact disk read only memory) 등을 들 수 있다. 광디스크 상에 열 지어 기록되어 있는 미세한 홈(pit)을 이용하여 읽어 내는 것을 바탕으로 설계되어 있는 광디스크 시스템은 불규칙한 잡음 및 변조(modulation) 잡음이 없고, 랜덤 액세스(random access)가 가능하며, 먼지와 흠에 따른 영향이 비교적 적으며, 정보가 장기간 안정적으로 기억되고 있고, 비디오 정보의 경우 재생시 화면상의 가상(ghost) 현상이 없으며, 반영구적인 재생 및 기억이 가능하다는 주요 특징이 있다. 따라서 이와 같은 장점을 바탕으로 데이터, 오디오, 비디오 정보의 광디스크를 이용한 저장 및 재생은 널리 확대되는 추세이다. 그러나 광디스크 표면에 발생한 여러 종류의 흠은 정보의 보전성에 커다란 영향을 미치게 된다. 따라서 광저장 매체를 갖는 디지털 저장 시스템 또는 재생기를 설계할 경우 반드시 고려해야 할 핵심 요소 기술은 오류정정기술이다.

## II. RS 부호

RS 부호는 유한체 GF(q) 상에서 부호의 길이가 n인 순회부호이며, 생성다항식은 식 (1)과 같다.

$$g(x) = (x - \alpha^b)(x - \alpha^{b+1})(x - \alpha^{b+2})(x - \alpha^{b+3}) \dots (x - \alpha^{b+2t-1}) \quad (1)$$

RS (n, k) 부호라고 했을 때  $n - k = \deg g(x) = 2t$  이라면 Dimension은 k 이고 MDS(maximum distance separable)가 d 인 RS부호가 된다.[2][3] 이때 k 와 d사이는 식(2),(3)과 같다.

$$k = n - 2t \quad (2)$$

$$d = n - k + 1 = 2t + 1 \quad (3)$$

(n, k) RS 부호에서  $q = 2^m$  인 경우에, 유한체 GF(q) 상의 각 심벌을 바이너리 m(bit)으로 나타낸다면 코드워드 C = (C<sub>0</sub>, C<sub>1</sub>, ..., C<sub>n-1</sub>)은 바이너리 mn 벡터로 나타낼 수 있고, d = n - k + 1 인 부호가 된다. 또 이 RS 부호는 심벌 오류정정능력이 t인, 버스트 에러의 길이가 m(t-1)+1 이하의 어떠한 심벌에러도 정정할 수 있게 된다.[2][3] t는 식 (4)와 같이 나타낼 수 있다.

$$t = \lfloor \frac{n-k}{2} \rfloor \quad (4)$$

RS 부호는 심벌내의 어떠한 비트에 대한 에러도 그 심벌에 대한 에러로 간주하게 되므로, 랜덤 에러에 대한 에러정정 보다 버스트에러에 대한 에러정정에 적합하다고 할 수 있다.

## III. 부호기/복호기의 설계

### 1. RS 부호기 설계

(32, 28) RS 부호기는 그림 1과 같이 간단하게 구현할 수 있는데, GF(2<sup>8</sup>)상에서의  $\alpha^{76}$ ,  $\alpha^{251}$ ,  $\alpha^{81}$ ,  $\alpha^{10}$  을 곱하는 곱셈 회로와 덧셈을 행하는 XOR, 또 지연을 위한 레지스터로 간단하게 구현할 수 있다.

각 8비트의 정보 심벌 28개가 들어오는 부분과 패리티를 위한 부분으로 나누어 볼 수 있다

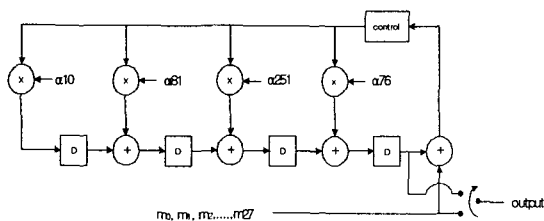


그림 1. RS 부호기의 블록도

### 2. RS 복호기 설계

RS 부호의 복호 알고리즘으로는 Chien의 알고리즘 [2][3]을 이용한 PGZ (Peterson-Gorenstein-Zieler) 복호 알고리즘, Berlekamp의 반복 알고리즘[4], Euclid 알고리즘[5], 변환 복호 알고리즘, 그리고 오증요소와 오류치(error value)와의 대수학적 관계를 이용한 직접 복호 알고리즘 등이 있다.

본 논문에서는 오디오 정보의 저장을 위한 광학식 디스크 시스템용 오류정정부호로 표준화되어 있는 GF(2<sup>8</sup>) 상에서의 2중 오류정정 RS부호의 효율적인 복호기를 설계했다.

컴팩트 디스크 시스템에서 발생하는 오류를 정정하기 위한 2중 오류정정 (32, 28) 단축 RS부호는 산발 오류뿐만 아니라 연접오류까지도 정정이 가능한 강력한 오류정정능력을 갖는 부호이다. RS부호를 복호하기 위해서는 2원 BCH 부호의 복호시 요구하지 않던 오류치(error value)를 구하는 과정이 추가로 요구되므로 복호과정의 매우 복잡하다. 특히, 오류정정능력, 유한체의 크기 및 유한체 연산기의 형태 및 실현 알고리즘 등은 복호기의 복잡도에 영향을 미치며, 복호 시간(decoding time) 역시 그에 비례하여 크게 달라진다. 그러므로 간단한 구조를 가지며 신속하게 복호를 수행하는 복호 알고리즘은 복호기뿐 아니라 전체 시스템의 성능에 커다란 영향을 미치게 된다. RS 부호의 복호기 성능에 영향을 미치는 요소는 일반적으로 오류개수 판별식 알고리즘, 오류위치(error location) 및 오류치 계산 알고리즘, 그리고 구현되는 유한체 원소간의 승, 제산기 등에 대한 연산 알고리즘 등이다.

오류정정 RS 부호의 복호 과정은 다음과 같이 5단계로 구성된다.

- 1) 오증 요소 (syndrome component) 계산
- 2) 오류위치 다항식 (error locator polynomial) 계산
- 3) 오류위치 번호(error location number) 계산
- 4) 오류치 (error value) 계산
- 5) 오류정정 (error correction) 수행

본 복호기는 (32, 28) 단축 Reed Solomon 복호기로 심벌당 8비트로 이루어진 32개의 심벌 중에서 2개 심벌의 오류정정의 성능을 가지는 복호기로 원시다항식(primitive polynomial)은  $1 + x^2 + x^3 + x^4 + x^8$  이며, 생성다항식은 식(5)과 같다.

$$g(x) = (x+1)(x+\alpha)(x+\alpha^2)(x+\alpha^3) = x^4 + \alpha^{76}x^3 + \alpha^{251}x^2 + \alpha^{81}x + \alpha^{10} \quad (5)$$

유한체 연산에 기초한 알고리즘에 의해서 복호가 수

행되는 방식을 사용하였다.

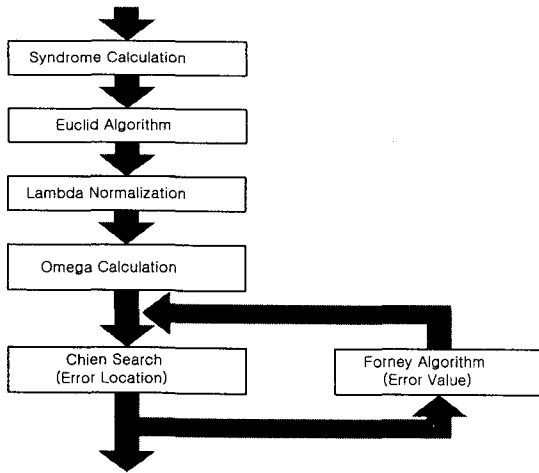


그림 2. Reed Solomon 복호기의 흐름도.

복호기의 복호과정에 대한 적용 알고리즘의 흐름도는 그림 2과 같고, Berlekamp-Massey 알고리즘 보다 하드웨어로의 구현이 용이하다고 알려진 Euclid 알고리즘을 기초로 하여 구현하였다. 각 오류의 위치를 구하기 위해, 오류치를 구하는 Chien 알고리즘과 Forney 알고리즘을 반복하여 사용하였다[6].

본 복호기에서 모든 연산은 GF(2<sup>8</sup>)에서의 8비트 심볼을 위한 연산을 이용하여 이루어지고, 복호의 마지막 과정에서는 수신된 신호의 복호된 결과에 대한 평가를 보여주는 신호를 덧붙였다.

그림 3에서는 각 블록별로 적용한 알고리즘과 신호의 흐름을 간략하게 도시하였다. 유한체에서 복호에 필요한 여러 가지 알고리즘의 수행과정에서의 요소를 처리하기 위한 변수로 오증(syndrome)은 S, 정규화 시킨 오류위치 다항식은  $\sigma$ , 오류평가 다항식은  $\Omega$ 로 나타내었다.

2.1. Syndrome Calculation 블록

오증은 식(6)와 같이 표현되고 2개의 8비트 심벌의 오류정정을 위해서는 4개의 오증이 필요하게 된다.

$$S_i = r(\alpha^i) = e(\alpha^i) = r_0 + r_1\alpha^i + r_2\alpha^{2i} + \dots + r_{n-1}\alpha^{(n-1)i} = r_0 + \alpha^i (r_1 + \alpha^i (r_2 + \alpha^i (r_3 + \dots + \alpha^i (r_{n-2} + r_{n-1}\alpha^i) \dots)) \dots) \quad (6)$$

그림 4은 오증요소 계산의 블록도를 보여주고 있고, 4개의 오증 요소의 계산 블록에서 오증이 계산된다.

입력되어지는 수신 신호에는 코드워드와 신호가 전송

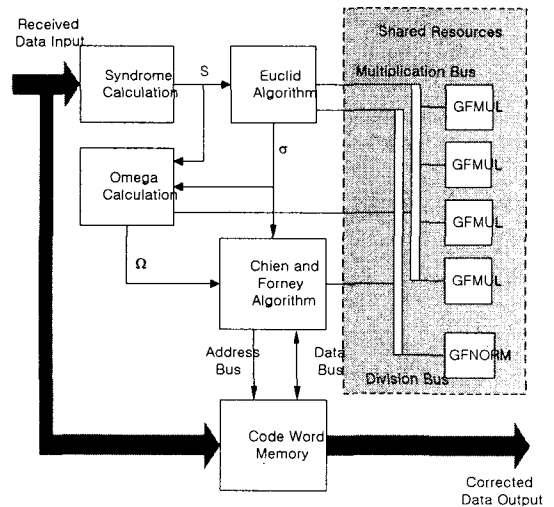
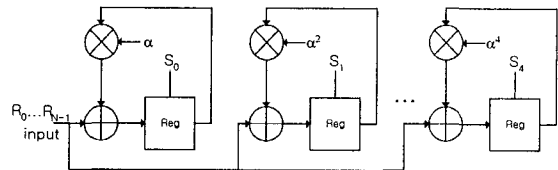


그림 3. Reed Solomon 복호기의 블록도.

되면서 채널에서 생기는 에러신호가 더해져서 들어오게 된다. 레지스터에는 전 심벌의 오증이 저장되어 있게 되고 현재의 신호와 유한체상의 연산이 이루어진



다.

그림 4. Syndrome Calculation 블록도

2.2. Euclid Algorithm 블록

Euclid Algorithm은 빠르게 GCD(great common divisor)를 찾는 알고리즘으로 알고리즘은 에러위치 다항식과 에러평가 다항식으로 식(7)과 같이 표현된다. S(x)는 오증을 나타내며 t<sub>i</sub>는 에러위치다항식, q<sub>i</sub>는 몫을 나타낸다

$$r_{-1}(x) = x^{2t+1}, r_0(x) = 1 + S(x) \quad (7)$$

$$t_{-1}(x) = 0, t_0(x) = 1$$

$$r_i(x) = r_{i-2}(x) - q_i(x)r_{i-1}(x)$$

그리고 r<sub>i</sub>의 차수는 t<sub>i</sub>의 차수보다 크지 않다.

오증으로 부터 오류위치다항식 sigma를 구하게 된다.

2.3. Omega Calculation 블록

오류평가 다항식 Omega를 구하는 블록으로 Syndrome Calculation 블록으로부터 오증 요소로부터 식(8)과 같이 계산되어 진다.

$$\sigma(x)[1+S(x)] = \Omega(x) \text{ mod } x^{2t+1} \quad (8)$$

2.4. Chien Search and Forney Algorithm 블록  
GF(2<sup>8</sup>) 유한체상에서 오류위치 다항식의 근과 에러의 크기를 구하는 블록이다.

$$e_{ik} = \frac{-X_k \Omega(X_k^{-1})}{\sigma'(X_k^{-1})} \quad (9)$$

Chien Search 블록은 Syndrome Calculation 블록과 유사한 구조를 가지며 오류위치다항식과 오류평가다항식으로부터 근과 오류의 크기를 계산한다.

2.5. GFMUL 블록  
GF(2<sup>8</sup>) 유한체상의 곱셈기의 역할을 하는 블록으로 Omega Calculation 블록으로부터 Ω값을 구하는데 4개의 GFMUL 블록을 사용하며, 또 Euclid Algorithm 블록으로부터 σ값을 구할 때에도 1개의 GFMUL 블록을 사용한다.

2.6. GFNORM 블록  
Chien Search and Forney Algorithm 블록에서 오류위치다항식의 근과 오류의 크기를 구할 때 쓰이는 회로이고 계산기의 기능을 하는 블록이다.

IV. 시뮬레이션 및 구현

RS 복호기의 구현에 있어서 각 블록을 VHDL로 기술하였고, SYNOPSIS CAD Tool을 사용하여 IDEC C-631 라이브러리를 이용하고 RAM을 내장하는 구조로 합성하여 구현하였다. 복호기는 XILINX FPGA XC4000EX으로의 구현을 목표로 설계하였다.

시뮬레이션은 쉽게 검증이 가능하도록 모두 제로로 가지는 코드워드를 이용하여 여러 심벌을 각각 다른 위치에 삽입하여 테스트하는 방법으로 수행하였고, 이때 복호된 코드워드는 모두 제로가 되어야 한다. 만약 모두 제로가 아닌 워드가 발생하는 경우는 복호를 실패한 경우가 되도록 하였다.

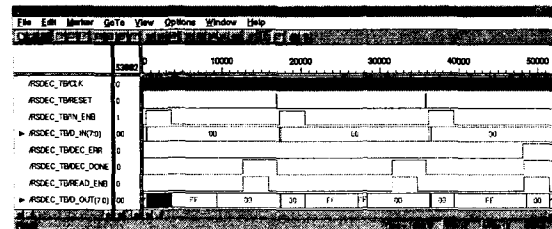


그림 5. 시뮬레이션 결과 파형

그림 5에서는 Synopsys Tool을 이용한 기능 시뮬레이션 결과이다. 결과의 일부분이지만 32심벌 중에서 1심벌, 2심벌 에러를 정정하고 3심벌 에러에 대해서는

에러를 검출하기는 하지만 정정하지 못하는 파형을 보여주고 있다.

V. 결론

Synopsys Tool을 사용하여 합성한 결과, GF(2<sup>8</sup>)유한체상에서 (32, 28) 부호기와 복호기는 각각 990 게이트와 11,000 게이트 정도이며 4개 블록과 5개의 공유하는 블록으로 총 9개의 블록으로 구성하였다. FPGA로 구현하기 위하여 XILINX XC4000EX을 이용하였다.

본 논문에서 설계한 복호기는 복호 성능에 있어서 효율적인 Euclid 알고리즘과 Chien 알고리즘, Forney 알고리즘을 적용한 결과, 수신되는 32개의 8비트 심벌 중에서 1개의 심벌과 2개의 심벌까지의 오류의 검출과 정정 가능한 구조를 가지고 있고, 적은 칩 면적으로 구현이 가능하여 고속의 컴팩트 디스크에 적용하여 사용할 수 있다.

참고문헌

[1] Heung Youl Youn, Jae Moon Kim, Man Young Rheen "New Decoding Techniques of RS Codes for Optical Disks", Journal of the KITE V.30-A,N,3 1993. PP. 16-33  
 [2] Stephen B. Wicker and Vijay K. bhargava, Reed-Solomon Codes and Their Applications, New York, IEEE Press, 1994.  
 [3] Stephen B. Wicker, Error Control Systems for Digital Communication and Storage, Prentice Hall, 1995.  
 [4] Yong Hwan Kim, Young Mo chung, Sang Uk Lee, "A Study on a VLSI Architecture for Reed-Solomon Decoder Based on the Berlekamp Algorithm", Journal of the KITE V.30-B, N.11, 1993. PP. 17-26  
 [5] Hyunman Chang, Sunwoo. M.H, "Design of an area efficient Reed-Solomon decoder ASIC chip", IEEE Workshop on Signal Processing Systems, 1999 . PP. 578 -585  
 [6] Schuler, C. "Code generation tools for hardware implementation of FEC circuits", The 6th IEEE International Conference on Electronics, Circuits and Systems, vol. 1. pp. 221-224, 1994