工學碩士 學位論文

EUROFIX 시스템에 적용하기 위한 Reed-Solomon 부호 설계

Design of Reed-Solomon Code for
Application to EUROFIX System

指導敎授　趙 炯 來

2002年 2月

韓國海洋大學校 大學院

電 波 工 學 科

金　玟　芝

本 論文을 金玟芝의 工學碩士 學位論文으로 認准함.

委員長 : 工學博士 鄭 世 謨 印

委 員 : 工學博士 金 東 一 印

委 員 : 工學博士 趙 炯 來 印

2002年 2月

韓國海洋大學校 大學院

電波工學科

金 玟 芝

<차 례>

Abstract

The function of the channel encoder is to introduce, in a controlled manner, some redundancy in the binary information sequence at the receiver. It is also required that the channel encoder can overcome the effects of noise and interference encountered in the transmission of the signal through the channel.

Thus, the added redundancy serves to increase the reliability of the received data and improves the fidelity of the received signal. In effect, redundancy in the information sequence aids the receiver in decoding the desired information sequence.

In this thesis, the Reed-Solomon code was designed, which is used in an EUROFIX system. The EUROFIX system uses the Loran-C system for information transmission.

As a result, this thesis proposed a design and implementation method of the (14, 9) RS encoder. In the decoder, the Galois Field Fourier Transform method was adopted for reducing the decoding time.

It is concluded that the decoding time get shorter clock times than a conventional method by the proposed one.

제 1 장 서 론

통신 시스템에서 통신로는 여러 형태의 잡음, 왜곡, 그리고 간섭 등에 영향을 받아 전송 중 발생하는 오류로 인해 일반적으로 수신 데이터는 송신 데이터와 서로 다를 수 있다.

따라서 통신 시스템에서 처리되는 광대한 양의 데이터에 대한 오류를 제어하기 위한 수단이 필요하다. 그 수단으로 오류정정 부호나 정정 알고리즘이 대두되었으며, 실제 현대 통신망과 컴퓨터 주기억 장치의 설계 및 장치화에 중요한 적용요소가 되고 있다.

특히, 오류 정정 부호는 여러 통신 분야 중 위성 측위 시스템 활용분야는 항법분야에서 해상분야는 물론이고 선박의 안전항해, 지리정보시스템(GIS)분야, 우주항법분야, 그리고 농업, 산림, 일반레저(등산, 낚시)에서 다양하고 유용하게 사용되고 있다.

본 논문에서는 이러한 오류 정정 부호 중 하나인 Reed-Solomon 코딩을 EUROFIX에 적용하는 연구에 대해서 기술하였다.

EUROFIX는 최근에 위성시스템의 비상수단으로서 인식되는 Loran-C 시스템을 이용하여 DGNSS(Differential Global Navigation Satellite Systems) 정보를 전송하는 통합위치결정 시스템을 말한다.

따라서 한국에서는 EUROFIX의 도입에 관한 연구를 진행중에 있으며, 이에 따라 EUROFIX 정보전송의 부호화과정에서 Reed-Solomon 코드의 부호화 및 복호화에 대해서 연구하였다.

본 논문의 제 2 장에서 EUROFIX 시스템과 Reed-Solomon code에 대해 기술하였으며, 제 3 장에서는 EUROFIX 시스템에서 RS Code의 부호화 과정을 이론적으로 전개하고 제 4 장에서는 EUROFIX 시스템에서 RS Code의 복호기를 설계하여 시뮬레이션 하였다. 그 결과 설계된 복호기는 성능이 우수하여 실용화 가능성을 확인하였으며 제 5 장에서 본 연구의 결과로부터 얻어진 결론을 정리하였다.

제 2 장 EUROFIX 시스템과 Reed-Solomon Code

2.1 Loran-C에 기초한 EUROFIX 데이터링크 모델

2.1.1 EUROFIX의 개요

EUROFIX는 DGNSS와 Loran-C를 기초로 하는 통합위치결정 시스템으로서, LORAN-C 신호 펄스열들의 펄스 위치 변조에 의한 LORAN-C 전송을 통해서 데이터 통신을 한다. EUROFIX는 DGNSS 정보 외에 완전한 LF 무선 항해 시스템을 제공하는데 이 시스템은 이 분야에 있어서 다른 모든 서비스와 자체 서비스를 구분한다.

EUROFIX를 이용하여 통신되는 메시지들은 주국과 부국간의 Loran-C/Chiayka 동기신호(BALTICA), DGNSS 데이타 , Loran-C/Chayka의 부가적인 공중파 요소(ASF)및 조난상황에서의 긴급상황 메시지 등이 있으며, 부가적인 시간-변조를 통해 전송된다[9].

그림 2.1은 EUROFIX DGNSS의 개념도에 대해서 나타낸다[11].

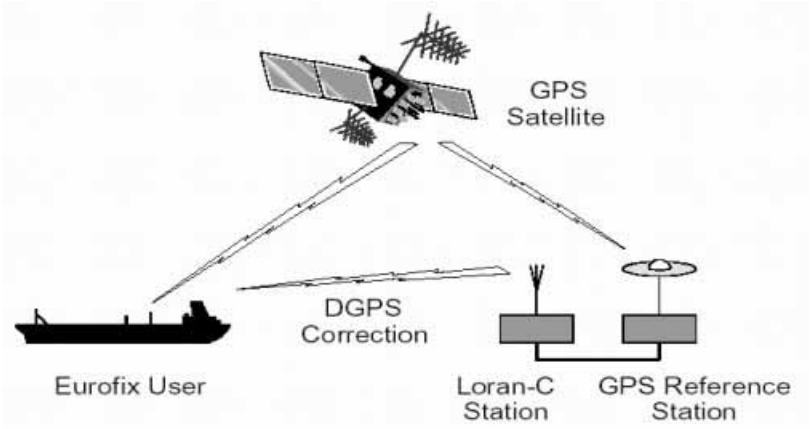

그림 2.1 EUROFIX DGNSS 의 개념도
Figure 2.1 The Concept of EUROFIX DGNSS.

Loran-C는 1958년부터 운영되어 온 전파항법의 일종이며, 하나의 주국과 2국 이상의 종국이 서로 동기된 신호를 90~110kHz의 대역으로 송신하며, 이동체에 설치된 수신기는 이 신호를 수신하여 그 펄스의 포락선(Envelop)과 싸이클을 비교하여, 주국과 종국의 신호의 시간차를 측정하며 마이크로초(μs)의 소수점이하까지 정확히 측정되는 시스템이다[11].

전파항법시스템은 위성계와 지상계의 2계통으로 분류할 수 있다. 위성계로서 현용으로는 전세계적 측위시스템인 GPS(Global Positioning System)와 러시아가 운영하는 GLONASS(Global Navigation Satellite System)가 있다.

또 지상계시스템으로는 Loran-C, 데카 등이 있다.

세계적으로는 위성계는 GPS가 실용적인 시스템으로 정착돼 가고 있고, 지상계는 Loran-C를 주체로 운용하고저 하는 경향이 있다.

2.1.2 일반적인 데이터링크 모델과 파라미터들

EUROFIX 데이터링크의 목적은 Loran-C국에서부터 사용자에게까지 정보를 수송하는 것이다. 원칙적으로 정보는 어떠한 종류의 것이라도 될 수 있다.

여기서는 Loran-C 데이터 채널을 위한 일반적인 모델과 필요 조건을 만족하는 성능 파라미터부터 먼저 살펴보도록 하겠다.

다음의 그림 2.2은 일반적인 데이터링크 모델을 보여준다.

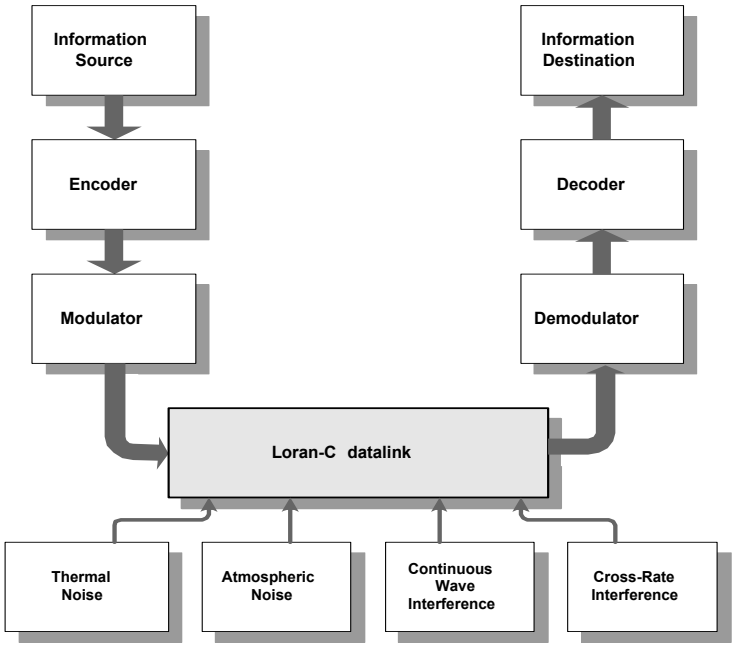

그림 2.2 Loran-C 데이터링크 모델

Figure 2.2 The Loran-C datalink model.

송신단에서 정보는 DGPS와 CRC 신호 및 RS 코드 신호로 인코딩 되고 변조된다.

전송되는 동안에 신호들은 연속파(Continuous wave interference)나 교차율 간섭(Cross-rate Wave)등의 노이즈에 의해 변형될 수 있다.

마지막으로 수신된 신호들은 복조되고 복호(decoding)되어 원 정보(source)를 얻을 수 있다.

2.2 EUROFIX 변조 과정

EUROFIX의 변조 과정은 Loran-C/Chayka 신호를 변조하여 데이터 전송을 가능하게 만들어 준다.

이 정의는 변조 패턴과 다른 데이터 표현간의 관계와 위치 표시를 위해 Loran-C/Chayka 사용의 평가절하를 최소화하는 변조 전략인 낮은 레벨의 변조타입을 채택한다.

2.2.1 펄스 변조

(1) 타이밍

3s-PPM(3-state Pulse Position Modulation)은 각 펄스그룹에서 3번째부터 8번째까지의 펄스에 적용되어진다. 변조는 변조되지 않은 펄스에 대하여 변조펄스의 1μs의 시간이동으로 구성된다.

변조의 3가지 가능한 상태가 표 2.1에 주어져 있다.

표 2.1 변조 상태

Table 2.1 States of the Modulation.

Pulse state	Transmission time minus time of reference pulse(μs)	Indication
Advanced pulse	-1	-
Prompt pulse	0	0
Delayed pulse	+1	+

(2) 변조의 균형

1개의 펄스그룹내의 한 채널상의 전진 혹은 지연 펄스들의 수는 서로 같아야만 한다. 한 펄스그룹내의 6개 펄스들의 변조는 141개의 가능한 균형화 된 패턴들로 분해된다.

(3) 타이밍의 정확성

변조된 신호의 타이밍의 정확성은 변조되지 않은 신호에 관한 한, 동일한 타이밍의 정확성에 따라야 한다.

(4) “데이타의 무전송” 패턴

패턴 ‘0 0 0 0 0 0’은 전송되어지는 데이터가 없음을 나타낸다.

(5) 메시지의 구조

1개의 3s-PPM 메시지는 30개의 연속적인 펄스 그룹으로 구성된다.

(6) blanking

블랭크(억제되어 발사되지 않음)된 펄스그룹은 변조목적을 위해서 전송되어졌다고 판단되어진다는 것을 고려해야 한다.

2.2.2 변조 구조

변조되는 과정을 살펴보면, 먼저 블링킹을 위해서 각 반복구간의 처음 2개의 펄스가 남겨진다. 남겨진 나머지 6개의 펄스들은 3가지 레벨로 위치 변조된다. 즉 726개(3^6)가운데 141개의 균형을 이룬 패턴들 중 6자리 디지털의 3가지 레벨상태인 패턴들만이 원하지 않은 추적 바이어스를 방지하기 위하여 사용된다. 이 패턴들이 패리티 체크를 하도록 한다. 이 과정이 아래 표 2.2에 도식화되어 있다.

표 2.2 패턴의 변조 조합과정

Table 2.2 Total number of balanced patterns in EUROFIX3-level modulation.

Modulation Pattern Combination	Example "0" is prompt pulse "+" is delayed pulse "−" is advanced pulse	Number of Combinations number= $\binom{6}{zero} \cdot \binom{6-zero}{plus} \cdot \binom{6-zero}{\min us}$
6 · zero 0 · plus 0 · minus	0 0 0 0 0 0	$\binom{6}{6} \cdot \binom{0}{0} \cdot \binom{0}{0} = 1$
4 · zero 1 · plus 1 · minus	0 0 0 0 + −	$\binom{6}{4} \cdot \binom{2}{1} \cdot \binom{1}{1} = 30$
2 · zero 2 · plus 2 · minus	0 0 + + − −	$\binom{6}{2} \cdot \binom{4}{2} \cdot \binom{2}{2} = 90$
0 · zero 3 · plus 3 · minus	+ + + − − −	$\binom{6}{0} \cdot \binom{6}{3} \cdot \binom{3}{3} = 20$
		total = 141

141개 중 128개의 패턴이 7비트의 이진 데이터를 표현하기 위해 선택된다.

7비트 심벌을 펄스 위치 변조 신호로 변경하는 예를 아래 표 2.3에 나타내었다.

표 2.3 7비트 이진데이터의 표현 예

Table 2.3 Example of representation for 7bit binary data.

Modulation Pattern	Bit Representation
- - 0 0 + +	1 0 0 0 0 0 0
- - 0 + 0 +	0 1 0 0 0 0 0
- - 0 + + 0	0 0 1 0 0 0 0
- - + 0 0 +	0 0 0 1 0 0 0
: :	: :

이 변환은 두 가지 중요한 영향력을 다음과 같이 가진다.

- 각 7비트 심벌은 6개의 전송 펄스들로 펼쳐진다. 어떤 잘못된 펄스의 수신은 그 심벌 내에서 여러 비트들에게 영향을 미친다.
- 처음부터 균형화 된 요구사항들이 에러 결정의 몇 가지 형태에 도입된다. 만약 수신된 변조 패턴이 균형화 되지 않는다면, 그 패턴은 비트로 변환될 수 없다. 그 비트는 제거용 심벌로 선언된다. 이 제거심벌(eraser)은 부가적인 순방향 에러 정정 코드로 정정되어져야 한다.

정규적인 Loran-C 사용에 대한 이러한 형태의 변조의 영향은 이전의 변조 구조와 비교해서 매우 적다. 계산결과는 이런

형태의 변조가 이전의 변조 패턴들[3,4]에 의해 소개된 손실보다 0.55dB이 적은 0.79dB정도의 신호 손실을 가진다는 것을 보여준다. 따라서 Loran-C 신호들의 부가적인 변조는 정규적인 Loran-C의 성능에 어떠한 변화를 초래하는 것이 극히 힘들다.

EUROFIX 변조 지식을 가지고 있는 미래의 Loran-C 수신기들은 응용 변조를 위해서 이전에 복조된 혹은 재변조된 펄스들을 쉽게 정정할 수 있다.

교차율 간섭과 블링킹의 영향에 주목해보면, Loran-C 신호 구조의 선택에 따른 고유 현상이 더 큰 신호 저하를 발생시킨다는 것을 알 수 있다.

40ms에서 100ms사이에 변화하는 Loran-C GRI(Group Reception Interval)를 가진 원시 비트 율은 175에서 70bps까지의 데이터 전송범위를 가능하게 한다. 앞에서 기술했듯이, 이러한 원시 비트율의 한 부분은 데이터링크의 이용성과 통합성을 확실하게 하기 위해서 순방향 에러 정정 코드를 사용해야만 한다.

2.3 순방향 에러 정정 코드

사용자가 전송된 메시지를 정확하게 수신했는지 아닌지를 승인하지 못할 때, FEC 코드들이 수시로 발생하는 에러를 정정하고 데이터 통합성을 정확하게 하는 효율적인 수단을 제공하여 이용성을 개선시킨다.

더욱이 적극적인 Loran-C 신호 환경은 순방향 에러 정정 코드를 절대적으로 필요로 한다.

2.3.1 EUROFIX의 에러 정정 구조

Loran-C 데이터 채널은 주로 다음과 같은 세 가지 정도의 에러 소스들에 의해서 교란된다.

- 대기 노이즈(잡음)
- 연속파 간섭
- 서로 다른 반복율 신호간 간섭

정규적인 Loran-C 수신기에서는 이러한 에러 소스들의 영향이 평균을 통한 확실한 범위로 감소되어 질 수 있다. 하지만, 만약 데이터가 Loran-C 채널을 통해 전송되면, 모든 펄스들은 복구되어야만 하는 정보를 전달한다.

복조이전에 데이터 신호의 품질을 향상시키기 위한 평균법의 사용은 따라서 불가능하다. 특히 멀리 떨어진 Loran-C 국들의 신호들은 서로 다른 반복율 신호간 간섭에 의해서 아주 많이 교란되어진다.

EUROFIX에서의 Reed-Solomon 코드를 이용한 간단한 패리티 체크의 연결은 우수한 결합이라 할 수 있다. 이런 패리티 체크는 이미 앞 절에서 설명된 균형화 된 변조 패턴에서 제공된다.

그림 2.3은 63비트 정보를 구성하는 메시지의 인코딩 과정을 보여준다.

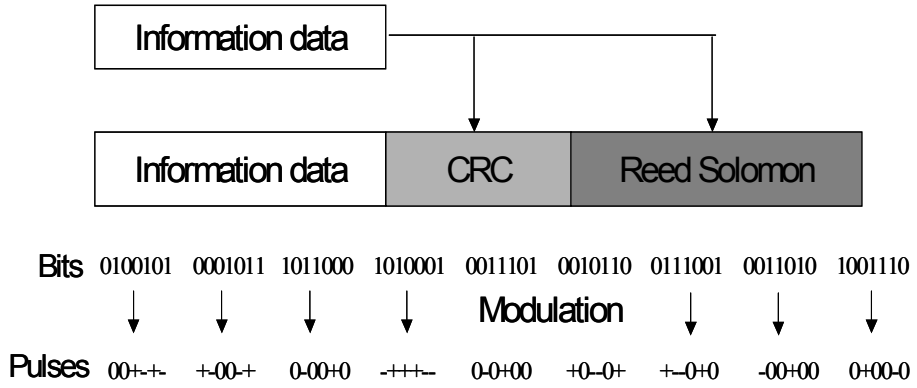

그림 2.3 EUROFIX 데이터 링크 상에서의
정보의 부호화 및 변조과정
Figure 2.3 Encoding and Modulation
for the Loran-C data channel.

정보 비트는 7 비트의 각 그룹으로 나누어진다. 따라서 단 한 개의 변조 패턴으로 표현되어진다. 그리하여, 이들 9개의 그룹들은 Reed-Solomon 인코더로 피드백 되어진다. 9개의 정보 심벌에 기초한 인코더는 패리티 심벌을 추가한다. 각각은 7 비트의 길이를 가진다.

전체 메시지의 길이는 적합한 Reed-Solomon 코드의 세기에 의존한다. 결국, 각 심벌은 유일한 균형화 된 변조 패턴을 이용하여 Loran-C 신호로 변조되어 사용자에게 전송된다.

현재는, 14, 20 또는 30개의 GRI로 인코드된 9개의 정보 GRI를 포함하는 EUROFIX 메시지가 전송된다[9][10].

2.4 Reed-Solomon 부호의 부호화

Reed-Solomon 부호는 CD(Compact Disk), digital VCR, DAT(Digital Audio Tape) 및 HDB등의 시스템에서 오류 정정 부호로 사용되고 있다.

이 부호는 특수한 조건을 만족하는 비2진(nonbinry) BCH 부호이다.

즉, BCH 부호와 마찬가지로 길이가 q^m-1인 q^m-ary 부호어이다. 따라서 $GF(q^m)$에서 정의된다는 점등은 BCH 부호의 특성을 그대로 보여주고 있다.

Reed-Solomon Code 이론에 들어가기 전에 수학적 개념들을 정의한다. 그것들은 다음과 같다[7][8].

2.4.1 Finite Galois Field with q elements : GF(q)

GF(q)는 가감승제 할 수 있는 수의 집합이다. finite field의 예로는 정수를 소수 q로 나눈 나머지를 나타내는데, GF(2)={0,1}, GF(3)={0,1,2}, GF(5)={0,1,2,3,4} 등등이 된다. 우리는 $GF(2^m)$, m=1,2,3,4,....에 대해서 알아야 하는데 이것은 정수를 2^m으로 나눈 나머지이다.

$GF(2^m)$에서 중요한 것은 primitive element α 인데, 이 때의 $GF(2^m)$의 원소는 다음과 같다.

$$GF(2^m) = \{ 0, 1, \alpha, \alpha^2, \alpha^3, \cdots, \alpha^{2^m-2} \} \cdots (2.1)$$

2.4.2 Field F_2에서 m차 다항식

Field $GF(2) = F_2$에서

만일 $f_i \in F_2$이고, $\forall i,\ 0 \le i \le m,$ 그리고 $f_m \neq 0$이라고 하자. 이 때 다항식

$$f(x) = f_0 + f_1 x + f_2 x^2 + \cdots + f_m x^m \cdots\cdots\cdots\cdots\cdots (2.2)$$

를 field F_2에서의 m차의 다항식(Polynomial of m power)이라고 한다.

2.4.3 Primitive element

한 finite field $GF(q)$에서 0아닌 한 element g의 order가 $q-1$이면 g를 $GF(q)$의 primitive element라 한다. 따라서 primitive element의 전력(power)으로서 $GF(q)$의 0아닌 모든 element를 나타낼 수 있게 된다. 또한 모든 finite field는 primitive element를 갖고 있다.

차수가 m차인 primitive polynomial $p(x)$를 이용하여 element α^j를 다음과 같이 나타낸다.

$$\alpha^j = X^j \bmod p(x),\ (j = 0, 1, \cdots, 2^m - 2) \cdots\cdots\cdots\cdots (2.3)$$

2.4.4 다항식의 근

$f(\alpha)=0$일 때, 이 α를 다항식의 근(Polynomial's root)이라고 하고, $f(x)$가 원시 다항식이면 $\alpha^2, \alpha^{2^2}, \alpha^{2^3}, \cdots$도 $f(x)$의 근이 된다. 이 때 $\alpha^2, \alpha^{2^2}, \alpha^{2^3}, \cdots$를 α의 conjugate라고 한다.

2.4.5 RS 부호의 생성 다항식

$$g(x)=\prod_{i=1}^{2^m-1-k}(x+\alpha^{b+i})=\sum_{i=0}^{2t}g_i x^i \cdots\cdots\cdots\cdots\cdots (2.4)$$

여기서 α는 $GF(2^m)$의 primitive element이고 b는 양의 정수이다[2].

Field $GF(p)$에서의 (n,k) RS-Code 는 n개의 심벌(symbol)들을 원소로 갖고 있다. 여기서는 $p=2^m$인 경우만은 고려한다. $GF(2^m)$의 각 원소는 m개의 비트로 이루어져 있으며 다음과 같은 변수들로 구성되어진다.

첫째, 부호장 n은 $5\sim 2^m-1$ 심벌들로 구성되어 있고 이것을 비트(bit)로 바꾸면 $5m\sim m(2^m-1)$ 비트들로 이루어진다.

둘째, 정보장 k는 $n-2t$ 심벌들로 구성되며 즉, $m(n-2t)$ 비트들로 이루어진다.

여기서 t는 심볼 에러 정정을 말한다. 다음과 같이 표현된다.

$$t = \left\lfloor \frac{n-k}{2} \right\rfloor \qquad \cdots\cdots (2.5)$$

$\lfloor x \rfloor$는 x를 넘지 않는 최대 정수를 의미한다.

셋째, 검사장 $n-k$는 $2t$ 심벌들로 되어있고 이것은 $m2t$ 비트들로 되어있다.

넷째, 최소거리 $d_{\min}$은 $2t+1$이다[1][6].

다음 그림 2.4은 RS code를 이용하여 메시지를 부호화하는 과정을 나타낸 것이다.

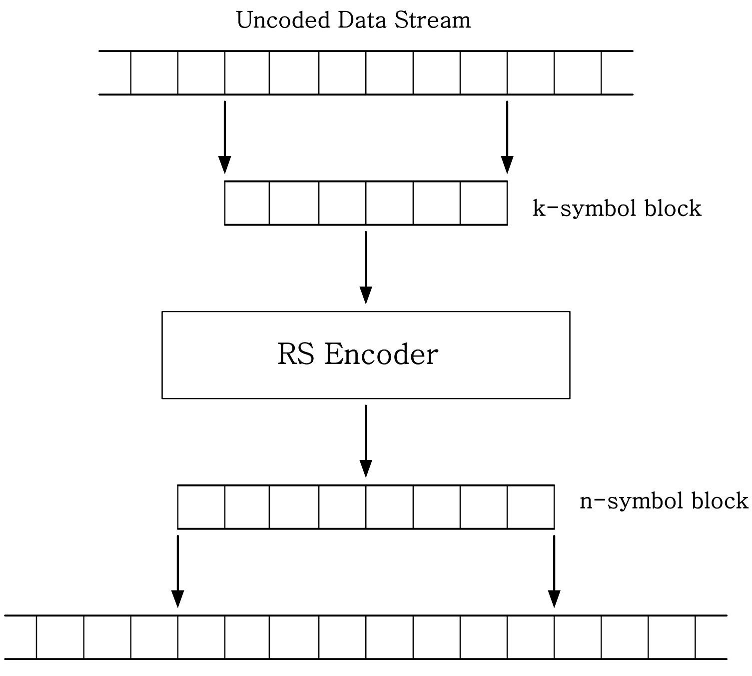

그림 2.4 RS code를 사용하여 부호화 하는 과정
Figure 2.4 Encoding using Reed-Solomon code.

2.5 유한체 Fourier 변환을 이용한 RS 부호의 복호법

2.5.1 RS 부호의 일반적인 복호법

수신벡터는 전송로상에서 잡음의 간섭여하에 따라 원래의 부호 벡터와 같을 수도 있고 다를 수도 있다.

전송한 부호다항식 $c(x)$, 수신된 부호다항식을 $r(x)$라 하면 다음과 같이 표현된다.

$$r(x) = g(x)q(x) + s(x) \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots (2.6)$$

여기서, $q(x)$는 몫다항식이고, $s(x)$는 나머지 잉여다항식이다.

식 (2.6)에서 $s(x)$를 오증(syndrom)이라 하며, $n-k-1$차 이하가 된다. 오증 $s(x)$가 영(zero)일때는 $r(x)$가 $g(x)$의 배수이므로 $r(x) = c(x)$가 되어 전송벡터와 일치하게 된다.

그러나, $s(x) \neq 0$일 때는 전송로의 불량으로 인해 원래의 부호다항식이 변형된 상태 즉, $r(x) \neq c(x)$이므로 오류정정을 시행해야 한다.

일반적으로 RS 부호의 복호 순서를 정리하면 다음과 같다.

첫째, 수신 벡터 $r(x)$로부터 오증 $s = (s_0, s_1, \cdots, s_{2t})$, 를 계산한다(syndrome computation).

둘째, 알고리즘을 이용하여 오증 요소 s_i, $(1 \leq i \leq 2t)$로부터 오류위치다항식을 구한다(error location polynomial).

셋째, 오류위치 다항식의 근을 구함으로써 오류위치 (error location) 및 오류치 (error value)를 계산한다.

넷째, 오류형태 $e(x)$를 결정하고 $c(x) = r(x) + e(x)$에 의해 오류를 정정한다(error correction).

다음 그림 2.5는 일반적인 RS 부호의 부호과정을 나타낸 것이다[2][5].

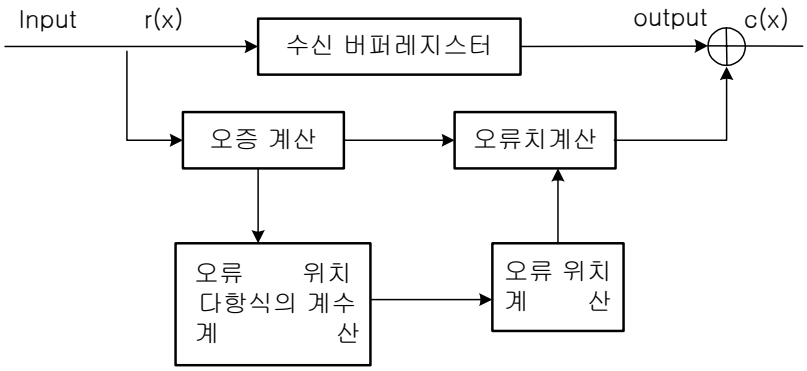

그림 2.5 일반적인 RS 부호의 복호기

Figure 2.5 Decoder of general RS code.

2.5.2 유한체의 Fourier 변환

RS 부호의 복호법은 여러 가지가 있다. 그 중 본 논문에서는 유한체 Fourier 변환을 이용한 복호법을 사용하였다.

Fourier 변환은 유한체 $GF(q)$의 n차원 벡터 공간에서도 정의된다.

실수 벡터의 이산 Fourier변환(DFT)은 $e^{-j2\pi/n}$을 변환의 중심으로 하여 정의된다.

실수벡터를 $v=(v_o, v_1, \cdots\cdots v_{n-1})$이라 하고, 변환벡터를 $V=(V_o, V_1, \cdots\cdots V_{n-1})$이라 하면 두 벡터의 관계는 다음과 같다.

$$V_k = \sum_{i=0}^{n-1} e^{-j2\pi ik/n} v_i, \quad 0 \le k \le n-1 \quad \cdots\cdots\cdots\cdots\cdots (2.7)$$

같은 방법으로 유한체 원소를 성분으로 갖는 n차원 벡터에 대하여 유한체의 Fourier변환을 다음과 같이 정의할 수 있다.

$c=\{c_i | i=0,1,\cdots,n-1\}$를 $GF(q)$상의 $n\ (n=q^m-1)$차원 벡터라 하고, α를 $GF(q^m)$내의 위수가 n인 원소라 할 때 c에 대한 유한체 Fourier 변환 $C=\{C_j | j=0,1,\cdots,n-1\}$는 다음과 같이 정의된다.

$$C_j = \sum_{i=0}^{n-1} \alpha^{ij} c_i \ , \quad 0 \le j \le n-1 \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots (2.8)$$

식 (2.8)에서 두 벡터 c와 C는 변환쌍이며 c를 시간영역함수, C를 변환영역함수 또는 Spectrum 이라 한다.

일반적으로 $GF(q)$상의 벡터를 변환하면 확대체 (extension field)인 $GF(q^m)$상의 원소로 변환영역 벡터가 구성되며, 벡터와는 다항식 형태로 다음과 같이 표현할 수 있다.

$$c(x) = \sum_{i=0}^{n-1} c_i x^i = c_0 + c_1 x + c_2 x^2 + \cdots + c_{n-1} x^{n-1} \quad \cdots\cdots (2.9)$$

$$C(x) = \sum_{j=0}^{n-1} c_j x^j = C_0 + C_1 x + C_2 x^2 + \cdots + C_{n-1} x^{n-1} \quad \cdots\cdots (2.10)$$

이 때 식 (2.9)과 식 (2.10)의 각 다항식 계수는 다음과 같다.

$$c_i = \sum_{j=0}^{n-1} C_j (\alpha^{-i})^j = C(\alpha^{-i}),\ 0 \le i \le n-1 \quad \cdots\cdots (2.11)$$

$$C_j = \sum_{i=0}^{n-1} c_i (\alpha^j)^i = C(\alpha^j)\ ,\ 0 \le i \le n-1 \quad \cdots\cdots (2.12)$$

식 (2.11)와 식 (2.12)으로부터 $C_j = 0$이면 $c(x) = \alpha^j$를 근으로 갖고, 역으로 $C_i = 0$이면 $c(x) = \alpha^{-i}$를 근으로 갖는다는 것을 의미한다.

2.5.3 유한체 Fourier 변환을 이용한 RS 부호의 복호과정

수신벡터는 부호벡터와 오류벡터의 합이므로 다음 식으로 표현된다.

$$r_i = c_i + e_i\ ,\ 0 \le i \le n-1 \quad \cdots\cdots (2.13)$$

$$R_j = C_j + E_j\ ,\ 0 \le i \le n-1 \quad \cdots\cdots (2.14)$$

위의 식 (2.13)과 식 (2.14)에서 $C_j = c(\alpha^j)$이므로 $c(x)$가 $\alpha, \alpha^2, \cdots, \alpha^{2t}$등의 근을 갖는 부호다항식이면 $c(x)$의 계수 중, $c_1, c_2, \cdots, c_{2t}$는 <0>이다. 따라서, 식 (2.15)은 다음과 같다.

$$R_j = E_j = e(\alpha^j) \ , \ 1 \le j \le 2t \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots (2.15)$$

수신 다항식 중 $R_1, R_2, \cdots, R_{2t}$는 오류 형태의 변환만으로 표현할 수 있다. 또한 식 (2.16)는 다음과 같이 된다.

$$S_j = R_j = E_j \ , \ 1 \le j \le 2t \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots (2.16)$$

그림 2.6은 이러한 관계를 나타낸 것이다.

c_0	0	• • •	0	c_{2t-1}	• • •	c_{n-1}
E_0	E_i	• • •	E_{2t}	E_{2t+1}	• • •	E_{n-1}
R_0	R_i	• • •	R_{2t}	R_{2t+1}	• • •	R_{n-1}
0	S_i	• • •	S_{2t}	0	• • •	0

그림 2.6 변환 영역에서의 부호계열, 오류, 수신계열간의 관계
Figure 2.6 Connection of code, error, syndrom in transformation domain.

만일 $v \le t$인 v개의 오류가 $i_k (k = 1, 2, \cdots, v)$의 위치에 발생하였다면 변환영역에서의 오류 위치다항식은 다음과 같이 정의된다.

$$\Phi(x) = \prod_{k=1}^{v} (1 - x\alpha^{i_k}), \quad (0 \le i_k \le n-1)$$

$$= 1 + \Phi_1 x + \Phi_2 x^2 + \cdots + \Phi_v x^v \cdots\cdots\cdots\cdots\cdots (2.17)$$

식 (2.17)을 시간영역에서 오류위치다항식으로 표현하면 다음과 같다.

$$\sigma_i = \sum_{j=1}^{n-1} \Phi_j \alpha^{-ij} = \Phi(\alpha^{-i})$$

$$= \prod_{k=1}^{v} (1 - \alpha^{-i}\alpha^{i_k}), \quad (0 \le i \le n-1) \cdots\cdots\cdots\cdots (2.18)$$

식 (2.18)에서 $i = i_k$이면 $\sigma = 0$이고, $i \neq i_k$이면 $\sigma \neq 0$이므로 다음 관계가 성립한다.

$$\sigma_i e_i = 0 \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots (2.19)$$

식 (2.19)은 시간영역에서의 곱이므로 변환영역에서는 Convolution과 같으므로 다음으로 표현되게 된다.

$$\Phi * E = 0 \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots (2.20)$$

또한 $\Phi_0 = 1$이고, $\Phi_j = 0 \ (j > t)$이므로 식 (2.20)는 다시 다음과 같이 표현된다.

$$E_j = -\sum_{k=1}^{t} \Phi_t E_{j-k} \ , \ 1 \le j \le n-1 \cdots\cdots\cdots\cdots\cdots (2.21)$$

식 (2.21)은 $\Phi(x)$의 계수와 E의 성분으로 된 n개의 방정식 집합이다.

이중 t개의 방정식은 다음과 같이 나타낸다.

$$S_j = -\sum_{k=1}^{t} \Phi_k S_{j-k} \ , \ t+1 \le j \le 2t \cdots\cdots\cdots\cdots\cdots (2.22)$$

식 (2.22)에서 Φ_k는 식 (2.21)에 표현된 것처럼 오증

$E_j(1 \le k \le t)$와 $E_k(0 \le k \le t)$로만 구성되므로 구할 수 있다. 따라서 오증 이외의 E성분들도 식 (2.21)를 이용하여 모두 구할 수 있다.

모든 E_j가 구해지면 $C_j = R_j - E_j(0 = j \le n-1)$이므로 변환영역에서 복호가 완료되며 이를 다시 역변환하면 시간영역에서의 복호가 종료된다. 이러한 일련의 과정을 거쳐 오류를 정정하는 장치를 변환복호기라 하며 그림 2.7는 변환복호기의 블록도이다[3][4][5].

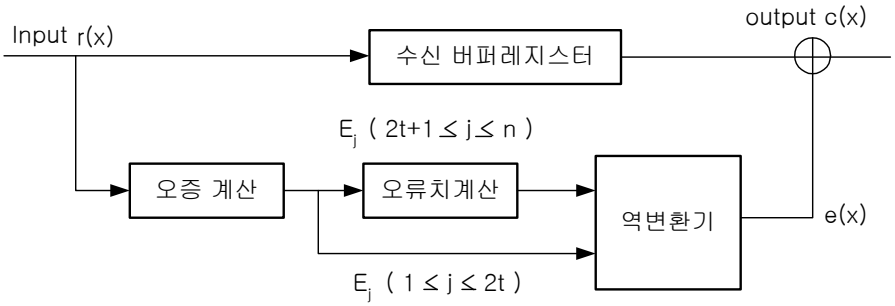

그림 2.7 변환 복호기

Figure 2.7 Transformation Decoder.

제 3 장 EUROFIX 시스템에서 RS Code 부호화

3.1 $GF(2^7)$에서 (14, 9) 2중 오류 정정 RS 부호 과정

본 논문에서는 EUROFIX에서 사용되는 $GF(2^7)$에서 부호화의 최대 128개까지의 심벌들 중에서 NELS에서 테스트 모델로 이용한 (14, 9), (20, 9), (30, 9)의 3 모델 중 가장 간단하게 구현할 수 있는 (14, 9)를 선택하여 2중 오류 정정 RS 부호기를 설계 하였다.

다음 그림 3.1는 부호화 과정을 나타낸 것이다.

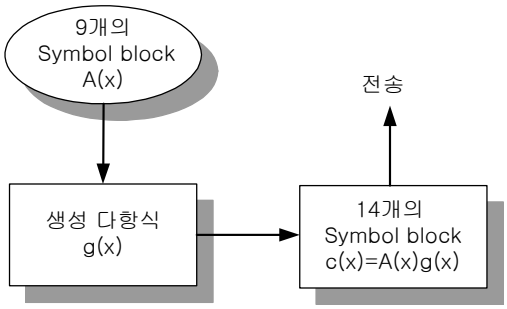

그림 3.1 RS 부호 과정

Figure 3.1 Encoding of Reed-Solomon code.

그림 3.1에서 보듯이 먼저 생성 다항식을 결정해야 한다.

$GF(2^7)$에서 사용된 원시 다항식은 다음과 같다.

$$p(\alpha) = 1 + \alpha^3 + \alpha^7 = 0 \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots (3.1)$$

식 (3.1)의 $GF(2^7)$에서의 다항식, 벡터 표현들은 부록을 참조한다.

식 (2.4)로부터 생성다항식은 다음과 같이 표현된다.

$$g(x) = (x+\alpha)(x+\alpha^2)(x+\alpha^3)(x+\alpha^4)$$
$$= \alpha^{10} + \alpha^{99}x + \alpha^{41}x^2 + \alpha^{94}x^3 + x^4 \cdots\cdots\cdots\cdots (3.2)$$

또한, 단일 입력 심벌은 다음과 같다.

$$A(\alpha) = a_0 + a_1\alpha + a_2\alpha^2 + a_3\alpha^3 + a_4\alpha^4 + a_5\alpha^5 + a_6\alpha^6$$
$$a_i \in GF(2) \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots (3.3)$$

식 (3.2)와 식 (3.3)의 각 계수와의 곱은 다음과 같이 나타난다.

$$\alpha^{10}A(\alpha) = (a_1 + a_4 + a_5) + (a_2 + a_5 + a_6)\alpha + (a_3 + a_6)\alpha^2$$
$$+ (a_0 + a_1 + a_5)\alpha^3 + (a_1 + a_2 + a_6)\alpha^4$$
$$+ (a_2 + a_3)\alpha^5 + (a_0 + a_3 + a_4)\alpha^6 \cdots\cdots\cdots\cdots (3.4)$$

$$\alpha^{99}A(\alpha) = (a_0 + a_1 + a_2 + a_3 + a_4)$$
$$+ (a_0 + a_1 + a_2 + a_3 + a_4 + a_5)\alpha$$
$$+ (a_0 + a_1 + a_2 + a_3 + a_4 + a_5 + a_6)\alpha^2$$
$$+ (a_0 + a_5 + a_6)\alpha^3 + (a_0 + a_1 + a_6)\alpha^4$$
$$+ (a_0 + a_1 + a_2)\alpha^5 + (a_0 + a_1 + a_2 + a_3)\alpha^6$$
$$\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots (3.5)$$

$$\alpha^{41}A(\alpha)=(a_0+a_1+a_3+a_5)+(a_1+a_2+a_4+a_6)\alpha$$
$$+(a_2+a_3+a_5)\alpha^2+(a_1+a_4+a_5+a_6)\alpha^3$$
$$+(a_0+a_2+a_5+a_6)\alpha^4+(a_1+a_3+a_6)\alpha^5$$
$$+(a_0+a_2+a_4)\alpha^6 \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots (3.6)$$

$$\alpha^{94}A(\alpha)=(a_3+a_4+a_5+a_6)+(a_0+a_4+a_5+a_6)\alpha^2$$
$$+(a_0+a_1+a_2+a_3+a_4+a_5)\alpha^3$$
$$+(a_0+a_1+a_2+a_3+a_4+a_5+a_6)\alpha^4$$
$$+(a_1+a_2+a_3+a_4+a_5+a_6)\alpha^5$$
$$+(a_2+a_3+a_4+a_5+a_6)\alpha^6 \cdots\cdots\cdots\cdots\cdots (3.7)$$

3.2 (14, 9) RS 부호기 설계

위의 식 (3.4),(3.5),(3.6),(3.7)들을 이용하여 (14, 9) RS 부호기를 설계하였다.

그림 3.2와 같이 나타난다.

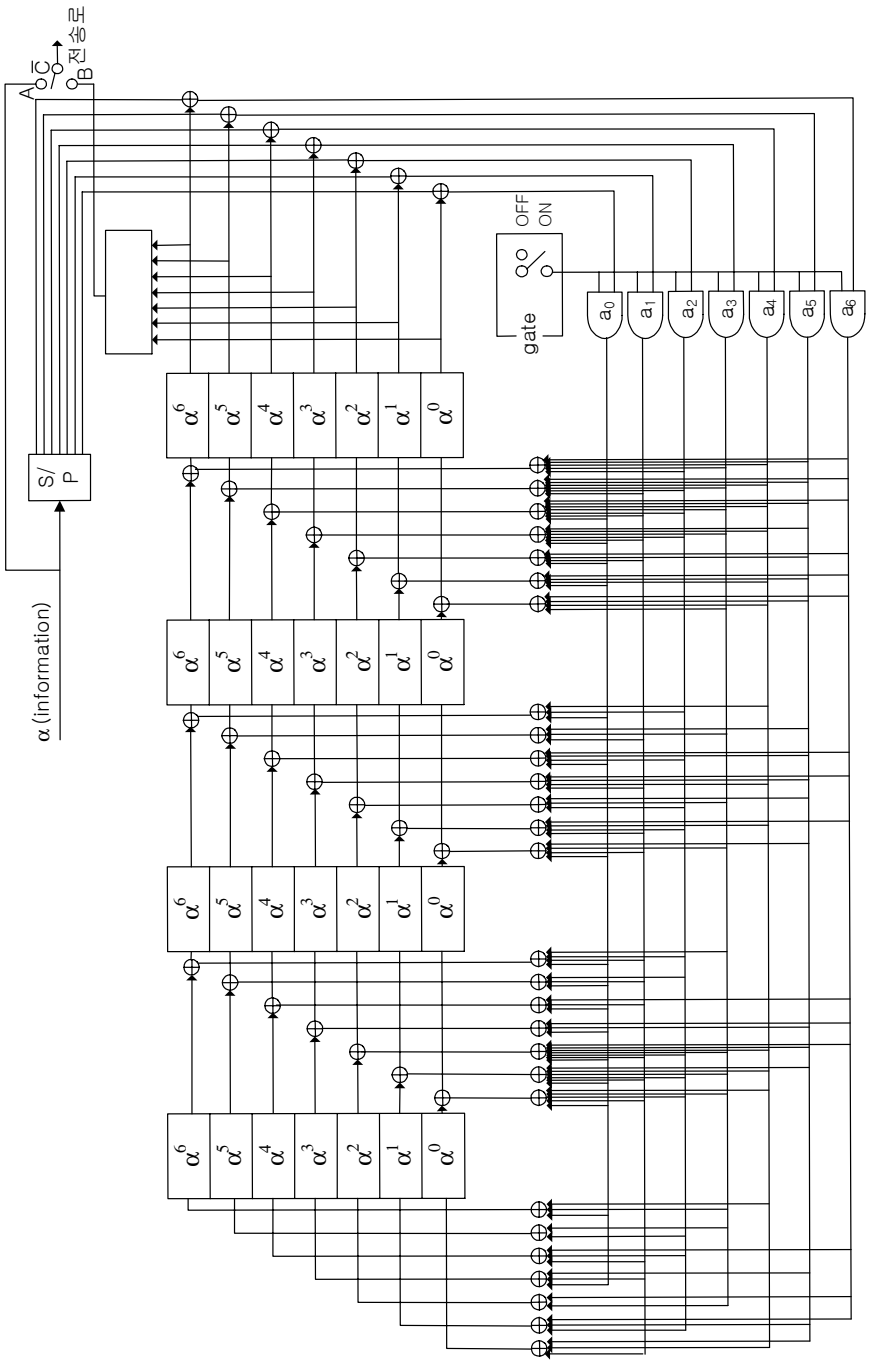

Figure 3.2 (14, 9) Encoder of RS code.

3.3 RS Encoding 시뮬레이션

위의 그림 3.2와 같이 설계된 부호기를 시뮬레이션 해 보았다.

Galois Field $GF(2^7)$에서 나올 수 있는 최대의 n값 127을 사용하였다. 그 결과는 그림 3.4와 같이 나타난다. 그림 3.4에 대한 설명은 다음의 그림 3.3과 같다.

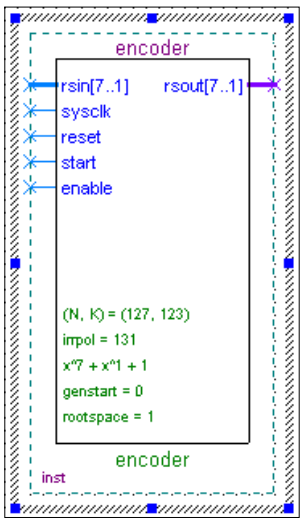

그림 3.3 RS encoder
Figure 3.3 RS encoder.

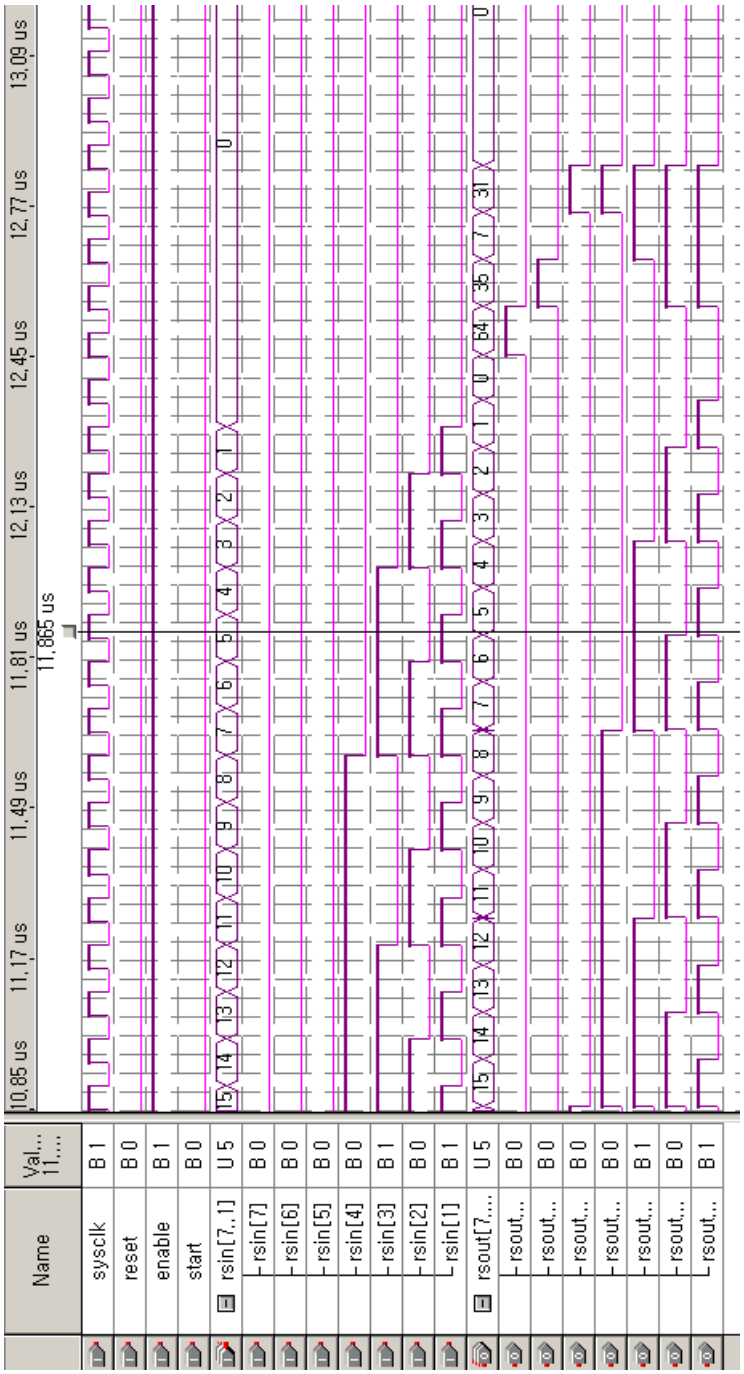

그림 3.4 RS 부호화 시뮬레이션 결과
Figure 3.4 Results of RS encoding's simulation.

제 4 장 EUROFIX 시스템에서 RS Code 복호화

본 논문에서는 EUROFIX의 특성에 맞추어 $GF(2^7)$에서의 (14, 9) 2중 오류 정정 RS Code 복호기를 설계하였다.

또한, RS 복호 방법 중, 유한체 Fourier 변환을 이용한 복호법을 이용하였다[1][5].

4.1 (14, 9) RS 부호의 변환 복호법을 이용한 복호 과정

(14, 9) RS 부호의 오증은 수신계열로부터 구할 수 있으며 $2t=4$개의 오증 요소는 다음과 같다.

$$S_1 = E_1 = \sum_{i=0}^{13} \alpha^i r^i = r(\alpha)$$

$$S_2 = E_2 = \sum_{i=0}^{13} \alpha^{2i} r^i = r(\alpha^2)$$

$$S_3 = E_3 = \sum_{i=0}^{13} \alpha^{3i} r^i = r(\alpha^3)$$

$$S_4 = E_4 = \sum_{i=0}^{13} \alpha^{4i} r^i = r(\alpha^4) \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots (4.1)$$

그 중 오류가 발생한 경우 식 (2.21)로부터 다음과 같이 나타난다.

$$E_j = -\sum_{k=1}^{2} \Phi_k E_{j-k}\ , \quad 0 \le j \le 13 \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots (4.2)$$

$j=3,4$일 때 E_3 , E_4는 각각 다음과 같다.

$$-E_3 = \Phi_1 E_2 + \Phi_2 E_1 \cdots\cdots (4.3)$$

$$-E_4 = \Phi_1 E_3 + \Phi_2 E_2 \cdots\cdots (4.4)$$

식 (4.3), (4.4)을 행렬형으로 표현하면 다음과 같다.

$$\begin{bmatrix} \Phi_1 \\ \Phi_2 \end{bmatrix} = \frac{1}{E_1 E_3 + E_2^{\,2}} \begin{bmatrix} E_3 & -E_2 \\ -E_2 & E_1 \end{bmatrix} \cdot \begin{bmatrix} -E_3 \\ -E_4 \end{bmatrix} \cdots (4.5)$$

식 (4.5)에서 Φ_i, Φ_2를 구하면 다음과 같다.

$$\Phi_1 = \frac{E_2 E_3 - E_1 E_4}{E_1 E_3 + E_2^{\,2}} \cdots\cdots (4.6)$$

$$\Phi_2 = \frac{E_2 E_4 - E_3^{\,2}}{E_1 E_3 + E_2^{\,2}} \cdots\cdots (4.7)$$

이러한 계산과정은 그림 4.1과 같이 설계함으로써 오증으로부터 지연없이 오류위치 다항식의 계수를 구할 수 있다.

Ⓜ은 $GF(2^7)$상의 승산기, Ⓘ는 $GF(2^7)$상의 역원기를 나타낸다.

식 (4.2)를 이용하여 오증과 오류위치 다항식의 계수로부터 변환영역에서의 오류를 모두 구할 수 있다. 식 (4.8)은 단일 오류, 식 (4.9)는 2중 오류의 경우이다.

$$-E_5 = \Phi_1 E_4, \quad -E_6 = \Phi_1 E_5$$

$$\vdots$$

$$-E_{12} = \Phi_1 E_{11}, \quad -E_{13} = \Phi_1 E_{12}$$

$$-E_0 = \Phi_1 E_{13} \cdots\cdots (4.8)$$

$$-E_5 = \Phi_1 E_4 + \Phi_2 E_3$$

$$-E_6 = \Phi_1 E_5 + \Phi_2 E_4$$

$$\vdots$$

$$-E_{13} = \Phi_1 E_{12} + \Phi_2 E_{11}$$

$$-E_0 = \Phi_1 E_{13} + \Phi_2 E_{12} \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots (4.9)$$

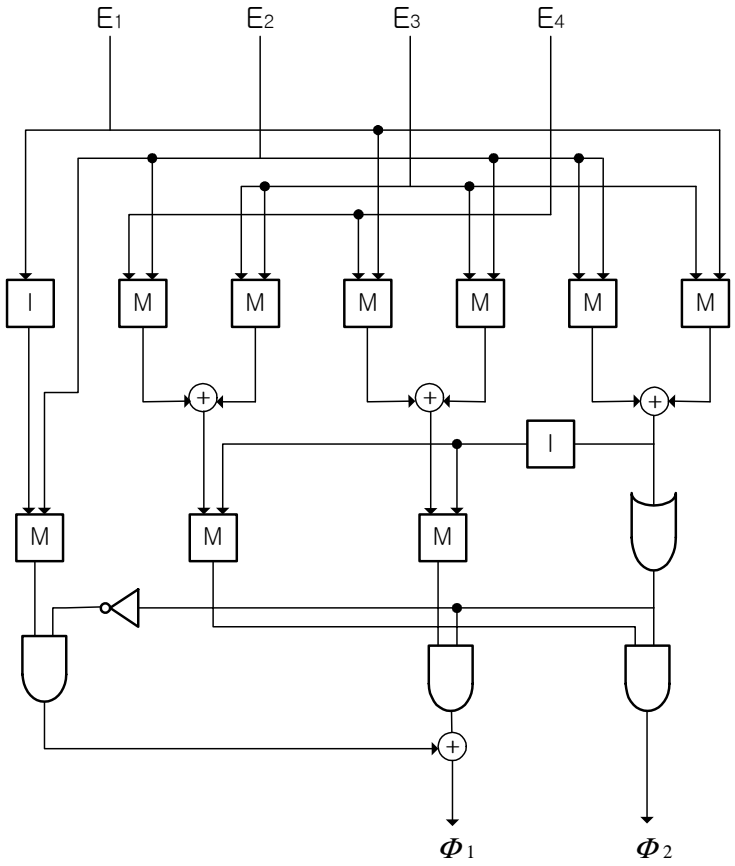

그림 4.1 오류 위치 다항식의 계수결정 회로
Figure 4.1 Circuit of coefficient determination for polynomial of error location.

그림 4.2는 식 (4.8)과 식 (4.9)를 수행하는 회로를 설계한 것으로 수신 보호의 마지막 심벌이 복호기에 입력되는 순간 가장 빠르게 모든 오류를 얻을 수 있는 회로로서 소자들의 지연시간 후에는 바로 복호된 신호를 출력할 수 있도록 설계한 것이다.

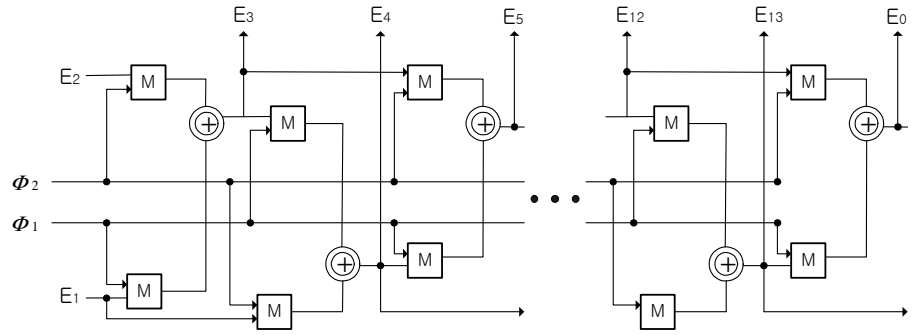

그림 4.2 변환 영역에서의 오류구성회로
Figure 4.2 Circuit of error constitution in transformation domain.

변환영역에서의 오류를 역변환시키므로써 시간영역에서의 오류 $e_i(0 \le i \le 14)$를 차례로 얻을 수 있다. 즉, 식 (2.11),(2.12)을 이용하면 다음과 같이 나타난다.

$$e_{13} = \sum_{j=0}^{13} \alpha^{-13j} E_j = E(\alpha^{-13}) = E(\alpha) \cdots\cdots\cdots\cdots (4.10)$$

$$e_{12} = \sum_{j=0}^{13} \alpha^{-12j} E_j = E(\alpha^{-12}) = E(\alpha^{2})$$

$$\vdots$$

$$e_1 = \sum_{j=0}^{13} \alpha^{-j} E_j = E(\alpha^{-13}) = E(\alpha^{13})$$

$$e_0 = \sum_{j=0}^{13} E_j = E(\alpha^{-13}) = E(1) \cdots\cdots\cdots\cdots\cdots\cdots (4.11)$$

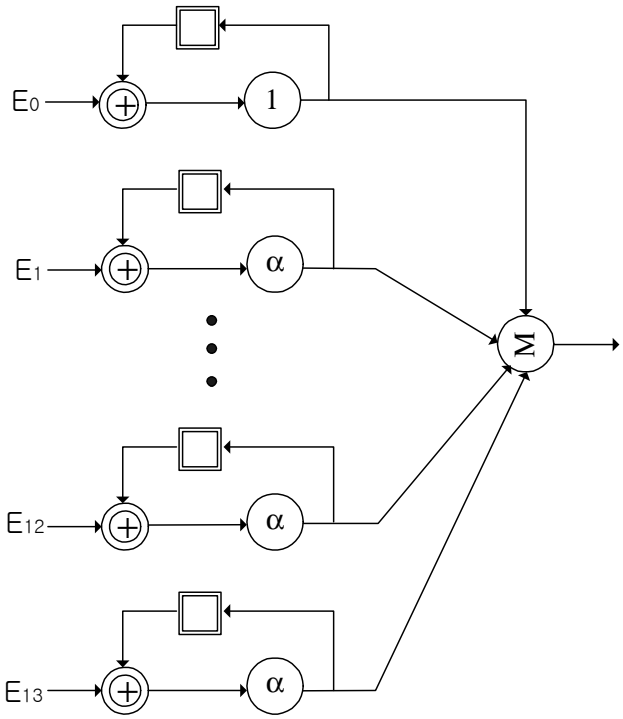

그림 4.3 역변환기 회로

Figure 4.3 Circuit of Inverse transform.

그림 4.3은 역변환 과정을 위한 회로를 설계한 것이다. 변환 영역의 모든 오류가 입력되는 수간 e_{14}가 계산되어 출력되고, 동시에 귀환되어 filp-flop에 저장된다. 순차적 과정에 e_0이 출력되는 순간 모든 flip-flop을 clear 하여 다음 수신계열의 오류를 받을 준비를 한다.

4.2 (14, 9) RS 부호의 변환 복호기 설계

(14, 9) 2중 오류 정정 RS 부호의 변환복호기의 블록도는 그림 4.4와 같다. 14단의 수신계열 저장 레지스터에 마지막 심벌이 저장되는 순간 E_1, E_2, E_3, E_4가 계산되는 동시에 Φ_1, Φ_2가 출력된다. 그리고 Φ_1, Φ_2, E_1, E_2를 이용하여 $E_5, E_6, \cdots E_{13}, E_0$를 구하고 이를 역변환 시켜 $e_{13}, e_{12}, \cdots e_1, e_0$를 순서대로 출력하여 오류 정정을 수행한다.

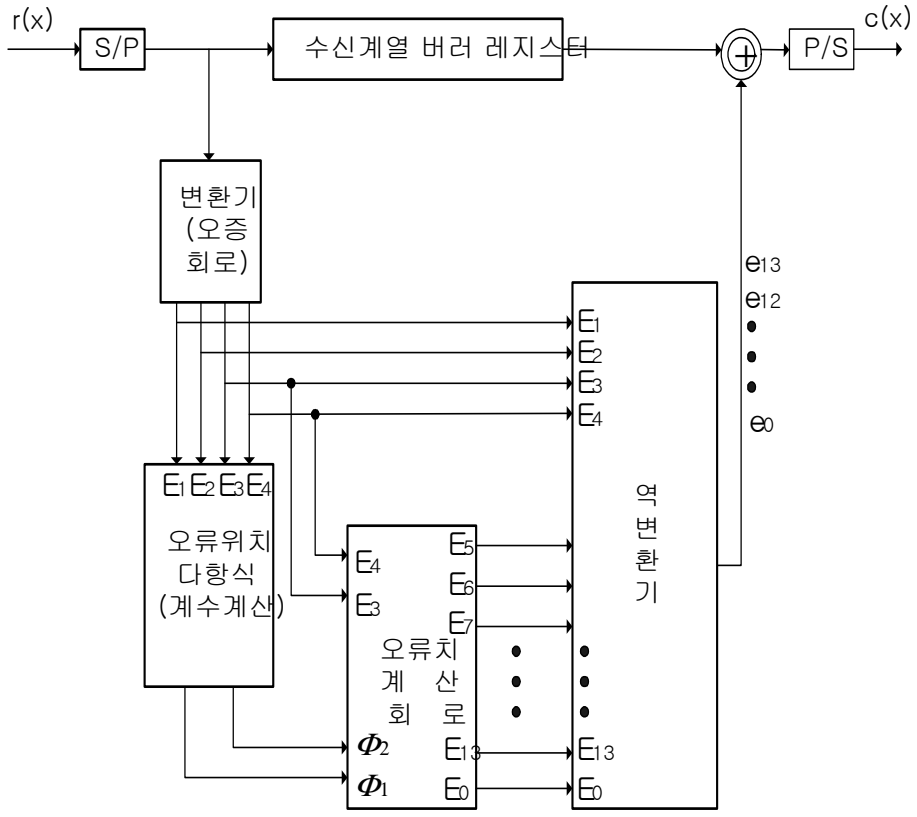

그림 4.4 (14, 9) RS 부호의 변환 복호기

Figure 4.4 (14, 9) Transformation Decoder of RS code.

4.3 RS decoding 시뮬레이션

그림 4.4에서 설계한 RS 복호기를 시뮬레이션 해 보았다.

본 논문에서 사용한 $GF(2^7)$의 (14, 9)를 사용하였고 그 결과는 그림 4.6과 같다. 또한 그림 4.6에 대한 설명은 그림 4.5에 나타난다.

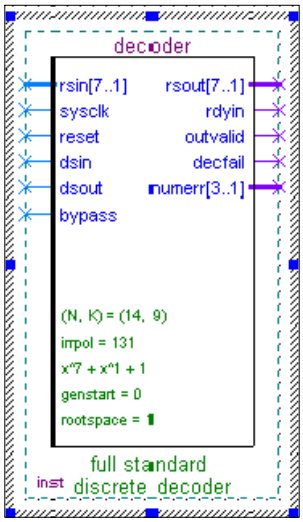

그림 4.5 RS decoder

Figure 4.5 RS decoder.

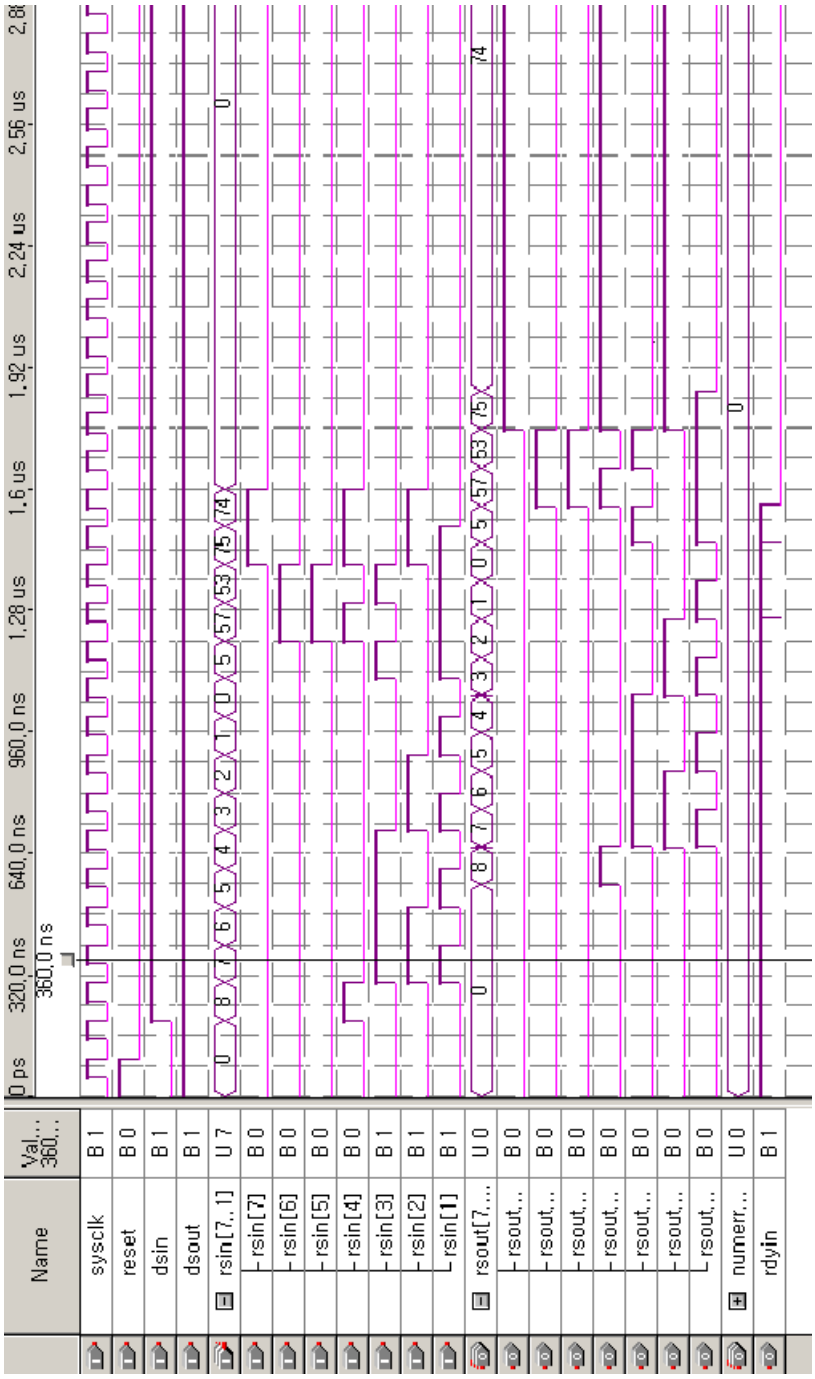

그림 4.6 RS 복호화 시뮬레이션 결과
Figure 4.6 Results of RS decoding's simulation.

제 5 장 결 론

여기서 구현한 RS 부호의 복호법은 유한체 Fourier 변환을 적용한 복호법이다.

Fourier 변환을 부호 이론에 적용하면 전자통신공학의 여러 분야에서 이미 사용되고 있어 부호이론에 대한 접근이 용이하고, 변환영역에서 복호기의 구조와 실현이 시간 영역의 경우보다 간단해 진다.

또한, 다른 일반 복호 방법에 비해 복호 시간을 단축 처리할 수 있게 된다. 즉, 변환 복호 방법은 유한체 $GF(2^m)$의 크기에 무관하게 복호가 완료되는 반면, 일반적인 복호 방법의 경우에는 $GF(2^m)$상의 차수(order)인 m의 크기에 따라 민감하게 변화된다.

따라서, 본 논문에서는 다른 일반 복호법에 비해 많은 장점을 가진 유한체 Fourier 변환 복호법을 적용한 EUROFIX 시스템을 제안하였다.

그리고 EUROFIX의 데이터 전송에 적용되는 Reed-Solomon Code에서 7비트를 하나의 심벌로 보내는 (14, 9)을 기본 모델로 하여 부호기와 복호기를 설계하였으며 시뮬레이션을 통하여 성능의 우수성을 확인 하였다.

앞으로 연구할 과제는 다양한 알고리즘의 분석과 제작을 통하여 시뮬레이션과 측정한 데이터의 비교 분석이다.

참고 문헌

[1] M.Y.Rhee, *Error Correcting Coding Theory,* McGraw-Hill, New York, 1989.

[2] A. Hocquenghem, "Codes correcteurs d'erreurs",*Chifferes,* Vol.2, pp.147～156, 1959.

[3] R. C. Bose and D. K. Ray-Chaudhuri, "On a Class of Error-correcting Binary Group codes," *Information and control,* vol3, pp.68～79, 1960.

[4] Blahut, R.E., "Transform Techniques for Error-control Codes," *IBM J. Res,* Dev., 23, pp.299～315, 1970.

[5] 이만영, BCH 부호와 Reed-Solomon 부호, 민음사, 1990.

[6] http://www.altera.com/products/ip/altera/m-ham-rs-dec-mini.html

[7] 이문호, 실용 정보이론, 복두 출판사, 1998.

[8] 강창언, 디지털통신입문, 북두 출판사, pp. 168～180, 1996.

[9] Willigen, D. van, E.J Breeuwer, G.W.A Offermans, J. Sierenveld and J. de Zwart, "EUROFIX Information

Paper," *TVS Memorandum*, No, REP9606A, June, 1996.

[10] Willigen, D. van, "Eurofix," *The Journal of Navigation*, Vol. 42, No. 3, September 1989.

[11] Willigen, D. van, "*Eurofix : Differential Hybridized Integrated Navigation,*" Proceedings of the 18th Annual Technical Meeting of the Wild Goose Association , Hyannis, MA, October 29-November 1, 1989.

부 록

$p(\alpha) = 1 + \alpha^3 + \alpha^7 = 0$인 Galois Field $GF(2^7)$

멱	다항식 표현	벡터 표현
0		0000000
α^0	1	1000000
α^1	α	0100000
α^2	α^2	0010000
α^3	α^3	0001000
α^4	α^4	0000100
α^5	α^5	0000010
α^6	α^6	0000001
α^7	$1 + \alpha^3$	1001000
α^8	$\alpha + \alpha^4$	0100100
α^9	$\alpha^2 + \alpha^5$	0010010
α^{10}	$\alpha^3 + \alpha^6$	0001001
α^{11}	$1 + \alpha^3 + \alpha^4$	1001100
α^{12}	$\alpha + \alpha^4 + \alpha^5$	0100110
α^{13}	$\alpha^2 + \alpha^5 + \alpha^6$	0010011
α^{14}	$1 + \alpha^6$	1000001
α^{15}	$1 + \alpha + \alpha^3$	1101000
α^{16}	$\alpha + \alpha^2 + \alpha^4$	0110100
α^{17}	$\alpha^2 + \alpha^3 + \alpha^5$	0011010
α^{18}	$\alpha^3 + \alpha^4 + \alpha^6$	0001101
α^{19}	$1 + \alpha^3 + \alpha^4 + \alpha^5$	1001110
α^{20}	$\alpha + \alpha^4 + \alpha^5 + \alpha^6$	0100111
α^{21}	$1 + \alpha^2 + \alpha^3 + \alpha^5 + \alpha^6$	1011011
α^{22}	$1 + \alpha + \alpha^4 + \alpha^6$	1100101
α^{23}	$1 + \alpha + \alpha^2 + \alpha^3 + \alpha^5$	1111010
α^{24}	$\alpha + \alpha^2 + \alpha^3 + \alpha^4 + \alpha^6$	0111101
α^{26}	$\alpha + \alpha^3 + \alpha^5 + \alpha^6$	0101011
α^{26}	$\alpha + \alpha^3 + \alpha^5 + \alpha^6$	0101011
α^{27}	$1 + \alpha^2 + \alpha^3 + \alpha^4 + \alpha^6$	1011101
α^{28}	$1 + \alpha + \alpha^4 + \alpha^5$	1100110
α^{29}	$\alpha + \alpha^2 + \alpha^5 + \alpha^6$	0110011
α^{30}	$1 + \alpha^2 + \alpha^6$	1010001
α^{31}	$1 + \alpha$	1100000
α^{32}	$\alpha + \alpha^2$	0110000
α^{33}	$\alpha^2 + \alpha^3$	0011000
α^{34}	$\alpha^3 + \alpha^4$	0001100
α^{35}	$\alpha^4 + \alpha^5$	0000110
α^{36}	$\alpha^5 + \alpha^6$	0000011
α^{37}	$1 + \alpha^3 + \alpha^6$	1001001
α^{38}	$1 + \alpha + \alpha^3 + \alpha^4$	1101100
α^{39}	$\alpha + \alpha^2 + \alpha^4 + \alpha^5$	0110110
α^{40}	$\alpha^2 + \alpha^3 + \alpha^5 + \alpha^6$	0011011

멱	다항식 표현	벡터 표현
α^{41}	$1 + \alpha^4 + \alpha^6$	1000101
α^{42}	$1 + \alpha + \alpha^3 + \alpha^5$	1101010
α^{43}	$\alpha + \alpha^2 + \alpha^4 + \alpha^6$	0110101
α^{44}	$1 + \alpha^2 + \alpha^5$	1010010
α^{45}	$\alpha + \alpha^3 + \alpha^6$	0101001
α^{46}	$1 + \alpha^2 + \alpha^3 + \alpha^4$	1011100
α^{47}	$\alpha + \alpha^3 + \alpha^4 + \alpha^5$	0101110
α^{48}	$\alpha^2 + \alpha^4 + \alpha^5 + \alpha^6$	0010111
α^{49}	$1 + \alpha^5 + \alpha^6$	1000011
α^{50}	$1 + \alpha + \alpha^3 + \alpha^6$	1101001
α^{51}	$1 + \alpha + \alpha^2 + \alpha^3 + \alpha^4$	1111100
α^{52}	$\alpha + \alpha^2 + \alpha^3 + \alpha^4 + \alpha^5$	0111110
α^{53}	$\alpha^2 + \alpha^3 + \alpha^4 + \alpha^5 + \alpha^6$	0011111
α^{54}	$1 + \alpha^4 + \alpha^5 + \alpha^6$	1000111
α^{55}	$1 + \alpha + \alpha^2 + \alpha^3 + \alpha^4 + \alpha^5 + \alpha^6$	1101011
α^{56}	$1 + \alpha + \alpha^3 + \alpha^5 + \alpha^6$	1111101
α^{57}	$1 + \alpha + \alpha^2 + \alpha^4 + \alpha^5$	1110110
α^{58}	$\alpha + \alpha^2 + \alpha^3 + \alpha^5 + \alpha^6$	0111011
α^{59}	$1 + \alpha^2 + \alpha^4 + \alpha^6$	1010101
α^{60}	$1 + \alpha + \alpha^5$	1100010
α^{61}	$\alpha + \alpha^2 + \alpha^6$	0110001
α^{62}	$1 + \alpha^2$	1010000
α^{63}	$\alpha + \alpha^3$	0101000
α^{64}	$\alpha^2 + \alpha^4$	0010100
α^{65}	$\alpha^3 + \alpha^5$	0001010
α^{66}	$\alpha^4 + \alpha^6$	0000101
α^{67}	$1 + \alpha^3 + \alpha^5$	1001010
α^{68}	$\alpha + \alpha^4 + \alpha^6$	0100101
α^{69}	$1 + \alpha^2 + \alpha^3 + \alpha^5$	1011010
α^{70}	$\alpha + \alpha^3 + \alpha^4 + \alpha^6$	0101101
α^{71}	$1 + \alpha^2 + \alpha^3 + \alpha^4 + \alpha^5$	1011110
α^{72}	$\alpha + \alpha^3 + \alpha^4 + \alpha^5 + \alpha^6$	0101111
α^{73}	$1 + \alpha^2 + \alpha^3 + \alpha^4 + \alpha^5 + \alpha^6$	1011111
α^{74}	$1 + \alpha + \alpha^4 + \alpha^5 + \alpha^6$	1100111
α^{75}	$1 + \alpha + \alpha^2 + \alpha^3 + \alpha^5 + \alpha^6$	1111011
α^{76}	$1 + \alpha + \alpha^2 + \alpha^4 + \alpha^6$	1110101
α^{77}	$1 + \alpha + \alpha^2 + \alpha^5$	1110010
α^{78}	$\alpha + \alpha^2 + \alpha^3 + \alpha^6$	0111001
α^{79}	$1 + \alpha^2 + \alpha^4$	1010100
α^{80}	$\alpha + \alpha^3 + \alpha^5$	0101010
α^{81}	$\alpha^2 + \alpha^4 + \alpha^6$	0010101
α^{82}	$1 + \alpha^5$	1000010

멱	다항식 표현	벡터 표현
α^{83}	$\alpha + \alpha^6$	0100001
α^{84}	$1 + \alpha^2 + \alpha^3$	1011000
α^{85}	$\alpha + \alpha^3 + \alpha^4$	0101100
α^{86}	$\alpha^2 + \alpha^4 + \alpha^5$	0010110
α^{87}	$\alpha^3 + \alpha^5 + \alpha^6$	0001011
α^{88}	$1 + \alpha^3 + \alpha^4 + \alpha^6$	1011011
α^{89}	$1+ \alpha + \alpha^3 + \alpha^4 + \alpha^5$	1101110
α^{90}	$\alpha+ \alpha^2 + \alpha^4 + \alpha^5 + \alpha^6$	0110111
α^{91}	$1 + \alpha^2 + \alpha^5 + \alpha^6$	1010011
α^{92}	$1+ \alpha + \alpha^6$	1100001
α^{93}	$1+ \alpha+ \alpha^2 + \alpha^3$	1111000
α^{94}	$\alpha+ \alpha^2 + \alpha^3 + \alpha^4$	0111100
α^{95}	$\alpha^2 + \alpha^3 + \alpha^4 + \alpha^5$	0011110
α^{96}	$\alpha^3 + \alpha^4 + \alpha^5 + \alpha^6$	0001111
α^{97}	$1 + \alpha^3 + \alpha^4 + \alpha^5 + \alpha^6$	1001111
α^{98}	$1+ \alpha + \alpha^3 + \alpha^4 + \alpha^5 + \alpha^6$	1101111
α^{99}	$1+ \alpha+ \alpha^2 + \alpha^3 + \alpha^4 + \alpha^5 + \alpha^6$	1111111
α^{100}	$1+ \alpha+ \alpha^2 + \alpha^4 + \alpha^5 + \alpha^6$	1110111
α^{101}	$1+ \alpha+ \alpha^2 + \alpha^5 + \alpha^6$	1110011
α^{102}	$1+ \alpha+ \alpha^2 + \alpha^6$	1110001
α^{103}	$1+ \alpha+ \alpha^2$	1110000
α^{104}	$\alpha+ \alpha^2 + \alpha^3$	0111000

멱	다항식 표현	벡터 표현
α^{105}	$\alpha^2 + \alpha^3 + \alpha^4$	0011100
α^{106}	$\alpha^3 + \alpha^4 + \alpha^5$	0001110
α^{107}	$\alpha^4 + \alpha^5 + \alpha^6$	0000111
α^{108}	$1 + \alpha^3 + \alpha^5 + \alpha^6$	1001011
α^{109}	$1+ \alpha + \alpha^3 + \alpha^4 + \alpha^6$	1101101
α^{110}	$1+ \alpha+ \alpha^2 + \alpha^3 + \alpha^4 + \alpha^5$	1111110
α^{111}	$\alpha+ \alpha^2 + \alpha^3 + \alpha^4 + \alpha^5 + \alpha^6$	0111111
α^{112}	$1 + \alpha^2 + \alpha^4 + \alpha^5 + \alpha^6$	1010111
α^{113}	$1+ \alpha + \alpha^5 + \alpha^6$	1100011
α^{114}	$1+ \alpha+ \alpha^2 + \alpha^3 + \alpha^6$	1111001
α^{115}	$1+ \alpha+ \alpha^2 + \alpha^4$	1110100
α^{116}	$\alpha+ \alpha^2 + \alpha^3 + \alpha^5$	0111010
α^{117}	$\alpha^2 + \alpha^3 + \alpha^4 + \alpha^6$	0011101
α^{118}	$1 + \alpha^4 + \alpha^5$	1000110
α^{119}	$\alpha + \alpha^5 + \alpha^6$	0100011
α^{120}	$1 + \alpha^2 + \alpha^3 + \alpha^6$	1011001
α^{121}	$1+ \alpha + \alpha^4$	1100100
α^{122}	$\alpha+ \alpha^2 + \alpha^5$	0110010
α^{123}	$\alpha^2 + \alpha^3 + \alpha^6$	0011001
α^{124}	$1 + \alpha^4$	1000100
α^{125}	$\alpha + \alpha^5$	0100010
α^{126}	$\alpha^2 + \alpha^6$	0010001

감사의 글

먼저 오늘이 있기까지 묵묵히 지켜봐 주시고 보살펴 주신 조형래 지도교수님께 깊이 감사드립니다.

또한 보다 좋은 논문이 될 수 있도록 따뜻한 조언과 세심한 검토를 해 주신 정세모 교수님과 김동일 교수님께도 깊은 감사를 드립니다.

짧지 않은 학교 생활동안, 많은 가르침을 주신 정지원 교수님, 김기만 교수님, 강인호 교수님, 민경식 교수님께 진심으로 감사드리며, 어렵고 힘들 때 항상 격려해주신 장원일 교수님께도 깊이 감사드립니다.

한편, 공부하는 동안 함께 부딪기고 연구한 이동통신실험실의 태경이 선배, 철성이 선배, 후배님들과 그리고 같이 수업받고 생활한 전파공학과 선후배, 동기들에게도 고마움을 전합니다.

끝으로 오늘의 결실을 맺기까지 항상 격려와 걱정을 해주신 나의 가족들과 이 기쁨을 함께 나누고 싶습니다.