

단축 차집합 순회부호 (272,190)에 기반한 DARC 오류정정 복호기 설계 (The Design of DARC Error Correction Decoder Based on (272,190) Shortened Difference Set Cyclic Code)

심 병 섭* 박 형 근** 김 환 용***
(Byoung-Sup Shim) (Hyoung-Keun Park) (Hwan-Yong Kim)

요 약

본 논문에서는 FM 부가방송 시스템을 위한 오류 정정 복호기에서 오류 검출 및 오류 정정을 위해 다수결 논리 복호가 가능한 (272,190) 단축 차집합 순환 부호를 사용하였다. 블록과 프레임 구조상에서 행 방향과 열 방향 오류정정의 결과를 저장할 수 있고, 행과 열 방향 오류 플래그 메모리 구조를 갖는 오류정정 복호기는 정정될 비트로부터 이미 정정된 비트의 영향을 제거시켜 다수결 논리에 의해 결정된 출력을 무효화할 수 있도록 설계되었다. 행 방향 오류정정의 성공 결과를 나타내는 오류 플래그에 의해 오류정정이 완벽하게 수행되었다면 열 방향의 오류정정은 수행하지 않도록 하여 행과 열 방향의 오류정정을 모두 수행하는 기존의 오류정정 복호기에 비해 오류정정 방식을 개선하였다.

ABSTRACT

In this paper, DARC(Data Radio Channel) error correction decoder for the FM Subcarrier Broadcasting System is designed of using $(272,190) \times (272,190)$ product code based on (272,190) shortened difference set cyclic code. This decoder has error flag of column and row direction that can store the result of the error correction of column and row direction in the block and frame structure, is designed to be of no benefit the output with majority logic determination to cancel the corrected and determined bit, and can improve by using the error correction method that no error correction of the row direction is performed, if error correction of the column direction is completely performed by error flag.

1. 서론

다양한 통신 매체의 급격한 변화와 경제활동이 고도화되어감에 따라 새로운 통신 기술의 개발과 경제성 있는 통신 수단이 강구되고 있다. 또한 방대한 양의 정보 전송이 요구됨에 따라 전파자원의 고갈로 인해 기존의 주파수 및 전송시스템을 사용하는 부가정보방송 서비스가 점점 중요하게 되었다[1][2].

FM 부가방송은 새로운 주파수 채널의 할당 없이 뉴스, 일기예보, 주식정보, 프로그램 정보 등을 기존의 FM 라디오 채널에 디지털 형태의 정보를 부가하여 전송하는 방식이다. 이 방식은 수신시 무선채널의 특성상 잡음, 왜곡, 간섭 등의 영향을 많이 받기 때문에 데이터가 목적지까지 도달하는 과정에서 오류

* 학생회원 : 원광대학교 전자공학과 박사과정
** 정회원 : 원광대학교 BK21 산학협력교육사업팀
*** 정회원 : 원광대학교 전기전자 및 정보공학부 교수

논문접수 : 2001. 5. 28.
심사완료 : 2001. 6. 11.

가 발생할 수 있다. 따라서 이러한 오류를 최소화하고, 최상의 서비스 품질을 얻기 위해서는 원래의 정보를 정확하게 복원할 수 있는 오류정정 부호가 필요하다[3][4].

대표적인 FM 부가방송 시스템은 RDS(Radio Data Service)[5]와 DARC(Data Radio Channel)[6]가 있다. 그러나 RDS 방식의 서비스는 낮은 정보 전송률로 인해 다양한 서비스 제공이 불가능하여 세계적으로도 DARC 방식의 FM 부가방송 서비스를 시험 연구 및 도입하고 있는 추세이며, 이용자들의 다양한 서비스 욕구 해결, 새로운 방식의 기술 확보 등을 토대로 DARC 방식의 새로운 FM 부가방송 서비스가 부각되고 있다[7][8][9].

DARC 방식에서는 (272,190) 단축 차집합 부호를 기본으로 하는 (272,190)×(272,190) 적부호를 오류정정 부호로 사용하고 있으며 복호방식에서는 다수결 논리 복호법을 이용하고 있다.[10] 기존의 다수결 논리 복호법에서 신드롬 레지스터(syndrome register)는 각 비트의 정정 결과에 영향을 받으므로 다수결 논리 결정에 따라 정정되는 각 비트에 잘못 적용될 수 있다. 본 논문에서는 오류정정의 성공 여부를 확인하는 오류 플래그와 CRC(Cyclic Redundancy Check)를 이용하여 정정이 필요치 않을 경우 다수결 논리 회로의 출력을 무효화 할 수 있는 새로운 구조의 다수결 논리 복호법을 이용한 오류정정 방식을 제안하였다.

2. 다수결 논리 복호 알고리즘

랜덤 오류(random error)를 정정하기 위한 가장 간단한 방법이 다수결 논리 복호 부호(majority logic decodable codes)이다. 다수결 논리 복호 알고리즘을 적용할 수 있는 부호를 직교 가능 부호(orthogonalizable code)라 한다. 이 복호법은 순회 부호의 복호법과 마찬가지로 부호의 순회성을 이용하여 수신어를 순회 치환하면서 복호가 이루어지므로 복호기의 구성이 간단하다.[11] 따라서, (272,190) 단축 차집합 부호의 비트 수만 적을 뿐 (21,11) 단축 차집합 순회부호(shortened difference set cyclic code)를 이용하여 다수결 논리 복호(majority logic decoding) 알고리즘을 설명한다. (21,11) 단축 차집합 순회부호

의 생성 다항식은 $g(x)=1+x^2+x^4+x^6+x^7+x^{10}$, 최소 거리(minimum distance)는 6, 그리고 직교수 J는 5이다.[12][13] 패리티 검사 행렬(parity check matrix) H에서 열의 선형조합으로 얻어진 (21,11) 부호의 패리티 검사 합행렬(parity check sum matrix) H1은 식(1)과 같다.

$$H_1 = \begin{bmatrix} 0000000000000110000101 \\ 010000000001001100001 \\ 000010100000000010001 \\ 100001010000000001001 \\ 001100001010000000001 \end{bmatrix} \quad (1)$$

H₁ 행렬은 21번째 열에서 직교하며 다른 열이 단지 하나의 '1'을 가지는 동안에 21번째 열은 모두 '1'로 구성되는 것이다. 수신된 신호 $r=(r_1, r_2, r_3, \dots, r_{21})$ 의 내적과 H₁ 행렬의 열은 다수결 논리 복호에서 첫 번째로 얻어진다. 여기에서 r_i는 경판정(hard decision)의 경우에 0과 1이다. 다음으로 내적의 산술합이 얻어진다. 그 합이 3을 초과하면 r₂₁은 정정되는 것이고 그렇지 않으면 r₂₁은 정정되지 않는다. 즉, 판정은 내적에 대한 '1'의 수를 구하면 된다. 그런 다음 r₁에서 r₂₀까지의 정정은 수신된 신호 r을 주기적으로 쉬프트 시키고 위의 동작을 반복함으로써 수행된다. 패리티 검사 합 A_k는 식(2)와 같고, 식(3)은 다섯 개의 패리티 검사 합을 나타낸다.

$$A_k = \sum_{i=1}^5 r_i h_{ki} \pmod{-2}, \quad k=1.2.\dots 5 \quad (2)$$

$$\begin{aligned} A_1 &= r_{21} \oplus r_{19} \oplus r_{14} \oplus r_{13} \oplus r_{10} \\ A_2 &= r_{21} \oplus r_{16} \oplus r_{15} \oplus r_{12} \oplus r_2 \\ A_3 &= r_{21} \oplus r_{20} \oplus r_{17} \oplus r_7 \oplus r_5 \\ A_4 &= r_{21} \oplus r_{18} \oplus r_8 \oplus r_6 \oplus r_1 \\ A_5 &= r_{21} \oplus r_{11} \oplus r_9 \oplus r_4 \oplus r_3 \end{aligned} \quad (3)$$

여기서, h_{ki}는 H₁행렬의 k개의 열과 i개 행의 구성 요소이며 ⊕는 mod-2 덧셈을 나타낸다. 복호방법은 $\sum_{k=1}^5 A_k > T$ 를 만족하면 r₂₁은 정정되고, $\sum_{k=1}^5 A_k \leq T$ 를 만족하면 정정되지 않는다[3].

DARC 복호기는 (272,190)×(272,190) product 코드를 사용해 수신시 발생하는 오류를 검출하고 정정한다. (272, 190) 단축순환코드의 차집합 다항식

$\theta^*(x)$ 는 다음과 같다.

$$\theta^*(X) = x^{201} + x^{183} + x^{177} + x^{155} + x^{151} + x^{134} + x^{98} + x^{89} + x^{86} + x^{75} + x^{73} + x^{42} + x^{35} + x^{34} + x^{15} + x^5 + 1 \quad (4)$$

생성 다항식 $g(x)$ 는 다음과 같다.

$$g(x) = x^{82} + x^{77} + x^{76} + x^{71} + x^{67} + x^{66} + x^{56} + x^{52} + x^{48} + x^{40} + x^{36} + x^{34} + x^{24} + x^{22} + x^{18} + x^{10} + x^4 + 1 \quad (5)$$

신드롬 검사 합 다항식은 다음과 같다.

$$\begin{aligned} A_1 &= S_{72} + S_7 \\ A_2 &= S_{18} \\ A_3 &= S_6 + S_{24} \\ A_4 &= S_{22} + S_{28} + S_{46} \\ A_5 &= S_4 + S_{26} + S_{32} + S_{50} \\ A_6 &= S_{17} + S_{21} + S_{43} + S_{49} + S_{67} \\ A_7 &= S_{36} + S_{53} + S_{57} + S_{79} \\ A_8 &= S_9 + S_{45} + S_{62} + S_{66} \\ A_9 &= S_3 + S_{12} + S_{48} + S_{65} + S_{69} \\ A_{10} &= S_{11} + S_{14} + S_{23} + S_{59} + S_{76} + S_{80} \\ A_{11} &= S_2 + S_{13} + S_{16} + S_{25} + S_{61} + S_{78} + S_{82} \\ A_{12} &= S_{31} + S_{33} + S_{44} + S_{47} + S_{56} \\ A_{13} &= S_7 + S_{38} + S_{40} + S_{51} + S_{54} + S_{63} \\ A_{14} &= S_1 + S_8 + S_{39} + S_{41} + S_{52} + S_{55} + S_{64} \end{aligned} \quad (6)$$

$$A_{15} = S_{19} + S_{20} + S_{27} + S_{58} + S_{60} + S_{71} + S_{74}$$

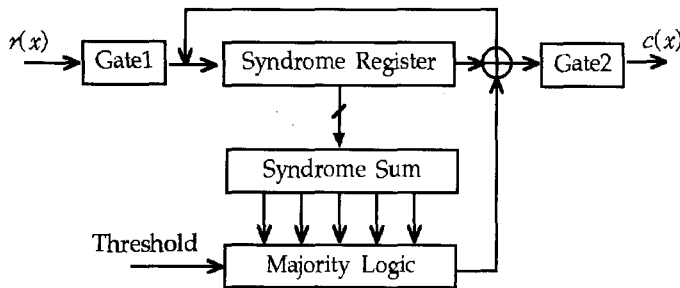
$$A_{16} = S_{10} + S_{29} + S_{30} + S_{37} + S_{68} + S_{70} + S_{81}$$

$$A_{17} = S_5 + S_{15} + S_{34} + S_{35} + S_{42} + S_{73} + S_{75}$$

기존의 오류정정 복호기의 경우 272 비트의 수신 데이터 $r(x)$ 는 데이터 레지스터에 입력되고 신드롬 레지스터는 $r(x)$ 의 입력이 시작되는 시점에서 리셋된 후 데이터가 입력된다. 272 비트의 데이터가 모두 데이터 레지스터에 입력되면 데이터 레지스터 앞단의 스위치에 의해 순회되면서 신드롬 레지스터로부터 계산된 신드롬 합이 다수결 논리 회로로 입력되어 XOR 게이트에 의해 신드롬을 산술 연산한다.

만일 272비트에 대한 연산이 다 수행된 후 임계값(threshold)이 '9'가 될 때까지 임계값을 반복 산출한다. 만일 오류정정 수행 후 신드롬 레지스터의 모든 내용이 '0'이 아닐 경우 오류정정이 성공적으로 수행되지 않았음을 의미하며 오류 플래그가 오류 플래그 회로로부터 출력된다. [그림 1]은 다수결 논리 복호기의 블록도이다.

이와 같은 기존의 방법에서는 오류정정이 적부호의 행과 열에 대해 개별적으로 이루어지므로 잘못된 정정의 가능성이 커지고 오류정정 능력이 저하된다. 특히 오류정정 복호기에서 이용되는 (272, 190) 단축 차집합 순회 부호에 대한 다수결 논리 복호법에 있어서 신드롬 레지스터는 이와 같이 결정된 비트의 정정 결과에 영향을 받으므로 오류정정은 다수결 논리 결정에 따라 정정되었던 비트에 잘못 적용될 수 있고, 이 영향이 다음 비트들까지 확장된다.



[그림 1] 다수결 논리 복호기 블록도

[Fig. 1] Block diagram of the majority logic decoder

3. 제안된 오류정정 복호기 설계

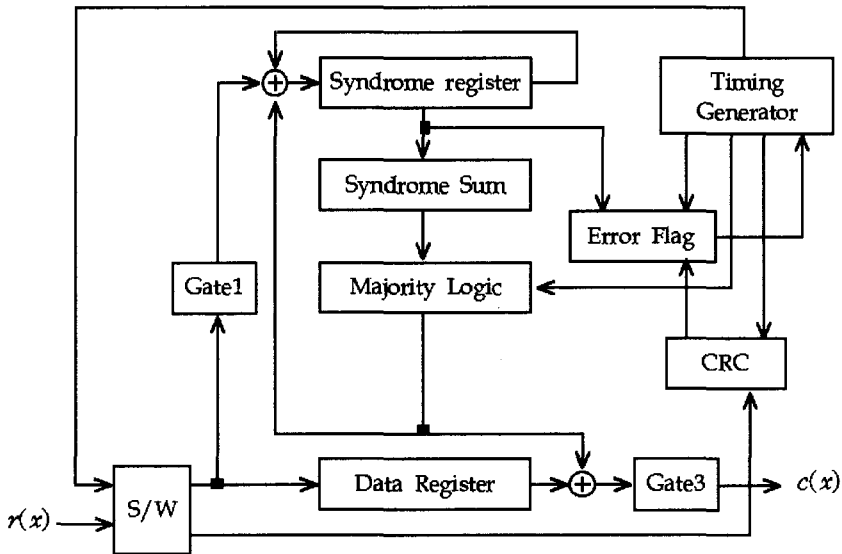
오류정정 복호기는 부호화된 수신정보에 대해 전송 상에서 발생한 오류를 정정하여 원래의 정보를 복원하는 기능을 수행한다. 본 논문에서는 FM 부가 방송 수신기에 적합한 DARC 오류정정 복호기를 다수결 논리 복호법의 원리를 적용하여 단축 차집합 부호에 의해 부호화된 적부호(product code)를 복호하기 위한 오류정정 복호기를 설계하였다. 본 논문에서는 기존의 다수결 논리 복호기에 오류정정의 성공 여부를 확인하는 오류 플래그를 와 오류의 존재여부를 검사하기 위한 CRC를 이용하여 잘못된 오류정정의 가능성을 줄일 수 있는 구조의 다수결 논리 복호기를 제안하였으며 블록도는 [그림 2]와 같다.

272 비트의 데이터가 모두 입력되면 오류정정 복호기는 신드롬 레지스터로부터 신드롬 검사합을 생성하여 다수결 논리 회로에서 초기의 임계값과 비교한 후 각각의 비트에 대해 행 방향과 열 방향의 오류정정을 수행하고 그 결과를 오류 플래그에 저장한다. 열방향 오류정정이 수행될 때 행방향 오류정정 결과가 성공적으로 결정되면 다수결 논리 회로의 출

력을 강제적으로 무효화시켜 열방향 오류정정이 수행되지 않는다. 이러한 이유는 DARC 데이터 구조에 있어서 14 비트의 CRC가 행방향으로 더해지므로 행방향 오류정정 능력은 열방향의 오류정정 능력에 비해 매우 크게 된다. 또한 첫 번째 행방향 오류정정의 정정 결과에 따라 열방향 오류정정이 수행되기 때문에 첫 번째 행방향 오류정정이 성공적인 비트는 열방향 오류정정이 수행되는 동안 변화하지 않는다. 그리고 정정된 비트들의 수에 따라 복호 성공조건 변화는 다수결 논리 복호에 의해 복호되기 때문에 신드롬 레지스터의 내용이 '0'이고 CRC의 내용이 '0'이면 복호된 데이터가 출력된다.

3.1 타이밍 발생기(timing generator) 블록

타이밍 발생기블록은 오류정정 복호기를 위한 제어신호를 생성하며, 복호시작 신호 start_decoder가 입력되면 복호기의 초기화 신호 clear_decoder를 발생한다. 273비트마다 발생하는 count272_clear 신호에 의해 초기화되는 9 비트 동기식 계수기 count_272 블록은 클럭의 상승 에지(positive edge)에서 동작하고, 신호 count272_clear는 반전되어 계수기



[그림 2] 제안된 다수결 논리 복호기 블록도

[Fig. 2] Block diagram of proposed majority logic decoder

count_16의 상향 계수(up count) 동작 신호가 된다. 출력 신호 중 신드롬 레지스터에 인가되는 클럭신호 syn_reg_clk은 상승 에지에서 신드롬 레지스터를 한 비트씩 순회시키고 데이터 레지스터에 인가되는 data_reg_clk은 상승 에지에서 데이터 레지스터를 순회시킨다. majority_logic_enable 신호는 다수결 논리 블록을 enable 시키고 error_flag_enable 신호는 오류 플래그 블록을 enable 시킨다. [그림 3]은 타이밍 발생기 회로이다.

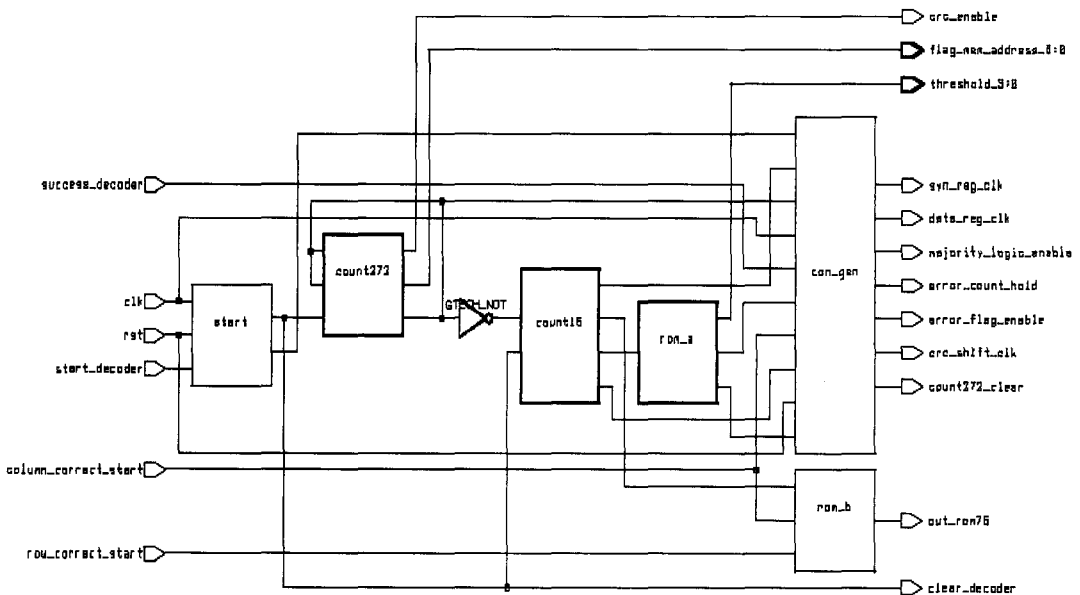
타이밍 발생기 블록은 count_16, ROM_A 및 ROM_B 등의 하위 블록을 포함하고 있다. count_16 블록은 count_272 블록이 '272'를 지시할 때마다 계수되므로 count_16은 데이터 레지스터의 순회 수를 계수한다. count_16 블록의 값에 의해서 결정된 어드레스를 가지고 있는 ROM_A 블록은 <표 1>에 따라 순회 수에 일치하는 다수결 논리회로의 임계값 threshold[3:0]을 출력한다.

<표 1> ROM_A 블록의 내용

<Table 1> Contents of ROM_A block

input from count_16	threshold[3:0]	majority_logic_enable	crc_shift_clk
0	15	0	0
1	15	0	0
2	15	1	1
3	14	1	1
4	13	1	1
5	12	1	1
6	11	1	1
7	10	1	1
8	9	1	1
9	15	0	1
10	15	0	0

오류정정 모드는 <표 2>와 같이 ROM_B의 내용에 따라 첫 번째 행 방향의 오류정정을 위한 mode 1, 열 방향의 오류정정을 하는 mode 2, 두 번째 행 방향의 오류정정을 mode 3 등 세 가지 mode가 있으며 각각의 mode가 순차적으로 오류정정을 수행한다.



[그림 3] 타이밍 발생기 회로
[Fig. 3] Timing generator circuit

<표 2> ROM_B 블록의 내용
 <Table 2> Contents of ROM_B block

mode	count_16의 출력[1000]	column_correct_start	row_correct_start	out_rom
mode 1	don't care	0	0	1
mode 2	don't care	1	don't care	0
mode 3	0	0	1	0
mode 3	1	0	1	1

mode 1에서는 count_16 블록의 내용에 상관없이 column_correct_start와 row correct_start 신호가 모두 '0'이고 ROM_B의 출력인 out_rom을 '1'로 하여 첫 번째 행 방향의 오류를 정정한다. mode 2에서는 count_16 블록의 내용과 row_correct_start 신호에 상관없이 column_correct_start 신호가 '1'인 경우로 out_rom 을 '0'으로 하여 열 방향의 오류를 정정한다. mode 3에서는 column_correct_start는 '0'이고 row_correct_start는 '1'이며 만일 count_16 블록의 출력이 '0'인 경우 out_rom을 '0'으로 하고 count_16 블록의 출력이 '1'인 경우 out_rom을 '1'로 하여 두 번째 행 방향의 오류를 정정한다. 여기에서 count_16 블록의 계수 값이 '8'인 경우 출력은 '1'이 되고 계수 값이 '8'이외의 값을 나타내는 경우 출력이 '0'이 된다.

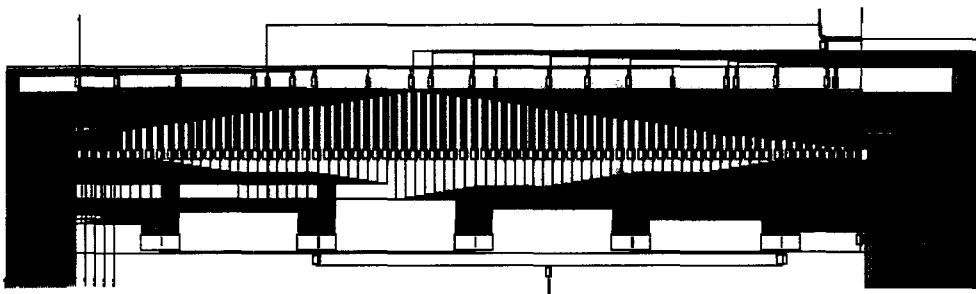
3.2 신드롬 블록

신드롬블록은 신드롬 레지스터와 신드롬 검사 합을 계산하는 블록으로 구성되며 272 비트의 데이터가 (272,190) 부호의 부호어인 경우 82 비트의 신드

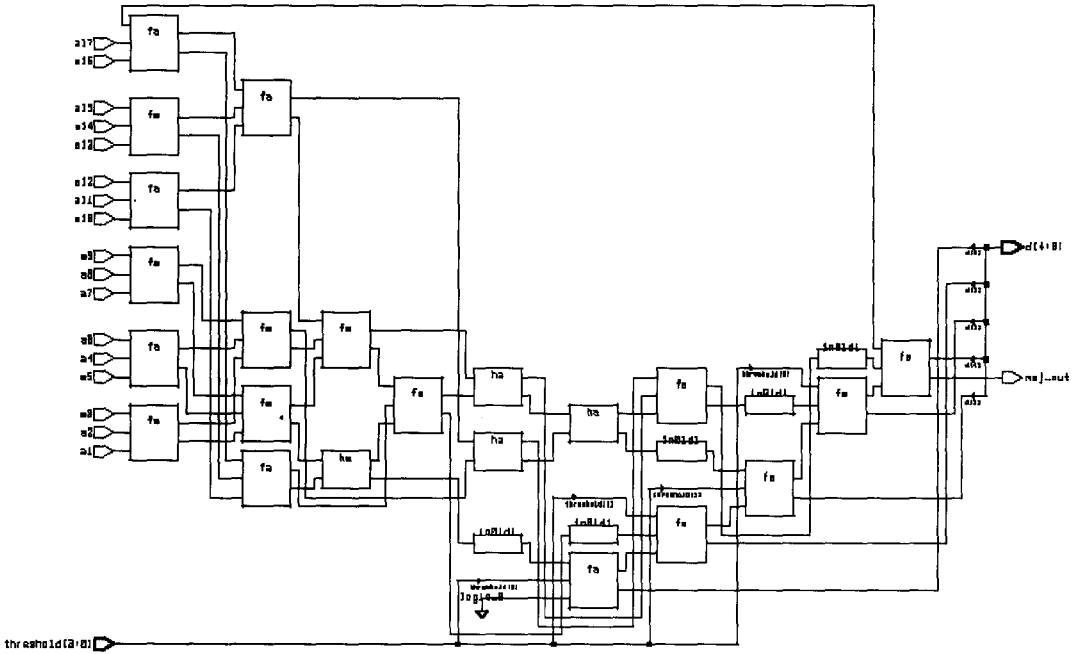
롬 레지스터의 내용은 모두 '0'이다. 신드롬 레지스터를 구성하는 다항식 $g(x)$ 는 식(5)와 같고 신드롬 합 블록은 신드롬 레지스터로부터 계산된 신드롬 검사 값의 합은 식(6)의 A1~A17까지의 값을 구하여 그 결과를 다수결 논리 블록으로 출력한다.

3.3 다수결 논리(majority logic) 블록

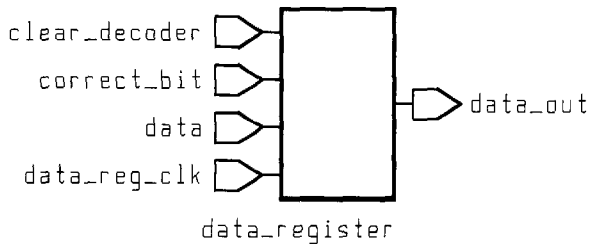
다수결 논리 블록은 신드롬 합 회로로부터 계산된 값과 타이밍 발생기의 ROM_A에서 출력되는 임계값 threshold[3:0]을 비교하여 임계값보다 작을 경우에는 '0'(low level)을 출력하고 임계값보다 클 경우에는 '1'(high level)을 출력하여 오류정정이 이루어지도록 한다. 그러나 오류 플래그가 오류정정의 성공을 나타내는 경우에는 다수결 논리출력이 강제로 무효화되며 어떠한 오류정정도 수행되지 않는다. 다수결 논리 회로는 [그림 5]와 같다.



[그림 4] 신드롬 회로
 [Fig. 4] Syndrome circuit



[그림 5] 다수결 논리 회로
 [Fig. 5] Majority logic circuit



[그림 6] 데이터 레지스터 회로
 [Fig. 6] Data register circuit

3.4 데이터 레지스터(data register) 블록

데이터 레지스터블록은 272비트의 시프트 레지스터로 구성되었으며 데이터의 입력은 타이밍 발생기의 출력 신호 data_reg_clk의 상승 에지에서 시프트되어 입력된다. [그림 6]은 데이터 레지스터 블록이다.

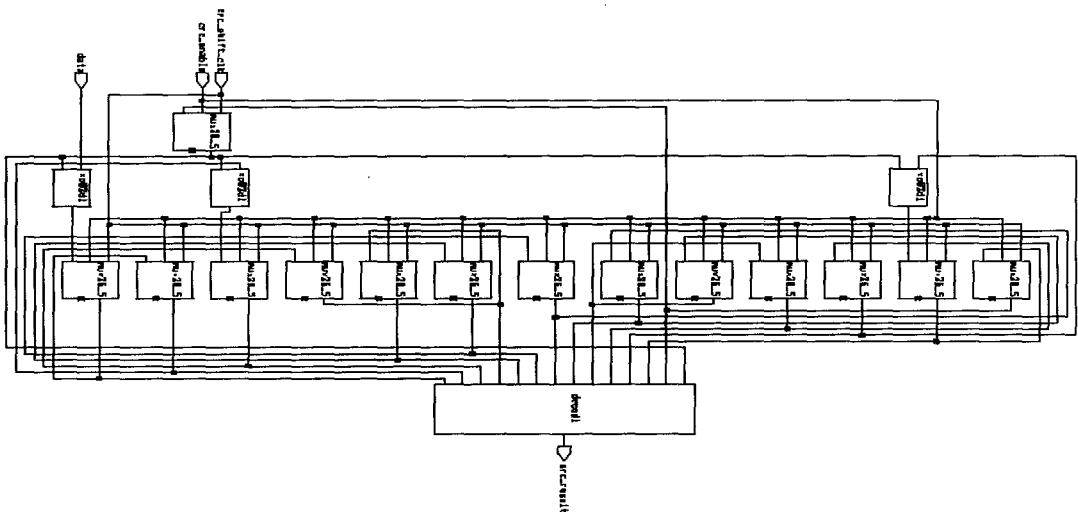
3.5 CRC 블록

CRC는 14비트로 구성되며 176개의 정보 비트에 대해 하나의 CRC가 계산된다. CRC 회로는 위의 생성 다항식에 의한 쉬프트 레지스터와 Error Detect 블록으로 구성된다. count_272의 계수 값이 입력 비트의 계수를 위해 '0'~'189'를 나타내는 한 주기동안 CRC 연산 결과 오류가 없음을 확인하면 오류 플래그 회로로 신호 crc_result를 '1'로 하여 출력한다. CRC를 구성하는 다항식 $g(x)$ 는 식(6)과 같고 [그림 7]은 CRC 회로이다.

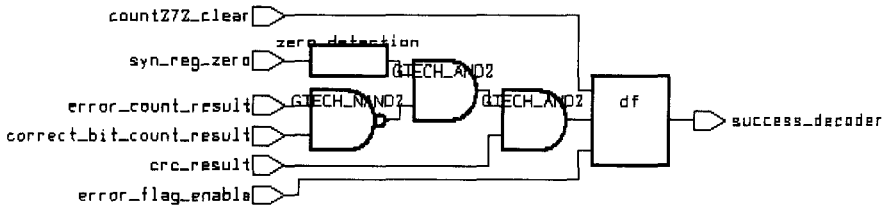
$$g(x) = x^{14} + x^{11} + x^2 + 1 \tag{7}$$

3.7 오류 플래그(error flag) 블록

기존의 오류 플래그 회로는 신드롬 레지스터의 내용이 모두 '0'이 될때 '1'을 출력하는 영 검출(zero detection)회로의 출력에 근거하여 복호의 성공여부를 결정하고 CRC 블록의 출력 신호 crc_result가 입력된다. 오류 플래그 회로의 복호 성공 조건은 행 방향의 오류정정 결과와 열 방향의 오류정정 결과에 따라 영향을 받는다. 다수의 데이터에 대한 복호가 성공적이지 못한 첫 번째 행 방향의 오류정정 후에 열 방향의 오류정정을 수행하는데 있어서 오류계수 결과신호 error_count_result는 '1'이 되고, 열 방향의 오류정정에서 정정된 비트의 수가 미리 결정된 값보다 더 크면 정정비트 결과신호 correct_bit_count_result는 '1'로 되므로 NAND 게이트의 출력은 '0'이 된다. 따라서 비록 기존의 복호성공 조건들을 만족한다 하더라도 이는 아직 오류가 있는 경우이다. [그림 8]은 오류 플래그 회로이다.



□림 7] CRC 회로
[Fig. 7] CRC circuit



[그림 8] 오류 플래그 회로
[Fig. 8] Error flag circuit

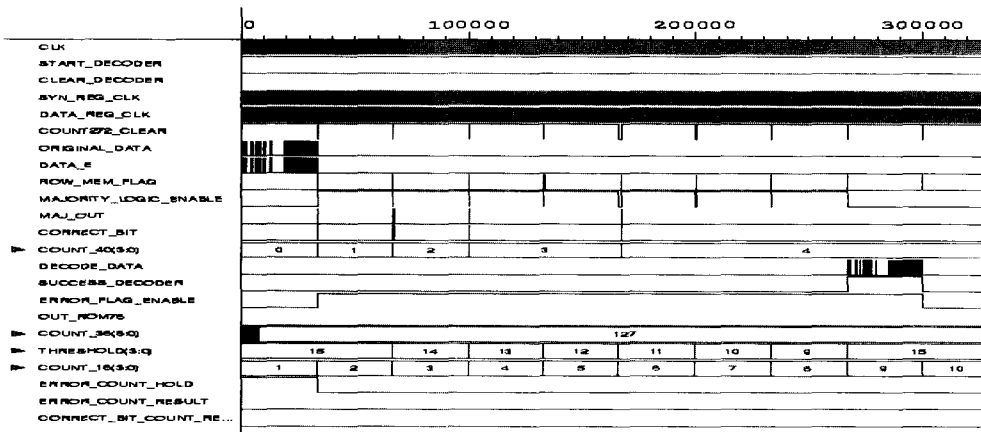
4. 모의 실험 및 고찰

모의실험의 입력 데이터로 부호화된 (272, 190) 데이터에 오류를 발생시킨 후 사용하였다. 오류 없는 데이터와 오류가 섞인 데이터를 <표 3>에 나타냈다.

[그림 9]는 설계된 오류정정 복호기의 오류정정 진행 상태의 모의 실험 결과이다.

<표 3> 272 비트의 입력 데이터
<Table 3> 272 bits input data

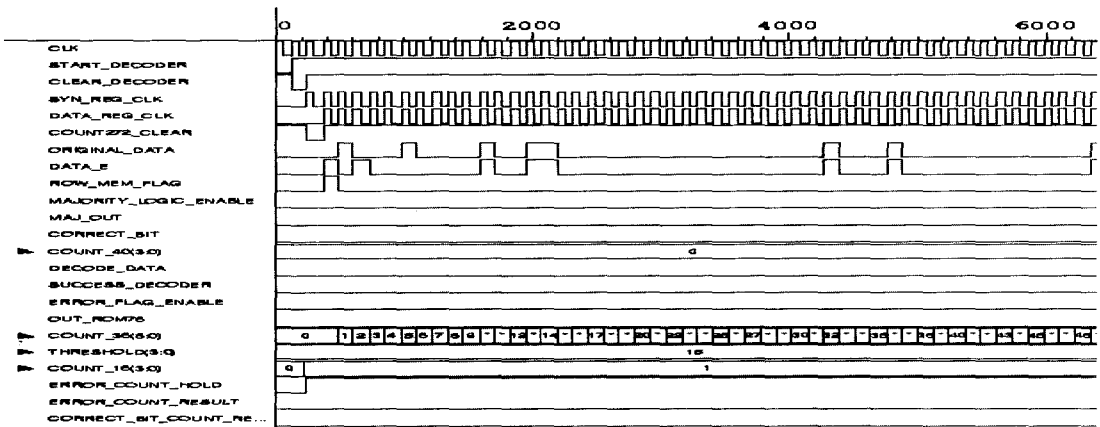
original data	22	64	00	00	01	00	06	10	1E	30	64	01	40	01	00	00	00
	00	00	03	8B	99	C5	48	14	75	D6	55	DB	F0	2B	61	8D	33
error data	05	64	00	00	11	00	06	10	1E	30	64	01	40	01	00	00	00
	00	00	03	8B	99	C5	48	14	75	D6	55	DB	F0	2B	61	8D	33



[그림 9] 오류정정 진행 상태의 모의 실험
[Fig. 9] Simulation of error correction decoding state

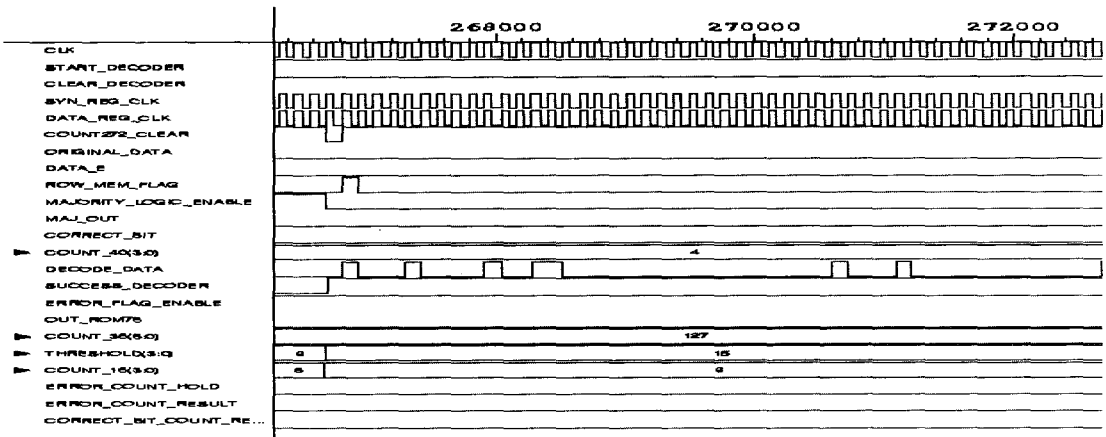
[그림 9]에 보여진 것처럼 신호 out_rom은 '0'으로 되며 오류정정은 플래그 메모리의 결과에 따라 수행된다. 다수결 논리 회로는 신호 malority_logic_enable에 의해 인에이블되고 다수결 논리 회로의 출력 maj_out은 플래그 메모리의 출력과 AND 연산된다. 그 결과로서 correct_bit는 '1'이 출력되고 count_40은 계수된다. count_36은 error_count_hold신호에 의해 인에이블되고 row_mem_flag로부터 출력된 실패한 비트를 계수한다. 오류 플래그 회로에서 신호 error_flag_enable에 의해 나타내어지는 오류정정 처

리 주기 동안에 데이터의 체크는 신드롬 레지스터로부터 출력되는 신호 syn_reg_zero와 CRC 결과 신호인 crc_result에 따라 수행된다. [그림 10]과 [그림 11]은 오류정정 복호기의 각각 입력과 출력 데이터이다.



[그림 10] 정정할 입력 데이터

[Fig. 10] Input data to error correction



[그림 11] 정정된 데이터의 출력 모의 실험 결과

[Fig. 11] Simulation result of output data after error correction

5. 결론

기존의 다수결 논리 복호법에 있어서 신드롬 레지스터는 각 비트의 정정 결과를 입력받으므로 다수결 논리 결정에 따라 정정되는 각 비트에 잘못 적용될 수 있고, 오류정정이 적부호의 행과 열에 대해 개별적으로 이루어지므로 행과 열에 대한 잘못된 정정의 가능성이 커질 뿐 만 아니라 오류정정 능력이 감소하게 된다.

따라서, 본 논문에서는 오류정정의 성공 여부를 확인하는 오류 플래그와 행과 열 방향의 오류 플래그를 저장하는 플래그 메모리, 오류계수기, 정정비트 계수기를 적용하여 정정이 필요치 않을 경우 다수결 논리 회로의 출력을 무효화 할 수 있는 새로운 구조의 다수결 논리 복호법을 이용한 오류정정 복호기의 오류정정 방식을 제안하였다. 제안된 방식은 행과 열 방향의 오류정정을 모두 수행하는 기존의 오류정정 복호기에 비해 행 방향의 오류정정의 성공 결과를 나타내는 오류 플래그 메모리에 의해 오류정정이 완벽하게 수행되었을 경우 불필요한 열 방향의 오류정정은 수행하지 않도록 하여 오류정정 방식을 개선시켰다.

※ 참고문헌

- [1] Linda Zeger, "Analysis and Simulation of Multipath Interference of FM Subcarrier Digital Signals" Proc. of the third IEEE Sym. on Computer and Comm., pp.35-41, June 1998.
- [2] Irving S. Reed, Xuemin Chen, "Error-Control coding for Data Networks", KAP, 1999.
- [3] Shu Lin, Danilel j. Costello Jr., "Error Control Coding : Fundamentals and Applications", Prentice Hall, 1983.
- [4] Man Young Rhee, "Error-Correcting Coding Theory", McGraw-Hill, 1989.
- [5] J-P. M. G. Linartz, "Spectrum Efficiency of Radio Data System(RDS)," IEEE Trans. Broadcasting, Vol 39, No. 3, pp.331-334, Sep. 1993.
- [6] Kurada, T., M. Takada, T. Isobe, O. Yamada, "Transmission scheme of High Capacity FM Multiplex Broadcasting System", IEEE Trans. Broadcasting, Vol. 42, No. 3, pp 245-250, Sept. 1996
- [7] 김기근, 류홍균, "RDS 수신 시스템에서 동기식 신호복원과 에러정정에 관한 연구", 대한전자공학회논문지, Vol. 29-A, No. 8, aug. 1992.
- [8] P. Scomazzon, R. Andersson, "SWIFT EU 1197-A multi-applicative services using a high rate data system implemented in the Terrestrial FM Radio Network", Proc. of the 5th IEEE Inter. Sym. on the Personal Indoor and Mobile Radio Comm., Vol. 1, Sep. 1995.
- [9] 박진우, "FM 다중방송 기술 및 현황", 대한전자공학회지, Vol. 22, No. 1, 1995.
- [10] "FM 다중방송 연구보고서", MBC, 1996.
- [11] George C. Clark, J. Bibb Cain, "Error-Correction Coding for Digital Communications", Plenum Press, 1982.
- [12] E. J. Weldon, "Difference-Set Cyclic Codes", BSTJ, Vol. 45, No. 7. pp.1045-1055, Sep. 1963.
- [13] R. L. Graham, J. Macwilliams, "On the Number of Information Symbols in Difference-Set Cyclic Codes", BSTJ, Vol. 48, Sep. 1996.

심 병 섭



1998년 2월 원광대학교
전자공학과 (공학사)
2000년 2월 원광대학교 대학원
전자공학과 (공학석사)
현재 원광대학교 대학원
전자공학과 (박사과정)
주관심 분야 : 신호처리,
멀티미디어 통신,
통신용 ASIC 설계

박 형 근



1994년 2월 원광대학교
전자공학과 (공학사)
1996년 2월 원광대학교 대학원
전자공학과 (공학석사)
2000년 2월 원광대학교 대학원
전자공학과 (공학박사)
현재 원광대학교 BK21
산학협력교육사업팀
주관심 분야 : 신호처리,
이동통신

김 환 용



1974년 2월 전북대학교
전자공학과 (공학사)
1978년 2월 전북대학교 대학원
전자공학과 (공학석사)
1984년 2월 전북대학교 대학원
전자공학과 (공학박사)
1979년 3월~현재 원광대학교
전기전자 및 정보공학부 교수
주관심 분야 : 신호처리,
이동통신