DBpia

UWB 시스템을 위한 RS(23,17) 복호기 최적 설계

정회원 강 성 진*, 김 한 종*

An Optimized Design of RS(23,17) Decoder for UWB

Sung-jin Kang*, Han-jong Kim* ***Regular Members***

요 약

본 논문에서는 UWB 시스템에서 사용되는 RS(23,17)부호의 복호기를 최적화하여 설계하였다. 제안된 복호기는 파이프 라인 구조를 갖는 수정된 유클리드(pipeline structured - modified Euclidean) 알고리즘을 사용한다. 먼저, 기존의 PE 블록 구조를 수정하여 효율적인 PE 블록 구조를 제안하고, 차수(degree) 계산이 필요 없는 복호 알고리즘을 제안한다. 또한, Chien 탐색 알고리즘, Forney 알고리즘, FIFO 크기를 UWB 규격에 최적화 시켜, 작은 복호 지연(latency) 및 하드웨어 복잡도를 가지도록 하였다. 제안된 복호기는 Verilog HDL을 사용하여 구현되었고, 삼성 65nm library를 이용하여 합성한 결과, 실제 ASIC을 제작했을 경우에 250MHz정도까지는 동작이 보장된다고 볼 수 있으며, gate count는 17,628이다.

Key Words : Reed-Solomon Decoder, UWB, Modified Euclidean, Processing Element

ABSTRACT

In this paper, we present an optimized design of RS(23,17) decoder for UWB, which uses the pipeline structured-modified Euclidean(PS-ME) algorithm. Firstly, the modified processing element(PE) block is presented in order to get rid of degree comparison circuits, registers and MUX at the final PE stage. Also, a degree computationless decoding algorithm is proposed, so that the hardware complexity of the decoder can be reduced and high-speed decoder can be implemented. Additionally, we optimize Chien search algorithm, Forney algorithm, and FIFO size for UWB specification. Using Verilog HDL, the proposed decoder is implemented and synthesized with Samsung 65nm library. From synthesis results, it can operate at clock frequency of 250MHz, and gate count is 17,628.

Ⅰ. 서 론

미국 연방 통신 위원회(FCC)는 UWB(Ultra Wide-Band)를 '중심주파수의 20% 이상의 점유대역폭을 가지거나 500MHz 이상의 점유대역폭을 차지하는 무선전송기술'로 정의하였으며, 최근에는 광대역을 사용하여 짧은 거리에서 고속의 데이터를 전송할 수 있는 WPAN 기술로 주목을 받고 있다. 2007년 3월에 MB-OFDM 기술을 사용하는 UWB 시스템이 ISO/IEC의 국제 표준으로 채택되었다[1]. MB-OFDM UWB의 프레임(PPDU)은 크게 PLCP preamble, PLCP heaer, PSDU로 구성되며, PLCP header는 PHY header, MAC header, HCS, tail bit, RS parity로 구성되며, Reed-Solomon(RS)(23,17)부호와 convolution 부호를 사용하여 PLCP header를 보호하고 있다[1]. PHY header와 MAC header는 송수신시에 사용되는 중요한 정보들이 포함되어 있다. 특히, PHY header에는 payload의 전

* 한국기술교육대학교 정보기술공학부 (sjkang@kut.ac.kr, hjkim@kut.ac.kr)
논문번호 : KICS2008-06-285, 접수일자 : 2008년 6월 23일, 최종논문접수일자 : 2008년 7월 17일

www.dbpia.co.kr

송속도, 길이 등의 정보가 들어있기 때문에, 복호 지연이 짧아야하고, 고속으로 동작해야한다.

RS 부호는 연집 오류에 대하여 우수한 오류 정정 능력을 가지고 있어서, 많은 통신시스템에서 널리 사용되고 있다. 일반적인 RS(n,k,t) 부호에서 t = (n-k)/2는 RS 부호의 오류 정정 능력을 나타낸다[2]. RS 복호기는 그림 1과 같이 신드롬 연산(syndrome computation), 키 방정식 연산(KES, Key Equation Solver), Chien 탐색, Forney 알고리즘, 오류 정정 블록 및 FIFO로 구성된다[3-6]. 이중에서 오류 위치 다항식(error locator polynomial)과 오류 크기 다항식(error value polynomial)을 찾기 위한 KES 블록이 가장 많은 연산을 필요로 하며, 하드웨어 복잡도가 가장 높다. RS 복호기에 관한 연구는 대부분 키 방정식 연산 알고리즘에 관한 것이며, 많은 복호 알고리즘과 복호기 구조가 연구되어 왔다[3-8]. 이 중에서 ME(modified Euclidean)알고리즘이 하드웨어 규칙성이 우수하여 구현에 적합하기 때문에 많이 사용된다[7,8].

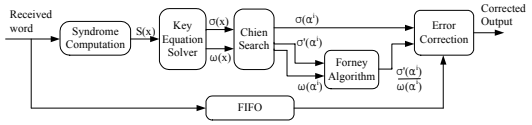

그림 1. 일반적인 RS 복호기 구조

파이프라인 구조의 ME(PS-ME) 알고리즘은 차수 계산과 다항식 연산을 하는 블록으로 구성되는 2t개의 processing element(PE)를 사용하여 계산되며, 하드웨어 규칙성 및 경로 지연(critical path)이 작아서 RS 복호기를 고속으로 구현할 수 있다[4-6]. [7]에서는 차수 계산이 필요치 않는 DCME 알고리즘을 제안하지만, 각 기본 셀(basic cell)의 feedback되는 부분과 모든셀에 입력되는 leading coefficient a_i, b_i가 feedback되므로 상대적으로 고속 구현이 어렵게 된다.

본 논문에서는 PS-ME 알고리즘을 사용하는 RS(23,17)부호의 복호기에서 사용할 수 있는 차수 계산 및 비교가 필요 없는 PS-DCME(pipeline structured - degree computationless modified Euclidean) 알고리즘을 제안한다. 또한, UWB 규격에 맞게 최적화 시켜서 작은 복호 지연(latency)와 하드웨어 복잡도을 줄일 수 있을 뿐 만아니라, 고속의 복호기 설계가 가능하게 한다.

본 논문의 구성은 Ⅱ장에서 PS-ME 알고리즘 블록 구조에 관하여 기술하고, Ⅲ장에서 제안된 PS-DCME 알고리즘에 관하여 설명하고, Ⅳ장에서는 UWB 규격에 최적화된 복호기 설계에 관하여 다룬다. Ⅴ장에서는 복호기의 성능 평가 및 verilog HDL를 이용하여 구현된 결과를 제시하고, Ⅵ장에서 결론을 맺는다.

Ⅱ. PS-ME 알고리즘 블록 구조

2.1 MB-OFDM 시스템의 RS(23,17) 부호

MB-OFDM UWB 시스템에서는 RS(255,249)의 축약 형태인 Systematic RS(23,17)부호를 사용한다. Systematic RS(255,249) 부호의 발생다항식은 다음과 같다[1].

$$g(x)=\prod_{i=1}^{6}(x-\alpha^i) \\ =x^6+126x^5+4x^4+158x^3+58x^2+49x+117 \quad (1)$$

RS(255,249)부호는 RS 부호기에 249byte가 입력된 후, 식(2)와 같은 연산을 수행하여 RS 패리티 6byte를 구한다.

$$p(x)=\sum_{i=0}^{5}p_i x^i = x^6 m(x) \bmod g(x) \quad (2)$$

여기에서 $m(x)$는 정보 다항식(Information polynomial)이며, 아래 식 (3)과 같다.

$$m(x)=\sum_{i=0}^{248}m_i x^i \quad (3)$$

여기에서, m_i는 i번째 메시지 정보이다. RS(23,17) 부호는 식 (3)에서 식 (4)와 같이 m_{248}~m_{17}을 0으로 하는 메시지 벡터 $\boldsymbol{m}$을 사용하여 계산된다.

$$\boldsymbol{m}=(0,\cdots,0,m_{16},\cdots,m_0) \quad (4)$$

따라서, UWB시스템의 RS(23,17)부호기의 출력 코드워드(codeword)는 식 (5)와 같이 정의된다.

$$\boldsymbol{c}=(c_{22},c_{21},\cdots,c_0) \\ =(m_{16},m_{15},\cdots,m_0,p_5,p_4,\cdots,p_0) \quad (5)$$

여기에서, c_i는 8bit이며, GF(2^8)의 원소이다.

2.2 ME 알고리듬

일반적인 RS 복호기의 구조는 앞서 설명한 바와 같이, 그림 1의 구조를 가지며 KES 블록은 키 방정식 $S(x)\sigma(x)=\omega(x) \bmod x^{2t}$을 계산하여, 오류위치

www.dbpia.co.kr

다항식 $\sigma(x)$와 오류값 다항식 $\omega(x)$을 찾는다[4,5]. 여기에서 $S(x)$는 신드롬 다항식을 나타내며, 식 (6)과 같이 표현된다.

$$S(x)=\sum_{i=0}^{2t-1}s_i x^i \tag{6}$$

여기에서 $s_i=r(\alpha^i)$이고, $r(x)$는 수신된 코드워드 다항식, $t=\lfloor (n-k)/2 \rfloor$는 RS(n,k) 부호의 오류 정정 능력을 나타낸다. ME 알고리즘은 식 (7)~(10)으로 표현된다.

$$\begin{aligned}R_i(x)=&\left[\sigma_{i-1}b_{i-1}R_{i-1}(x)+\overline{\sigma_{i-1}}a_{i-1}Q_{i-1}(x)\right]\\&-x^{l_{i-1}}\left[\sigma_{i-1}a_{i-1}Q_{i-1}(x)+\overline{\sigma_{i-1}}b_{i-1}R_{i-1}(x)\right]\end{aligned} \tag{7}$$

$$Q_i(x)=\sigma_{i-1}Q_{i-1}(x)+\overline{\sigma_{i-1}}R_{i-1}(x) \tag{8}$$

$$\begin{aligned}L_i(x)=&\left[\sigma_{i-1}b_{i-1}L_{i-1}(x)+\overline{\sigma_{i-1}}a_{i-1}U_{i-1}(x)\right]\\&-x^{l_{i-1}}\left[\sigma_{i-1}a_{i-1}U_{i-1}(x)+\overline{\sigma_{i-1}}b_{i-1}L_{i-1}(x)\right]\end{aligned} \tag{9}$$

$$U_i(x)=\sigma_{i-1}U_{i-1}(x)+\overline{\sigma_{i-1}}L_{i-1}(x) \tag{10}$$

여기에서, a_{i-1}, b_{i-1}는 각각 $R_{i-1}(x)$와 $Q_{i-1}(x)$의 leading coefficients이다. 그리고, l_{i-1}과 σ_{i-1}는 아래 식과 같다.

$$l_{i-1}=\deg(R_{i-1}(x))-\deg(Q_{i-1}(x)) \tag{11}$$

$$\sigma_{i-1}=\begin{cases}1, & \text{if } l_{i-1}\geq 0\\ 0, & \text{if } l_{i-1}<0\end{cases} \tag{12}$$

여기에서, $\deg(\cdot)$는 다항식의 차수를 나타낸다. ME알고리즘은 $\deg(R_i(x))<t$가 만족될 때 까지 반복된다.

2.3 PS-ME 알고리듬 블록 구조

ME 알고리즘은 식 (11)의 l_{i-1}값에 따라서 다항식 차수가 다양하게 바뀌기 때문에, 하드웨어 설계를 어렵게 한다. [4-6]의 PS-ME 알고리즘 블록의 동작 원리는 파이프라인 구조를 사용하여 각 PE에서 입력 $R_{i-1}(x)$ 또는 $Q_{i-1}(x)$의 차수가 1씩 감소하게 하여, 하드웨어의 규칙성을 갖게 만들었으며, 이로 인해 고속으로 복호기 설계가 가능하다. 즉, PE에서 $R_{i-1}(x)$의 차수가 $Q_{i-1}(x)$의 차수가 같거나 크면($\sigma_{i-1}=1$), 다항식 연산을 통해 $R_{i-1}(x)$의 최고차 항을 제거하고 남은 다항식 $R_i(x)$을 출력하므로, $\deg(R_i(x))=\deg(R_{i-1}(x))-1$이 되고, $Q_i(x)=Q_{i-1}(x)$가 된다. 만약 $R_{i-1}(x)$의 차수가 $Q_{i-1}(x)$차수보다 작으면($\sigma_{i-1}=0$), 다항식 스위치를 해서 같은 과정을 반복한다. 그리고, 만약에 $Q_{i-1}(x)$의 leading coefficient가 '0'이면, 다항식 연산을 하지 않고, shift 연산을 하여 '0'인 $Q_{i-1}(x)$의 최고차 항을 제거 한다. 이 알고리즘은 $\deg(R_i(x))<t$ 또는 $\deg(Q_i(x))<t$를 만족하면, stop 신호가 발생하여 이후의 PE은 shift 연산만을 수행하게 된다.

Ⅲ. PS-DCME 알고리즘

3.1 제안된 PS-ME 블록 구조

UWB 시스템에서 사용되는 RS(23,17)부호에 대하여 PS-ME 블록은 t=3이므로, PE1 ~ PE6까지 6개의 PE 블록을 사용한다[6]. 최종단의 PE6의 출력에서는 식 (13), (14)와 같이 R_6와 Q_6의 차수를 비교하여, $\sigma(x)$와 $\omega(x)$를 얻는다.

$$\sigma(x)=\begin{cases}U_6(x), & \text{if } \deg(R_6(x))>\deg(Q_6(x))\\ L_6(x), & otherwise\end{cases} \tag{13}$$

$$\omega(x)=\begin{cases}Q_6(x), & \text{if } \deg(R_6(x))>\deg(Q_6(x))\\ R_6(x), & otherwise\end{cases} \tag{14}$$

그리고, 짝수번째 PE 블록의 출력 $R_i(x)$와 $L_i(x)$의 출력은 원래 차수보다 1이 큰 $xR_i(x)$,$xL_i(x)$가 출력된다. 따라서, 최종단 PE6의 출력 $R_6(x)$와 $L_6(x)$에 레지스터를 두어, $R_6(x)$, $Q_6(x)$, $L_6(x)$, $U_6(x)$의 차수가 맞게 정렬을 시켜줘야한다. 만약 각 PE의 출력에서 항상 $\deg(R_i(x))\geq\deg(Q_i(x))$가 만족되도록 PE의 구조를 바꾼다면, 식 (13), (14)는 각각 식(15), (16)과 같이 간단하게 됨을 알 수 있다. 이렇게 하면, 출력 단에서 차수를 맞추기 위한 레지스터, 차수 비교 회로, MUX 등이 필요하지 않게 된다.

$$\sigma(x)=U_6(x) \tag{15}$$

$$\omega(x)=Q_6(x) \tag{16}$$

본 논문에서는 각 PE의 출력에서 항상 $\deg(R_i(x))\geq\deg(Q_i(x))$을 만족하도록 하기 위해, $R_{i-1}(x)$, $Q_{i-1}(x)$, $L_{i-1}(x)$, $U_{i-1}(x)$로 부터, 식 (7)~(10)의 다항식 연산을 한 이후에, $\deg(R_i(x))$와 $\deg(Q_i(x))$를 비교하여 출력 다항식을 스위치할지 여부를 결정한다. 이러한 구조를 그림 3에 나타내었

www.dbpia.co.kr

고, ⊗는 GF(2^8) 곱셈기, ⊕는 GF(2^8) 덧셈기를 나타낸다.

그림 3에서 leadR, leadQ는 각각 $R_{i-1}(x)$와 $Q_{i-1}(x)$의 leading coefficient를 나타내며, Stopi 신호는 식 (17)과 같다. PE cell 내부 제어신호 sw, cntrA, cntrB는 식 (18)에 정의되어있다.

$$Stop_i = \begin{cases} 1, & \text{if } (\deg(R_i) < t) \text{ or } (\deg(Q_i) < t) \\ 0, & otherwise \end{cases} \quad (17)$$

$$sw = \begin{cases} 1, & \text{if } \deg(R_i) < \deg(Q_i) \\ 0, & otherwise \end{cases} \quad (18)$$

$$zq = \begin{cases} 1, & \text{if } leadQ = 0 \\ 0, & otherwise \end{cases}$$

$$cntrA = Stop_{i-1} \text{ or } zq$$
$$cntrB = Stop_{i-1} \text{ or } (not\ zq)$$

3.2 PS-DCME 알고리즘

앞서 설명한 바와 같이, PS-ME 구조의 각 PE에서 $R_{i-1}(x)$ 또는 $Q_{i-1}(x)$의 차수가 1씩 감소하게 된다. 또한, 본 논문에서 제안한 그림 3의 PE 블록의 다항식 연산 블록 구조에서는 입력에서 항상

$$\deg(R_{i-1}(x)) \geq \deg(Q_{i-1}(x))$$

이 성립하기 때문에, 입력 다항식에 대하여 차수를 비교하는 대신에, $|\deg(R_i(x)) - \deg(Q_i(x))|$를 관찰함으로써 PE 블록의 제어를 할 수 있다. 또한, 각 PE에서 $|\deg(R_i(x)) - \deg(Q_i(x))|$ 값의 변화는 1이 증가하거나 혹은 1이 감소하므로, 스테이트 머신(state machine)으로 표현할 수 있으며, 각 상태 S_k는 식 (19)와 같이 정의할 수 있다.

$$S_k = |\deg(R_i(x)) - \deg(Q_i(x))| \quad (19)$$

여기에서, $k = 0, \cdots, t$이다.

PS-ME알고리즘에서 $\deg(R_0(x)) = 2t$이고, ME 알고리즘 동작이 완료되는 조건이 식(17)과 같기 때문에, 식 (19)의 상태는 $S_0 \sim S_t$까지 가능하다. 그림 2는 RS(23,17) 복호기에 대한 상태 천이도를 나타낸 것이며, $S_0 \sim S_3$의 4가지 상태가 가능하다.

그림 2의 상태 천이도에서 zq는 식(18)에 정의된 것과 같고, sw는 식 (18)에 정의된 것과 달리, 현재 상태와 zq값에 따라 결정된다. i번째 PE의 입력 상태가 S_0이면, $R_{i-1}(x)$와 $Q_{i-1}(x)$의 차수가 같다는 것을 의미하기 때문에, zq값과 무관하게, i번째 PE의 출력 상태는 S_1이 된다. 이 때, 만약 zq=1이면

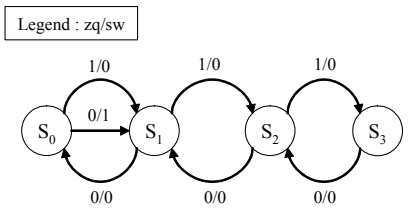

그림 2. RS(23,17) 복호기에 대한 상태 천이도

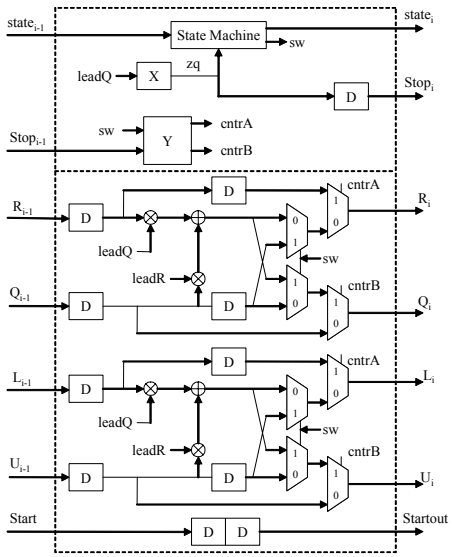

그림 3. PS-DCME 알고리즘을 위한 PE 블록 구조

$Q_{i-1}(x)$의 leading coefficient가 '0'이므로, shift 동작만을 하게 되고, zq=0이면 다항식 연산을 통해 $R_{i-1}(x)$의 차수가 $Q_{i-1}(x)$보다 1이 작아지므로, sw=1이 되고, i번째 PE출력은 $R_i(x)$와 $Q_i(x)$가 switch되어 출력된다. S_1, S_2에서는 zq=1이면, 상태가 1증가하고, zq=0이면 상태가 1감소한다. S_1, S_2일 때는 zq와 무관하게 다항식 switch를 할 필요가 없다. S_3에서는 zq=1인 경우는 발생하지 않는다. 이와 같이 그림 2의 상태도를 이용하면 각 PE블럭을 제어할 수 있다.

PS-DCME 알고리즘을 위한 PE 블록 구조를 그림 3에 나타내었다. 그림 3에서 Stopi 신호는 zq와 같고, 'X' 는 zq계산하는 회로를 표현한 것이며, 식 (18)의 zq와 같다. sw 신호는 그림 2의 상태도에 따라 출력되고, cntrA, cntrB 신호를 발생하는 사각형 'Y'는 식 (18)과 같다. 그림 3에서 state machine은 그림 2로부터 아래 식 (20)과 같이 간단하게 표현됨을 알 수 있다.

www.dbpia.co.kr

$$state_i = \begin{cases} state_{i-1}+1, & \text{if } state_{i-1}=S_0 \text{ or } zq=1 \\ state_{i-1}-1, & otherwise \end{cases} \quad (20)$$

그림 4는 그림 3의 PE 블록 6개를 이용하여 RS(23,17) 복호기를 위한 PS-DCME 알고리즘 블록도이다. RS(23,17)의 복호기에서 신드롬 s_5, s_4, s_3이 모두 동시에 0인 경우는 발생하지 않으므로, 항상 PE1, PE2, PE3, PE4는 모든 오류 패턴에 대하여 항상 동작을 해야한다. 따라서, Stop0, Stop1, Stop2, Stop3은 항상 '0'을 입력한다. 그림 4의 초기 상태 $state_0$는 S_1이고, PE6의 출력은 식 (15), (16)이 된다.

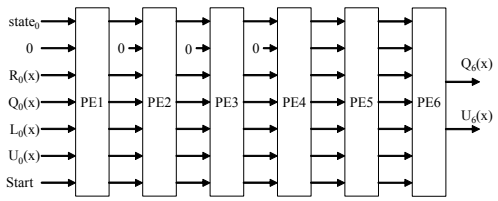

그림 4. PS-DCME 블록도

Ⅳ. UWB 시스템에서 복호기 최적 설계

4.1 Reserved bit 처리

그림 5는 UWB 규격에서 PHY header의 비트 할당을 나타낸다. 그림 5에서 R로 표현된 비트 0 ~ 2, 20 ~ 21, 24 ~ 25, 32 ~ 39는 reserved bit로 총 15bit이며 항상 '0'을 송신해야만 한다[1]. 따라서, RS(23,17)복호기의 입력되는 codeword를 신드롬 계산 블록에 입력시키기 전에, reserved bit는 '0'으로 정정을 한 후에 RS 복호기에 입력시킴으로써, 3byte를 초과하여 에러가 발생한 경우 중에서 reserved bit에서 에러가 난 경우는 복호기가 올바르게 오류를 정정하는 경우가 생기므로, 복호기의 비트오류확률 성능이 개선될 것으로 예측할 수 있다.

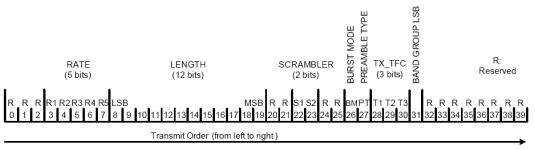

그림 5. PHY header의 비트 할당

4.2 KES 블록

RS(23,17) 복호기의 KES 블록은 그림 4와 같이 구현할 수 있다. 각 PE 블록은 2 clock이 지연되기 때문에, KES 블록에서 총 12 clock의 지연이 발생한다[6]. 그림 4의 PE1의 입력 신호의 초기치는 아래 식 (21)과 같은 초기치를 사용한다.

$$\begin{aligned} &state_0 = S_1 \\ &R_0(x) = x^6, Q_0(x) = xS(x) \\ &L_0(x) = 0, U_0(x) = x \end{aligned} \quad (21)$$

따라서, PE1의 출력 신호는 $Q_0(x)$의 leading coefficient(s_5)가 0인지 아닌지에 따라, 즉 zq 신호에 따라 다음과 같이 표현할 수 있다.

$$\begin{aligned} &\text{if } zq=1, \textit{ then} \\ &\quad state_1 = S_2, \\ &\quad R_1(x) = x^6, \\ &\quad Q_1(x) = s_4x^4 + s_3x^3 + s_2x^2 + s_1x^1 + s_0, \\ &\quad L_1(x) = 0,\ U_1(x) = 1 \\ &else \\ &\quad state_1 = S_0 \\ &\quad R_1(x) = s_4x^5 + s_3x^4 + s_2x^3 + s_1x^2 + s_0x, \\ &\quad Q_1(x) = S(x), \\ &\quad L_1(x) = x,\ U_1(x) = 1 \end{aligned}$$

이러한 결과를 이용하면, PE1은 그림 6과 같이 MUX로 구현할 수 있으며, KES 블록은 PE2~PE6까지 5개만으로 구현이 가능하게 되고, 지연이 2 clock이 줄어들어 총 10 clock의 latency를 가지며, gate count도 줄어들게 된다.

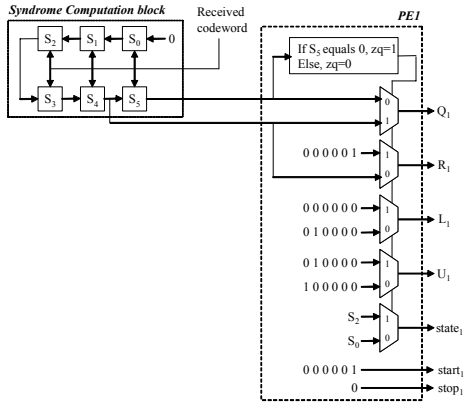

그림 6. 신드롬 계산 블록과 PE1 구조

4.3 Chien 탐색 및 Forney 알고리즘

KES 블록의 최종단 PE6의 출력은 식 (15), (16)과 같이 오류위치 다항식 $\sigma(x)$와 오류값 다항식 $\omega(x)$가 된다. RS(23,17) 부호는 t=3 이므로, $\sigma(x) = \sigma_3x^3 + \sigma_2x^2 + \sigma_1x + \sigma_0$, $\omega(x) = \omega_2x^2 + \omega_1x + \omega_0$로

www.dbpia.co.kr

주어진다. 수신된 코드워드의 오류를 정정하기 위해서는 $\sigma(x)$, $\omega(x)$로부터 정확한 오류 위치와 오류값을 계산해야하는데, Chien search 알고리즘과 Forney 알고리즘이 가장 효율적인 방법이며, 식 (22)와 같다[2,5,9].

$$\alpha c_{255-i} = \begin{cases} r_{255-i} + \dfrac{\omega(\alpha^i)}{\sigma'(\alpha^i)}, & \text{if } (\sigma(\alpha^i)=0) \\ r_{255-i}, & otherwise \end{cases} \quad (22)$$

여기에서, αc_i는 오류가 정정된 i번째 코드워드 심볼이며, RS(255,249)에 대하여 $i=1,\cdots,255$이다. $\sigma'(x)$는 $\sigma(x)$의 미분을 나타낸다. RS(23,17) 부호에서는 식 (4), (5)와 같이, c_{255}~c_{23}은 전송을 하지 않기 때문에, 오류 정정을 할 필요가 없다. 또한, parity 비트 $c_5 \sim c_0$에 대해서도 정보 비트가 아니므로 오류를 정정할 필요가 없기 때문에, 식(22)의 Chien search와 Forney 알고리즘은, $i=233,\cdots,249$에 대해서만 수행하면 된다. 따라서, Chien search알고리즘 그림 7과 같은 구조로 설계할 수 있고, Forney 알고리즘은 그림 8와 같이 설계할 수 있다.

그림 7과 그림 8에서 네모 박스 안의 다항식은 각 알고리즘 블록이 동작할 때 초기값을 나타내며, 식 (22)에서 $i=233$에 대하여 $\sigma(\alpha^i)$, $\sigma'(\alpha^i)$, $\omega(\alpha^i)$를 계산하기 위한 초기 값이다.

일반적인 RS 복호기에서는 복호기 전체 지연에 해당하는 크기 만큼의 FIFO를 가지고 있어야 하지만, UWB시스템에서는 PLCP header에서만 RS 부호를 사용하고 다른 곳에서는 사용되지 않기 때문에 전체 지연만큼의 FIFO 대신, 그림 8에서와 같이 정보 심볼의 길이인 17Byte FIFO만 있으면 되며, 메모리를 사용하지 않고 register로 구현하였다.

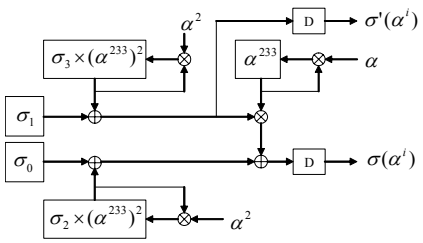

그림 7. Chien 탐색 알고리즘 블록

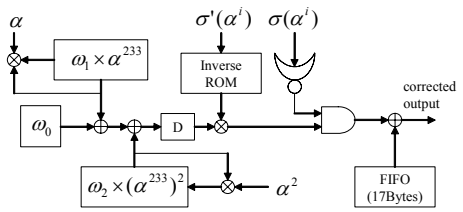

그림 8. Forney 알고리즘 및 오류정정 블록

V. 성능 평가

그림 9는 RS(23,17)부호를 PS-DCME 알고리즘을 이용하여 복호기를 구성했을 때 비트 오류 확률 성능을 나타낸다. C++ 언어를 이용한 시뮬레이션 프로그램을 작성하여 성능 평가를 하였으며, BPSK 변조를 사용했을 때, AWGN 채널에서 복호기 성능이다. RS 부호기의 성능을 비교하기 위하여, 부호화되지 않은 BPSK 변조 방식의 성능곡선을 같이 표시하였다. RS(23,17)부호는 비트오류확률 10^{-5}기준에서 약 2dB의 부호이득을 가진다.

그림 9에는 수신된 코드워드에 reserved bit를 '0'으로 정정하는 mask 블록이 있을 때의 비트오류확률의 성능 곡선도 같이 보여주고 있다. AWGN 채널에서 Reserved bit mask(RBM) 블록으로 인해, 비트오류확률이 약 0.15dB정도 성능개선이 이루어지고 있음을 볼 수 있다. 일반적으로 RS 부호의 Erasure 복호는 AWGN 채널에서는 큰 효과가 없지만, 페이딩 채널에서 성능 개선을 얻을 수 있는 것으로 잘 알려져 있다[2]. 따라서, RBM 블록으로 인한 성능 개선은 랜덤오류(random error)가 발생하는 AWGN 채널보다는 연집 오류(burst error)가 발생하는 페이딩 채널에서 효과가 더 클 것으로 예측된다.

그림 10은 Verilog로 구현된 RS(23,17) 복호기의 functional simulation결과이고, ModelSim을 이용하여 시뮬레이션을 수행하였다. 이 결과로부터 알 수 있듯이, 구현된 복호기는 수신 심볼 23byte가 신드롬 계산 블록에 입력되고 나서, 14clock 뒤에 오류 정정된 정보 심볼이 출력되는 것을 알 수 있다.

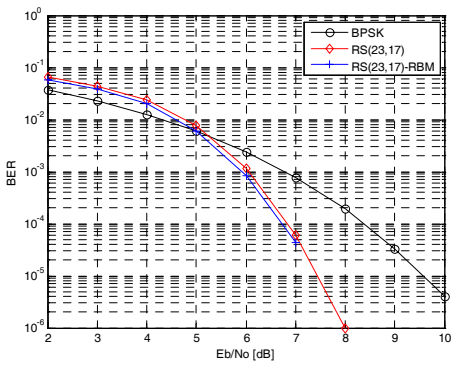

그림 9. AWGN채널에서 RS(23,17) 복호기의 비트오류확률

www.dbpia.co.kr

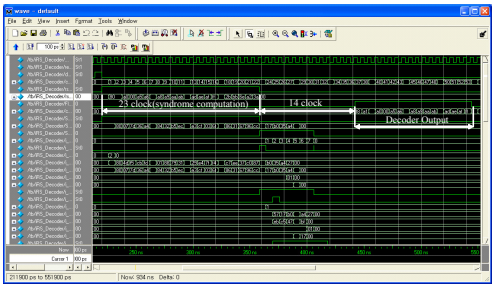

그림 10. ModelSim 시뮬레이션 결과

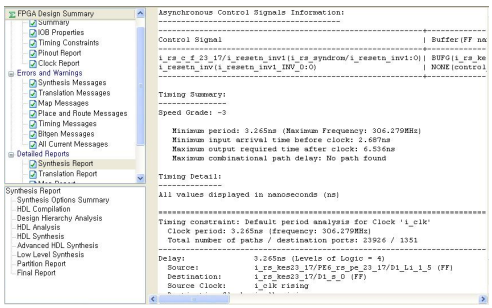

그림 11. Xilix FPGA (XC5VLX110) 합성 결과

그림 11은 Xilinx FPGA인 XC5VLX110-3FF676을 타겟으로 하여 ISE 9.1i에서 합성한 결과이다. 그림 11에서 동작주파수는 약 306MHz로 나타냈으며, 제안된 알고리즘으로 매우 고속의 복호기 설계가 가능하다는 것을 알 수 있다.

삼성 65nm library를 이용하여 합성한 결과, 400MHz(2.5nsec)에서 timing closure되었기 때문에, 실제 ASIC을 제작했을 경우에 250MHz정도까지는 동작이 보장된다고 볼 수 있으며, area estimation이 22564이므로 gate count는 대략 17,628(=22,564/1.28)로 추정된다.

[6]에서는 PS-ME 알고리즘을 구현하여 TSMC 0.18um library에서 합성한 결과, 동작 주파수는 232MHz이고, gate count는 27K 이다. 이 결과와 본 논문의 합성결과를 직접적으로 비교할 수는 없지만, 제안된 알고리즘을 이용하여 복호기를 설계할 때, 동작 속도 및 하드웨어 복잡도가 개선됨을 알 수 있다. [6]에서는 복호 지연(latency)도 46 clock인 반면, 제안된 복호기는 37clock의 latency를 갖는다.

Ⅵ. 결 론

본 논문에서는 UWB 시스템에서 사용되는 RS(23,17)부호의 복호기를 설계하였다. 제안된 복호기는 파이프 라인 구조를 갖는 수정된 유클리드 (pipeline structured - modified Euclidean) 알고리즘을 사용한다. 제안된 PS-DCME 알고리즘에서 PE 블록 구조는 차수 계산을 하지 않고, state machine으로 ME 알고리즘 계산을 할 수 있다. 이로 인해, 하드웨어 복잡도도 감소될 뿐 만 아니라, 고속의 복호기 설계가 가능하다. 또한, PS-DCME알고리즘의 마지막 단계의 PE 블록 출력에서, 항상 Q(x)는 오류값 다항식이 되고, U(x)는 오류 위치 다항식이 되기 때문에, 최종 출력단에서도 차수 비교를 하지 않아도 되고, MUX도 필요하지 않는다.

제안된 복호기는 Chien 탐색 알고리즘, Forney 알고리즘, FIFO 크기를 UWB 규격에 최적화 시켜, 작은 복호 지연(latency) 및 하드웨어 복잡도를 가지도록 하였다. 제안된 복호기는 Verilog HDL을 사용하여 구현되었고, 삼성 65nm library를 이용하여 합성한 결과, 실제 ASIC을 제작했을 경우에 250MHz정도까지는 동작이 보장된다고 볼 수 있으며, gate count는 17,628이고, 복호기의 latency는 37 clock이다.

본 논문에서는 RS(23,17) 부호에 대해서만 고려를 했지만, 제안된 PS-DCME 알고리즘은 일반적인 RS(n,k,t) 부호에 대해서도 성립할 것으로 보이며, 고속의 RS 복호기가 필요한 곳에 응용될 수 있다.

참 고 문 헌

〔1〕 International Standard, ISO/IEC 26907:2007(E), "*Information technology - Telecommunications and information exchange between systems - High Rate Ultra Wideband PHY and MAC Standard*"

〔2〕 S. B. Wicker, *Error Control Systems for Digital Communication and Storage*, Englewood Cliffs, NJ, Prentice-Hall, 1995.

〔3〕 H. M. Shao, T. K. Truong, L. J. Deutsch, J. H. Yuen, and I. S. Reed, "A VLSI design of a pipeline Reed-Solomon decoder", *IEEE Trans. Comput.*, Vol.C-34, No.5, pp.393-403, May, 1985.

〔4〕 H. Lee, "Modified Euclidean algorithm block for high-speed Reed-Solomon decoder", *Electron. Lett.*, 37, pp. 903-904, 2001.

〔5〕 H. Lee, "High-speed VLSI architecture for paral-

www.dbpia.co.kr

lel Reed-Solomon decoder", *IEEE Trans. VLSI Syst.,* Vol.11, No.2, pp.288-294, Apr., 2003.

〔6〕 S. W. Choi, S. S. Choi, H. Lee, "RS decoder architecture for UWB," *IEEE ICACT 2006,* pp. 805-808, 2006.

〔7〕 J. H. Baek and M. H. SunWoo, "New degree computationless modified Euclid's algorithm and architecture for Reed-Solomon decoder", *IEEE Trans. VLSI Syst.,* Vol.14, No.8, pp.915-920, Aug. 2006.

〔8〕 Hyeong-Ju Kang, In-Cheol Park, "A high-speed and low-latency Reed-Solomon decoder based on a dual-line structure", *IEEE ICASSP,* Vol.3, pp.3180-3183, May, 2002.

〔9〕 S. Lin, D. J. Costello, *Error Control Coding: Fundamentals and Applications,* Prentice- Hall, 1983.

강 성 진 (Sung-jin Kang) 정회원

2002년 9월 전자부품연구원 통신네트워크연구센터 책임연구원
2007년 3월~현재 한국기술교육대학교 정보기술공학부 조교수
<관심분야> WPAN, WLAN, MODEM SoC

김 한 종 (Han-jong Kim) 정회원

1994년 9월~현재 한국기술교육대학교 정보기술공학부 교수
<관심분야> 방송 및 이동통신 시스템, 변복조 및 채널부호

www.dbpia.co.kr