(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2020-0046938
(43) 공개일자 2020년05월07일

(51) 국제특허분류(Int. Cl.)
H03M 13/15 (2015.01) *H03M 13/31* (2006.01)
(52) CPC특허분류
H03M 13/1515 (2013.01)
H03M 13/31 (2013.01)
(21) 출원번호 10-2018-0128772
(22) 출원일자 2018년10월26일
심사청구일자 2018년10월26일

(71) 출원인
인하대학교 산학협력단
인천광역시 미추홀구 인하로 100(용현동, 인하대학교)
(72) 발명자
강진구
서울특별시 서초구 양재대로2길 109, 101동 1104호(우면동, 서초참누리에코리치)
박재필
경기도 용인시 처인구 이동읍 경기동로 732
(74) 대리인
양성보

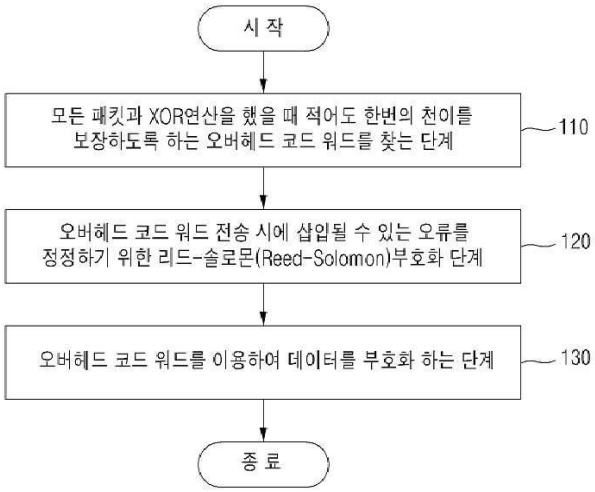
전체 청구항 수 : 총 7 항

(54) 발명의 명칭 **고속직렬인터페이스용 송수신 오류 정정기법이 포함된 오버헤드최소화 코딩 기법과 하드웨어 구현 방법**

(57) 요 약

고속직렬인터페이스용 송수신 오류 정정기법이 포함된 오버헤드최소화 코딩 방법 및 장치가 제시된다. 본 발명에서 제안하는 고속직렬인터페이스용 송수신 오류 정정기법이 포함된 오버헤드최소화 코딩 방법은 모든 패킷과 XOR 연산을 했을 때 적어도 한번의 천이를 보장하도록 하는 오버헤드 코드 워드를 찾는 단계, 오버헤드 코드 워드 전송 시에 삽입될 수 있는 오류를 정정하기 위한 리드-솔로몬(Reed-Solomon) 부호화 단계 및 오버헤드 코드 워드를 이용하여 데이터를 부호화 하는 단계를 포함한다.

대 표 도 - 도1



이 발명을 지원한 국가연구개발사업

과제고유번호	1711074193(세부과제번호: 10080285)
부처명	과학기술정보통신부
연구관리전문기관	한국산업기술평가관리원
연구사업명	전자정보디바이스산업원천기술개발사업
연구과제명	[민간] 8k/4k 디스플레이를 위한 interface 표준화 및 IP기술 개발 (2차년도)
기 여 율	1/1
주관기관	한양대학교산학협력단
연구기간	2018.04.01 ~ 2018.12.31

명 세 서

청구범위

청구항 1

모든 패킷과 XOR연산을 했을 때 적어도 한번의 천이를 보장하도록 하는 오버헤드 코드 워드를 찾는 단계;

오버헤드 코드 워드 전송 시에 삽입될 수 있는 오류를 정정하기 위한 리드-솔로몬(Reed-Solomon) 부호화 단계; 및

오버헤드 코드 워드를 이용하여 데이터를 부호화 하는 단계

를 포함하는 오버헤드최소화 코딩 방법.

청구항 2

제1항에 있어서,

모든 패킷과 XOR연산을 했을 때 적어도 한번의 천이를 보장하도록 하는 오버헤드 코드 워드를 찾는 단계는,

N-bits의 입력(A_i)이 들어올 경우, MSB 1bit로 반전 여부를 판단하여, MSB가 1이면, LSB (N-1)-bits의 반전된 값을, 0이면 LSB (N-1)-bits값을 E_i라고 하면, $D_i = 2^{E_i}$인 (2^{N-1})-bits 크기의 D_i를 생성하고, 상기 D_i를 (2^{N-1})-bits 크기 B_i에 저장하고, 상기 D_i을 생성하여 B_i에 저장하는 단계를 $2^{N-1}-2$번 반복하는

오버헤드최소화 코딩 방법.

청구항 3

제2항에 있어서,

B_i의 LSB bit부터 MSB bit까지 0인 bit가 적어도 하나 이상 존재하고, 0의 위치의 값이 2^C인 경우 C는 1번째부터 $2^{N-1}-2$까지의 모든 데이터 A_i에 대해 $A_i \neq C$ 및 $A_i \neq \sim C$ 이고, $Z = C\ xor\ A_i$로 적어도 한번의 천이를 보장하는 Z를 얻고, 사용 가능한 C의 값 중 하나를 오버헤드 최소화 코드로 선택하는

오버헤드최소화 코딩 방법.

청구항 4

제1항에 있어서,

오버헤드 코드 워드 전송 시에 삽입될 수 있는 오류를 정정하기 위한 리드-솔로몬(Reed-Solomon) 부호화 단계는,

채널에서 오버헤드 코드 워드에 잡음오류가 생길 때, $2^{N-1}-2$개의 데이터 모두 손실됨을 방지하고, 리드-솔로몬 코드(n, k)는 총 n개의 심볼(Symbol)로 k개의 메시지 심볼(Message Symbol)과 n-k개의 패리티 심볼(Parity Symbol)을 가지며, (n-k)/2개 심볼의 오류 정정능력을 갖는

오버헤드최소화 코딩 방법.

청구항 5

모든 패킷과 XOR연산을 했을 때 적어도 한번의 천이를 보장하도록 하는 오버헤드 코드 워드를 찾는 오버헤드 최소화 코드 생성부;

오버헤드 코드 워드 전송 시에 삽입될 수 있는 오류를 정정하기 위한 리드-솔로몬 부호화하는 오류 정정부; 및

오버헤드 코드 워드를 이용하여 데이터를 부호화 하는 데이터 처리부

를 포함하는 오버헤드최소화 코딩 장치.

청구항 6

제5항에 있어서,

오버헤드 최소화 코드 생성부는,

N-비트 크기의 $2^{N-1}-2$ 개의 데이터로부터 적어도 한번의 천이를 보장하는 오버헤드 최소화 코드를 생성하는

오버헤드최소화 코딩 장치.

청구항 7

제5항에 있어서,

오류 정정부는,

리드-솔로몬 부호화기(Reed-Solomon Encoder)를 통해 오류정정을 가능하도록 부호화하고, 부호화된 데이터를 입력 받아 리드-솔로몬 복호화기(Reed-Solomon Decoder)를 통해 오버헤드 최소화 코드의 오류를 정정 및 복호하는

오버헤드최소화 코딩 장치.

발명의 설명

기 술 분 야

[0001] 본 발명은 제안된 고속직렬인터페이스용 오류 정정기법이 포함된 오버헤드최소화 코딩기법과 하드웨어 구현 방법 및 장치에 관한 것이다.

배 경 기 술

[0002] 최근 광통신, 디지털 비디오 방송 및 메모리, 데이터 저장 시스템의 데이터 전송에 있어서, 전송 중에 발생하는 다수의 오류를 찾고 정정할 수 있는 순방향 오류 정정 기술에 리드-솔로몬(RS: Reed-Solomon) 부호와 BCH 부호가 널리 사용되고 있다.

[0003] 순방향 오류 정정 기술은 원 신호에 원 신호의 모양에 대한 정보를 담고 있는 패리티(parity)라 불리는 부가적인 정보를 삽입한 후 전송하여, 원 신호와 패리티에 담긴 정보를 토대로 일정량의 오류를 수신단에서 정정하는 오류 제어 기술이다.

[0004] BCH 부호를 나타내는 전형적인 형태는 BCH(n,k,t)이다. n은 원 데이터와 패리티 전체를 포함한 부호화되는 전체 부호어(codeword) 심볼(symbol)의 수이고, t는 부호어 중에 정정할 수 있는 최대 오류 심볼의 수를 의미하며, k는 실제 데이터의 심볼 수를 의미한다. 일반적으로 BCH 부호라 하면 각 심볼이 1비트(bit)로 이루어진 바이너리(binary) BCH 부호를 의미한다.

[0005] DWDM(Dense-Wavelength Division Multiplexing) 기술을 이용한 광대역 장거리 광통신 시스템 분야에서는 전송거리가 짧게는 수백 km에서 길게는 수천 km에 이르며 데이터 처리량 또한 100Gb/s에 이르고 있다. 이러한 광대역 장거리 광통신 시스템에서는 전송 중에 많은 양의 데이터가 전송 경로의 채널환경에 따라 변형되며 오류를 일으킨다. 전송 중에 발생하는 오류의 양을 경감시키기 위해서 현재까지는 8 바이트(byte) 오류정정 RS(255,239,8) 부호가 국제통신연합(ITU-T)에 의해 표준 순방향 오류 정정 기술로 채택되어 광통신 시스템의 필수 요소로 자리잡게 되었다. 그러나 RS(255,239,8) 부호만으로는 장거리 광통신 시스템이 필요로 하는 코딩이 성능을 만족하지 못하는 문제점이 있다.

[0006] 제안된 고속직렬인터페이스용 송수신 오류 정정기법이 포함된 오버헤드 최소화 코딩기법은 데이터 전송 시, 오버헤드를 최소화하여 데이터 전송 효율을 극대화하는 패킷 단위의 코딩기법이다.

발명의 내용

해결하려는 과제

[0007] 본 발명이 이루고자 하는 기술적 과제는 오버헤드를 최소화하며, 한 패킷당 최소 한번의 천이를 보장하여, 클럭

이 내장된 데이터 전송으로 클럭 복원을 가능하게 함으로써 데이터 전송 효율을 극대화하는 패킷 단위의 코딩 방법 및 장치를 제공하는데 있다.

과제의 해결 수단

[0008] 일 측면에 있어서, 본 발명에서 제안하는 고속직렬인터페이스용 송수신 오류 정정기법이 포함된 오버헤드최소화 코딩 방법은 모든 패킷과 XOR연산을 했을 때 적어도 한번의 천이를 보장하도록 하는 오버헤드 코드 워드를 찾는 단계, 오버헤드 코드 워드 전송 시에 삽입될 수 있는 오류를 정정하기 위한 리드-솔로몬(Reed-Solomon) 부호화 단계 및 오버헤드 코드 워드를 이용하여 데이터를 부호화 하는 단계를 포함한다.

[0009] 모든 패킷과 XOR연산을 했을 때 적어도 한번의 천이를 보장하도록 하는 오버헤드 코드 워드를 찾는 단계는 N-bits의 입력(A_i)이 들어올 경우, MSB 1bit로 반전 여부를 판단하여, MSB가 1이면, LSB (N-1)-bits의 반전된 값을, 0이면 LSB (N-1)-bits값을 E_i라고 하면, $D_i = 2^{E_i}$인 (2^{N-1})-bits 크기의 D_i를 생성하고, 상기 D_i를 (2^{N-1})-bits 크기 B_i에 저장하고, 상기 D_i을 생성하여 B_i에 저장하는 단계를 $2^{N-1}-2$번 반복한다.

[0010] B_i의 LSB bit부터 MSB bit까지 0인 bit가 적어도 하나 이상 존재하고, 0의 위치의 값이 2^C인 경우 C는 1번째부터 $2^{N-1}-2$까지의 모든 데이터 A_i에 대해 $A_i \neq C$ 및 $A_i \neq \sim C$ 이고, $Z = C\ xor\ A_i$로 적어도 한번의 천이를 보장하는 Z를 얻고, 사용 가능한 C의 값 중 하나를 오버헤드 최소화 코드로 선택한다.

[0011] 오버헤드 코드 워드 전송 시에 삽입될 수 있는 오류를 정정하기 위한 리드-솔로몬(Reed-Solomon) 부호화 단계는 채널에서 오버헤드 코드 워드에 잡음오류가 생길 때, $2^{N-1}-2$개의 데이터 모두 손실됨을 방지하고, 리드-솔로몬 코드(n, k)는 총 n개의 심볼(Symbol)로 k개의 메시지 심볼(Message Symbol)과 n-k개의 패리티 심볼(Parity Symbol)을 가지며, (n-k)/2개 심볼의 오류 정정능력을 갖는다.

[0012] 또 다른 일 측면에 있어서, 본 발명에서 제안하는 고속직렬인터페이스용 송수신 오류 정정기법이 포함된 오버헤드최소화 코딩 장치는 모든 패킷과 XOR연산을 했을 때 적어도 한번의 천이를 보장하도록 하는 오버헤드 코드 워드를 찾는 오버헤드 최소화 코드 생성부, 오버헤드 코드 워드 전송 시에 삽입될 수 있는 오류를 정정하기 위한 리드-솔로몬 부호화하는 오류 정정부 및 오버헤드 코드 워드를 이용하여 데이터를 부호화 하는 데이터 처리부를 포함한다.

[0013] 오버헤드 최소화 코드 생성부는 N-비트 크기의 $2^{N-1}-2$개의 데이터로부터 적어도 한번의 천이를 보장하는 오버헤드 최소화 코드를 생성한다.

[0014] 오류 정정부는 리드-솔로몬 부호화기(Reed-Solomon Encoder)를 통해 오류정정을 가능하도록 부호화하고, 부호화된 데이터를 입력 받아 리드-솔로몬 복호화기(Reed-Solomon Decoder)를 통해 오버헤드 최소화 코드의 오류를 정정 및 복호한다.

발명의 효과

[0015] 본 발명의 실시예들에 따른 고속직렬인터페이스용 송수신 오류 정정기법이 포함된 오버헤드최소화 코딩기법은 오버헤드를 최소화하며, 한 패킷당 최소 한번의 천이를 보장하여, 클럭이 내장된 데이터 전송으로 클럭 복원을 가능하게 함으로써 데이터 전송 효율을 극대화할 수 있다.

도면의 간단한 설명

[0016] 도 1은 본 발명의 일 실시예에 따른 고속직렬인터페이스용 송수신 오류 정정기법이 포함된 오버헤드최소화 코딩 방법을 설명하기 위한 흐름도이다.

도 2는 본 발명의 일 실시예에 따른 오버헤드 최소화 코딩 부호기 구조를 나타내는 도면이다.

도 3은 본 발명의 일 실시예에 따른 오버헤드 최소화 코딩 복호기 구조를 나타내는 도면이다.

도 4는 본 발명의 일 실시예에 따른 오버헤드 최소화 코드 생성부의 구조를 나타내는 도면이다.

도 5는 본 발명의 일 실시예에 따른 오류 정정기능 시뮬레이션 결과를 나타내는 도면이다.

도 6은 본 발명의 일 실시예에 따른 오류 정정기능이 포함된 오버헤드최소화 코딩기법 시뮬레이션 결과를 나타내는 도면이다.

발명을 실시하기 위한 구체적인 내용

[0017] 제안된 고속직렬인터페이스용 송수신 오류 정정기법이 포함된 오버헤드 최소화 코딩기법은 데이터 전송 시, 오버헤드를 최소화하여 데이터 전송 효율을 극대화하는 패킷 단위의 코딩기법이다. 이는 오버헤드를 최소화하며, 한 패킷당 최소 한번의 천이를 보장하여, 클럭이 내장된 데이터 전송으로 클럭 복원을 가능하게 하는 코딩기법이다. 이 코딩기법은 오버헤드 코드 워드를 찾는 과정, 오류 정정기법 부호화 하는 과정, 데이터 부호화 과정으로 나누어 진다. 오버헤드 코드 워드를 찾는 과정은 일정 수의 모든 패킷과 XOR연산을 했을 때, 적어도 한번의 천이를 보장하도록 하는 특정 코드 워드를 찾는 과정이다. 오류 정정기법 부호화는 오버헤드 코드 워드 전송시에 삽입될 수 있는 오류를 정정하기 위한 리드-솔로몬(Reed-Solomon) 부호화 과정이다. 데이터 부호화 방법은 오버헤드 코드 워드를 이용해 데이터를 부호화 하는 과정이다. 이하, 본 발명의 실시 예를 첨부된 도면을 참조하여 상세하게 설명한다.

[0019] 도 1은 본 발명의 일 실시예에 따른 고속직렬인터페이스용 송수신 오류 정정기법이 포함된 오버헤드최소화 코딩 방법을 설명하기 위한 흐름도이다.

[0020] 제안하는 고속직렬인터페이스용 송수신 오류 정정기법이 포함된 오버헤드최소화 코딩 방법은 모든 패킷과 XOR연산을 했을 때 적어도 한번의 천이를 보장하도록 하는 오버헤드 코드 워드를 찾는 단계(110), 오버헤드 코드 워드 전송 시에 삽입될 수 있는 오류를 정정하기 위한 리드-솔로몬(Reed-Solomon) 부호화 단계(120) 및 오버헤드 코드 워드를 이용하여 데이터를 부호화 하는 단계(130)를 포함한다.

[0021] 단계(110)에서, 모든 패킷과 XOR연산을 했을 때 적어도 한번의 천이를 보장하도록 하는 오버헤드 코드 워드를 찾는다.

[0022] 제안하는 고속직렬인터페이스용 송수신 오류 정정기법이 포함된 오버헤드최소화 코딩 장치는 오버헤드 최소화 코드 생성부, 오류 정정부, 데이터 처리부를 포함한다.

[0023] 오버헤드 최소화 코드 생성부는 모든 패킷과 XOR연산을 했을 때 적어도 한번의 천이를 보장하도록 하는 오버헤드 코드 워드를 찾는다.

[0024] 오류 정정부는 오버헤드 코드 워드 전송 시에 삽입될 수 있는 오류를 정정하기 위한 리드-솔로몬 부호화를 수행한다.

[0025] 데이터 처리부는 오버헤드 코드 워드를 이용하여 데이터를 부호화 한다.

[0026] 제안하는 오류 정정기법이 포함된 오버헤드최소화 코딩기법의 부호기와 복호기의 구조는 각각 도 2와 도 3과 같다.

[0028] 도 2는 본 발명의 일 실시예에 따른 오버헤드 최소화 코딩 부호기 구조를 나타내는 도면이다.

[0029] 도 3은 본 발명의 일 실시예에 따른 오버헤드 최소화 코딩 복호기 구조를 나타내는 도면이다.

[0030] 오버헤드 최소화 코드(Overhead Minimized Code; OM Code) 생성부는 N-bits 크기의 $2^{N-1}-2$개의 데이터로부터 최소 한번의 천이를 보장하는 OM 코드를 생성한다. 생성된 OM 코드는 리드-솔로몬 부호화기(Reed-Solomon Encoder)를 통해 오류정정을 가능하도록 부호화 하고, 데이터 입력은 OM 코드가 xor연산을 통해 부호화한다. 복호기는 부호화된 데이터를 입력 받아 리드-솔로몬 복호화기(Reed-Solomon Decoder)를 통해 OM 코드를 오류정정 및 복호하고, 데이터는 OM 코드와 xor연산을 통해 복호화한다. 부호화기는 $2^{N-1}-2$개의 부호화된 데이터마다 RS 부호화된 OM 코드를 출력한다. 복호화기는 RS복호화된 하나의 OM 코드를 이용해 $2^{N-1}-2$의 부호화된 데이터를 복호한다.

[0031] 한 패킷당 최소 한번의 천이를 보장하는 오버헤드최소화 코딩기법의 기본원리는 다음과 같다. N비트 크기의 X,

Y, Z에 대해 $Z = X\,xor\,Y$일 때, $\sum_{i=1}^{N} Z_i = 0$ 또는 $\prod_{i=1}^{N} Z_i = 1$ 이면 $X = Y$ 또는 $X = \sim Y$이다(Z_i: Z의 i번째 비트). 즉, Z의 성분이 모두 '0'이거나 '1'이면, X와 Y는 동일 하거나 보수 관계이다. 위 원리를 역으로 이용하면, $X \neq Y$이고 $X \neq \sim Y$의 관계인 X와 Y를 XOR 연산하면 N비트 크기의 한 패킷에서 최소 한번의 천이를 보장하는 Z를 얻을 수 있다. N-bits 크기의 $2^{N-1}-2$개의 데이터 입력을 xor연산 했을 때 최소 한번의 천이를 보장 하게하는 OM 코드를 찾는 알고리즘은 아래와 같으며 제안하는 OM 코드 생성부의 구조는 도 4와 같다.

[0033] 도 4는 본 발명의 일 실시예에 따른 오버헤드 최소화 코드 생성부의 구조를 나타내는 도면이다.

[0034] N-bits의 입력(A_i)이 들어올 경우, MSB 1bit로 반전 여부를 판단하여, MSB가 1이면, LSB (N-1)-bits의 반전된 값을, 0이면 LSB (N-1)-bits값을 E_i라고 하면, $D_i = 2^{E_i}$인 (2^{N-1})-bits 크기의 D_i를 생성한다.

[0035] D_i를 (2^{N-1})-bits 크기 B_i에 저장한다.($B_i = B_{i-1} or D_i$) (1)~(2)의 동작을 $2^{N-1}-2$번 반복한다.

[0036] B의 LSB bit부터 MSB bit까지 0인 bit가 최소 한 개 존재한다. 0의 위치의 값이 2^C인 경우 C는 1번째부터 $2^{N-1}-2$까지의 모든 데이터 A_i에 대해 $A_i \neq C$ 및 $A_i \neq \sim C$ 이다. $Z = C\,xor\,A_i$로 최소 한번의 천이를 보장하는 Z를 얻는다. 이 때 사용 가능한 C의 값 중 하나를 OM 코드로 선택한다.

[0038] 다음으로, 단계(120)에서, 오버헤드 코드 워드 전송 시에 삽입될 수 있는 오류를 정정하기 위한 리드-솔로몬(Reed-Solomon) 부호화를 수행한다.

[0039] 오류 정정기법은 리드-솔로몬 부호(Reed-Solomon Code)를 사용한다. 이는 채널에서 OM 코드에 잡음오류가 생길 때, $2^{N-1}-2$개의 데이터 모두 손실됨을 방지하기 위해 존재한다. RS 코드(n, k)는 총 n개의 심볼(Symbol)로 k개의 메시지 심볼(Message Symbol)과 n-k개의 패리티 심볼(Parity Symbol)을 가지며, (n-k)/2개 심볼의 오류 정정능력을 갖는다. 제안하는 구조에서 OM 코드를 오류 정정기법 복/부호화하여 채널에서의 오류를 방지한다.

[0040] 마지막으로, 단계(130)에서, 오버헤드 코드 워드를 이용하여 데이터를 부호화를 수행한다.

[0042] 도 5는 본 발명의 일 실시예에 따른 오류 정정기능 시뮬레이션 결과를 나타내는 도면이다.

[0043] 제안하는 오류정정 기능이 포함된 오버헤드최소화 코딩기법은 Verilog HDL로 설계하였으며 Xilinx Vivado 2017.5로 시뮬레이션을 진행하였다. 채널오류를 정정하여 복호화 되는 것을 확인하기 위해서 가상의 오류를 생성하여, RS(7, 3)부호화, 복호화를 확인하였다. Figure 4는 가상의 채널오류를 구성한 오류정정기능 확인 시뮬레이션 파형이다. 오류가 생긴 RS부호화된 데이터를 오류 정정하는 것을 확인했다. Figure 5는 오류 정정기능이 포함된 오버헤드최소화 코딩기법 시뮬레이션 결과이다. 이는 한 패킷을 6-bits길이로 설계하였다. 데이터 소스에 임의의 데이터를 인가하여, 오류정정기능이 포함된 오버헤드최소화 코딩이 부호화 되어, 올바르게 복호화 되는 것을 확인하였다.

[0045] 도 6은 본 발명의 일 실시예에 따른 오류 정정기능이 포함된 오버헤드최소화 코딩기법 시뮬레이션 결과를 나타내는 도면이다.

[0046] 제안된 오류정정기능이 포함된 오버헤드소화 코딩기법은 오버헤드비율을 현저히 감소시키고 클럭을 내장하여, 고속직렬인터페이스에서의 데이터전송효율을 높일 수 있다. 또한 복호화에 필요한 OM 코드에 오류 정정기능을 추가하여 채널 잡음에 의한 오류를 정정할 수 있다. 이러한 장점들로 기존 디스플레이포트, HDMI, 화면 내부 인터페이스 등에 사용되는 블록코딩을 대체할 수 있으며, 같은 속도의 데이터전송에서 더 많은 데이터를 송수신 할 수 있다.

[0048] 이상에서 설명된 장치는 하드웨어 구성요소, 소프트웨어 구성요소, 및/또는 하드웨어 구성요소 및 소프트웨어 구성요소의 조합으로 구현될 수 있다. 예를 들어, 실시예들에서 설명된 장치 및 구성요소는, 예를 들어, 프로세서, 콘트롤러, ALU(arithmetic logic unit), 디지털 신호 프로세서(digital signal processor), 마이크로컴퓨터, FPA(field programmable array), PLU(programmable logic unit), 마이크로프로세서, 또는 명령(instruction)을 실행하고 응답할 수 있는 다른 어떠한 장치와 같이, 하나 이상의 범용 컴퓨터 또는 특수 목적 컴퓨터를 이용하여 구현될 수 있다. 처리 장치는 운영 체제(OS) 및 상기 운영 체제 상에서 수행되는 하나 이상의 소프트웨어 애플리케이션을 수행할 수 있다. 또한, 처리 장치는 소프트웨어의 실행에 응답하여, 데이터를 접근, 저장, 조작, 처리 및 생성할 수도 있다. 이해의 편의를 위하여, 처리 장치는 하나가 사용되는 것으로 설명된 경우도 있지만, 해당 기술분야에서 통상의 지식을 가진 자는, 처리 장치가 복수 개의 처리 요소(processing element) 및/또는 복수 유형의 처리 요소를 포함할 수 있음을 알 수 있다. 예를 들어, 처리 장치는 복수 개의 프로세서 또는 하나의 프로세서 및 하나의 콘트롤러를 포함할 수 있다. 또한, 병렬 프로세서(parallel processor)와 같은, 다른 처리 구성(processing configuration)도 가능하다.

[0049] 소프트웨어는 컴퓨터 프로그램(computer program), 코드(code), 명령(instruction), 또는 이들 중 하나 이상의 조합을 포함할 수 있으며, 원하는 대로 동작하도록 처리 장치를 구성하거나 독립적으로 또는 결합적으로(collectively) 처리 장치를 명령할 수 있다. 소프트웨어 및/또는 데이터는, 처리 장치에 의하여 해석되거나 처리 장치에 명령 또는 데이터를 제공하기 위하여, 어떤 유형의 기계, 구성요소(component), 물리적 장치, 가상 장치(virtual equipment), 컴퓨터 저장 매체 또는 장치에 구체화(embody)될 수 있다. 소프트웨어는 네트워크로 연결된 컴퓨터 시스템 상에 분산되어서, 분산된 방법으로 저장되거나 실행될 수도 있다. 소프트웨어 및 데이터는 하나 이상의 컴퓨터 판독 가능 기록 매체에 저장될 수 있다.

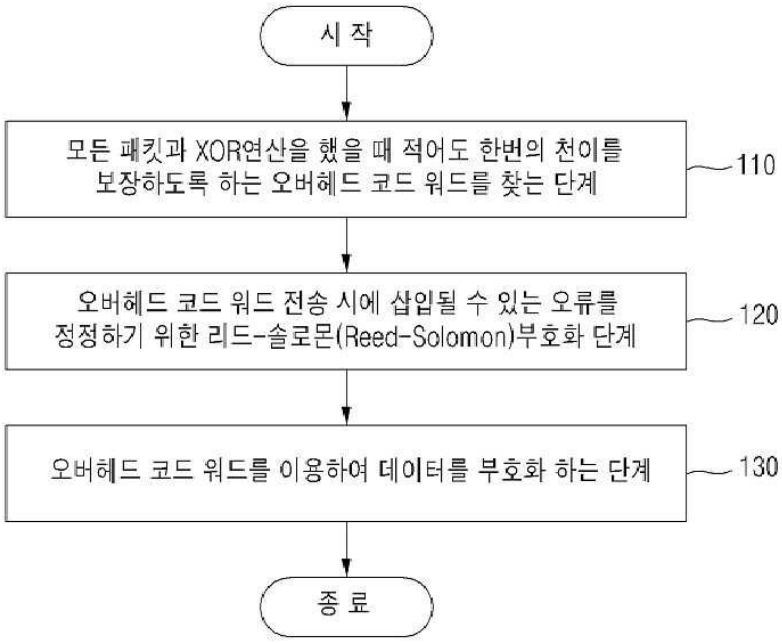
[0050] 실시예에 따른 방법은 다양한 컴퓨터 수단을 통하여 수행될 수 있는 프로그램 명령 형태로 구현되어 컴퓨터 판독 가능 매체에 기록될 수 있다. 상기 컴퓨터 판독 가능 매체는 프로그램 명령, 데이터 파일, 데이터 구조 등을 단독으로 또는 조합하여 포함할 수 있다. 상기 매체에 기록되는 프로그램 명령은 실시예를 위하여 특별히 설계되고 구성된 것들이거나 컴퓨터 소프트웨어 당업자에게 공지되어 사용 가능한 것일 수도 있다. 컴퓨터 판독 가능 기록 매체의 예에는 하드 디스크, 플로피 디스크 및 자기 테이프와 같은 자기 매체(magnetic media), CD-ROM, DVD와 같은 광기록 매체(optical media), 플롭티컬 디스크(floptical disk)와 같은 자기-광 매체(magneto-optical media), 및 롬(ROM), 램(RAM), 플래시 메모리 등과 같은 프로그램 명령을 저장하고 수행하도록 특별히 구성된 하드웨어 장치가 포함된다. 프로그램 명령의 예에는 컴파일러에 의해 만들어지는 것과 같은 기계어 코드뿐만 아니라 인터프리터 등을 사용해서 컴퓨터에 의해서 실행될 수 있는 고급 언어 코드를 포함한다.

[0051] 이상과 같이 실시예들이 비록 한정된 실시예와 도면에 의해 설명되었으나, 해당 기술분야에서 통상의 지식을 가진 자라면 상기의 기재로부터 다양한 수정 및 변형이 가능하다. 예를 들어, 설명된 기술들이 설명된 방법과 다른 순서로 수행되거나, 및/또는 설명된 시스템, 구조, 장치, 회로 등의 구성요소들이 설명된 방법과 다른 형태로 결합 또는 조합되거나, 다른 구성요소 또는 균등물에 의하여 대치되거나 치환되더라도 적절한 결과가 달성될 수 있다.

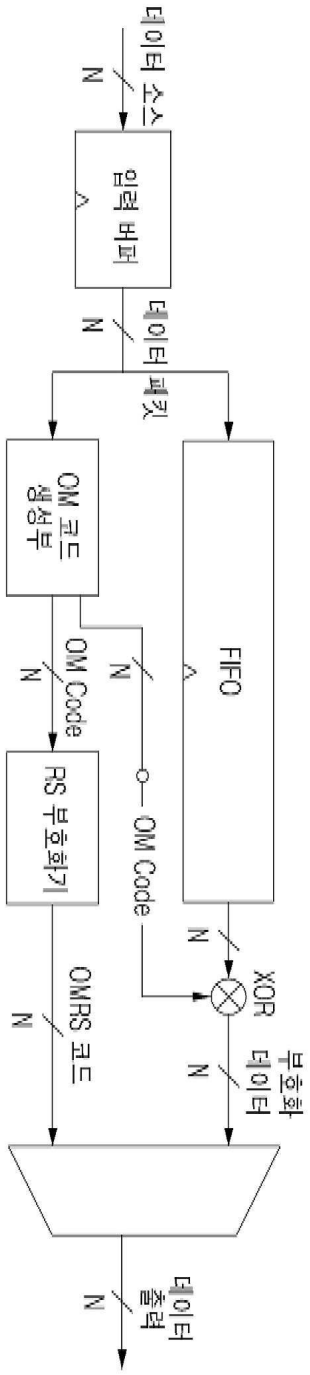
[0052] 그러므로, 다른 구현들, 다른 실시예들 및 특허청구범위와 균등한 것들도 후술하는 특허청구범위의 범위에 속한다.

도면

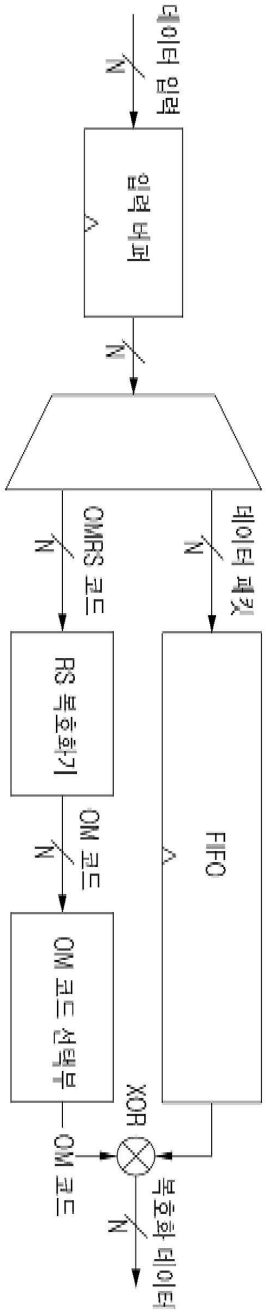
도면1


시 작
모든 패킷과 XOR연산을 했을 때 적어도 한번의 천이를 보장하도록 하는 오버헤드 코드 워드를 찾는 단계
110
오버헤드 코드 워드 전송 시에 삽입될 수 있는 오류를 정정하기 위한 리드-솔로몬(Reed-Solomon)부호화 단계
120
오버헤드 코드 워드를 이용하여 데이터를 부호화 하는 단계
130
종 료

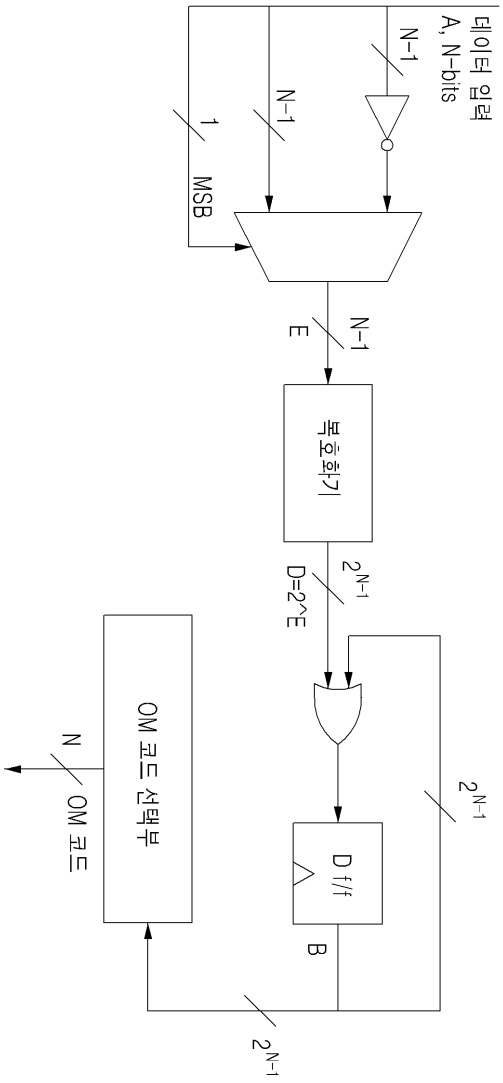
도면2


데이터 소스
N
입력 버퍼
N
데이터 패킷
OM 코드 생성부
FIFO
N
OM Code
N
OM Code
RS 부호화기
N
XOR
N
OMRS 코드
부호화 데이터
N
데이터 출력

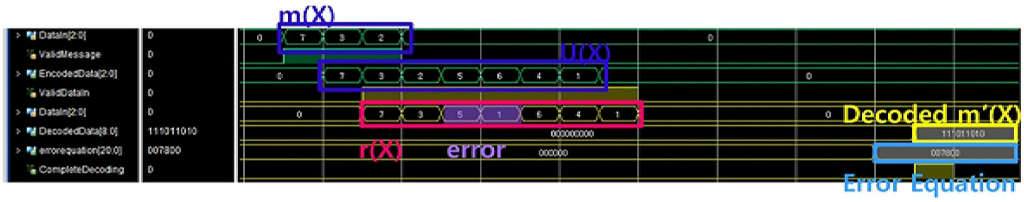
도면3


데이터 입력
N
입력 버퍼
N
CMRS 코드
N
데이터 패킷
N
RS 복호화기
FIFO
OM 코드
N
OM 코드 선택부
XOR
OM 코드
복호화 데이터
N

도면4


데이터 입력
A, N-bits
N-1
N-1
1
MSB
N-1
E
복호화기
2^N-1
D=2^E
OM 코드 선택부
N
OM 코드
D f/f
B
2^N-1
2^N-1

도면5


m(X)
U(X)
r(X)
error
Decoded m'(X)
Error Equation

도면6

