



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2013년05월29일
(11) 등록번호 10-1268997
(24) 등록일자 2013년05월23일

(51) 국제특허분류(Int. Cl.)

G06F 11/26 (2006.01) G06F 11/10 (2006.01)

(21) 출원번호 10-2012-0002924

(22) 출원일자 2012년01월10일

심사청구일자 2012년01월10일

(56) 선행기술조사문헌

KR1020090017382 A*

KR1020100133194 A*

*는 심사관에 의하여 인용된 문헌

(73) 특허권자

조선대학교산학협력단

광주광역시 동구 서석동 375 조선대학교 내

(72) 발명자

이정아

서울특별시 마포구 상암동 휴먼시아아파트 203동 102호

소마순다람, 나타르잔

사이트 엔오. 10비 나가파 콜로니 제2크로스거리 카분담팔라얌, 코임바토레 641030

(74) 대리인

특허법인아이엠

전체 청구항 수 : 총 24 항

심사관 : 이정은

(54) 발명의 명칭 프로그램머블 불 연산 유닛을 위한 확장형 오류검출코드 기반의 오류 검출 장치 및 그 오류 검출 장치를 포함하는 자가검사 프로그램머블 불 연산 유닛

(57) 요약

본 발명은 프로그램머블 불 연산 유닛을 위한 확장형 오류검출코드 기반의 오류 검출 장치 및 그 오류 검출 장치를 포함하는 자가검사 프로그램머블 불 연산 유닛에 관한 것으로, 보다 구체적으로는 입력 데이터들에 대해 불 연산을 수행하여 출력 데이터를 출력하는 프로그램머블 불 연산 유닛의 단방향 연산오류를 최소한의 하드웨어 추가로 연속적으로 검출할 수 있고, 입력 데이터 및 출력 데이터의 길이에 따라 오류검출코드의 길이를 유동적으로 확장할 수 있으며, 오류검출시간을 최소화할 수 있는 프로그램머블 불 연산 유닛을 위한 확장형 오류검출코드 기반의 오류 검출 장치 및 그 오류 검출 장치를 포함하는 자가검사 프로그램머블 불 연산 유닛에 관한 것이다.

대표도 - 도6

113

Operations	opBO	2-bit SEDC Encoded Boolean Operations ($C_{(2BO)}, C_{(2BO)} = SEDC_{2bit}(A(A_i, A_j), B(B_i, B_j), opBO)$)
Logic 0	0	$SEDC_2(00)$
A NOR B	1	$SEDC_2(\bar{A}) + SEDC_2(\bar{B}) - SEDC_2(\bar{A} \text{ OR } \bar{B})$
(NOT A)AND B	2	$SEDC_2(\bar{A}) + SEDC_2(B) - SEDC_2(\bar{A} \text{ OR } B)$
NOT A	3	$SEDC_2(\bar{A})$
A AND (NOT B)	4	$SEDC_2(A) + SEDC_2(\bar{B}) - SEDC_2(A \text{ OR } \bar{B})$
NOT B	5	$SEDC_2(\bar{B})$
A XOR B	6	$SEDC_2(A) + SEDC_2(B) - (2 \times SEDC_2(A \text{ AND } B)) + 3$
A NAND B	7	$SEDC_2(\bar{A}) + SEDC_2(\bar{B}) - SEDC_2(\bar{A} \text{ AND } \bar{B})$
A AND B	8	$SEDC_2(A) + SEDC_2(B) - SEDC_2(A \text{ OR } B)$
A XNOR B	9	$SEDC_2(\bar{A}) + SEDC_2(\bar{B}) - (2 \times SEDC_2(\bar{A} \text{ AND } \bar{B})) + 3$
B	10	$SEDC_2(B)$
(NOT A)OR B	11	$SEDC_2(\bar{A}) + SEDC_2(B) - SEDC_2(\bar{A} \text{ AND } B)$
A	12	$SEDC_2(A)$
A OR (NOT B)	13	$SEDC_2(A) + SEDC_2(\bar{B}) - SEDC_2(A \text{ AND } \bar{B})$
A OR B	14	$SEDC_2(A) + SEDC_2(B) - SEDC_2(A \text{ AND } B)$
Logic 1	15	$SEDC_2(3)$

이 발명을 지원한 국가연구개발사업

과제고유번호 2011-0017914

부처명 교과부

연구사업명 중견연구자지원사업(핵심연구)

연구과제명 오류 모듈에 안정적으로 대처하는 재구성 가능한 가상실험장치 구조 동시설계 연구

주관기관 조선대학교

연구기간 2010.05.01 ~ 2013.04.30

특허청구의 범위

청구항 1

삭제

청구항 2

적어도 두 개의 이진 입력 데이터에 대해 불 연산(Boolean operation)을 수행하여 연산 결과인 이진 출력 데이터를 출력하는 프로그래머블 불 연산 유닛(Programmable Boolean Operation Unit)의 오류를 검출하는 오류 검출 장치로써,

상기 이진 입력 데이터들을 입력받고, 불 연산의 종류에 따라 상기 이진 입력 데이터들에 대한 오류검출코드인 입력 오류검출코드를 생성하는 입력 오류검출코드 생성기; 및

상기 이진 출력 데이터 및 상기 입력 오류검출코드를 입력받고, 상기 이진 출력 데이터에 대한 오류검출코드인 출력 오류검출코드를 생성하며, 상기 출력 오류검출코드와 상기 입력 오류검출코드의 동일성을 판단하여 오류검출결과를 출력하는 오류 검출기;를 포함하고,

상기 입력 오류검출코드 및 상기 출력 오류검출코드는 상기 이진 입력 데이터 또는 상기 이진 출력 데이터의 길이에 따라 가변하는 확장형 오류검출코드(SED_C:Scalable Error Detection Coding)이며,

상기 입력 오류검출코드 생성기:는

2비트 데이터에 대응하는 2비트 오류검출코드들(SED_{C2})을 저장한 2비트 오류검출코드 진리테이블(truth table); 및

상기 이진 입력 데이터들이 2비트 입력 데이터일 경우, 상기 2비트 오류검출코드들(SED_{C2}) 단독이나 복수 개의 산술연산으로, 상기 2비트 입력 데이터들의 불 연산 결과에 대응하는 2비트 입력 오류검출코드(SED_{C2-in})를 계산하고, 상기 2비트 입력 오류검출코드(SED_{C2-in})를 상기 입력 오류검출코드로 생성하는 2비트 입력 오류검출코드 생성수단;을 포함하는 것을 특징으로 하는 오류 검출 장치.

청구항 3

제 2 항에 있어서,

상기 2비트 오류검출코드들(SED_{C2})의 최상위 비트는 2비트 데이터 '00', '01', '10' 또는 '11'에 대해 비트들의 '0'의 개수가 '1'의 개수보다 크거나 같을 경우 '1'로 계산되어 저장되고, 그 외의 경우에는 '0'으로 계산되어 저장되며, 최하위 비트는 '0'의 개수와 '1'의 개수가 동일할 경우 '0'으로 계산되어 저장되고, 그 외의 경우에는 '1'로 계산되어 저장되거나,

최상위 비트는 '0'의 개수와 '1'의 개수가 동일할 경우 '0'으로 계산되어 저장되고, 그 외의 경우에는 '1'로 계산되어 저장되고, 최하위 비트는 '0'의 개수가 '1'의 개수보다 크거나 같을 경우 '1'로 계산되어 저장되고, 그 외의 경우에는 '0'으로 계산되어 저장되는 것을 특징으로 하는 오류 검출 장치.

청구항 4

제 3 항에 있어서,

상기 이진 입력 데이터들은 제1 입력 데이터(A) 및 제2 입력 데이터(A)의 두 개의 입력 데이터로 이루어지고,

상기 입력 오류검출코드 생성기:는

상기 입력 데이터들을 조합할 수 있는 경우에 대응하여,

상기 제1 입력 데이터 및 상기 제2 입력 데이터의 논리곱에 대응하는 2비트 오류검출코드($SEDC_2$)인 2비트 논리곱 오류검출코드($SEDC_2(A \text{ AND } B)$),

상기 제1 입력 데이터 및 상기 제2 입력 데이터의 논리합에 대응하는 2비트 오류검출코드($SEDC_2$)인 2비트 논리합 오류검출코드($SEDC_2(A \text{ OR } B)$) 및

상기 제1 입력 데이터 및 상기 제2 입력 데이터의 논리곱에 대응하는 2비트 오류검출코드($SEDC_2$)를 왼쪽 시프트한 3비트의 오류검출코드인 2비트 시프트곱 오류검출코드($2 \times SEDC_2(A \text{ AND } B)$)를,

2비트 기본 오류검출코드($SEDC_{2-\text{basic}}$)로 저장하는 2비트 기본 오류검출코드 진리테이블;을 더 포함하는 것을 특징으로 하는 오류 검출 장치.

청구항 5

제 4 항에 있어서,

상기 2비트 입력 오류검출코드 생성수단은 상기 2비트 오류검출코드들($SEDC_2$) 또는 상기 2비트 기본 오류검출코드들($SEDC_{2-\text{basic}}$)의 각 단독이나 산술연산 조합에 의해 상기 입력 데이터들의 불 연산 결과에 대응하는 2비트 입력 오류검출코드($SEDC_{2-\text{in}}$)를 상기 입력 오류검출코드로 생성하는 것을 특징으로 하는 오류 검출 장치.

청구항 6

제 5 항에 있어서,

상기 2비트 입력 오류검출코드 생성수단은,

상기 불 연산이 로직 '0' 연산(Logic 0)일 경우, 데이터 '00'의 2비트 오류검출코드($SEDC_2(00)$)를 상기 2비트 입력 오류검출코드($SEDC_{2-\text{in}}$)로 생성하고,

상기 불 연산이 상기 제1 입력 데이터(A) 및 상기 제2 입력 데이터(B)의 부정 논리합연산($A \text{ NOR } B$)일 경우, 상기 제1 입력 데이터(A)의 논리 부정 값($\text{NOT } A$)의 2비트 오류검출코드($SEDC_2(\text{NOT } A)$)에 상기 제2 입력데이터(B)의 논리 부정 값($\text{NOT } B$)의 2비트 오류검출코드($SEDC_2(\text{NOT } B)$)를 산술합(+)하고, 상기 제1 입력 데이터(A)의 논리 부정 값($\text{NOT } A$)과 상기 제2 입력데이터(B)의 논리 부정 값($\text{NOT } B$)의 2비트 논리합 오류검출코드($SEDC_2((\text{NOT } A) \text{ OR } (\text{NOT } B)))$ 를 산술차(-)하여 상기 2비트 입력 오류검출코드($SEDC_{2-\text{in}}$)를 생성하고,

상기 불 연산이 상기 제1 입력 데이터(A)의 논리 부정 값($\text{NOT } A$)과 상기 제2 입력 데이터(B)의 논리곱연산($(\text{NOT } A) \text{ AND } B$)일 경우, 상기 제1 입력 데이터(A)의 논리 부정 값($\text{NOT } A$)의 2비트 오류검출코드($SEDC_2(\text{NOT } A)$)에 상기 제2 입력데이터(B)의 2비트 오류검출코드($SEDC_2(B)$)를 산술합(+)하고, 상기 제1 입력 데이터(A)의 논리 부정 값($\text{NOT } A$)과 상기 제2 입력데이터(B)의 2비트 논리합 오류검출코드($SEDC_2((\text{NOT } A) \text{ OR } B))$ 를 산술차(-)하여 상기 2비트 입력 오류검출코드($SEDC_{2-\text{in}}$)를 생성하고,

상기 불 연산이 상기 제1 입력 데이터(A)의 논리부정연산($\text{NOT } A$)일 경우, 상기 제1 입력 데이터(A)의 논리 부정 값($\text{NOT } A$)의 2비트 오류검출코드($SEDC_2(\text{NOT } A)$)를 상기 2비트 입력 오류검출코드($SEDC_{2-\text{in}}$)로 생성하고,

상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 논리 부정값($\text{NOT } B$)의 논리곱연산($A \text{ AND } (\text{NOT } B)$)일 경우, 상기 제1 입력 데이터(A)의 2비트 오류검출코드($SEDC_2(A)$)에 상기 제2 입력데이터(B)의 논리 부정값($\text{NOT } B$)의 2비트 오류검출코드($SEDC_2(\text{NOT } B)$)를 산술합(+)하고, 상기 제1 입력 데이터(A)와 상기 제2 입력데이터(B)의 논리 부정값($\text{NOT } B$)의 2비트 논리합 오류검출코드($SEDC_2(A \text{ OR } (\text{NOT } B)))$ 를 산술차(-)하여 상기 2

비트 입력 오류검출코드($SEDC_{2-in}$)를 생성하고,

상기 불 연산이 상기 제2 입력 데이터(B)의 논리부정연산($NOT\ B$)일 경우, 상기 제2 입력 데이터(B)의 논리 부정값($NOT\ B$)의 2비트 오류검출코드($SEDC_2(NOT\ B)$)를 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)로 생성하고,

상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 배타적 논리합연산($A\ XOR\ B$)일 경우, 상기 제1 입력 데이터(A)의 2비트 오류검출코드($SEDC_2(A)$)에 상기 제2 입력데이터(B)의 논리 부정값($NOT\ B$)의 2비트 오류검출코드($SEDC_2(NOT\ B)$)를 산술합(+)하고, 상기 제1 입력 데이터(A)와 상기 제2 입력데이터(B)의 2비트 시프트곱 오류검출코드($2 \times SEDC_2(A\ AND\ B)$)를 산술차(-)한 값에, 십진수 '3'을 산술합(+)하여 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)로 생성하고,

상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 부정 논리곱연산($A\ NAND\ B$)일 경우, 상기 제1 입력 데이터(A)의 논리 부정 값($NOT\ A$)의 2비트 오류검출코드($SEDC_2(NOT\ A)$)에 상기 제2 입력데이터(B)의 논리 부정 값($NOT\ B$)의 2비트 오류검출코드($SEDC_2(NOT\ B)$)를 산술합(+)하고, 상기 제1 입력 데이터(A)의 논리 부정 값($NOT\ A$)과 상기 제2 입력데이터(B)의 논리 부정 값($NOT\ B$)의 2비트 논리곱 오류검출코드($SEDC_{2-in}((NOT\ A)\ AND\ (NOT\ B))$)를 산술차(-)하여 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)를 생성하고,

상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 논리곱연산($A\ AND\ B$)일 경우, 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 2비트 논리곱 오류검출코드($SEDC_2(A\ AND\ B)$)를 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)로 생성하거나, 상기 제1 입력 데이터(A)의 2비트 오류검출코드($SEDC_2(A)$)에 상기 제2 입력데이터(B)의 2비트 오류검출코드($SEDC_2(B)$)를 산술합(+)하고, 상기 제1 입력 데이터(A)와 상기 제2 입력데이터(B)의 2비트 논리합 오류검출코드($SEDC_2(A\ OR\ B)$)를 산술차(-)하여 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)를 생성하고,

상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 배타적 부정 논리합연산($A\ XNOR\ B$)일 경우, 상기 제1 입력 데이터(A)의 논리 부정 값($NOT\ A$)의 2비트 오류검출코드($SEDC_2(NOT\ A)$)에 상기 제2 입력데이터(B)의 2비트 오류검출코드($SEDC_2(NOT\ B)$)를 산술합(+)하고, 상기 제1 입력 데이터(A)의 논리 부정값($NOT\ A$)과 상기 제2 입력데이터(B)의 2비트 시프트곱 오류검출코드($2 \times SEDC_2((NOT\ A)\ AND\ B)$)를 산술차(-)한 값에, 십진수 '3'을 산술합(+)하여 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)를 생성하고,

상기 불 연산이 상기 제2 입력 데이터(B)의 출력연산(B)일 경우, 상기 제2 입력 데이터(B)의 2비트 오류검출코드($SEDC_2(B)$)를 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)로 생성하고,

상기 불 연산이 상기 제1 입력 데이터(A)의 논리 부정값($NOT\ A$)과 상기 제2 입력 데이터(B)의 논리합연산($(NOT\ A)\ OR\ B$)일 경우, 상기 제1 입력 데이터(A)의 논리 부정값($NOT\ A$)의 2비트 오류검출코드($SEDC_2(NOT\ A)$)와 상기 제2 입력데이터(B)의 2비트 오류검출코드($SEDC_2(B)$)를 산술합(+)한 값에, 상기 제1 입력 데이터(A)의 논리 부정값($NOT\ A$)과 상기 제2 입력데이터(B)의 2비트 논리곱 오류검출코드($SEDC_2((NOT\ A)\ AND\ B)$)를 산술차(-)하여 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)를 생성하고,

상기 불 연산이 상기 제1 입력 데이터(A)의 출력연산(A)일 경우, 상기 제2 입력 데이터(A)의 2비트 오류검출코드($SEDC_2(A)$)를 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)로 생성하고,

상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 논리 부정값($NOT\ B$)의 논리합연산($A\ OR\ (NOT\ B)$)일 경우, 상기 제1 입력 데이터(A)의 2비트 오류검출코드($SEDC_2(A)$)와 상기 제2 입력데이터(B)의 논리 부정값($NOT\ B$)의 2비트 오류검출코드($SEDC_2(NOT\ B)$)를 산술합(+)하고, 상기 제1 입력 데이터(A)와 상기 제2 입력데이터(B)의 논리 부정값($NOT\ B$)의 2비트 논리곱 오류검출코드($SEDC_2(A\ AND\ (NOT\ B))$)를 산술차(-)하여 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)를 생성하고,

상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 논리합연산($A\ OR\ B$)일 경우, 상기 제1

입력 데이터(A)와 상기 제2 입력 데이터(B)의 2비트 논리합 오류검출코드($SEDC_2(A \text{ OR } B)$)를 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)로 생성하거나, 상기 제1 입력 데이터(A)의 2비트 오류검출코드($SEDC_2(A)$)에 상기 제2 입력 데이터(B)의 2비트 오류검출코드($SEDC_2(B)$)를 산술합(+)하고, 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 2비트 논리곱 오류검출코드($SEDC_2(A \text{ AND } B)$)를 산술차(-)하여 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)를 생성하고,

상기 불 연산이 로직 '1' 연산(Logic 1)일 경우, 데이터 '11'의 2비트 오류검출코드($SEDC_2(11)$)를 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)로 생성하는 것을 특징으로 하는 오류 검출 장치.

청구항 7

제 6 항에 있어서,

상기 2비트 입력 오류검출코드 생성수단은,

상기 제1 입력 데이터(A)와 제1 제어신호(S_1)의 배타적 논리합 연산을 수행하는 제1 XOR게이트;

이진 데이터 '00'을 출력하는 '00'출력기;

상기 제1 XOR게이트와 상기 '00'출력기의 출력을 입력받고 제3 제어신호(S_3)에 의해 입력되는 신호들 중 어느 하나의 신호를 출력하는 제1 2비트 2×1 멀티플렉서(MUX);

상기 제1 2비트 2×1 멀티플렉서의 출력을 입력받아 2비트 오류검출코드를 출력하는 제1 2비트 오류검출코드 생성기;

상기 제1 2비트 오류검출코드 생성기의 출력과 제5 제어신호(S_5)를 입력받아 논리곱 연산을 수행하는 제1 AND게이트;

상기 제2 입력 데이터(B)와 제2 제어신호(S_2)의 배타적 논리합 연산을 수행하는 제2 XOR게이트;

이진 데이터 '11'을 출력하는 제1 '11'출력기;

상기 제2 XOR게이트와 상기 '11'출력기의 출력을 입력받고 상기 제3 제어신호(S_3)에 의해 입력되는 신호들 중 어느 하나의 신호를 출력하는 제2 2비트 2×1 멀티플렉서;

상기 제2 2비트 2×1 멀티플렉서의 출력을 입력받아 2비트 오류검출코드를 출력하는 제2 2비트 오류검출코드 생성기;

상기 제2 2비트 오류검출코드 생성기의 출력과 제6 제어신호(S_6)를 입력받아 논리곱 연산을 수행하는 제2 AND게이트;

이진 데이터 '11'을 출력하는 제2 '11'출력기;

상기 제2 '11'출력기의 출력과 제7 제어신호(S_7)를 입력받아 논리곱 연산을 수행하는 제3 AND게이트;

상기 제1 2비트 2×1 멀티플렉서와 상기 제2 2비트 2×1 멀티플렉서의 출력을 논리합 연산하는 OR게이트;

상기 제1 2비트 2×1 멀티플렉서와 상기 제2 2비트 2×1 멀티플렉서의 출력을 논리곱 연산하는 제4 AND게이트;

상기 OR게이트와 상기 제4 AND게이트의 출력을 입력받고 제4 제어신호(S_4)에 의해 입력되는 신호들 중 어느 하나의 신호를 출력하는 제3 2비트 2×1 멀티플렉서;

상기 제3 2비트 2×1 멀티플렉서의 출력을 입력받아 2비트 오류검출코드를 출력하는 제3 2비트 오류검출코드 생성기;

상기 제3 2비트 오류검출코드 생성기의 출력과 제8 제어신호(S_8)를 논리곱 연산하는 제4 AND게이트;

상기 제4 AND게이트의 출력을 입력받고 상기 제7 제어신호(S_7)에 의해 상기 제4 AND게이트의 출력을 왼쪽 시프트 연산하는 Left Shift게이트;

상기 제1 AND게이트, 상기 제2 AND게이트 및 상기 제3 AND게이트의 출력을 산술합(+)하고 상기 Left Shift게이

트의 출력을 산술차(-)하여 상기 입력 오류검출코드를 출력하는 가산기(Block Adder); 및
 상기 불 연산의 종류에 따라 상기 제어신호들($S_1, S_2, S_3, S_4, S_5, S_6, S_7, S_8$)을 출력하는 제어신호 생성기;를 포함하는 논리회로로 구성되는 것을 특징으로 하는 오류 검출 장치.

청구항 8

제 6 항에 있어서,

상기 이진 입력 데이터가 1비트 입력 데이터일 경우, 상기 2비트 입력 오류검출코드 생성수단은 상기 1비트 입력 데이터를 2비트 입력 데이터들로 가정하여, 최상위 비트는 '0'으로 설정하고, 최하위 비트는 상기 1비트 입력 데이터로 설정하여 상기 각 불 연산의 경우에 해당하는 오류검출코드의 계산을 수행하고, 계산된 오류검출코드 중, 최하위 비트만을 상기 입력 오류검출코드로 생성하되,

상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 배타적 논리합연산(A XOR B) 또는 배타적 부정 논리합연산(A XNOR B)일 경우에는, 상기 십진수 '3'이 아닌 십진수 '1'을 산술합(+)한 값의 최하위 비트를 상기 입력 오류검출코드로 생성하는 것을 특징으로 하는 오류 검출 장치.

청구항 9

제 8 항에 있어서,

상기 오류 검출기는 상기 이진 입력 데이터가 1비트 입력 데이터일 경우, 상기 입력 오류검출코드의 최하위 비트와 상기 출력 오류검출코드의 최하위 비트를 서로 비교하여 오류검출결과를 출력하는 것을 특징으로 하는 오류 검출 장치.

청구항 10

제 6 항에 있어서,

상기 입력 오류검출코드 생성기:는

상기 이진 입력 데이터들이 3비트 입력 데이터일 경우 2비트인 3비트 입력 오류검출코드($SEDC_{3-in}$)를 상기 입력 오류검출코드로 생성하는 3비트 입력 오류검출코드 생성수단;을 더 포함하고,

상기 3비트 입력 오류검출코드 생성수단은 상기 3비트 입력 데이터의 하위 두 비트를 입력으로 하는 상기 2비트 입력 오류검출코드 생성수단의 2비트 입력 오류검출코드($SEDC_{2-in}$)를 과도 2비트 입력 오류검출코드로 생성하고, 선택된 불 연산의 종류 및 상기 이진 입력 데이터들 각각의 최상위 비트의 값에 따라 상기 과도 2비트 입력 오류검출코드 자체 또는 상기 과도 2비트 입력 오류검출코드에 '1'을 산술차(-)하여 상기 3비트 입력 오류검출코드($SEDC_{3-in}$)를 생성하는 것을 특징으로 하는 오류 검출 장치.

청구항 11

제 10 항에 있어서,

상기 3비트 입력 오류검출코드 생성수단이 상기 과도 2비트 입력 오류검출코드에 '1'을 산술차(-)하여 상기 3비트 입력 오류검출코드($SEDC_{3-in}$)를 생성하는 경우는,

상기 불 연산이 상기 제1 입력 데이터(A) 및 상기 제2 입력 데이터(B)의 부정 논리합연산(A NOR B)이고, 상기 제1 입력 데이터(A) 및 상기 제2 입력 데이터(B)의 최상위 비트가 '0'인 경우,

상기 불 연산이 상기 제1 입력 데이터(A)의 논리 부정 값(NOT A)과 상기 제2 입력 데이터(B)의 논리곱연산((NOT A) AND B)이고, 상기 제1 입력 데이터(A)의 최상위 비트가 '0'이고 상기 제2 입력 데이터(B)의 최상위 비트가

'1'인 경우,

상기 불 연산이 상기 제1 입력 데이터(A)의 논리부정연산(NOT A)이고, 상기 제2 입력 데이터(B)의 최상위 비트에 관계없이 상기 제1 입력 데이터(A)의 최상위 비트가 '0'인 경우,

상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 논리 부정값(NOT B)의 논리곱연산(A AND (NOT B))이고, 상기 제1 입력 데이터(A)의 최상위 비트가 '1'이고, 상기 제2 입력 데이터(B)의 최상위 비트가 '0'인 경우,

상기 불 연산이 상기 제2 입력 데이터(B)의 논리부정연산(NOT B)이고, 상기 제1 입력 데이터(A)의 최상위 비트에 관계없이 상기 제2 입력 데이터(B)의 최상위 비트가 '0'인 경우,

상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 배타적 논리합연산(A XOR B)이고, 상기 제1 입력 데이터(A)의 최상위 비트가 '1'이고, 상기 제2 입력 데이터(B)의 최상위 비트가 '0'이거나, 상기 제1 입력 데이터(A)의 최상위 비트가 '0'이고, 상기 제2 입력 데이터(B)의 최상위 비트가 '1'인 경우,

상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 부정 논리곱연산(A NAND B)이고, 상기 제1 입력 데이터(A) 및 상기 제2 입력 데이터(B)의 최상위 비트가 '1'이 아닌 경우,

상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 논리곱연산(A AND B)이고, 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 최상위 비트가 '1'인 경우,

상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 배타적 부정 논리합연산(A XNOR B)이고, 상기 제1 입력 데이터(A)의 최상위 비트가 '1'이고, 상기 제2 입력 데이터(B)의 최상위 비트가 '0'이거나, 상기 제1 입력 데이터(A)의 최상위 비트가 '0'이고, 상기 제2 입력 데이터(B)의 최상위 비트가 '1'인 경우,

상기 불 연산이 상기 제2 입력 데이터(B)의 출력연산(B)이고, 상기 제1 입력 데이터(A)의 최상위 비트에 관계없이 상기 제2 입력 데이터(B)의 최상위 비트가 '1'인 경우,

상기 불 연산이 상기 제1 입력 데이터(A)의 논리 부정값(NOT A)과 상기 제2 입력 데이터(B)의 논리합연산((NOT A) OR B)이고, 상기 제1 입력 데이터(A)의 최상위 비트가 '1'이고 상기 제2 입력 데이터(B)의 최상위 비트가 '0'이 아닌 경우,

상기 불 연산이 상기 제1 입력 데이터(A)의 출력연산(A)이고, 상기 제2 입력 데이터(B)의 최상위 비트에 관계없이 상기 제1 입력 데이터(A)의 최상위 비트가 '1'인 경우,

상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 논리 부정값(NOT B)의 논리합연산(A OR (NOT B))이고, 상기 제1 입력 데이터(A)의 최상위 비트가 '0'이고 상기 제2 입력 데이터(B)의 최상위 비트가 '1'이 아닌 경우,

상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 논리합연산(A OR B)이고, 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 최상위 비트가 '0'이 아닌 경우 또는

상기 제1 입력 데이터(A) 및 상기 제2 입력 데이터(B)의 최상위 비트에 관계없이 상기 불 연산이 로직 '1'(Logic 1)연산일 경우인 것을 특징으로 하는 오류 검출 장치.

청구항 12

제 11 항에 있어서,

상기 입력 오류검출코드 생성기:는

상기 이진 입력 데이터들이 4비트 입력 데이터일 경우, 3비트인 4비트 입력 오류검출코드(SED_{C4-in})를 상기 입력 오류검출코드로 생성하는 4비트 입력 오류검출코드 생성수단;을 더 포함하고,

상기 4비트 입력 오류검출코드 생성수단은 상기 4비트 입력 데이터의 하위 세 비트를 입력으로 하는 상기 3비트 입력 오류검출코드 생성수단의 3비트 입력 오류검출코드(SED_{C3-in})를 상기 4비트 입력 오류검출코드(SED_{C4-in})의 하위 두 비트로 생성하고, 상기 4비트 입력 오류검출코드(SED_{C4-in})의 최상위 비트는 선택된 불 연산의 종류 및 상기 이진 입력 데이터들 각각의 최상위 비트의 값에 따라 '1' 또는 '0' 값으로 생성하는 것을 특징으로 하는

오류 검출 장치.

청구항 13

제 12 항에 있어서,

상기 4비트 입력 오류검출코드($SEDC_{4-in}$)의 최상위 비트가 '1'로 생성되는 경우는,

상기 3비트 입력 오류검출코드 생성수단이 상기 과도 2비트 입력 오류검출코드에 '1'을 산술차(-)하는 경우들 이외의 경우인 것을 특징으로 하는 오류 검출 장치.

청구항 14

제 12 항에 있어서,

상기 이진 입력 데이터들이 n비트 입력 데이터일 경우 상기 n비트 입력 데이터에 대한 오류검출코드인 n비트 입력 오류검출코드($SEDC_{n-in}$)를 상기 입력 오류검출코드로 생성하는 n비트 입력 오류검출코드 생성수단;을 더 포함하고,

상기 n비트 입력 오류검출코드 생성수단은 상기 입력 오류검출코드 생성수단들 각각의 오류검출코드($SEDC_{2-in}, SEDC_{3-in}, SEDC_{4-in}$)나, 상기 입력 오류검출코드 생성수단들 중 어느 하나의 입력 오류검출코드 생성수단의 오류검출코드와 하나 또는 복수 개의 3비트 입력 오류검출코드($SEDC_{3-in}$)의 조합을 상기 입력 오류검출코드로 생성하는 것을 특징으로 하는 오류 검출 장치.

청구항 15

제 2 항 내지 제 14 항 중 어느 한 항에 있어서,

상기 오류 검출기:는

상기 이진 출력 데이터가 2비트 출력 데이터일 경우, 상기 2비트 출력 데이터에 대한 2비트의 오류검출코드인 2비트 출력 오류검출코드($SEDC_{2-out}$)를 상기 출력 오류검출코드로 생성하는 2비트 출력 오류검출코드 생성수단; 및

상기 입력 오류검출코드와 상기 출력 오류검출코드의 동일성을 비교하여 동일하지 않을 경우 오류 발생신호를 출력하는 코드비교수단;을 포함하는 것을 특징으로 하는 오류 검출 장치.

청구항 16

제 15 항에 있어서,

상기 2비트 출력 오류검출코드 생성수단은,

상기 2비트 출력 오류검출코드들($SEDC_{2-out}$)의 최상위 비트는 2비트 데이터 '00', '01', '10' 또는 '11'에 대해 비트들의 '0'의 개수가 '1'의 개수보다 크거나 같을 경우 '1'로 생성하고, 그 외의 경우에는 '0'으로 생성하며, 최하위 비트는 '0'의 개수와 '1'의 개수가 동일할 경우 '0'으로 생성하고, 그 외의 경우에는 '1'로 생성하거나, 최상위 비트는 '0'의 개수와 '1'의 개수가 동일할 경우 '0'으로 생성되고, 그 외의 경우에는 '1'로 생성하며, 최하위 비트는 '0'의 개수가 '1'의 개수보다 크거나 같을 경우 '1'로 생성하고, 그 외의 경우에는 '0'으로 생성하되,

상기 2비트 출력 오류검출코드($SEDC_{2-out}$)는 동일한 입력에 대해 상기 2비트 오류검출코드($SEDC_2$)와 동일한 오류검출코드로 생성되는 것을 특징으로 하는 오류 검출 장치.

청구항 17

제 15 항에 있어서,

상기 오류 검출기는,

상기 이진 출력 데이터가 3비트 출력 데이터일 경우, 상기 3비트 출력 데이터에 대한 2비트의 오류검출코드인 3비트 출력 오류검출코드($SEDC_{3-out}$)를 상기 출력 오류검출코드로 생성하는 3비트 출력 오류검출코드 생성수단;을 더 포함하는 것을 특징으로 하는 오류 검출 장치.

청구항 18

제 17 항에 있어서,

상기 3비트 출력 오류검출코드 생성수단은 상기 3비트 출력 데이터의 최상위 비트가 '0'일 경우에는, 중간비트 및 최하위 비트를 입력으로 하는 상기 2비트 출력 오류검출코드 생성수단의 2비트 출력 오류검출코드($SEDC_{2-out}$)와 동일하게 상기 3비트 출력 오류검출코드($SEDC_{3-out}$)를 생성하고, 상기 3비트 출력 데이터의 최상위 비트가 '1'일 경우에는 중간비트 및 최하위 비트의 논리 부정값을 입력으로 하는 상기 2비트 출력 오류검출코드 생성수단의 2비트 출력 오류검출코드($SEDC_{2-out}$)의 논리 부정값을 상기 3비트 출력 오류검출코드($SEDC_{3-out}$)로 생성하는 것을 특징으로 하는 오류 검출 장치.

청구항 19

제 18 항에 있어서,

상기 오류 검출기는,

상기 이진 출력 데이터가 4비트 출력 데이터일 경우, 상기 4비트 출력 데이터에 대한 3비트의 오류검출코드인 4비트 출력 오류검출코드($SEDC_{4-out}$)를 상기 출력 오류검출코드로 생성하는 4비트 출력 오류검출코드 생성수단;을 더 포함하는 것을 특징으로 하는 오류 검출 장치.

청구항 20

제 19 항에 있어서,

상기 4비트 출력 오류검출코드 생성수단은 상기 4비트 출력 오류검출코드($SEDC_{4-out}$)의 하위 2비트는 상기 4비트 출력 데이터 중, 최상위 비트를 제외한 3비트를 입력으로 하는 상기 3비트 출력 오류검출코드 생성수단의 3비트 출력 오류검출코드($SEDC_{3-out}$)로 생성하고, 상기 4비트 출력 오류검출코드($SEDC_{4-out}$)의 최상위 비트는 상기 4비트 출력 데이터의 최상위 비트의 논리 부정 값으로 생성하는 것을 특징으로 하는 오류 검출 장치.

청구항 21

제 20 항에 있어서,

상기 오류 검출기는,

상기 이진 출력 데이터가 n비트 출력 데이터일 경우, 상기 n비트 출력 데이터에 대한 오류검출코드인 n비트 출력 오류검출코드($SEDC_{n-out}$)를 상기 출력 오류검출코드로 생성하는 n비트 출력 오류검출코드 생성수단;을 더 포함하고,

상기 n 비트 출력 오류검출코드 생성수단은 상기 출력 오류검출코드 생성수단들 각각의 오류검출코드($SEDC_{2-out}$, $SEDC_{3-out}$, $SEDC_{4-out}$)나, 상기 출력 오류검출코드 생성수단들 중 어느 하나의 출력 오류검출코드 생성수단의 오류검출코드와 하나 또는 복수 개의 3비트 출력 오류검출코드($SEDC_{3-out}$)의 조합을 상기 출력 오류검출코드로 생성하는 것을 특징으로 하는 오류 검출 장치.

청구항 22

제 21 항의 오류 검출 장치; 및

적어도 두 개의 이진 입력 데이터에 대해 불 연산(Boolean operation)을 수행하여 연산 결과인 이진 출력 데이터를 출력하는 프로그래머블 불 연산 유닛;을 포함하는 자가검사 프로그래머블 불 연산 유닛(Self-checking Programmable Boolean Operations Unit).

청구항 23

제 21 항의 오류 검출 장치; 및

적어도 두 개의 이진 입력 데이터에 대해 불 연산(Boolean operation)을 수행하여 연산 결과인 이진 출력 데이터를 출력하는 프로그래머블 산술/논리 연산유닛을 포함하는 자가검사 프로그래머블 산술/논리 연산유닛(Self-checking Programmable Arithmetic and Logic Unit).

청구항 24

컴퓨터를 제 21 항의 오류 검출 장치로 기능하게 하는 오류 검출 프로그램이 저장된 컴퓨터로 읽을 수 있는 기록매체.

청구항 25

제 24 항의 오류 검출 프로그램이 저장되고 통신망을 통해 상기 오류 검출 프로그램을 전송할 수 있는 서버 시스템.

명세서

기술분야

[0001] 본 발명은 프로그래머블 불 연산 유닛을 위한 확장형 오류검출코드 기반의 오류 검출 장치 및 그 오류 검출 장치를 포함하는 자가검사 프로그래머블 불 연산 유닛에 관한 것으로, 보다 구체적으로는 입력 데이터들에 대해 불 연산을 수행하여 출력 데이터를 출력하는 프로그래머블 불 연산 유닛의 단방향 연산오류를 최소한의 하드웨어 추가로 연속적으로 검출할 수 있고, 입력 데이터들 및 출력 데이터의 길이에 따라 오류검출코드의 길이를 유동적으로 확장할 수 있으며, 오류검출시간을 최소화할 수 있는 프로그래머블 불 연산 유닛을 위한 확장형 오류검출코드 기반의 오류 검출 장치 및 그 오류 검출 장치를 포함하는 자가검사 프로그래머블 불 연산 유닛에 관한 것이다.

배경기술

[0002] 결함허용(fault-tolerance)이란 시스템에서 이나 연산환경에서 결함이 발생하였을 때, 어떠한 제약도 없이 의도된 기능을 수행할 수 있게 하는 능력을 말한다.

[0003] 또한, 결함허용을 수행하기 위해서는 기본 연산 블록들(basic computational blocks)이 결함을 인식할 수 있어

야하고, 시스템이 기능하는 도중에도 결함을 복구하기 위한 신호를 출력할 수 있어야한다.

- [0004] 오류검출방법은 크게 Built-In-Self-Test(BIST)기법, 로빙테스트 기법, 중복기법, 논리값 기반의 기법, 오류검출코딩기법 등으로 분류된다.
- [0005] 먼저, BIST 기법은 별도의 테스트장비를 요구하지 않는 정적오류검출기법으로 동적결함검출 기능을 필요로 하는 재구성가능 내장형 시스템에는 적합하지 않다.
- [0006] 또한, 로빙테스트 기법은 계산용 데이터의 배열이 동일 크기의 영역으로 분할되며, 이 영역들 중 하나가 BIST를 수행할 수 있도록 구성되며, 나머지 영역들은 본래의 의도된 기능을 수행하도록 구성된다.
- [0007] 또한, 중복기법은 구조적 또는 시간적 중복성에 기반하여 오류를 검출하며 구조적 중복성을 이용한 오류검출기법은 동일한 기능블럭들을 복수 개로 구성하여 오류를 검출하며, 시간적 중복성을 이용한 오류검출기법은 동일한 기능블럭에서 복수 회 기능을 수행하여 오류를 검출한다. 그러나 이러한 중복기법은 시간적, 공간적 추가비용을 초래하는 단점이 있다.
- [0008] 또한, 논리값 기반의 기법은 오류를 검출하고자 하는 회로의 넷리스트를 입력으로 하여 논리값 패턴에 대한 회로의 모든 내부 노드들을 탐색하고, 오류의 발생가능성이 있는 회로의 노드패턴이 검색될 경우, 추가적인 검출 모듈을 추가하는 방식이다. 그러나, 이러한 논리값 기반의 기법은 모든 오류를 검출할 수 없는 단점이 있다.
- [0009] 또한, 오류검출코딩기법은 회로의 면적, 동작속도, 검출 오류의 범위 등에서 다른 기법들과 비교하여 매우 효율적이며, 이러한 방법은 에러검출코딩 알고리즘을 이용하여 이진데이터를 코딩함으로써 해당 이진데이터를 표현하는 코드를 검증하여 단방향 오류를 검출해 낼 수 있다.
- [0010] 또한, 오류검출코딩기법은 패리티코드(parity code), 해밍코드(hamming code), 리드솔로몬코드(Reed-Solomon code), 버저코드(berger code), 보스코드(Bose-lin code) 등과 같이 단방향 오류를 검출해 낼 수 있는 코딩기법들로 연구되어 왔으며, 그중 가장 간단하고 적은 비용으로 오류를 검출할 수 있는 코드는 원래의 이진데이터에 한 비트의 오류확인비트를 추가한 패리티코드이다.
- [0011] 이러한 패리티코드의 오류확인비트는 전체 데이터 비트에 '1'의 개수가 홀수개인지 짝수개인지에 따라 결정되며 매우 적은 하드웨어의 면적추가만으로도 신속한 오류검출기능을 수행할 수 있다. 그러나 패리티코드는 원래의 이진데이터에서 단 한 비트의 오류가 발생했을 경우와 홀수 비트에서 오류가 발생한 경우 이외에 다른 경우에는 오류검출기능을 수행할 수 없는 단점이 있다.
- [0012] 또한, 해밍코드는 에러정정기능을 제공하는 최초의 오류검출코드으로써, 해밍체크비트를 생성하기 위해 각각 다른 비트들에 패리티코딩을 수행한다. 즉, 해밍코드는 오류정정기능을 수행하기 위한 비트가 추가되므로 패리티코드에 비해 약간의 추가적인 하드웨어가 필요하다. 또한, 패리티코드와 유사하게 단일비트오류와 두 비트의 오류만을 검출할 수 있으므로 모든 이진 데이터들의 단방향 오류들을 검출할 수는 없다.
- [0013] 또한, 리드솔로몬코드는 오류검출과 오류정정기능을 함께 제공하는 다항식기반의 오류정정기법이다. 이 방법은 해밍코드와 비교하여 더 많은 하드웨어의 추가가 요구되며 이 역시 모든 이진데이터들의 단방향 오류를 검출할 수는 없다.
- [0014] 또한, 버저코드는 모든 단방향 오류들을 검출할 수는 있으나 오류정정기능을 제공하지 않는다. 이 코딩기법은 원래 이진데이터비트들의 논리 값 '0' 또는 '1'의 개수를 계산하여 그 값을 이진값으로 코딩하여 수행된다.
- [0015] 또한, 보스코드는 논리값 '0'또는 '1'의 개수를 계수하여 코드를 생성한다는 점에서 버저코드와 유사하나, 단방향 오류를 검출하기 위한 모듈(modulo 4 or modulo 8)로 연산을 수행한다는 점이 서로 다르다. 또한, 보스코드는 t-단방향오류를 검출할 수 있으며, 여기서 't'는 원래 이진데이터의 두 비트 혹은 세 비트 오류를 의미한다. 또한, 보스코드는 버저코드와 마찬가지로, 오류정정기능을 제공하지 않는다.
- [0016] 이상에서 살펴본 바와 같이 오류검출코드 기법들 중에서 버저코드만이 입력 이진데이터들의 모든 비트에 대해 단방향 오류를 검출할 수 있다. 그러나 이 기법은 카운터회로를 위한 순차논리회로를 구성하는데 추가적인 하드웨어 면적부담이 초래되며 복수의 클럭사이클이 필요하며, 긴 임계경로로 인한 계산 지연시간이 초래되는 문제점이 있다.

발명의 내용

해결하려는 과제

- [0017] 본 발명자들은 결함 허용 재구성가능(fault-tolerant reconfigurable)하고, 연산 속도나 레이턴시(latency)와 같은 특성을 유지하면서도 입력 데이터의 길이에 따라 선형적으로 오류검출코드를 확장하여 자가검사(self-checking)할 수 있는 프로그래머블 불 연산 유닛을 위한 확장형 오류검출코드 기반의 오류검출장치 및 자가검사 프로그래머블 불 연산 유닛을 개발하기 위해 연구한 결과, 프로그래머블 불 연산 유닛의 기능은 유지시키되 입력 데이터의 연산오류를 검출하여 결함 허용 가능하고, 입력 데이터들과 출력 데이터들의 길이에 대해 선형적으로 오류검출코드를 확장하여 오류 검출을 수행할 수 있는 프로그래머블 불 연산 유닛을 위한 확장형 오류검출코드 기반의 오류검출장치와 상기 오류검출장치를 구비하여 자가검사를 수행할 수 있는 자가검사 프로그래머블 불 연산 유닛의 기술적 구성을 개발하게 되어 본 발명을 완성하게 되었다.
- [0018] 따라서, 본 발명의 목적은 프로그래머블 불 연산 유닛의 기능은 유지시키되 연산의 오류를 검출할 수 있는 결함 허용 재구성가능한 오류 검출 장치를 제공하는 것이다.
- [0019] 또한, 본 발명의 다른 목적은 프로그래머블 불 연산 유닛의 입력 데이터 및 출력데이터의 길이에 따라 선형적으로 오류검출코드를 확장하여 오류 검출을 수행할 수 있는 오류 검출 장치를 제공하는 것이다.
- [0020] 또한, 본 발명의 또 다른 목적은 상기 오류 검출 장치를 구비하여 자가 검사할 수 있는 자가검사 프로그래머블 불 연산 유닛을 제공하는 것이다.
- [0021] 본 발명의 목적들은 이상에서 언급한 목적들로 제한되지 않으며, 언급되지 않은 또 다른 목적들은 아래의 기재로부터 당업자에게 명확하게 이해될 수 있을 것이다.

과제의 해결 수단

- [0022] 상기의 목적을 달성하기 위하여 본 발명은 적어도 두 개의 이진 입력 데이터에 대해 불 연산(Boolean operation)을 수행하여 연산 결과인 이진 출력 데이터를 출력하는 프로그래머블 불 연산 유닛(Programmable Boolean Operation Unit)의 오류를 검출하는 오류 검출 장치로써, 상기 이진 입력 데이터들을 입력받고, 불 연산의 종류에 따라 상기 이진 입력 데이터들에 대한 오류검출코드인 입력 오류검출코드를 생성하는 입력 오류검출코드 생성기; 및 상기 이진 출력 데이터 및 상기 입력 오류검출코드를 입력받고, 상기 이진 출력 데이터에 대한 오류검출코드인 출력 오류검출코드를 생성하며, 상기 출력 오류검출코드와 상기 입력 오류검출코드의 동일성을 판단하여 오류검출결과를 출력하는 오류 검출기;를 포함하고, 상기 입력 오류검출코드 및 상기 출력 오류검출코드는 상기 이진 입력 데이터 또는 상기 이진 출력 데이터의 길이에 따라 가변하는 확장형 오류검출코드(SED_C:Scalable Error Detection Coding)인 것을 특징으로 하는 오류 검출 장치를 제공한다.
- [0023] 바람직한 실시예에 있어서, 상기 입력 오류검출코드 생성기:는 2비트 데이터에 대응하는 2비트 오류검출코드들(SED_{C2})을 저장한 2비트 오류검출코드 진리테이블(truth table); 및 상기 이진 입력 데이터들이 2비트 입력 데이터일 경우, 상기 2비트 오류검출코드들(SED_{C2}) 단독이나 복수 개의 산술연산으로, 상기 2비트 입력 데이터들의 불 연산 결과에 대응하는 2비트 입력 오류검출코드(SED_{C2-in})를 계산하고, 상기 2비트 입력 오류검출코드(SED_{C2-in})를 상기 입력 오류검출코드로 생성하는 2비트 입력 오류검출코드 생성수단:을 포함한다.
- [0024] 바람직한 실시예에 있어서, 상기 2비트 오류검출코드들(SED_{C2})의 최상위 비트는 2비트 데이터 '00', '01', '10' 또는 '11'에 대해 비트들의 '0'의 개수가 '1'의 개수보다 크거나 같을 경우 '1'로 계산되어 저장되고, 그 외의 경우에는 '0'으로 계산되어 저장되며, 최하위 비트는 '0'의 개수와 '1'의 개수가 동일할 경우 '0'으로 계산되어 저장되고, 그 외의 경우에는 '1'로 계산되어 저장되거나, 최상위 비트는 '0'의 개수와 '1'의 개수가 동일할 경우 '0'으로 계산되어 저장되고, 그 외의 경우에는 '1'로 계산되어 저장되고, 최하위 비트는 '0'의 개수가 '1'의 개수보다 크거나 같을 경우 '1'로 계산되어 저장되고, 그 외의 경우에는 '0'으로 계산되어 저장된다.
- [0025] 바람직한 실시예에 있어서, 상기 이진 입력 데이터들은 제1 입력 데이터(A) 및 제2 입력 데이터(A)의 두 개의 입력 데이터로 이루어지고, 상기 입력 오류검출코드 생성기:는 상기 입력 데이터들을 조합할 수 있는 경우에 대응하여, 상기 제1 입력 데이터 및 상기 제2 입력 데이터의 논리곱에 대응하는 2비트 오류검출코드(SED_{C2})인 2비

트 논리곱 오류검출코드($SEDC_2(A \text{ AND } B)$), 상기 제1 입력 데이터 및 상기 제2 입력 데이터의 논리합에 대응하는 2비트 오류검출코드($SEDC_2$)인 2비트 논리합 오류검출코드($SEDC_2(A \text{ OR } B)$) 및 상기 제1 입력 데이터 및 상기 제2 입력 데이터의 논리곱에 대응하는 2비트 오류검출코드($SEDC_2$)를 왼쪽 시프트한 3비트의 오류검출코드인 2비트 시프트곱 오류검출코드($2 \times SEDC_2(A \text{ AND } B)$)를, 2비트 기본 오류검출코드($SEDC_{2-\text{basic}}$)로 저장하는 2비트 기본 오류검출코드 진리테이블;을 더 포함한다.

[0026] 바람직한 실시예에 있어서, 상기 2비트 입력 오류검출코드 생성수단은 상기 2비트 오류검출코드들($SEDC_2$) 또는 상기 2비트 기본 오류검출코드들($SEDC_{2-\text{basic}}$)의 각 단독이나 산술연산 조합에 의해 상기 입력 데이터들의 불 연산 결과에 대응하는 2비트 입력 오류검출코드($SEDC_{2-\text{in}}$)를 상기 입력 오류검출코드로 생성한다.

[0027] 바람직한 실시예에 있어서, 상기 2비트 입력 오류검출코드 생성수단은, 상기 불 연산이 로직 '0' 연산(Logic 0)일 경우, 데이터 '00'의 2비트 오류검출코드($SEDC_2(00)$)를 상기 2비트 입력 오류검출코드($SEDC_{2-\text{in}}$)로 생성하고, 상기 불 연산이 상기 제1 입력 데이터(A) 및 상기 제2 입력 데이터(B)의 부정 논리합연산($A \text{ NOR } B$)일 경우, 상기 제1 입력 데이터(A)의 논리 부정 값($\text{NOT } A$)의 2비트 오류검출코드($SEDC_2(\text{NOT } A)$)에 상기 제2 입력데이터(B)의 논리 부정 값($\text{NOT } B$)의 2비트 오류검출코드($SEDC_2(\text{NOT } B)$)를 산술합(+)하고, 상기 제1 입력 데이터(A)의 논리 부정 값($\text{NOT } A$)과 상기 제2 입력데이터(B)의 논리 부정 값($\text{NOT } B$)의 2비트 논리합 오류검출코드($SEDC_2((\text{NOT } A) \text{ OR } (\text{NOT } B)))$ 를 산술차(-)하여 상기 2비트 입력 오류검출코드($SEDC_{2-\text{in}}$)를 생성하고, 상기 불 연산이 상기 제1 입력 데이터(A)의 논리 부정 값($\text{NOT } A$)과 상기 제2 입력 데이터(B)의 논리곱연산($(\text{NOT } A) \text{ AND } B$)일 경우, 상기 제1 입력 데이터(A)의 논리 부정 값($\text{NOT } A$)의 2비트 오류검출코드($SEDC_2(\text{NOT } A)$)에 상기 제2 입력데이터(B)의 2비트 오류검출코드($SEDC_2(B)$)를 산술합(+)하고, 상기 제1 입력 데이터(A)의 논리 부정 값($\text{NOT } A$)과 상기 제2 입력데이터(B)의 2비트 논리합 오류검출코드($SEDC_2((\text{NOT } A) \text{ OR } B))$ 를 산술차(-)하여 상기 2비트 입력 오류검출코드($SEDC_{2-\text{in}}$)를 생성하고, 상기 불 연산이 상기 제1 입력 데이터(A)의 논리부정연산($\text{NOT } A$)일 경우, 상기 제1 입력 데이터(A)의 논리 부정 값($\text{NOT } A$)의 2비트 오류검출코드($SEDC_2(\text{NOT } A)$)를 상기 2비트 입력 오류검출코드($SEDC_{2-\text{in}}$)로 생성하고, 상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 논리 부정값($\text{NOT } B$)의 논리곱연산($A \text{ AND } (\text{NOT } B)$)일 경우, 상기 제1 입력 데이터(A)의 2비트 오류검출코드($SEDC_2(A)$)에 상기 제2 입력데이터(B)의 논리 부정값($\text{NOT } B$)의 2비트 오류검출코드($SEDC_2(\text{NOT } B)$)를 산술합(+)하고, 상기 제1 입력 데이터(A)와 상기 제2 입력데이터(B)의 논리 부정값($\text{NOT } B$)의 2비트 논리합 오류검출코드($SEDC_2(A \text{ OR } (\text{NOT } B)))$ 를 산술차(-)하여 상기 2비트 입력 오류검출코드($SEDC_{2-\text{in}}$)를 생성하고, 상기 불 연산이 상기 제2 입력 데이터(B)의 논리부정연산($\text{NOT } B$)일 경우, 상기 제2 입력 데이터(B)의 논리 부정 값($\text{NOT } B$)의 2비트 오류검출코드($SEDC_2(\text{NOT } B)$)를 상기 2비트 입력 오류검출코드($SEDC_{2-\text{in}}$)로 생성하고, 상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 배타적 논리합연산($A \text{ XOR } B$)일 경우, 상기 제1 입력 데이터(A)의 2비트 오류검출코드($SEDC_2(A)$)에 상기 제2 입력데이터(B)의 논리 부정값($\text{NOT } B$)의 2비트 오류검출코드($SEDC_2(\text{NOT } B)$)를 산술합(+)하고, 상기 제1 입력 데이터(A)와 상기 제2 입력데이터(B)의 2비트 시프트곱 오류검출코드($2 \times SEDC_2(A \text{ AND } B)$)를 산술차(-)한 값에, 십진수 '3'을 산술합(+)하여 상기 2비트 입력 오류검출코드($SEDC_{2-\text{in}}$)로 생성하고, 상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 부정 논리곱연산($A \text{ NAND } B$)일 경우, 상기 제1 입력 데이터(A)의 논리 부정 값($\text{NOT } A$)의 2비트 오류검출코드($SEDC_2(\text{NOT } A)$)에 상기 제2 입력데이터(B)의 논리 부정 값($\text{NOT } B$)의 2비트 오류검출코드($SEDC_2(\text{NOT } B)$)를 산술합(+)하고, 상기 제1 입력 데이터(A)의 논리 부정 값($\text{NOT } A$)과 상기 제2 입력데이터(B)의 논리 부정 값($\text{NOT } B$)의 2비트 논리곱 오류검출코드($SEDC_{2-\text{in}}((\text{NOT } A) \text{ AND } (\text{NOT } B)))$ 를 산술차(-)하여 상기 2비트 입력 오류검출코드($SEDC_{2-\text{in}}$)를 생성하고, 상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 논리곱연산($A \text{ AND } B$)일 경우, 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 2비트 논리곱 오류검출코드($SEDC_2(A \text{ AND } B)$)를 상기 2비트 입력 오류검출코드($SEDC_{2-\text{in}}$)로 생성하거나, 상기 제1 입력 데이터(A)의 2비트 오류검출코드($SEDC_2(A)$)에 상기 제2 입력데이터(B)의 2비트 오류검출코드($SEDC_2(B)$)를 산술합(+)하고, 상기 제1 입력 데이터(A)와 상기 제2 입력데이터(B)의 2비트 논리합 오류검출코

드($SEDC_2(A \text{ OR } B)$)를 산술차(-)하여 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)를 생성하고, 상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 배타적 부정 논리합연산($A \text{ XNOR } B$)일 경우, 상기 제1 입력 데이터(A)의 논리 부정 값($\text{NOT } A$)의 2비트 오류검출코드($SEDC_2(\text{NOT } A)$)에 상기 제2 입력데이터(B)의 2비트 오류검출코드($SEDC_2(\text{NOT } B)$)를 산술합(+)하고, 상기 제1 입력 데이터(A)의 논리 부정값($\text{NOT } A$)과 상기 제2 입력데이터(B)의 2비트 시프트곱 오류검출코드($2 \times SEDC_2((\text{NOT } A) \text{ AND } B)$)를 산술차(-)한 값에, 십진수 '3'을 산술합(+)하여 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)를 생성하고, 상기 불 연산이 상기 제2 입력 데이터(B)의 출력연산(B)일 경우, 상기 제2 입력 데이터(B)의 2비트 오류검출코드($SEDC_2(B)$)를 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)로 생성하고, 상기 불 연산이 상기 제1 입력 데이터(A)의 논리 부정값($\text{NOT } A$)과 상기 제2 입력 데이터(B)의 논리합연산($(\text{NOT } A) \text{ OR } B$)일 경우, 상기 제1 입력 데이터(A)의 논리 부정값($\text{NOT } A$)의 2비트 오류검출코드($SEDC_2(\text{NOT } A)$)와 상기 제2 입력데이터(B)의 2비트 오류검출코드($SEDC_2(B)$)를 산술합(+)한 값에, 상기 제1 입력 데이터(A)의 논리 부정값($\text{NOT } A$)과 상기 제2 입력데이터(B)의 2비트 논리곱 오류검출코드($SEDC_2((\text{NOT } A) \text{ AND } B)$)를 산술차(-)하여 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)를 생성하고, 상기 불 연산이 상기 제1 입력 데이터(A)의 출력연산(A)일 경우, 상기 제2 입력 데이터(A)의 2비트 오류검출코드($SEDC_2(A)$)를 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)로 생성하고, 상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 논리 부정값($\text{NOT } B$)의 논리합연산($A \text{ OR } (\text{NOT } B)$)일 경우, 상기 제1 입력 데이터(A)의 2비트 오류검출코드($SEDC_2(A)$)와 상기 제2 입력데이터(B)의 논리 부정값($\text{NOT } B$)의 2비트 오류검출코드($SEDC_2(\text{NOT } B)$)를 산술합(+)하고, 상기 제1 입력 데이터(A)와 상기 제2 입력데이터(B)의 논리 부정값($\text{NOT } B$)의 2비트 논리곱 오류검출코드($SEDC_2(A \text{ AND } (\text{NOT } B))$)를 산술차(-)하여 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)를 생성하고, 상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 논리합연산($A \text{ OR } B$)일 경우, 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 2비트 논리합 오류검출코드($SEDC_2(A \text{ OR } B)$)를 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)로 생성하거나, 상기 제1 입력 데이터(A)의 2비트 오류검출코드($SEDC_2(A)$)에 상기 제2 입력데이터(B)의 2비트 오류검출코드($SEDC_2(B)$)를 산술합(+)하고, 상기 제1 입력 데이터(A)와 상기 제2 입력데이터(B)의 2비트 논리곱 오류검출코드($SEDC_2(A \text{ AND } B)$)를 산술차(-)하여 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)를 생성하고, 상기 불 연산이 로직 '1'연산(Logic 1)일 경우, 데이터 '11'의 2비트 오류검출코드($SEDC_2(11)$)를 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)로 생성한다.

[0028] 바람직한 실시예에 있어서, 상기 2비트 입력 오류검출코드 생성수단은, 상기 제1 입력 데이터(A)와 제1 제어신호(S_1)의 배타적 논리합 연산을 수행하는 제1 XOR게이트; 이진 데이터 '00'을 출력하는 '00'출력기; 상기 제1 XOR게이트와 상기 '00'출력기의 출력을 입력받고 제3 제어신호(S_3)에 의해 입력되는 신호들 중 어느 하나의 신호를 출력하는 제1 2비트 2×1 멀스(MUX); 상기 제1 2비트 2×1 멀스의 출력을 입력받아 2비트 오류검출코드를 출력하는 제1 2비트 오류검출코드 생성기; 상기 제1 2비트 오류검출코드 생성기의 출력과 제5 제어신호(S_3)를 입력받아 논리곱 연산을 수행하는 제1 AND게이트; 상기 제2 입력 데이터(B)와 제2 제어신호(S_2)의 배타적 논리합 연산을 수행하는 제2 XOR게이트; 이진 데이터 '11'을 출력하는 제1 '11'출력기; 상기 제2 XOR게이트와 상기 '11'출력기의 출력을 입력받고 상기 제3 제어신호(S_3)에 의해 입력되는 신호들 중 어느 하나의 신호를 출력하는 제2 2비트 2×1 멀스; 상기 제2 2비트 2×1 멀스의 출력을 입력받아 2비트 오류검출코드를 출력하는 제2 2비트 오류검출코드 생성기; 상기 제2 2비트 오류검출코드 생성기의 출력과 제6 제어신호(S_6)를 입력받아 논리곱 연산을 수행하는 제2 AND게이트; 이진 데이터 '11'을 출력하는 제2 '11'출력기; 상기 제2 '11'출력기의 출력과 제7 제어신호(S_7)를 입력받아 논리곱 연산을 수행하는 제3 AND게이트; 상기 제1 2비트 2×1 멀스와 상기 제2 2비트 2×1 멀스의 출력을 논리합 연산하는 OR게이트; 상기 제1 2비트 2×1 멀스와 상기 제2 2비트 2×1 멀스의 출력을 논리곱 연산하는 제4 AND게이트; 상기 OR게이트와 상기 제4 AND게이트의 출력을 입력받고 제4 제어신호(S_4)에 의해 입력되는 신호들 중 어느 하나의 신호를 출력하는 제3 2비트 2×1 멀스; 상기 제3 2비트 2×1 멀스의 출력을 입력받아 2비트 오류검출코드를 출력하는 제3 2비트 오류검출코드 생성기; 상기 제3 2비트 오류검출코드 생성기의 출력과 제8 제어신호(S_4)를 논리곱 연산하는 제4 AND게이트; 상기 제4 AND게이트의 출력을 입력받고 상기 제7

제어신호(S_7)에 의해 상기 제4 AND게이트의 출력을 왼쪽 시프트 연산하는 Left Shift게이트; 상기 제1 AND게이트, 상기 제2 AND게이트 및 상기 제3 AND게이트의 출력을 산술합(+)하고 상기 Left Shift게이트의 출력을 산술차(-)하여 상기 입력 오류검출코드를 출력하는 가산기(Block Adder); 및 상기 불 연산의 종류에 따라 상기 제어신호들($S_1, S_2, S_3, S_4, S_5, S_6, S_7, S_8$)을 출력하는 제어신호 생성기;를 포함하는 논리회로로 구성된다.

[0029] 바람직한 실시예에 있어서, 상기 이진 입력 데이터가 1비트 입력 데이터일 경우, 상기 2비트 입력 오류검출코드 생성수단은 상기 1비트 입력 데이터를 2비트 입력 데이터들로 가정하여, 최상위 비트는 '0'으로 설정하고, 최하위 비트는 상기 1비트 입력 데이터로 설정하여 상기 각 불 연산의 경우에 해당하는 오류검출코드의 계산을 수행하고, 계산된 오류검출코드 중, 최하위 비트만을 상기 입력 오류검출코드로 생성하되, 상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 배타적 논리합연산(A XOR B) 또는 배타적 부정 논리합연산(A XNOR B)일 경우에는, 상기 십진수 '3'이 아닌 십진수 '1'을 산술합(+)한 값의 최하위 비트를 상기 입력 오류검출코드로 생성한다.

[0030] 바람직한 실시예에 있어서, 상기 오류 검출기는 상기 이진 입력 데이터가 1비트 입력 데이터일 경우, 상기 입력 오류검출코드의 최하위 비트와 상기 출력 오류검출코드의 최하위 비트를 서로 비교하여 오류검출결과를 출력한다.

[0031] 바람직한 실시예에 있어서, 상기 입력 오류검출코드 생성기:는 상기 이진 입력 데이터들이 3비트 입력 데이터일 경우 2비트인 3비트 입력 오류검출코드(SED_{3-in})를 상기 입력 오류검출코드로 생성하는 3비트 입력 오류검출코드 생성수단;을 더 포함하고, 상기 3비트 입력 오류검출코드 생성수단은 상기 3비트 입력 데이터의 하위 두 비트를 입력으로 하는 상기 2비트 입력 오류검출코드 생성수단의 2비트 입력 오류검출코드(SED_{2-in})를 과도 2비트 입력 오류검출코드로 생성하고, 선택된 불 연산의 종류 및 상기 이진 입력 데이터들 각각의 최상위 비트의 값에 따라 상기 과도 2비트 입력 오류검출코드 자체 또는 상기 과도 2비트 입력 오류검출코드에 '1'을 산술차(-)하여 상기 3비트 입력 오류검출코드(SED_{3-in})를 생성한다.

[0032] 바람직한 실시예에 있어서, 상기 3비트 입력 오류검출코드 생성수단이 상기 과도 2비트 입력 오류검출코드에 '1'을 산술차(-)하여 상기 3비트 입력 오류검출코드(SED_{3-in})를 생성하는 경우는, 상기 불 연산이 상기 제1 입력 데이터(A) 및 상기 제2 입력 데이터(B)의 부정 논리합연산(A NOR B)이고, 상기 제1 입력 데이터(A) 및 상기 제2 입력 데이터(B)의 최상위 비트가 '0'인 경우, 상기 불 연산이 상기 제1 입력 데이터(A)의 논리 부정 값(NOT A)과 상기 제2 입력 데이터(B)의 논리곱연산((NOT A) AND B)이고, 상기 제1 입력 데이터(A)의 최상위 비트가 '0'이고 상기 제2 입력 데이터(B)의 최상위 비트가 '1'인 경우, 상기 불 연산이 상기 제1 입력 데이터(A)의 논리부정연산(NOT A)일 경우, 상기 제2 입력 데이터(B)의 최상위 비트에 관계없이 상기 제1 입력 데이터(A)의 최상위 비트가 '0'인 경우, 상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 논리 부정값(NOT B)의 논리곱연산(A AND (NOT B))이고, 상기 제1 입력 데이터(A)의 최상위 비트가 '1'이고, 상기 제2 입력 데이터(B)의 최상위 비트가 '0'인 경우, 상기 불 연산이 상기 제2 입력 데이터(B)의 논리부정연산(NOT B)이고, 상기 제1 입력 데이터(A)의 최상위 비트에 관계없이 상기 제2 입력 데이터(B)의 최상위 비트가 '0'인 경우, 상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 배타적 논리합연산(A XOR B)이고, 상기 제1 입력 데이터(A)의 최상위 비트가 '1'이고, 상기 제2 입력 데이터(B)의 최상위 비트가 '0'이거나, 상기 제1 입력 데이터(A)의 최상위 비트가 '0'이고, 상기 제2 입력 데이터(B)의 최상위 비트가 '1'인 경우, 상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 부정 논리곱연산(A NAND B)이고, 상기 제1 입력 데이터(A) 및 상기 제2 입력 데이터(B)의 최상위 비트가 '1'이 아닌 경우, 상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 논리곱연산(A AND B)이고, 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 최상위 비트가 '1'인 경우, 상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 배타적 부정 논리합연산(A XNOR B)이고, 상기 제1 입력 데이터(A)의 최상위 비트가 '1'이고, 상기 제2 입력 데이터(B)의 최상위 비트가 '0'이거나, 상기 제1 입력 데이터(A)의 최상위 비트가 '0'이고, 상기 제2 입력 데이터(B)의 최상위 비트가 '1'인 경우, 상기 불 연산이 상기 제2 입력 데이터(B)의 출력연산(B)이고, 상기 제1 입력 데이터(A)의 최상위 비트에 관계없이 상기 제2 입력 데이터(B)의 최상위 비트가 '1'인 경우, 상기 불 연산이 상기 제1 입력 데이터(A)의 논리 부정값(NOT A)과 상기 제2 입력 데이터(B)의 논리합연산((NOT A) OR B)이고, 상기 제1 입력 데이터(A)의 최상위 비트가 '1'이고 상기 제2 입력 데이터(B)의 최상위 비트가 '0'이 아닌 경우, 상기 불 연산이 상기 제1 입력 데이터(A)의 출력연산(A)이고, 상기 제2 입력 데이터(B)의 최상위 비트에 관계없이 상기 제1 입력 데이터(A)의 최상위 비트가 '1'인 경우, 상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 논리 부정값(NOT B)의 논리합연산(A OR (NOT B))이고, 상기 제1 입력 데이터(A)의 최상위 비트가 '0'이고

상기 제2 입력 데이터(B)의 최상위 비트가 '1'이 아닌 경우, 상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 논리합연산(A OR B)이고, 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 최상위 비트가 '0'이 아닌 경우 또는 상기 제1 입력 데이터(A) 및 상기 제2 입력 데이터(B)의 최상위 비트에 관계없이 상기 불 연산이 로직'1'(Logic 1)연산일 경우이다.

[0033] 바람직한 실시예에 있어서, 상기 입력 오류검출코드 생성기:는 상기 이진 입력 데이터들이 4비트 입력 데이터일 경우, 3비트인 4비트 입력 오류검출코드($SEDC_{4-in}$)를 상기 입력 오류검출코드로 생성하는 4비트 입력 오류검출코드 생성수단;을 더 포함하고, 상기 4비트 입력 오류검출코드 생성수단은 상기 4비트 입력 데이터의 하위 세 비트를 입력으로 하는 상기 3비트 입력 오류검출코드 생성수단의 3비트 입력 오류검출코드($SEDC_{3-in}$)를 상기 4비트 입력 오류검출코드($SEDC_{4-in}$)의 하위 두 비트로 생성하고, 상기 4비트 입력 오류검출코드($SEDC_{4-in}$)의 최상위 비트는 선택된 불 연산의 종류 및 상기 이진 입력 데이터들 각각의 최상위 비트의 값에 따라 '1' 또는 '0' 값으로 생성한다.

[0034] 바람직한 실시예에 있어서, 상기 4비트 입력 오류검출코드($SEDC_{4-in}$)의 최상위 비트가 '1'로 생성되는 경우는, 상기 3비트 입력 오류검출코드 생성수단이 상기 과도 2비트 입력 오류검출코드에 '1'을 산술차(-)하는 경우들 이외의 경우이다.

[0035] 바람직한 실시예에 있어서, 상기 이진 입력 데이터들이 n비트 입력 데이터일 경우 상기 n비트 입력 데이터에 대한 오류검출코드인 n비트 입력 오류검출코드($SEDC_{n-in}$)를 상기 입력 오류검출코드로 생성하는 n비트 입력 오류검출코드 생성수단;을 더 포함하고, 상기 n비트 입력 오류검출코드 생성수단은 상기 입력 오류검출코드 생성수단들 각각의 오류검출코드($SEDC_{2-in}$, $SEDC_{3-in}$, $SEDC_{4-in}$)나, 상기 입력 오류검출코드 생성수단들 중 어느 하나의 입력 오류검출코드 생성수단의 오류검출코드와 하나 또는 복수 개의 3비트 입력 오류검출코드($SEDC_{3-in}$)의 조합을 상기 입력 오류검출코드로 생성한다.

[0036] 바람직한 실시예에 있어서, 상기 오류 검출기:는 상기 이진 출력 데이터가 2비트 출력 데이터일 경우, 상기 2비트 출력 데이터에 대한 2비트의 오류검출코드인 2비트 출력 오류검출코드($SEDC_{2-out}$)를 상기 출력 오류검출코드로 생성하는 2비트 출력 오류검출코드 생성수단; 및 상기 입력 오류검출코드와 상기 출력 오류검출코드의 동일성을 비교하여 동일하지 않을 경우 오류 발생신호를 출력하는 코드비교수단;을 포함한다.

[0037] 바람직한 실시예에 있어서, 상기 2비트 출력 오류검출코드 생성수단은, 상기 2비트 출력 오류검출코드들($SEDC_{2-out}$)의 최상위 비트는 2비트 데이터 '00', '01', '10' 또는 '11'에 대해 비트들의 '0'의 개수가 '1'의 개수보다 크거나 같을 경우 '1'로 생성하고, 그 외의 경우에는 '0'으로 생성하며, 최하위 비트는 '0'의 개수와 '1'의 개수가 동일할 경우 '0'으로 생성하고, 그 외의 경우에는 '1'로 생성하거나, 최상위 비트는 '0'의 개수와 '1'의 개수가 동일할 경우 '0'으로 생성되고, 그 외의 경우에는 '1'로 생성하며, 최하위 비트는 '0'의 개수가 '1'의 개수보다 크거나 같을 경우 '1'로 생성하고, 그 외의 경우에는 '0'으로 생성하되, 상기 2비트 출력 오류검출코드($SEDC_{2-out}$)는 동일한 입력에 대해 상기 2비트 오류검출코드($SEDC_2$)와 동일한 오류검출코드로 생성된다.

[0038] 바람직한 실시예에 있어서, 상기 오류 검출기는, 상기 이진 출력 데이터가 3비트 출력 데이터일 경우, 상기 3비트 출력 데이터에 대한 2비트의 오류검출코드인 3비트 출력 오류검출코드($SEDC_{3-out}$)를 상기 출력 오류검출코드로 생성하는 3비트 출력 오류검출코드 생성수단;을 더 포함한다.

[0039] 바람직한 실시예에 있어서, 상기 3비트 출력 오류검출코드 생성수단은 상기 3비트 출력 데이터의 최상위 비트가 '0'일 경우에는, 중간비트 및 최하위 비트를 입력으로 하는 상기 2비트 출력 오류검출코드 생성수단($SEDC_{2-out}$)의 2비트 출력 오류검출코드와 동일하게 상기 3비트 출력 오류검출코드($SEDC_{3-out}$)를 생성하고, 상기 3비트 출력 데이터의 최상위 비트가 '1'일 경우에는 중간비트 및 최하위 비트의 논리 부정값을 입력으로 하는 상기 2비트 출력 오류검출코드 생성수단의 2비트 출력 오류검출코드($SEDC_{2-out}$)의 논리 부정값을 상기 3비트 출력 오류검출코드($SEDC_{3-out}$)로 생성한다.

[0040] 바람직한 실시예에 있어서, 상기 오류 검출기는, 상기 이진 출력 데이터가 4비트 출력 데이터일 경우, 상기 4비트 출력 데이터에 대한 3비트의 오류검출코드인 4비트 출력 오류검출코드($SEDC_{4-out}$)를 상기 출력 오류검출코드로 생성하는 4비트 출력 오류검출코드 생성수단;을 더 포함한다.

- [0041] 바람직한 실시예에 있어서, 상기 4비트 출력 오류검출코드 생성수단은 상기 4비트 출력 오류검출코드($SEDC_{4-out}$)의 하위 2비트는 상기 4비트 출력 데이터 중, 최상위 비트를 제외한 3비트를 입력으로 하는 상기 3비트 출력 오류검출코드 생성수단의 3비트 출력 오류검출코드($SEDC_{3-out}$)로 생성하고, 상기 4비트 출력 오류검출코드($SEDC_{4-out}$)의 최상위 비트는 상기 4비트 출력 데이터의 최상위 비트의 논리 부정 값으로 생성한다.
- [0042] 바람직한 실시예에 있어서, 상기 오류 검출기는, 상기 이진 출력 데이터가 n비트 출력 데이터일 경우, 상기 n비트 출력 데이터에 대한 오류검출코드인 n비트 출력 오류검출코드($SEDC_{n-out}$)를 상기 출력 오류검출코드로 생성하는 n비트 출력 오류검출코드 생성수단을 더 포함하고, 상기 n비트 출력 오류검출코드 생성수단은 상기 출력 오류검출코드 생성수단들 각각의 오류검출코드($SEDC_{2-out}, SEDC_{3-out}, SEDC_{4-out}$)나, 상기 출력 오류검출코드 생성수단들 중 어느 하나의 출력 오류검출코드 생성수단의 오류검출코드와 하나 또는 복수 개의 3비트 출력 오류검출코드($SEDC_{3-out}$)의 조합을 상기 출력 오류검출코드로 생성한다.
- [0043] 또한, 본 발명은 상기 오류 검출 장치와 프로그래머블 불 연산 유닛을 포함하는 자가검사 프로그래머블 불 연산 유닛(Self-checking Programmable Boolean Operations Unit)을 더 제공한다.
- [0044] 또한, 본 발명은 상기 오류 검출 장치 및 상기 프로그래머블 불 연산 유닛으로 기능하는 프로그래머블 산술/논리 연산유닛을 포함하는 자가검사 프로그래머블 산술/논리 연산유닛(Self-checking Programmable Arithmetic and Logic Unit)을 더 제공한다.
- [0045] 또한, 본 발명은 컴퓨터를 상기 오류 검출 장치로 기능하게 하는 오류 검출 프로그램이 저장된 컴퓨터로 읽을 수 있는 기록매체를 더 제공한다.
- [0046] 또한, 본 발명은 상기 오류 검출 프로그램이 저장되고 통신망을 통해 상기 확장형 오류 검출 프로그램을 전송할 수 있는 서버 시스템을 더 제공한다.

발명의 효과

- [0047] 본 발명은 다음과 같은 우수한 효과를 가진다.
- [0048] 먼저, 본 발명의 오류 검출 장치 및 자가 검사 프로그래머블 불 유닛에 의하면 입력 오류검출코드 생성기 및 오류 검출기가 각각 2비트, 3비트, 4비트 및 n비트의 입력에 대해 선형적이고 유동적인 확장형 오류검출코드를 생성하여 서로 불 연산의 결합을 검출할 수 있는 장점이 있다.
- [0049] 또한, 본 발명의 오류 검출 장치 및 자가 검사 프로그래머블 불 유닛에 의하면, 오류검출코드의 생성이 입력 데이터에 대한 불 연산과는 별개로 이루어지므로 입력 데이터의 불 연산의 속도나 레이턴시와 같은 성능의 하락 없이 결합 허용 가능 시스템을 구현할 수 있는 장점이 있다.

도면의 간단한 설명

- [0050] 도 1은 본 발명의 일 실시예에 따른 오류 검출 장치 및 자가검사 프로그래머블 불 연산 유닛을 설명하기 위한 도면,
- 도 2는 본 발명의 일 실시예에 따른 오류 검출 장치 및 자가검사 프로그래머블 불 연산 유닛의 2비트 입력 오류검출코드 진리 테이블을 보여주는 도면,
- 도 3은 본 발명의 일 실시예에 따른 오류 검출 장치 및 자가검사 프로그래머블 불 연산 유닛의 2비트 입력 오류검출코드를 생성하는 논리회로의 일례를 보여주는 도면,
- 도 4는 본 발명의 일 실시예에 따른 오류 검출 장치 및 자가검사 프로그래머블 불 연산 유닛의 2비트 입력 오류검출코드를 생성하는 논리회로의 다른 예를 보여주는 도면,
- 도 5는 본 발명의 일 실시예에 따른 오류 검출 장치 및 자가검사 프로그래머블 불 연산 유닛의 2비트 입력 기본 오류검출코드 진리테이블을 보여주는 도면,

도 6은 본 발명의 일 실시예에 따른 오류 검출 장치 및 자가검사 프로그래머블 불 연산 유닛의 불 연산 종류에 따른 2비트 입력 오류검출코드의 생성 로직을 설명하기 위한 도면,

도 7은 본 발명의 일 실시예에 따른 오류 검출 장치 및 자가검사 프로그래머블 불 연산 유닛의 불 연산 종류에 따른 1비트 입력 오류검출코드의 생성 로직을 설명하기 위한 도면,

도 8은 도 6의 생성 로직을 구현하는 논리회로를 보여주는 도면,

도 9는 도 8의 논리회로의 제어신호들을 설명하기 위한 도면,

도 10은 본 발명의 일 실시예에 따른 오류 검출 장치 및 자가검사 프로그래머블 불 연산 유닛의 불 연산 종류에 따른 3비트 입력 오류검출코드의 생성 로직을 설명하기 위한 도면,

도 11은 본 발명의 일 실시예에 따른 오류 검출 장치 및 자가검사 프로그래머블 불 연산 유닛의 불 연산 종류에 따른 4비트 입력 오류검출코드의 생성 로직을 설명하기 위한 도면,

도 12는 본 발명의 일 실시예에 따른 오류 검출 장치 및 자가검사 프로그래머블 불 연산 유닛의 2비트 출력 오류검출코드 진리 테이블을 보여주는 도면,

도 13은 본 발명의 일 실시예에 따른 오류 검출 장치 및 자가검사 프로그래머블 불 연산 유닛의 3비트 출력 오류검출코드 진리 테이블을 보여주는 도면,

도 14는 본 발명의 일 실시예에 따른 오류 검출 장치 및 자가검사 프로그래머블 불 연산 유닛의 3비트 출력 오류검출코드를 생성하기 위한 논리회로를 보여주는 도면,

도 15는 본 발명의 일 실시예에 따른 오류 검출 장치 및 자가검사 프로그래머블 불 연산 유닛의 4비트 출력 오류검출코드 진리 테이블을 보여주는 도면,

도 16은 본 발명의 일 실시예에 따른 오류 검출 장치 및 자가검사 프로그래머블 불 연산 유닛의 4비트 출력 오류검출코드를 생성하기 위한 논리회로를 보여주는 도면이다.

발명을 실시하기 위한 구체적인 내용

[0051] 본 발명에서 사용되는 용어는 가능한 현재 널리 사용되는 일반적인 용어를 선택하였으나, 특정한 경우는 출원인이 임의로 선정한 용어도 있는데 이 경우에는 단순한 용어의 명칭이 아닌 발명의 상세한 설명 부분에 기재되거나 사용된 의미를 고려하여 그 의미가 파악되어야 할 것이다.

[0052] 이하, 첨부한 도면에 도시된 바람직한 실시예들을 참조하여 본 발명의 기술적 구성을 상세하게 설명한다.

[0053] 그러나, 본 발명은 여기서 설명되는 실시예에 한정되지 않고 다른 형태로 구체화 될 수도 있다. 명세서 전체에 걸쳐 동일한 참조번호는 동일한 구성요소를 나타낸다.

[0054] 본 발명의 일 실시예에 따른 오류 검출 장치(100)는 불 연산(Boolean Operations)을 수행하는 프로그래머블 불 연산 유닛(Programmable Boolean Operations Unit, 10)의 입력 데이터(A,B)의 불 연산 결과에 대한 오류검출코드와 상기 프로그래머블 불 연산 유닛(10)의 출력 데이터(F)에 대한 오류검출코드를 각각 생성하여 서로 동일성을 비교함으로써 상기 프로그래머블 불 연산 유닛(10)에 결함이 있는지 여부를 판단한다.

[0055] 또한, 상기 오류 검출 장치(100)는 상기 프로그래머블 불 연산 유닛(10)과 함께 하나의 자가검사 프로그래머블 불 연산 유닛(Self-checking Programmable Boolean Operations Unit)으로 구성되어 제공될 수 있다.

[0056] 또한, 상기 프로그래머블 불 연산 유닛(10)은 불 연산을 수행하는 프로그래머블 산술/논리 연산유닛(Programmable Arithmetic and Logic Unit)으로 대체될 수 있으며, 이 경우, 상기 오류 검출 장치(100)는 상기 프로그래머블 산술/논리 연산유닛과 함께 하나의 자가검사 프로그래머블 산술/논리 연산유닛(Self-checking Programmable Arithmetic and Logic Unit)으로 구성되어 제공될 수 있다.

[0057] 또한, 상기 오류 검출 장치는 실질적으로 오류 검출 프로그램에 의해 기능하는 컴퓨터일 수 있고, 상기 오류 검출 프로그램은 컴퓨터로 읽을 수 있는 기록매체나 통신망을 통해 통신가능한 서버 시스템에 의해 상기 컴퓨터에 전송되고, 상기 컴퓨터는 상기 오류 검출 프로그램에 기능하여 불 연산에 대한 오류 검출을 수행할 수 있다.

- [0058] 이하에서는 본 발명의 일 실시예에 따른 오류 검출 장치(100)의 구성을 상세히 설명하기로 한다.
- [0059] 도 1을 참조하면, 본 발명의 일 실시예에 따른 오류 검출 장치(100)는 입력 오류검출코드 생성기(110) 및 오류 검출기(120)를 포함하여 이루어진다.
- [0060] 또한, 상기 입력 오류검출코드 생성기(110)는 상기 프로그래머블 불 연산 유닛(10)의 입력 데이터들(A,B)을 입력받아 선택된 불 연산의 결과에 대한 오류검출코드인 입력 오류검출코드를 생성하고, 상기 오류 검출기(120)는 상기 프로그래머블 불 연산 유닛(10)의 출력 데이터(F)를 입력받아 출력 데이터에 대한 오류검출코드인 출력 오류검출코드를 생성하며, 상기 입력 오류검출코드와 상기 출력 오류검출코드의 동일성을 판단하여 오류검출결과(Fault Detection Result)를 출력한다.
- [0061] 즉, 본 발명의 일 실시예에 따른 오류 검출 장치(100)는 상기 프로그래머블 불 연산 유닛(10)의 연산에 대해서 관여하지 않으므로 결함 허용 가능한 구조(fault-tolerant reconfigurable architecture)이다.
- [0062] 또한, 상기 입력 오류검출코드 생성기(110)와 상기 오류 검출기(120)는 상기 입력 데이터들(A,B)이나 상기 출력 데이터(F)의 길이에 선형적으로 가변하는 확장형 오류검출코드(SED_C:scalable error detection coding)를 생성한다.
- [0063] 상기 입력 오류검출코드 생성기(100)는 2비트 오류검출코드 진리테이블(truth table, 111) 및 2비트 입력 오류검출코드 생성수단을 포함하여 이루어진다.
- [0064] 또한, 도 2를 참조하면, 상기 2비트 오류검출코드 진리테이블(111)은 2비트 데이터(A, 111a)에 대응하는 2비트의 오류검출코드인 2비트 오류검출코드들(SED_{C2}, 111b, 111c)이 계산되어 저장된다.
- [0065] 여기서, 상기 2비트 데이터(111a)는 상기 프로그래머블 불 연산 유닛의 입력 데이터들(A,B)과 각각 대응하는 데이터로써, 상기 2비트 오류검출코드들(SED_{C2})은 상기 프로그래머블 불 연산 유닛의 입력 데이터들(A,B)의 불 연산 결과에 대해 생성되는 2비트 입력 오류검출코드(SED_{C2-in})와는 서로 구분된다.
- [0066] 또한, 상기 2비트 오류검출코드들(SED_{C2})은 미리 계산되어 저장될 수 있고, 도 3 및 도 4에 도시한 논리회로에 의해 오류검출시 실시간으로 계산될 수 있다.
- [0067] 또한, 상기 2비트 오류검출코드들(SED_{C2})은 서로 다른 두 가지 케이스(Case1, Case2)로 계산될 수 있다.
- [0068] 또한, 상기 2비트 오류검출코드(SED_{C2})의 첫 번째 케이스(Case1, 111b)는 아래의 수학적식1에 의해 계산된다.

수학적식 1

$$C_1 = \begin{cases} 1, & \text{if number of 0's in } A \geq \text{number of 1's in } A \\ 0, & \text{otherwise} \end{cases}$$

$$C_0 = \begin{cases} 0, & \text{if number of 0's in } A = \text{number of 1's in } A \\ 1, & \text{otherwise} \end{cases}$$

[0069]

- [0070] 즉, 첫 번째 케이스(111b)의 경우, 2비트 오류검출코드(SED_{C2})의 최상위 비트(C₁)는 상기 2비트 데이터(111a) '00', '01', '10' 또는 '11'에 대해 비트들의 '0'의 개수가 '1'의 개수보다 크거나 같을 경우 '1'로 계산되어 저장되고, 그 외의 경우에는 '0'으로 계산되어 저장되며, 최하위 비트(C₀)는 '0'의 개수와 '1'의 개수가 동일할 경우 '0'으로 계산되어 저장되고, 그 외의 경우에는 '1'로 계산되어 저장된다.

- [0071] 또한, 상기 첫 번째 케이스(111b)는 도 3에 도시한 제1 2비트 오류검출코드 생성 논리회로(111-1)에 의해 생성되며, 상기 제1 2비트 오류검출코드 생성 논리회로(111-1)은 2비트 입력에 대해 exclusive-NOT-OR 연산을 수행하는 하나의 XNOR게이트(111-1a)와 2비트 입력에 대해 NOT-AND 연산을 수행하는 하나의 NAND게이트(111-1b)를 이용하여 구현하였다.

- [0072] 그러나, 상기 제1 2비트 오류검출코드 생성 논리회로(111-1)의 구성은 예시일 뿐, 다양한 논리 게이트의 조합으로 구현할 수 있음은 물론이다.
- [0073] 또한, 상기 2비트 오류검출코드($SEDC_2$)의 두 번째 케이스(Case2, 111c)는 첫 번째 케이스(111b)와 비교하여 상위 비트와 하위 비트가 서로 역전된 코드이다.
- [0074] 즉, 상기 두 번째 케이스(111c)의 경우, 2비트 오류검출코드($SEDC_2$)의 최상위 비트(C_1)는 최상위 비트는 '0'의 개수와 '1'의 개수가 동일할 경우 '0'으로 계산되어 저장되고, 그 외의 경우에는 '1'로 계산되어 저장되고, 최하위 비트(C_0)는 '0'의 개수가 '1'의 개수보다 크거나 같을 경우 '1'로 계산되어 저장되고, 그 외의 경우에는 '0'으로 계산되어 저장된다.
- [0075] 또한, 상기 두 번째 케이스(111c)의 경우, 도 4에 도시한 제2 2비트 오류검출코드 생성 논리회로(111-1)에 의해 생성되며, 제1 2비트 오류검출코드 생성 논리회로(111-1)와 구성은 서로 동일하나, XNOR게이트(111-2a)의 출력이 상기 2비트 오류검출코드($SEDC_2$)의 상위 비트로 생성되고, NAND게이트(111-2b)의 출력이 상기 2비트 오류검출코드($SEDC_2$)의 하위 비트로 생성된다.
- [0076] 또한, 상기 제2 2비트 오류검출코드 생성 논리회로(111-2) 역시, 다양한 논리 게이트의 조합으로 구현될 수 있다.
- [0077] 도 4를 참조하면, 상기 입력 오류검출코드 생성기(110)는 상기 2비트 오류검출코드들($SEDC_2$)의 기본 연산에 대한 오류검출코드인 2비트 기본 오류검출코드($SEDC_{2-basic}$)를 미리 계산하여 저장하는 2비트 기본 오류검출코드 진리테이블(112)을 더 포함한다.
- [0078] 또한, 상기 2비트 기본 오류검출코드 진리테이블(112)은 상기 이진 입력 데이터(A,B)가 두 개의 입력 데이터일 경우에, 제1 입력 데이터(A)와 제2 입력 데이터(B)가 조합될 수 있는 경우(112a, 112b)에 대해 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 논리곱 연산 결과에 대응하는 2비트 논리곱 오류검출코드($SEDC_2(A \text{ AND } B)$, 112c), 논리합 연산 결과에 대응하는 2비트 논리합 오류검출코드($SEDC_2(A \text{ OR } B)$, 112d) 및 시프트곱 연산 결과에 대응하는 2비트 시프트곱 오류검출코드($2 \times SEDC_2(A \text{ AND } B)$, 112e)가 미리 저장된다.
- [0079] 즉, 상기 입력 오류검출코드 생성시 상기 2비트 기본 오류검출코드들($SEDC_{2-basic}$)을 재구성하여 생성하게 함으로써 오류검출코드의 연산시간을 줄일 수 있다.
- [0080] 그러나, 상기 2비트 기본 오류검출코드($SEDC_{2-basic}$)는 논리곱, 논리합 및 시프트곱 연산 이외에 배타적 논리합 연산의 결과에 대응하는 2비트 배타적 논리합 오류검출코드($SEDC_2(A \text{ XOR } B)$, 112f)와 같은 다양한 기본 연산에 대응하는 오류검출코드들이 미리 계산되어 저장될 수 있다.
- [0081] 도 6은 상기 2비트 입력 오류검출코드 생성수단이 2비트 입력 오류검출코드($SEDC_{2-in}$)($C_{1(2bit)}$, $C_{0(2bit)}$)를 생성하는 로직(113)을 설명하기 위한 것으로, 상기 2비트 입력 오류검출코드 생성수단은 상기 제1 입력 데이터(A) 및 상기 제2 입력 데이터(B)를 입력받고 불 연산의 종류(opBO)에 따라 2비트 입력 오류검출코드($SEDC_{2-in}$)를 생성한다.
- [0082] 또한, 상기 불 연산은 열여섯 개의 불 연산으로 이루어지며, 상기 2비트 오류검출코드들($SEDC_2$)이나 상기 2비트 기본 오류검출코드들($SEDC_{2-basic}$) 각 단독 또는 산술 조합에 의해 생성된다.
- [0083] 또한, 이하에서 설명하는 각 불 연산의 종류에 따른 2비트 입력 오류검출코드($SEDC_{2-in}$)의 생성 로직(113)은 본 발명의 실시예에 따른 예일 뿐, 상기 2비트 오류검출코드들($SEDC_2$)이나 상기 2비트 기본 오류검출코드들($SEDC_{2-basic}$) 각 단독 또는 산술 조합으로 선택된 불 연산의 연산결과에 대응하는 2비트 오류검출코드($SEDC_2$)를 생성할 수 있는 로직이라면 어떠한 로직도 가능하다.
- [0084] 먼저, 상기 불 연산이 로직'0'연산(Logic 0)일 경우, 데이터 '00'의 2비트 오류검출코드($SEDC_2(00)$)를 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)로 생성한다.

- [0085] 이 경우는 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)의 생성이 상기 입력 데이터들(A,B)의 값과 관계가 없으므로 상기 2비트 오류검출코드($SEDC_2$)를 단독으로 이용하여 계산할 수 있다.
- [0086] 다음, 상기 불 연산이 상기 제1 입력 데이터(A) 및 상기 제2 입력 데이터(B)의 부정 논리합연산($A \text{ NOR } B$)일 경우, 상기 제1 입력 데이터(A)의 논리 부정 값($\text{NOT } A$)의 2비트 오류검출코드($SEDC_2(\text{NOT } A)$)에 상기 제2 입력데이터(B)의 논리 부정 값($\text{NOT } B$)의 2비트 오류검출코드($SEDC_2(\text{NOT } B)$)를 산술합(+)하고, 상기 제1 입력 데이터(A)의 논리 부정 값($\text{NOT } A$)과 상기 제2 입력데이터(B)의 논리 부정 값($\text{NOT } B$)의 2비트 논리합 오류검출코드($SEDC_2((\text{NOT } A) \text{ OR } (\text{NOT } B)))$)를 산술차(-)하여 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)를 생성한다.
- [0087] 여기서, 산술합(+)과 산술차(-)는 비트합 및 비트차로도 표현되며 논리합과는 구분된다.
- [0088] 다음, 상기 불 연산이 상기 제1 입력 데이터(A)의 논리 부정 값($\text{NOT } A$)과 상기 제2 입력 데이터(B)의 논리곱연산($(\text{NOT } A) \text{ AND } B$)일 경우, 상기 제1 입력 데이터(A)의 논리 부정 값($\text{NOT } A$)의 2비트 오류검출코드($SEDC_2(\text{NOT } A)$)에 상기 제2 입력데이터(B)의 2비트 오류검출코드($SEDC_2(B)$)를 산술합(+)하고, 상기 제1 입력 데이터(A)의 논리 부정 값($\text{NOT } A$)과 상기 제2 입력데이터(B)의 2비트 논리합 오류검출코드($SEDC_2((\text{NOT } A) \text{ OR } B))$)를 산술차(-)하여 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)를 생성한다.
- [0089] 다음, 상기 불 연산이 상기 제1 입력 데이터(A)의 논리부정연산($\text{NOT } A$)일 경우, 상기 제1 입력 데이터(A)의 논리 부정 값($\text{NOT } A$)의 2비트 오류검출코드($SEDC_2(\text{NOT } A)$)를 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)로 생성한다.
- [0090] 이 경우에도 하나의 입력 데이터에 대해 오류검출코드를 생성하므로 상기 2비트 오류검출코드($SEDC_2$)를 단독으로 이용하여 계산할 수 있다.
- [0091] 다음, 상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 논리 부정값($\text{NOT } B$)의 논리곱연산($A \text{ AND } (\text{NOT } B)$)일 경우, 상기 제1 입력 데이터(A)의 2비트 오류검출코드($SEDC_2(A)$)에 상기 제2 입력데이터(B)의 논리 부정값($\text{NOT } B$)의 2비트 오류검출코드($SEDC_2(\text{NOT } B)$)를 산술합(+)하고, 상기 제1 입력 데이터(A)와 상기 제2 입력데이터(B)의 논리 부정값($\text{NOT } B$)의 2비트 논리합 오류검출코드($SEDC_2(A \text{ OR } (\text{NOT } B)))$)를 산술차(-)하여 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)를 생성한다.
- [0092] 다음, 상기 불 연산이 상기 제2 입력 데이터(B)의 논리부정연산($\text{NOT } B$)일 경우, 상기 제2 입력 데이터(B)의 논리 부정 값($\text{NOT } B$)의 2비트 오류검출코드($SEDC_2(\text{NOT } B)$)를 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)로 생성한다.
- [0093] 다음, 상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 배타적 논리합연산($A \text{ XOR } B$)일 경우, 상기 제1 입력 데이터(A)의 2비트 오류검출코드($SEDC_2(A)$)에 상기 제2 입력데이터(B)의 논리 부정값($\text{NOT } B$)의 2비트 오류검출코드($SEDC_2(\text{NOT } B)$)를 산술합(+)하고, 상기 제1 입력 데이터(A)와 상기 제2 입력데이터(B)의 2비트 시프트곱 오류검출코드($2 \times SEDC_2(A \text{ AND } B)$)를 산술차(-)한 값에, 십진수 '3'을 산술합(+)하여 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)로 생성한다.
- [0094] 다음, 상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 부정 논리곱연산($A \text{ NAND } B$)일 경우, 상기 제1 입력 데이터(A)의 논리 부정 값($\text{NOT } A$)의 2비트 오류검출코드($SEDC_2(\text{NOT } A)$)에 상기 제2 입력데이터(B)의 논리 부정 값($\text{NOT } B$)의 2비트 오류검출코드($SEDC_2(\text{NOT } B)$)를 산술합(+)하고, 상기 제1 입력 데이터(A)의 논리 부정 값($\text{NOT } A$)과 상기 제2 입력데이터(B)의 논리 부정 값($\text{NOT } B$)의 2비트 논리곱 오류검출코드($SEDC_{2-in}((\text{NOT } A) \text{ AND } (\text{NOT } B)))$)를 산술차(-)하여 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)를 생성한다.
- [0095] 다음, 상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 논리곱연산($A \text{ AND } B$)일 경우, 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 2비트 논리곱 오류검출코드($SEDC_2(A \text{ AND } B)$)를 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)로 생성한다.
- [0096] 이 경우 상기 2비트 논리곱 오류검출코드($SEDC_2(A \text{ AND } B)$)를 단독으로 이용하여 상기 2비트 입력 오류검출코드

($SEDC_{2-in}$)를 생성할 수 있다.

- [0097] 그러나, 상기 제1 입력 데이터(A)의 2비트 오류검출코드($SEDC_2(A)$)에 상기 제2 입력데이터(B)의 2비트 오류검출코드($SEDC_2(B)$)를 산술합(+)하고, 상기 제1 입력 데이터(A)와 상기 제2 입력데이터(B)의 2비트 논리합 오류검출코드($SEDC_2(A \text{ OR } B)$)를 산술차(-)하여 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 논리곱연산(A AND B)에 대한 2비트 입력 오류검출코드($SEDC_{2-in}$)를 생성할 수 있다.
- [0098] 다음, 상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 배타적 부정 논리합연산(A XNOR B)일 경우, 상기 제1 입력 데이터(A)의 논리 부정 값(NOT A)의 2비트 오류검출코드($SEDC_2(\text{NOT } A)$)에 상기 제2 입력데이터(B)의 2비트 오류검출코드($SEDC_2(\text{NOT } B)$)를 산술합(+)하고, 상기 제1 입력 데이터(A)의 논리 부정값(NOT A)과 상기 제2 입력데이터(B)의 2비트 시프트곱 오류검출코드($2 \times SEDC_2((\text{NOT } A) \text{ AND } B)$)를 산술차(-)한 값에, 십진수 '3'을 산술합(+)하여 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)를 생성한다.
- [0099] 다음, 상기 불 연산이 상기 제2 입력 데이터(B)의 출력연산(B)일 경우, 상기 제2 입력 데이터(B)의 2비트 오류검출코드($SEDC_2(B)$)를 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)로 생성한다.
- [0100] 다음, 상기 불 연산이 상기 제1 입력 데이터(A)의 논리 부정값(NOT A)과 상기 제2 입력 데이터(B)의 논리합연산((NOT A) OR B)일 경우, 상기 제1 입력 데이터(A)의 논리 부정값(NOT A)의 2비트 오류검출코드($SEDC_2(\text{NOT } A)$)와 상기 제2 입력데이터(B)의 2비트 오류검출코드($SEDC_2(B)$)를 산술합(+)한 값에, 상기 제1 입력 데이터(A)의 논리 부정값(NOT A)과 상기 제2 입력데이터(B)의 2비트 논리곱 오류검출코드($SEDC_2((\text{NOT } A) \text{ AND } B)$)를 산술차(-)하여 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)를 생성한다.
- [0101] 다음, 상기 불 연산이 상기 제1 입력 데이터(A)의 출력연산(A)일 경우, 상기 제2 입력 데이터(A)의 2비트 오류검출코드($SEDC_2(A)$)를 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)로 생성한다.
- [0102] 다음, 상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 논리 부정값(NOT B)의 논리합연산(A OR (NOT B))일 경우, 상기 제1 입력 데이터(A)의 2비트 오류검출코드($SEDC_2(A)$)와 상기 제2 입력데이터(B)의 논리 부정값(NOT B)의 2비트 오류검출코드($SEDC_2(\text{NOT } B)$)를 산술합(+)하고, 상기 제1 입력 데이터(A)와 상기 제2 입력데이터(B)의 논리 부정값(NOT B)의 2비트 논리곱 오류검출코드($SEDC_2(A \text{ AND } (\text{NOT } B))$)를 산술차(-)하여 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)를 생성한다.
- [0103] 다음, 상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 논리합연산(A OR B)일 경우, 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 2비트 논리합 오류검출코드($SEDC_2(A \text{ OR } B)$)를 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)로 생성한다.
- [0104] 이 경우, 상기 2비트 논리합 오류검출코드($SEDC_2(A \text{ OR } B)$)를 단독으로 이용하여 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)로 생성할 수 있다.
- [0105] 그러나, 상기 제1 입력 데이터(A)의 2비트 오류검출코드($SEDC_2(A)$)에 상기 제2 입력데이터(B)의 2비트 오류검출코드($SEDC_2(B)$)를 산술합(+)하고, 상기 제1 입력 데이터(A)와 상기 제2 입력데이터(B)의 2비트 논리곱 오류검출코드($SEDC_2(A \text{ AND } B)$)를 산술차(-)하여 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 논리합연산(A OR B)에 대한 2비트 입력 오류검출코드($SEDC_{2-in}$)를 생성할 수 있다.
- [0106] 다음, 상기 불 연산이 로직'1'연산(Logic 1)일 경우, 데이터 '11'의 2비트 오류검출코드($SEDC_2(11)$)를 상기 2비트 입력 오류검출코드($SEDC_{2-in}$)로 생성한다.
- [0107] 도 7은 상기 입력 데이터들(A,B)이 1비트 입력 데이터일 경우, 1비트의 입력 오류검출코드인 1비트 입력 오류검출코드($SEDC_{1-in}(((C_{1(1bit)}), C_{0(1bit)}))$)를 생성하는 로직(114)을 보여주는 도면이다.

- [0108] 또한, 상기 1비트 입력 오류검출코드($SEDC_{1-in}$)는 상기 2비트 입력 오류검출코드 생성수단에 의해 생성될 수 있으며, 도 6에서 도시한 2비트 입력 오류검출코드($SEDC_{2-in}$) 생성 로직에서 상기 1비트 입력 데이터를 2비트 입력 데이터로 가정하여 계산된다.
- [0109] 더욱 자세하게는 가정된 2비트 입력 데이터의 최상위 비트는 '0'으로 가정하고, 최하위 비트는 상기 1비트 입력 데이터로 설정하여 상기 1비트 입력 데이터가 마치 2비트 입력 데이터인 것으로 가정하여 2비트 입력 오류검출코드($SEDC_{2-in}$)를 계산한다.
- [0110] 다만, 상기 1비트 입력 오류검출코드($SEDC_{1-in}$)를 생성 로직(114)은 상기 2비트 입력 오류검출코드($SEDC_{2-in}$) 생성 로직(113)과 비교하여 입력 데이터들의 배타적 논리합 연산($A \oplus B$) 및 배타적 부정 논리합연산($A \oplus \neg B$)일 경우, 십진수 '3'을 산술합하지 않고 십진수 '1'을 산술합한다는데 차이가 있다.
- [0111] 또한, 상기 1비트 입력 오류검출코드($SEDC_{1-in}$)는 도 7의 1비트 입력 오류검출코드($SEDC_{1-in}$) 생성 로직(114)에 의해 계산된 2비트 입력 오류검출코드($SEDC_{2-in}$) 중 최하위 비트만을 입력 오류검출코드로 생성할 수 있다.
- [0112] 그러나, 상기 1비트 입력 오류검출코드($SEDC_{1-in}$)는 최상위 비트가 '0'으로 가정된 2비트 입력 데이터에 대한 2비트 입력 오류검출코드($SEDC_{2-in}$)와 동일하게 2비트의 입력 오류검출코드로 생성될 수 있으며, 이 경우, 상기 오류 검출기는 입력 오류검출코드와 출력 오류검출코드의 최하위 비트만을 서로 비교하여 오류검출결과를 출력할 수 있다.
- [0113] 도 8은 상기 2비트 입력 오류검출코드($SEDC_{2-in}$) 생성 로직(113)을 구현한 논리회로(115)를 보여주는 도면으로써 상기 2비트 입력 오류검출코드 생성수단의 실질적인 구성이다.
- [0114] 또한, 상기 논리회로(115)는 복수 개의 논리 연산 게이트들, 2비트 오류검출코드 생성수단들($SEDC_2$) 및 불 연산의 종류에 따라 제어신호들($S_1, S_2, S_3, S_4, S_5, S_6, S_7, S_8$)을 출력하여 상기 논리 연산 게이트들과 1비트 오류검출코드 생성수단들($SEDC_1$)을 동작하게 하는 제어신호 생성기(115-20)를 포함하여 이루어진다.
- [0115] 더욱 자세하게는 상기 2비트 입력 오류검출코드($SEDC_{2-in}$) 생성 로직(113)은 상기 제1 입력 데이터(A)와 제1 제어신호(S_1)의 배타적 논리합 연산을 수행하는 제1 XOR게이트(115-1), 이진 데이터 '00'을 출력하는 '00'출력기(115-2), 상기 제1 XOR게이트(115-1)와 상기 '00'출력기(115-2)의 출력을 입력받고, 제3 제어신호(S_3)에 의해 입력되는 신호들 중 어느 하나의 신호를 출력하는 제1 2비트 2×1 멀스(MUX)(115-3), 상기 제1 2비트 2×1 멀스(115-3)의 출력을 입력받아 2비트 오류검출코드를 출력하는 제1 2비트 오류검출코드 생성기(115-4), 상기 제1 2비트 오류검출코드 생성기(115-4)의 출력과 상기 제어신호들 중, 제5 제어신호(S_5)를 입력받아 논리곱 연산을 수행하는 제1 AND게이트(115-5), 상기 제2 입력 데이터(B)와 제2 제어신호(S_2)의 배타적 논리합 연산을 수행하는 제2 XOR게이트(115-6), 이진 데이터 '11'을 출력하는 제1 '11'출력기(115-7), 상기 제2 XOR게이트(115-6)와 상기 '11'출력기(115-7)의 출력을 입력받고 상기 제3 제어신호(S_3)에 의해 입력되는 신호들 중 어느 하나의 신호를 출력하는 제2 2비트 2×1 멀스(115-8), 상기 제2 2비트 2×1 멀스(115-8)의 출력을 입력받아 2비트 오류검출코드를 출력하는 제2 2비트 오류검출코드 생성기(115-9), 상기 제2 2비트 오류검출코드 생성기(115-9)의 출력과 제6 제어신호(S_6)를 입력받아 논리곱 연산을 수행하는 제2 AND게이트(115-10), 이진 데이터 '11'을 출력하는 제2 '11'출력기(115-11), 상기 제2 '11'출력기(115-11)의 출력과 제7 제어신호(S_7)를 입력받아 논리곱 연산을 수행하는 제3 AND게이트(115-12), 상기 제1 2비트 2×1 멀스(115-3)와 상기 제2 2비트 2×1 멀스(115-8)의 출력을 논리합 연산하는 OR게이트(115-13), 상기 제1 2비트 2×1 멀스(115-3)와 상기 제2 2비트 2×1 멀스(115-8)의 출력을 논리곱 연산하는 제4 AND게이트(115-14), 상기 OR게이트(115-13)와 상기 제4 AND게이트(115-14)의 출력을 입력받고 제4 제어신호(S_4)에 의해 입력되는 신호들 중 어느 하나의 신호를 출력하는 제3 2비트 2×1 멀스(115-15), 상기 제3 2비트 2×1 멀스(115-15)의 출력을 입력받아 2비트 오류검출코드를 출력하는 제3 2비트 오류검출코드 생성기(115-16), 상기 제3 2비트 오류검출코드 생성기(115-16)의 출력과 제8 제어신호(S_8)를 논리곱 연산하는 제4 AND게이트(115-17), 상기 제4 AND게이트(115-17)의 출력을 입력받고 상기 제7 제어신호(S_7)에 의해 상기 제4 AND게이트(115-17)의 출력을 왼쪽 시프트 연산하는 Left Shift게이트(115-18)(상기 Left Shift게이트

(115-18)는 2비트의 입력을 왼쪽으로 시프트하여 3비트의 데이터를 출력하며, 최하위 비트는 '0'의 값을 갖는다.), 상기 제1 AND게이트(115-5), 상기 제2 AND게이트(115-10) 및 상기 제3 AND게이트(115-12)의 출력을 산술합(+)하고 상기 Left Shift게이트(115-18)의 출력을 산술차(-)하여 상기 입력 오류검출코드를 출력하는 가산기(Block Adder, 115-19)를 포함하여 이루어진다.

[0116] 도 9는 상기 제어신호 생성기(115-20)에서 출력되는 제어신호들의 생성 로직(116)을 보여주는 것으로 여덟 개의 제어신호들($S_1, S_2, S_3, S_4, S_5, S_6, S_7, S_8$)이 열여섯 개의 불 연산에 대해 어떠한 값을 출력하는지 보여준다.

[0117] 또한, 상기 제어신호 생성기(115-20)는 불 연산의 종류를 선택하는 불 연산 선택신호(opB0)를 입력받고 해당 불 연산에 따른 제어신호들을 출력하여 상기 입력 데이터들(A,B)에 대해 선택된 불 연산의 결과에 대응하는 2비트 입력 오류검출코드($SEDC_{2-in}$)가 생성되게 한다.

[0118] 또한, 상기 입력 오류검출 코드 생성기(110)는 상기 이진 입력 데이터들(A,B)이 3비트의 입력 데이터일 경우 2비트인 오류검출코드인 3비트 입력 오류검출코드($SEDC_{3-in}$)를 생성하는 3비트 입력 오류검출코드 생성수단을 더 포함한다.

[0119] 또한, 도 10은 상기 3비트 입력 오류검출코드 생성수단이 상기 3비트 입력 오류 검출코드($SEDC_{3-in}$)($C_{1(3bit)}, C_{0(3bit)}$)를 생성하는 생성 로직(117)을 보여주는 도면이다.

[0120] 도 10을 참조하면, 상기 3비트 입력 오류검출코드 생성수단은 3비트의 이진 입력 데이터들($A(A_2, A_1, A_0), B(B_2, B_1, B_0)$)과 불 연산 선택신호(opB0)를 입력받아 상기 3비트 입력 오류 검출코드($SEDC_{3-in}$)를 생성한다.

[0121] 먼저, 상기 3비트 입력 오류검출코드 생성수단은 상기 3비트 입력 데이터들($A(A_2, A_1, A_0), B(B_2, B_1, B_0)$) 각각의 하위 두 비트($(A_1, A_0), (B_1, B_0)$)를 입력으로 하는 상기 2비트 입력 오류검출코드 생성수단의 2비트 입력 오류검출코드($SEDC_{2-in}(A(A_1, A_0), B(B_1, B_0), opB0)$)를 과도 2비트 입력 오류검출코드($C_{1(2bit)}, C_{0(2bit)}$)로 생성한다.

[0122] 다음, 상기 불 연산 선택신호(opB0)의 종류 및 상기 입력 데이터들 각각의 최상위 비트의 값(A_2, B_2)에 따라 상기 과도 2비트 입력 오류검출코드 자체 또는 상기 과도 2비트 입력 오류검출코드에 '1'을 산술차(-)하여 상기 3비트 입력 오류검출코드($SEDC_{3-in}$)를 생성한다.

[0123] 여기서, '1'을 산술차(-)하는 경우는, 도 10에 도시한 바와 같이 상기 입력 데이터들 각각의 최상위 비트의 값(A_2, B_2)과 상기 불 연산 선택신호(opB0)에 의해 결정되며, 상기 불 연산이 상기 제1 입력 데이터(A) 및 상기 제2 입력 데이터(B)의 부정 논리합연산(A NOR B)이고, 상기 제1 입력 데이터(A) 및 상기 제2 입력 데이터(B)의 최상위 비트(A_2, B_2)가 '0'인 경우, 상기 불 연산이 상기 제1 입력 데이터(A)의 논리 부정 값(NOT A)과 상기 제2 입력 데이터(B)의 논리곱연산((NOT A) AND B)이고, 상기 제1 입력 데이터(A)의 최상위 비트(A_2)가 '0'이고 상기 제2 입력 데이터(B)의 최상위 비트(B_2)가 '1'인 경우, 상기 불 연산이 상기 제1 입력 데이터(A)의 논리부정연산(NOT A)이고, 상기 제2 입력 데이터(B)의 최상위 비트에 관계없이 상기 제1 입력 데이터(A)의 최상위 비트(A_2)가 '0'인 경우, 상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 논리 부정값(NOT B)의 논리곱연산(A AND (NOT B))이고, 상기 제1 입력 데이터(A)의 최상위 비트(A_2)가 '1'이고, 상기 제2 입력 데이터(B)의 최상위 비트(B_2)가 '0'인 경우, 상기 불 연산이 상기 제2 입력 데이터(B)의 논리부정연산(NOT B)이고, 상기 제1 입력 데이터(A)의 최상위 비트(A_2)에 관계없이 상기 제2 입력 데이터(B)의 최상위 비트(B_2)가 '0'인 경우, 상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 배타적 논리합연산(A XOR B)이고, 상기 제1 입력 데이터(A)의 최상위 비트(A_2)가 '1'이고, 상기 제2 입력 데이터(B)의 최상위 비트(B_2)가 '0'이거나, 상기 제1 입력 데이터(A)의 최상위 비트(A_2)가 '0'이고, 상기 제2 입력 데이터(B)의 최상위 비트(B_2)가 '1'인 경우, 상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 부정 논리곱연산(A NAND B)이고, 상기 제1 입력 데이터(A) 및 상기 제2 입력 데이터(B)의 최상위 비트(A_2, B_2)가 '1'이 아닌 경우, 상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 논리곱연산(A AND B)이고, 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 최상위 비트(A_2, B_2)가 '1'인 경우, 상기 불 연산이 상기 제1 입력 데

이터(A)와 상기 제2 입력 데이터(B)의 배타적 부정 논리합연산(A XNOR B)이고, 상기 제1 입력 데이터(A)의 최상위 비트(A₂)가 '0'이고, 상기 제2 입력 데이터(B)의 최상위 비트(B₂)가 '0'이거나, 상기 제1 입력 데이터(A)의 최상위(A₂) 비트가 '1'이고, 상기 제2 입력 데이터(B)의 최상위 비트(B₂)가 '1'인 경우, 상기 불 연산이 상기 제2 입력 데이터(B)의 출력연산(B)이고, 상기 제1 입력 데이터(A)의 최상위 비트(A₂)에 관계없이 상기 제2 입력 데이터(B)의 최상위 비트(B₂)가 '1'인 경우, 상기 불 연산이 상기 제1 입력 데이터(A)의 논리 부정값(NOT A)과 상기 제2 입력 데이터(B)의 논리합연산((NOT A) OR B)이고, 상기 제1 입력 데이터(A)의 최상위 비트(A₂)가 '1'이고 상기 제2 입력 데이터(B)의 최상위 비트(B₂)가 '0'이 아닌 경우, 상기 불 연산이 상기 제1 입력 데이터(A)의 출력연산(A)이고, 상기 제2 입력 데이터(B)의 최상위 비트(B₂)에 관계없이 상기 제1 입력 데이터(A)의 최상위 비트(A₂)가 '1'인 경우, 상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 논리 부정값(NOT B)의 논리합연산(A OR (NOT B))이고, 상기 제1 입력 데이터(A)의 최상위 비트(A₂)가 '0'이고 상기 제2 입력 데이터(B)의 최상위 비트(B₂)가 '1'이 아닌 경우, 상기 불 연산이 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 논리합연산(A OR B)이고, 상기 제1 입력 데이터(A)와 상기 제2 입력 데이터(B)의 최상위 비트(A₂, B₂)가 '0'이 아닌 경우 또는 상기 제1 입력 데이터(A) 및 상기 제2 입력 데이터(B)의 최상위 비트(A₂, B₂)에 관계없이 상기 불 연산이 로직'1'(Logic 1)연산일 경우이다.

[0124] 또한, 상기 입력 오류검출 코드 생성기(110)는 상기 이진 입력 데이터들(A,B)이 4비트의 입력 데이터일 경우 3비트인 오류검출코드인 4비트 입력 오류검출코드(SEDC_{4-in})를 생성하는 4비트 입력 오류검출코드 생성수단을 더 포함한다.

[0125] 또한, 도 11은 상기 4비트 입력 오류검출코드 생성수단이 상기 3비트 입력 오류 검출코드(SEDC_{4-in})(C₂(4bit), C₁(4bit), C₀(4bit))를 생성하는 생성 로직(118)을 보여주는 도면이다.

[0126] 도 11을 참조하면, 상기 4비트 입력 오류검출코드 생성수단은 4비트의 이진 입력 데이터들(A(A₃, A₂, A₁, A₀), B(B₃, B₂, B₁, B₀))과 불 연산 선택신호(opB0)를 입력받아 상기 4비트 입력 오류 검출코드(SEDC_{4-in})를 생성한다.

[0127] 먼저, 상기 4비트 입력 오류검출코드 생성수단은 상기 4비트 입력 데이터(A(A₃, A₂, A₁, A₀), B(B₃, B₂, B₁, B₀))의 하위 세 비트((A₂, A₁, A₀), (B₂, B₁, B₀))를 입력으로 하는 상기 3비트 입력 오류검출코드 생성수단의 3비트 입력 오류검출코드(SEDC_{3-in})((C₁(3bit), C₀(3bit)))를 상기 4비트 입력 오류검출코드(SEDC_{4-in})의 하위 두 비트(C₁(4bit), C₀(4bit))로 생성한다.

[0128] 다음, 상기 4비트 입력 오류검출코드(SEDC_{4-in})의 최상위 비트(C₂(4bit))는 선택된 불 연산의 종류 및 상기 이진 입력 데이터들(A,B) 각각의 최상위 비트(A₂, B₂)의 값에 따라 '1' 또는 '0' 값으로 하여 상기 4비트 입력 오류 검출코드(SEDC_{4-in})를 생성한다.

[0129] 또한, 상기 4비트 입력 오류검출코드(SEDC_{4-in})의 최상위 비트(C₂(4bit))가 '1'로 생성되는 경우는, 도 10에 도시한 상기 3비트 입력 오류검출코드 생성수단이 상기 과도 2비트 입력 오류검출코드에 '1'을 산술차(-)하는 경우들 이외의 경우이다.

[0130] 또한, 상기 입력 오류검출 코드 생성기(110)는 상기 이진 입력 데이터들(A,B)이 n비트(n≥1)의 입력 데이터일 경우 n비트인 오류검출코드인 n비트 입력 오류검출코드(SEDC_{n-in})를 생성하는 n비트 입력 오류검출코드 생성수단을 더 포함한다.

[0131] 또한, 상기 n비트 입력 오류검출코드 생성수단은 아래의 수학적식 2와 같이 입력되는 입력 데이터의 길이에 따라 생성될 입력 오류검출코드의 길이를 계산한다.

수학식 2

$$m = \lceil \log_2[n+1-(3 \times a)] \rceil + (2 \times a)$$

[0132] 여기서, m 은 상기 n 비트 입력 오류검출코드($SEDC_{n-in}$)의 길이이고, 파라미터 ' a '는 '0'에서부터 ' ∞ '까지의 정수 중 어느 하나이며, 상기 파라미터 ' a '는 아래의 수학식 3에 의해 계산된다.

수학식 3

$$a = \frac{n - \max(b)}{3}$$

[0133] 여기서, n 은 상기 입력데이터의 길이이고, 파라미터 ' b '는 '2', '3' 또는 '4'의 정수이며, $\max(b)$ 는 정수 '2', '3' 및 '4' 중, 상기 파라미터 ' a '가 정수가 되게 하는 최대값이다.

[0134] 예를 들어, 상기 입력 데이터(n)의 길이가 '8비트'일 경우, 상기 수학식 3의 $\max(b)$ 는 정수 '2'로 선택되고, 상기 파라미터 ' a '는 정수 '2'로 계산된다.

[0135] 또한, 상기 파라미터 ' a '가 정수 '2'로 계산되었으므로, 수학식 2에 대입하면, 가우스 ' $\log_2 3$ '은 정수 '2'가 되고, 여기에 정수 '4'를 더하면 정수 '6'이 되어, 상기 입력 데이터의 길이가 '8비트'일 경우, 오류검출코드의 길이는 '6비트'로 계산되는 것이다.

[0136] 즉, 상기 n 비트 입력 오류검출코드 생성수단은 상기 입력 데이터(A,B)가 '8비트'일 경우, 상기 2비트 입력 오류검출코드($SEDC_{2-in}$), 상기 3비트 입력 오류검출코드($SEDC_{3-in}$) 및 상기 4비트 입력 오류검출코드($SEDC_{4-in}$)의 조합하여 '8비트'의 입력오류검출코드를 생성하는 역할을 한다.

[0137] 따라서, 본 발명의 일 실시예에 따른 오류 검출 장치(100)는 입력 데이터의 길이에 선형적으로 오류검출코드를 확장할 수 있는 확장형 오류검출 기반의 오류 검출 장치이다.

[0138] 그러나, 상기 n 비트 입력 오류검출코드 생성수단은 상기 2비트 입력 오류검출코드($SEDC_{2-in}$), 상기 3비트 입력 오류검출코드($SEDC_{3-in}$) 및 상기 4비트 입력 오류검출코드($SEDC_{4-in}$)의 조합하지 않고, 상기 입력 데이터가 2비트, 3비트 또는 4비트의 입력 데이터일 경우, 상기 2비트 입력 오류검출코드($SEDC_{2-in}$), 상기 3비트 입력 오류검출코드($SEDC_{3-in}$) 및 상기 4비트 입력 오류검출코드($SEDC_{4-in}$)를 각각 상기 입력 오류검출코드로 생성할 수 있다.

[0139] 또한, 상기 n 비트 입력 오류검출코드 생성수단은 상기 2비트 입력 오류검출코드($SEDC_{2-in}$), 상기 3비트 입력 오류검출코드($SEDC_{3-in}$) 및 상기 4비트 입력 오류검출코드($SEDC_{4-in}$) 중, 상기 파라미터 ' b '의 숫자와 대응하는 어느 하나의 입력 오류검출코드와 상기 3비트 입력 오류검출코드($SEDC_{3-in}$) 하나 또는 복수 개의 조합으로 상기 입력 오류검출코드를 생성할 수 있다.

[0140] 예를 들면, 상기 입력 데이터(A,B)가 각각 '8비트'의 입력 데이터일 경우, 상기 파라미터 ' b '는 정수 '2'로 계산되고, 상기 입력 오류검출코드의 길이는 '6비트'로 계산됨을 앞서 보인바 있다.

[0141] 이 경우, 상기 n 비트 입력 오류검출코드 생성수단은 상기 파라미터 ' b '인 정수 '2'와 대응하는 2비트 입력 오류검출코드($SEDC_{2-in}$) 하나와 3비트 입력 오류검출코드($SEDC_{3-in}$) 두 개를 조합하여 총 '6비트'의 오류검출코드를 생성하는 것이다.

[0142] 그러나, 상기 n 비트 입력 오류검출코드 생성수단은 상기 입력 오류검출코드의 길이가 '6비트'일 경우, 상기 2비트 입력 오류검출코드($SEDC_{2-in}$) 세 개나 상기 3비트 입력 오류검출코드($SEDC_{3-in}$) 세 개, 또한, 상기 2비트 입력 오류검출코드($SEDC_{2-in}$) 두 개와 상기 3비트 입력 오류검출코드($SEDC_{3-in}$) 하나를 조합하거나, 상기 4비트 입력 오

류검출코드($SEDC_{4-in}$) 두 개를 조합하여 상기 입력 오류검출코드를 생성할 수도 있다.

[0145] 또한, 상기 n 비트 입력 오류검출코드 생성수단이 상기 입력 데이터가 2비트, 3비트 또는 4비트의 입력 데이터일 경우, 상기 2비트 입력 오류검출코드($SEDC_{2-in}$), 상기 3비트 입력 오류검출코드($SEDC_{3-in}$) 및 상기 4비트 입력 오류검출코드($SEDC_{4-in}$)를 각각 상기 입력 오류검출코드를 생성할 수 있는데, 예를 들면, 상기 입력 데이터(A,B)의 길이가 '2비트'일 경우, 상기 파라미터 'b'는 '2'가 되고 상기 파라미터 'a'는 '0'이 되며 입력 오류검출코드의 길이는 가우스 ' $\log_2 3$ '이 되어 '2비트'가 된다. 즉, 상기 n 비트 입력 오류검출코드 생성수단은 상기 파라미터 'b'가 '2'이므로 하나의 2비트 입력 오류검출코드($SEDC_{2-in}$)만으로 입력 오류검출코드를 생성할 수 있는 것이다.

[0146] 상기 오류 검출기(120)는 상기 프로그래머블 불 연산 유닛(10)에서 출력되는 연산결과인 출력 데이터(F)에 대해 출력 오류검출코드를 생성하며, 상기 입력 오류검출코드 생성기(110)에서 생성된 입력 오류검출코드와 상기 출력 오류검출코드를 서로 비교하여 동일하지 않을 경우 오류 발생 신호를 출력한다.

[0147] 또한, 상기 오류 검출기(120)는 상기 출력 데이터가 2비트의 출력데이터일 경우, 상기 출력 데이터에 대한 2비트의 오류검출코드인 2비트 출력 오류검출코드($SEDC_{2-out}$)를 상기 출력 오류검출코드로 출력하는 2비트 출력 오류검출코드 생성수단과 상기 입력 오류검출코드와 상기 출력 오류검출코드의 동일성을 비교하여 오류 발생 신호를 출력하는 코드비교수단을 포함한다.

[0148] 또한, 도 12는 상기 출력 데이터(F, 121a)에 대한 2비트 출력 오류검출코드($SEDC_{2-out}$)의 진리 테이블(121)을 보여주는 것으로 두 가지 케이스(121b, 121c)($C_1'_{(2bit)}, C_0'_{(2bit)}$)로 생성될 수 있으며, 상기 2비트 출력 오류검출코드($SEDC_{2-out}$)의 진리 테이블(121)은 도 2에서 보인 2비트 오류검출코드($SEDC_2$)의 진리 테이블(111)과 실질적으로 동일하므로 설명을 생략한다.

[0149] 다만, 상기 2비트 출력 오류검출코드 생성수단은 상기 입력 오류검출코드 생성기에서 사용되는 2비트 오류검출코드들($SEDC_2$)의 특정 케이스와 동일한 케이스의 2비트 출력 오류검출코드($SEDC_{2-out}$)를 상기 출력 오류검출코드로 생성하여야 한다.

[0150] 또한, 상기 오류 검출기(120)는 상기 출력 데이터가 3비트의 출력데이터일 경우, 상기 출력 데이터에 대한 2비트의 오류검출코드인 3비트 출력 오류검출코드($SEDC_{3-out}$)를 상기 출력 오류검출코드로 출력하는 3비트 출력 오류검출코드 생성수단을 더 포함한다.

[0151] 도 13은 상기 3비트 출력 오류검출코드($SEDC_{3-out}$)의 진리 테이블(122)을 보여주는 것으로 상기 3비트 출력 오류검출코드 생성수단은 아래의 수학적 식 4를 이용하여 상기 출력 데이터(F)를 3비트 출력 오류검출코드($C_1'_{(3bit)}, C_0'_{(3bit)}$)로 생성하며, 더욱 자세하게는, 상기 출력 데이터(F)의 최상위 비트($F_{2(3bit)}$)가 '0'일 경우에는, 상기 출력 데이터(F) 중간비트($F_{1(3bit)}$) 및 최하위 비트($F_{0(3bit)}$)를 입력으로 하는 상기 2비트 출력 오류검출코드 생성수단의 2비트 출력 오류검출코드($SEDC_{2-out}$)와 동일하게 상기 3비트 출력 오류검출코드($C_1'_{(3bit)}, C_0'_{(3bit)}$)를 생성하고, 상기 출력 데이터(F)의 최상위 비트($F_{2(3bit)}$)가 '1'일 경우에는 상기 출력 데이터(F) 중간비트($F_{1(3bit)}$) 및 최하위 비트($F_{0(3bit)}$)의 논리 부정값을 입력으로 하는 상기 2비트 출력 오류검출코드 생성수단의 2비트 출력 오류검출코드($SEDC_{2-out}$)의 논리 부정값을 상기 3비트 출력 오류검출코드($C_1'_{(3bit)}, C_0'_{(3bit)}$)로 생성한다.

수학적 식 4

$$(C_1'_{(3bit)}, C_0'_{(3bit)}) = \begin{cases} SEDC_{2-out}(F_{1(3bit)}, F_{0(3bit)}), & \text{if } F_{2(3bit)} = 0 \\ 1's \text{ complement}(SEDC_{2-out}(\overline{F_{1(3bit)}}), \overline{F_{0(3bit)}}), & \text{if } F_{2(3bit)} = 1 \end{cases}$$

[0153] 또한, 도 14는 상기 3비트 출력 오류검출코드 생성수단을 구현한 논리회로(122-1)를 보여주는 것으로 세 개의

입력에 대해 exclusive-NOT-OR 연산을 수행하는 하나의 XNOR게이트(122-1a), 네 개의 입력에 대해 or-and-invert 연산을 수행하는 하나의 OAI게이트(122-1b) 및 두 개의 입력에 대해 AND 연산을 수행하는 AND게이트(122-1c)로 설계하였다.

[0154] 그러나, 상기 3비트 출력 오류검출코드 생성수단은 3비트의 출력 데이터에 생성되는 오류검출코드가 도 13의 진리 테이블을 만족할 수 있다면, 어떠한 논리 게이트들의 조합으로도 설계가 가능하다.

[0155] 또한, 상기 오류 검출기(120)는 상기 출력 데이터가 4비트의 출력 데이터일 경우, 상기 출력 데이터에 대한 3비트의 오류검출코드인 4비트 출력 오류검출코드(SED_{C4-out})를 상기 출력 오류검출코드로 출력하는 4비트 출력 오류검출코드 생성수단을 더 포함한다.

[0156] 도 15는 상기 4비트 출력 오류검출코드(SED_{C4-out})의 진리 테이블(123)을 보여주는 것으로 상기 4비트 출력 오류검출코드 생성수단은 아래의 수학적 식 5를 이용하여 상기 출력 데이터(F)를 4비트 출력 오류검출코드(C_{2'}(4bit), C_{1'}(4bit), C_{0'}(4bit))로 생성하며, 더욱 자세하게는, 상기 4비트 출력 오류검출코드(SED_{C4-out})의 하위 2비트(C_{1'}(4bit), C_{0'}(4bit))는 상기 4비트 출력 데이터들 중, 최상위 비트(F_{3(4bit)})를 제외한 3비트(F_{2(4bit)}, F_{1(4bit)}, F_{0(4bit)})를 입력으로 하는 상기 3비트 출력 오류검출코드 생성수단의 3비트 출력 오류검출코드(SED_{C3-out})로 생성하고, 상기 4비트 출력 오류검출코드(SED_{C4-out})의 최상위 비트(C_{2'}(4bit))는 상기 4비트 출력 데이터의 최상위 비트(F_{3(4bit)})의 논리 부정 값으로 생성한다.

수학적 식 5

$$(C_1'(4bit), C_0'(4bit)) = \overline{SED_{C3-out}(F_{2(4bit)}, F_{1(4bit)}, F_{0(4bit)})}$$

$$C_2'(4bit) = F_{3(4bit)}$$

[0158] 또한, 도 16은 상기 4비트 출력 오류검출코드 생성수단을 구현한 논리회로(123-1)를 보여주는 것으로 두 개의 입력에 대해 exclusive-NOT-OR 연산을 수행하는 하나의 XNOR게이트(123-1a), 두 개의 입력에 대해 exclusive-OR 연산을 수행하는 하나의 XOR 게이트(123-1b), 두 개의 입력과 하나의 선택 입력에 대해 선택적 입력을 출력하는 하나의 MUX(123-1c), 하나의 입력에 대해 invert 연산을 수행하는 하나의 invert 게이트(123-1d)로 설계하였다.

[0159] 또한, 상기 오류 검출기(120)는 상기 이진 출력 데이터(F)가 n비트 출력 데이터일 경우, 상기 n비트 출력 데이터에 대한 오류검출코드인 n비트 출력 오류검출코드(SED_{Cn-out})를 상기 출력 오류검출코드로 생성하는 n비트 출력 오류검출코드 생성수단을 더 포함한다.

[0160] 또한, 상기 n비트 출력 오류검출코드 생성수단은 상기 출력 오류검출코드 생성수단들 각각의 오류검출코드(SED_{C2-out}, SED_{C3-out}, SED_{C4-out})나, 상기 출력 오류검출코드 생성수단들 중 어느 하나의 출력 오류검출코드 생성수단의 오류검출코드와 하나 또는 복수 개의 3비트 출력 오류검출코드(SED_{C3-out})의 조합을 상기 출력 오류검출코드로 생성한다.

[0161] 또한, 상기 n비트 출력 오류검출코드 생성수단이 출력 오류검출코드를 생성하는 방법은 상기 n비트 입력 오류검출코드 생성수단이 입력 오류검출코드를 생성하는 방법과 실질적으로 동일하므로 설명을 생략하기로 한다.

[0162] 이상에서 살펴본 바와 같이 본 발명은 바람직한 실시예를 들어 도시하고 설명하였으나, 상기한 실시예에 한정되지 아니하며 본 발명의 정신을 벗어나지 않는 범위 내에서 당해 발명이 속하는 기술분야에서 통상의 지식을 가진 자에 의해 다양한 변경과 수정이 가능할 것이다.

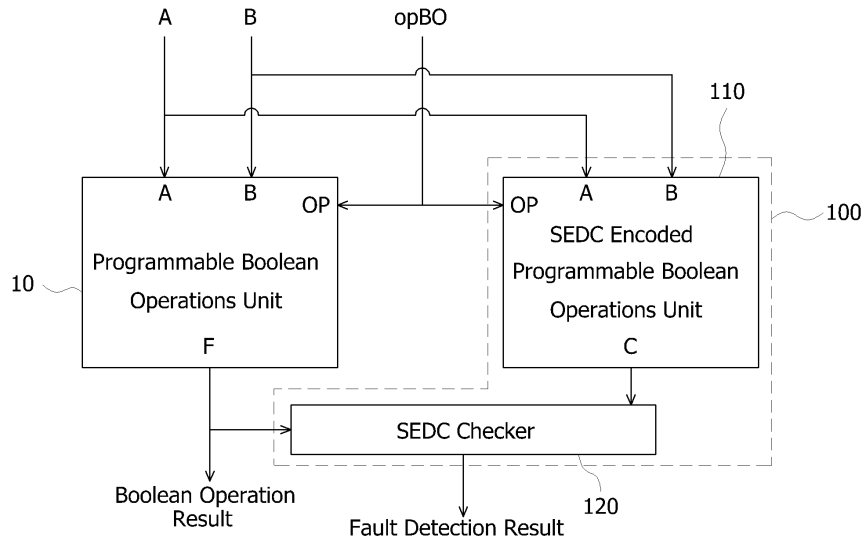
부호의 설명

[0163] 10: 프로그래머블 불 연산 유닛 100: 오류 검출 장치

110: 입력 오류검출코드 생성기 120: 오류 검출기

도면

도면1



도면2

111

A		SEDC ₂ (A) Case1		SEDC ₂ (A) Case2	
A ₁	A ₀	C ₁	C ₀	C ₁	C ₀
0	0	1	1	1	1
0	1	1	0	0	1
1	0	1	0	0	1
1	1	0	1	1	0

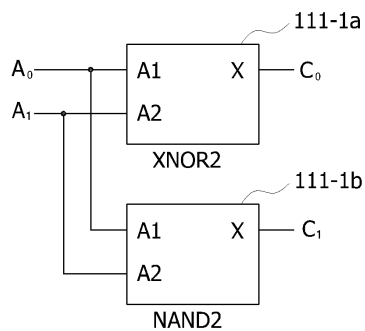
111a

111b

111c

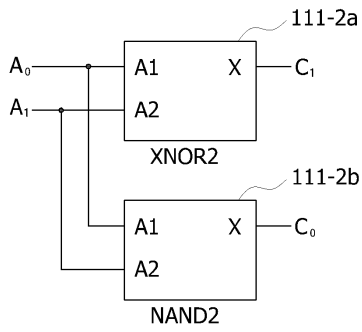
도면3

111-1



도면4

111-2



도면5

112

112a		112b		112c	112d	112e	112f
A	SEDC ₂ (A)	B	SEDC ₂ (B)	SEDC ₂ (A AND B)	SEDC ₂ (A OR B)	2 x SEDC ₂ (A AND B)	SEDC ₂ (A XOR B)
00	11	00	11	11	11	110	11
00	11	01	10	11	10	110	10
00	11	10	10	11	10	110	10
00	11	11	01	11	01	110	01
01	10	00	11	11	10	110	10
01	10	01	10	10	10	100	11
01	10	10	10	11	01	110	01
01	10	11	01	10	01	100	10
10	10	00	11	11	10	110	10
10	10	01	10	11	01	110	01
10	10	10	10	10	10	100	11
10	10	11	01	10	01	100	10
11	01	00	11	11	01	110	01
11	01	01	10	10	01	100	10
11	01	10	10	10	01	100	10
11	01	11	01	01	01	010	11

도면6

113

Operations	opBO	2-bit SEDC Encoded Boolean Operations
		$(C_{1(2bit)}, C_{0(2bit)}) = SEDC_{2-in}(A(A_1, A_0), B(B_1, B_0), opBO)$
Logic 0	0	$SEDC_2(00)$
A NOR B	1	$SEDC_2(\bar{A}) + SEDC_2(\bar{B}) - SEDC_2(\bar{A} \text{ OR } \bar{B})$
(NOT A)AND B	2	$SEDC_2(\bar{A}) + SEDC_2(B) - SEDC_2(\bar{A} \text{ OR } B)$
NOT A	3	$SEDC_2(\bar{A})$
A AND (NOT B)	4	$SEDC_2(A) + SEDC_2(\bar{B}) - SEDC_2(A \text{ OR } \bar{B})$
NOT B	5	$SEDC_2(\bar{B})$
A XOR B	6	$SEDC_2(A) + SEDC_2(B) - (2 \times SEDC_2(A \text{ AND } B)) + 3$
A NAND B	7	$SEDC_2(\bar{A}) + SEDC_2(\bar{B}) - SEDC_2(\bar{A} \text{ AND } \bar{B})$
A AND B	8	$SEDC_2(A) + SEDC_2(B) - SEDC_2(A \text{ OR } B)$
A XNOR B	9	$SEDC_2(\bar{A}) + SEDC_2(B) - (2 \times SEDC_2(\bar{A} \text{ AND } B)) + 3$
B	10	$SEDC_2(B)$
(NOT A)OR B	11	$SEDC_2(\bar{A}) + SEDC_2(B) - SEDC_2(\bar{A} \text{ AND } B)$
A	12	$SEDC_2(A)$
A OR (NOT B)	13	$SEDC_2(A) + SEDC_2(\bar{B}) - SEDC_2(A \text{ AND } \bar{B})$
A OR B	14	$SEDC_2(A) + SEDC_2(B) - SEDC_2(A \text{ AND } B)$
Logic 1	15	$SEDC_2(3)$

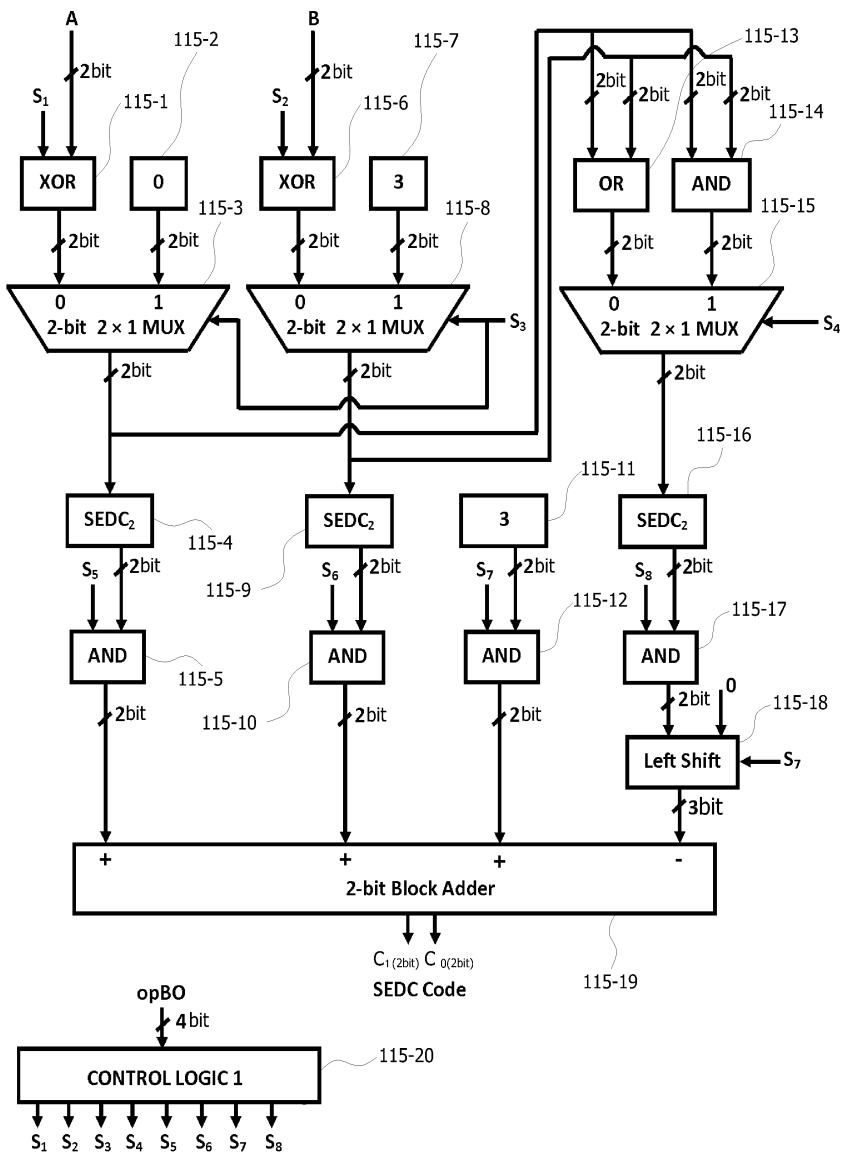
도면7

114

Operations	opBO	1-bit SEDC Encoded Boolean Operations
		$(C_{0(1bit)}) = SEDC_{1-in}(A, B, opBO)$
Logic 0	0	$lsb\{SEDC_2(00)\}$
A NOR B	1	$lsb\{SEDC_2(0[\bar{A}]) + SEDC_2(0[\bar{B}]) - SEDC_2(0[\bar{A} \text{ OR } \bar{B}])\}$
(NOT A)AND B	2	$lsb\{SEDC_2(0[\bar{A}]) + SEDC_2(0[B]) - SEDC_2(0[\bar{A} \text{ OR } B])\}$
NOT A	3	$lsb\{SEDC_2(0[\bar{A}])\}$
A AND (NOT B)	4	$lsb\{SEDC_2(0[A]) + SEDC_2(0[\bar{B}]) - SEDC_2(0[A \text{ OR } \bar{B}])\}$
NOT B	5	$lsb\{SEDC_2(0[\bar{B}])\}$
A XOR B	6	$lsb\{SEDC_2(0[A]) + SEDC_2(0[B]) - (2 \times SEDC_2(0[A \text{ AND } B])) + 1\}$
A NAND B	7	$lsb\{SEDC_2(0[\bar{A}]) + SEDC_2(0[\bar{B}]) - SEDC_2(0[\bar{A} \text{ AND } \bar{B}])\}$
A AND B	8	$lsb\{SEDC_2(0[A]) + SEDC_2(0[B]) - SEDC_2(0[A \text{ OR } B])\}$
A XNOR B	9	$lsb\{SEDC_2(0[\bar{A}]) + SEDC_2(0[B]) - (2 \times SEDC_2(0[\bar{A} \text{ AND } B])) + 1\}$
B	10	$lsb\{SEDC_2(0[B])\}$
(NOT A)OR B	11	$lsb\{SEDC_2(0[\bar{A}]) + SEDC_2(0[B]) - SEDC_2(0[\bar{A} \text{ AND } B])\}$
A	12	$lsb\{SEDC_2(0[A])\}$
A OR (NOT B)	13	$lsb\{SEDC_2(0[A]) + SEDC_2(0[\bar{B}]) - SEDC_2(0[A \text{ AND } \bar{B}])\}$
A OR B	14	$lsb\{SEDC_2(0[A]) + SEDC_2(0[B]) - SEDC_2(0[A \text{ AND } B])\}$
Logic 1	15	$lsb\{SEDC_2(01)\}$

도면8

115



도면9

116

CONTROL LOGIC 1									
Operations	opBO	S ₁	S ₂	S ₃	S ₄	S ₅	S ₆	S ₇	S ₈
Logic 0	0	×	×	1	×	1	0	0	0
A NOR B	1	1	1	0	0	1	1	0	1
(NOT A)AND B	2	1	0	0	0	1	1	0	1
NOT A	3	1	×	0	×	1	0	0	0
A AND (NOT B)	4	0	1	0	0	1	1	0	1
NOT B	5	×	1	0	×	0	1	0	0
A XOR B	6	0	0	0	1	1	1	1	1
A NAND B	7	1	1	0	1	1	1	0	1
A AND B	8	0	0	0	0	1	1	0	1
A XNOR B	9	1	0	0	1	1	1	1	1
B	10	×	0	0	×	0	1	0	0
(NOT A)OR B	11	1	0	0	1	1	1	0	1
A	12	0	×	0	×	1	0	0	0
A OR (NOT B)	13	0	1	0	1	1	1	0	1
A OR B	14	0	0	0	1	1	1	0	1
Logic 1	15	×	×	1	×	0	1	0	0

도면10

117

3-bit SEDC Encoded Boolean Operations				
$(C_{1(3bit)}, C_{0(3bit)}) = SEDC_{3-in}(A(A_2, A_1, A_0), B(B_2, B_1, B_0), opBO)$				
$(C_{1(2bit)}, C_{0(2bit)}) = SEDC_{2-in}(A(A_1, A_0), B(B_1, B_0), opBO)$ $(C_{1(3bit)}, C_{0(3bit)}) = \begin{cases} (C_{1(2bit)}, C_{0(2bit)}) - 1 & \text{if } S_9 = 1 \\ (C_{1(2bit)}, C_{0(2bit)}) & \text{otherwise} \end{cases}$				
Operations	opBO	A ₂	B ₂	S ₉
Logic 0	0	×	×	0
A NOR B	1	0	0	1
(NOT A)AND B	2	0	1	1
NOT A	3	0	×	1
A AND (NOT B)	4	1	0	1
NOT B	5	×	0	1
A XOR B	6	0	1	1
	6	1	0	1
A NAND B	7	1	1	0
A AND B	8	1	1	1
A XNOR B	9	0	0	1
	9	1	1	1
B	10	×	1	1
(NOT A)OR B	11	1	0	0
A	12	1	×	1
A OR (NOT B)	13	0	1	0
A OR B	14	0	0	0
Logic 1	15	×	×	1

도면11

118

4-bit SEDC Encoded Boolean Operations				
$(C_{2(4bit)}, C_{1(4bit)}, C_{0(4bit)}) = SEDC_{2-in}(A(A_3, A_2, A_1, A_0), B(B_3, B_2, B_1, B_0), opBO)$				
$(C_{1(4bit)}, C_{0(4bit)}) = (C_{1(3bit)}, C_{0(3bit)}) = SEDC_{2-in}(A(A_2, A_1, A_0), B(B_2, B_1, B_0), opBO)$ $C_{2(4bit)} = \begin{cases} 1, & \text{if } S_{10} = 0 \\ 0, & \text{otherwise} \end{cases}$				
Operations	opBO	A ₃	B ₃	S ₁₀
Logic 0	0	×	×	0
A NOR B	1	0	0	1
(NOT A)AND B	2	0	1	1
NOT A	3	0	×	1
A AND (NOT B)	4	1	0	1
NOT B	5	×	0	1
A XOR B	6	0	1	1
	6	1	0	1
A NAND B	7	1	1	0
A AND B	8	1	1	1
A XNOR B	9	0	0	1
	9	1	1	1
B	10	×	1	1
(NOT A)OR B	11	1	0	0
A	12	1	×	1
A OR (NOT B)	13	0	1	0
A OR B	14	0	0	0
Logic 1	15	×	×	1

도면12

121

F		SEDC _{2-out} (F) Case1		SEDC _{2-out} (F) Case2	
F _{1(2bit)}	F _{0(2bit)}	C _{1'(2bit)}	C _{0'(2bit)}	C _{1'(2bit)}	C _{0'(2bit)}
0	0	1	1	1	1
0	1	1	0	0	1
1	0	1	0	0	1
1	1	0	1	1	0

121a 121b 121c

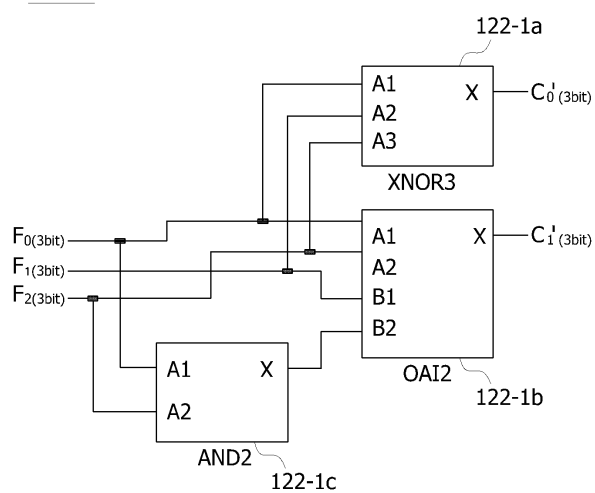
도면13

122

$F_2(3\text{bit})$	$F_1(3\text{bit})$	$F_0(3\text{bit})$	$C_1'(3\text{bit})$	$C_0'(3\text{bit})$
0	0	0	1	1
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	0	0

도면14

122-1



도면15

123

$F_3(4\text{bit})$	$F_2(4\text{bit})$	$F_1(4\text{bit})$	$F_0(4\text{bit})$	$C_2'(4\text{bit})$	$C_1'(4\text{bit})$	$C_0'(4\text{bit})$
0	0	0	0	1	1	1
0	0	0	1	1	1	0
0	0	1	0	1	1	0
0	0	1	1	1	0	1
0	1	0	0	1	1	0
0	1	0	1	1	0	1
0	1	1	0	1	0	1
0	1	1	1	1	0	0
1	0	0	0	0	1	1
1	0	0	1	0	1	0
1	0	1	0	0	1	0
1	0	1	1	0	0	1
1	1	0	0	0	1	0
1	1	0	1	0	0	1
1	1	1	0	0	0	1
1	1	1	1	0	0	0

도면16

123-1

