세포의 자가 치료 기능을 모사한 디지털 회로에서의 오류위치 확인 및 복구 알고리즘

김석환*, 허창우**
*,**목원대학교 전자공학과

An recovery algorithm and error position detection in digital circuit mimicking by self-repair on Cell

Seok-Hwan Kim*, Chang-Wu Hur**
* **Dept. of Electronics Engineering Mokwon University
E-mail : ksh63045@nave.com

요 약

본 연구에서는 세포의 자가 치료 기능을 모사하여 복잡한 디지털 회로를 기능별 분리시킨 구조에서 회로 동작 중 발생하는 오류 위치를 빠르게 찾고 복구 시키는 알고리즘 방법을 제안한다. 디지털 회로를 각 기능별로 9가지로 분리시켜 오류 난 디지털 회로의 기능블록 위치를 빠르게 검출할 수 있게 하며 복구 시키는 방법을 제안한다. 복잡한 구조의 디지털 회로에서도 각 디지털 회로의 기능별 위치에 대한 번호 및 좌표를 3x3 행렬 구조로 확대시켜 오류 위치에 대아여 검출 및 복구가 가능한 알고리즘이다.

Abstract

In this study, we propose an algorithm of the method of recovering quickly find the location of the error encountered during separate operations in the functional structure of complex digital circuits by mimicking the self-healing function of the cell. By the digital circuit was divided by 9 function block unit of function, proposes a method that It can quickly detect and recover the error position. It was the detection and recovery algorithms for the error location in the digital circuit of a complicated structure and could extended the number of function block for the 3x3 matrix structure on the digital circuit.

키워드

Self-repair, Error detection, Bio-inspired Engineering, Recovery algrorithm

Ⅰ. 서 론

생명체의 기본 단위는 세포이다. 세포는 자신이 필요로 하는 화학 물질이나 화학적 활성화에 있어서 매우 다양한 특성을 지닌다. 모든 생물에서 지니는 유전정보, 즉 유전자는 DNA 분자 내부에 저장 되어있다. DNA가 손상을 입을 경우 계속적으로 감시하고 회복하는 세포내 장치가 없다면 생물체의 존재 가치가 무의미해 진다. 세포는 염기쌍을 구조를 가지며 A(아데닌)-T(티민), G(구아닌)-C(시토신)가 틀린 염기쌍 구조가 수정되지 않고 DNA에 축적될 경우 세포가 죽기도 한다.

세포의 기능에 이상이 생겨 죽게 되면 새로운 세포가 발현하여 새로운 세포가 생성된다.[1][4]
본 연구에서는 생물의 기본단위인 세포의 특성을 모사하여 오류결합이 있을 경우 그 위치를 찾아내고 복구[3]하는 방법을 공학적인 방법을 적용한다.

II. 본론

1. 세포 기능 모사한 시스템 모델

본 연구는 세포가 지니는 자가 치료 기능을 모사하여 기능별 분리시킨 전체 디지털 회로를 각 기능별로 9가지로 나누어 디지털 회로에서 동작 중 발생 할 수 있는 오류 위치를 빠르게 찾아내서 정상 회로로 복구시키기 위한 알고리즘이다.
디지털 회로에서 오류 위치를 정확하게 찾아내기 위해서는 아래 그림과 같이 하나의 회로 내용을 회로 기능의 최소 단위로 나누고 이중 모듈화(Double Modular Redundancy, DMR)로 설계를 한다.[2] 그림 1은 오류 검출을 위한 이중 모듈화 블록구조이다.

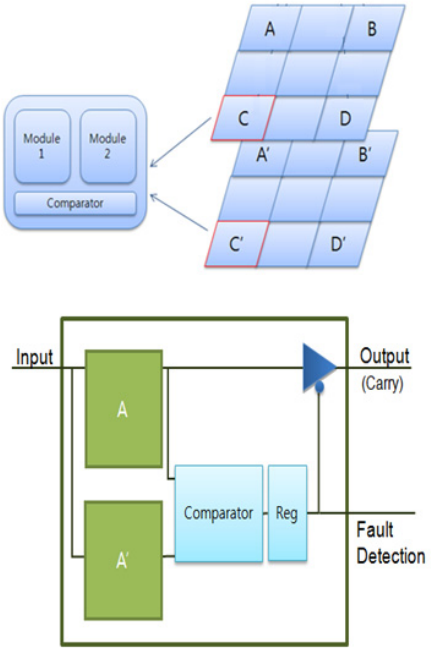

그림 1. 이중 모듈화 블록구조

이중화 모듈 구조로 디지털 회로를 설계하며 모듈에서 출력 단자를 ex-or회로로 상호 연결한다. 출력 값을 비교해 오류 위치를 찾아내는 기본 구조로 설계한다.

2. 시스템 배열

디지털 회로를 기준으로 기능별로 9가지 블록으로 나눈다. 본 연구에서는 9가지 블록에 대하여 아래 그림2와 같이 모듈 번호를 설정하였다. 1부터 9까지의 숫자를 배열할 경우 가로, 세로, 대각선의 어느 곳에서든지 더할 경우 15가 되도록 번호 배열을 하였다.

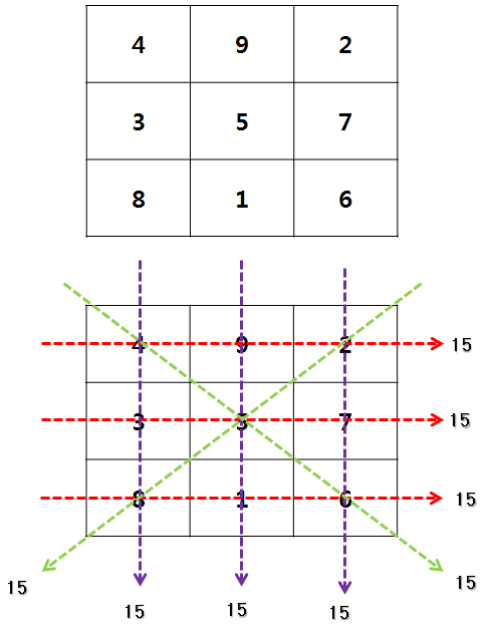

그림 2. 디지털 회로 기능별 분류

정상적으로 동작하는 회로가 있을 경우 출력단의 값이 다르게 됨을 인지할 경우 오류 위치를 빠르고 정확하게 찾아내며 그 위치의 회로부분의 출력을 차단시켜야 한다. 오류 위치 정보는 모듈의 기능을 조절하는 상위 모듈에 정보가 전달되고 오류 모듈의 출력을 차단시킨다. 그림 2는 디지털회로를 세분화 시켰을 때 하나의 모듈 배열을 나타내며 각 모듈의 번호를 합해서 합이 15가 되는 모듈의 번호 관계를 보여준다.

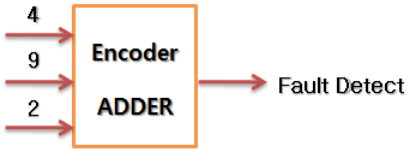

그림 3. 오류검출 모듈 배열

그림 3은 오류 검출관련 구현 블록구조이고 그림 4는 현재 제안한 회로 구성에 대한 오류 검출 모듈의 배열이다. 전체 시스템에 대한 디지털 회로는 전체 공통적으로 이 규칙을 지니며 오류난 모듈의 위치를 찾아낼 수 있다.

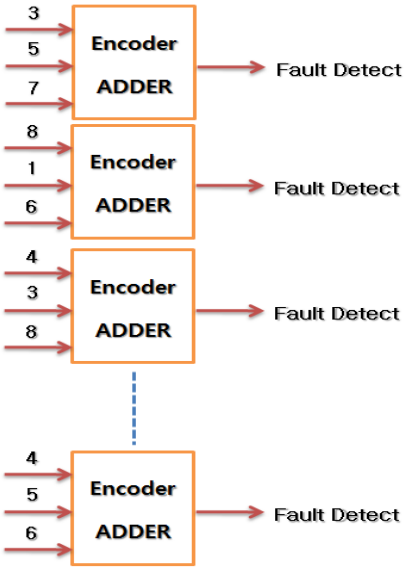

그림 4. 제안한 회로구성의 오류검출

Ⅲ. 오류 검출

1. 오류 검출 순서

정상적으로 동작하던 모듈에서 오류가 발생할 경우 가장 오류를 찾는 가장 기본적인 방법은 앞서 설명한 바와 같이 회로의 모듈을 이중화 구조(DMR)로 설계를 하는 것이다. 본 연구는 세포의 자가 복구 기능을 모사한 알고리즘이므로 세부적으로 기능을 나눈 블록단위별로 오류를 검출한다.

오류검출의 순서는 좌측 상부에 있는 모듈 4번을 기준으로 오류 검출 번호를 정한다. 먼저 모듈 4번을 기준으로 가로방향으로 위에서 아래로 세 가지, 세로방향으로 좌에서 우로 세 가지, 대각선 방향으로 좌측에서 우측으로 한 가지, 대각선 방향으로 우측에서 좌측으로 한 가지 총 8가지 검출방법이 있다.

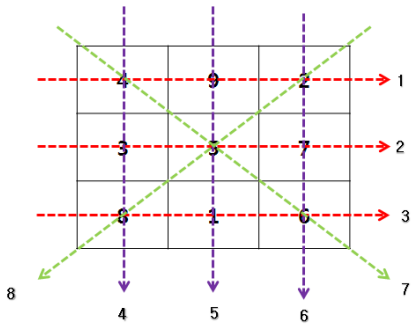

그림 5.오류 검출순서

2. 오류 검출 방법

이중화 구조로 설계된 각 모듈은 출력 단에 EX-OR Gate로 연결을 한다. 상호 출력 값이 같으면 “0”, 다르면 “1” 값이 검출되도록 설계한다. 세부적으로 나눈 모듈의 결과 값이 정확한지 확인하기 위한 이중화 구조에서 비교 대상은 똑같이 설계된 Master를 기준으로 한다.[5] 오류검출을 위한 회로 설계는 아래 그림 6과 같다. 여기서 모듈 4번을 기준으로 1번 방향을 기준으로 설명한다.

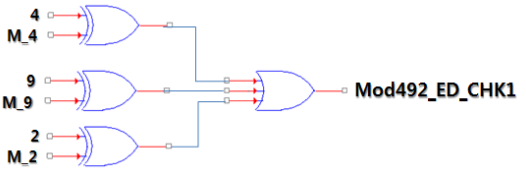

그림 6. 모듈4,9,2에 대한 오류검출 이중화 설계

여기서 M은 Master를 의미하고 Mod492(Module 4번, 9번, 2번), ED(Error Detect), CHK1(Check 1)을 의미한다. 오류가 발생한 모듈4을 기준으로 검출 방법은 3가지(가로방향-Mod492_ED_CHK1, 세로방향 -Mod438_ED_CHK4, 대각선방향 -Mod456_ED_CHK7)이므로 이 세 가지 비트에서 오류가 발생한다.

오류가 난 모듈의 위치를 찾을 수는 있지만 본 연구에서 제안하는 세포 기능의 모사는 모듈상의 비트 오류를 찾아야 한다. 그 방법은 다음과 같다. 각 모듈이 9까지 번호를 부여 되어있으므로 각 모듈 번호에 대하여 Bit 단위로 표현한다.

표 1. 모듈번호의 비트 표현

모듈 번호	Bit(4bit)
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001

오류가 발생한 모듈에서 어느 위치의 비트가 오류인지 검출 하는 방법은 각 비트별로 더하면 모두 “1” 이 되고 Carry가 발생하지 않으므로 이에 대한 진리표와 회로구성은 다음과 같다.

표 2. 모듈 4,9,2에 대한 가로방향 Sum과 Carry

Module 이름	Bit 4	Bit 3	Bit 2	Bit 1	Sum				Carry			
					S4	S3	S2	S1	C4	C3	C2	C1
Module 4	0	1	0	0	1	1	1	1	0	0	0	0
Module 9	1	0	0	1								
Module 2	0	0	1	0								

표 3. 모듈 4,3,8에 대한 세로방향 Sum과 Carry

Module 이름	Bit 4	Bit 3	Bit 2	Bit 1	Sum				Carry			
					S4	S3	S2	S1	C4	C3	C2	C1
Module 4	0	1	0	0	1	1	1	1	0	0	0	0
Module 3	0	0	1	1								
Module 8	1	0	0	0								

표 4. 모듈 4,5,6에 대한 대각선 방향 Sum과 Carry

Module 이름	Bit 4	Bit 3	Bit 2	Bit 1	Sum				Carry			
					S4	S3	S2	S1	C4	C3	C2	C1
Module 4	0	1	0	0	1	1	1	1	0	0	0	0
Module 5	0	1	0	1								
Module 6	0	1	1	0								

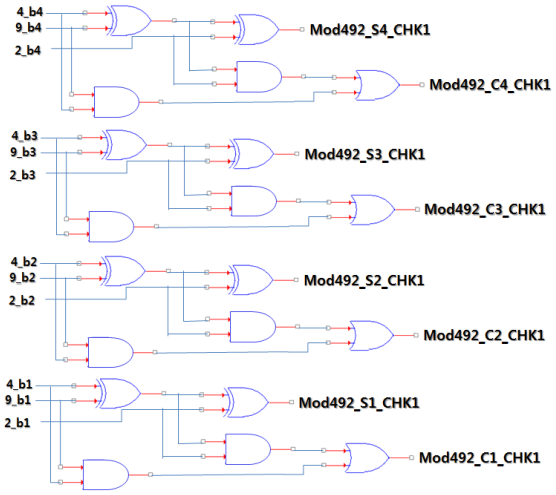

그림 7. 모듈4의 오류 검출 위한 디지털 회로 표현

모듈 4에 대한 오류 검출은 상기 진리표와 같이 3가지 방법으로 표현이 가능하며 각 비트별로 회로를 구성하게 되면 어느 모듈의 어느 비트 값에 오류가 발생했는지 정확하게 판단할 수 있다. 만일 모듈 4의 비트 4번에서 오류가 발생하여 정상 값 “0”이 “1”로 바뀌었다면 아래의 표와 같이 바뀌게 된다.

표 5. 오류가 발생한 모듈 4에서 모듈 4,9,2에 대한 가로방향 Sum과 Carry

Module 이름	Bit 4	Bit 3	Bit 2	Bit 1	Sum				Carry			
					S4	S3	S2	S1	C4	C3	C2	C1
Module 4	1	1	0	0								
Module 9	1	0	0	1	0	1	1	1	1	0	0	0
Module 2	0	0	1	0								

표 6. 오류가 발생한 모듈 4에서 모듈 4,9,2에 대한 세로방향 Sum과 Carry

Module 이름	Bit 4	Bit 3	Bit 2	Bit 1	Sum				Carry			
					S4	S3	S2	S1	C4	C3	C2	C1
Module 4	1	1	0	0								
Module 3	0	0	1	1	0	1	1	1	1	0	0	0
Module 8	1	0	0	0								

표 7. 오류가 발생한 모듈 4에서 모듈 4,9,2에 대한 대각선 방향 Sum과 Carry

Module 이름	Bit 4	Bit 3	Bit 2	Bit 1	Sum				Carry			
					S4	S3	S2	S1	C4	C3	C2	C1
Module 4	1	1	0	0								
Module 5	0	1	0	1	0	1	1	1	1	0	0	0
Module 6	0	1	1	0								

Ⅳ. 결론

본 연구에서는 세포의 자가 치료 기능을 모사하여 복잡한 디지털 회로를 기능별 분리시킨 구조에서 회로 동작 중 발생하는 오류 위치를 빠르게 찾고 복구 시키는 알고리즘 방법을 제안했다. 전체 시스템에 대한 디지털 회로의 동작 상태를 파악하기 위한 이중 모듈화 (Double Modular Redundancy, DMR)방법을 이용하여 구성하였으며 오류가 발생한 위치를 확인하는 기능을 가진다.

디지털 회로를 각 기능별로 9가지로 분리시켜 오류 난 디지털 회로의 기능블록 위치를 빠르게 검출할 수 있게 하며 복구 시키는 방법을 제안한다. 복잡한 구조의 디지털 회로에서도 각 디지털 회로의 기능별 위치에 대한 번호 및 좌표를 3x3 행렬 구조로 확대시켜 오류 위치에 대하여 검출 및 복구가 가능한 알고리즘이다.

본 연구에서는 추후 다른 방법에서는 이런 부분을 보안하는 새로운 알고리즘 방법을 제시하고자 한다.

참고 문헌

[1] 김석환, 허창우, “DNA 이중나선 구조에서의 오류 검출 및 복구”,한국정보통신학회논문지, vol.15,no.11,pp255-2504,2011.11

[2] Will Barker, David M. Halliday, Yann Thoma, Eduardo Sanchez, Gianluca Tempesti, and Andy M. Tyrell, “Fault Tolerance Using Dynamic Reconfiguration on the POEtic Tissue,” IEEE Trans. Evol. Comput., vol.11, no.5, pp. 666-684, Oct. 2007.

[3] Asso Yamauchi Toshihiro, Nakashima, “Evolvability of random polypeptides through functional selection within a small library“, Protein engineering vol.15 no.17,pp619-626,2002.

[4] Alberts, Bray저, 박상대 역 “필수 세포생물학”,교보문고, 2010.

[5] Garrison W. Greenwood, “ On the Practicality of Using Intrinsic Reconfiguration for Fault Recovery, “ IEEE Trans. Evol. Comput., vol. 9, no.4, pp. 398-405, 2005.