세포기능을 모사한 자가복구 알고리즘연구

김석환*, 허창우**
*, **목원대학교 전자공학과

A Study on Self Repairing Algorithm inspired of Cell

Soke-Hwan Kim*, Chang-Wu Hur**
*, **Mokwon University

요 약

살아있는 세포는 세포의 주기에 따라 생성되거나 사멸되는 과정을 반복하게 된다. 이 기간 동안 오류가 발생하면 세포가 생명을 유지하기 위해서는 주변 세포에서 오류가 부분을 찾아서 그 세포를 사멸시킬 수 있는 과정을 거치게 된다. 이런 세포 기능을 모사하여 회로 시스템에 적용하였다. 회로가 기능을 유지하다가 오류가 발생하였는지 확인하기 위해서는 DMR(Double Modular Redundancy)로 구현된 회로에 의해서 위치를 파악하고 빠르게 복구할 수 있도록 같은 회로를 구현하였다. 본 연구에서는 카운터를 구현하였으며 임의의 부분에 오류가 발생하도록 회로를 구성, 오류위치를 찾고 정상적으로 동작할 수 있도록 하였다.

Ⅰ. 서 론

컴퓨터와 통신기술의 발달로 인하여 정보의 양이 증가했을 뿐만 아니라 이를 뒷받침하기 위한 반도체 소자의 고속화 및 대용량을 요구되고 있다. 데이터를 에러없이 빠르고 정확하게 저장하고 전달하기 위한 회로의 동작의 유지 시간 및 데이터에[대한 신뢰성을 요구되는 시점이다.

이런 특성을 맞추기 위해서는 수백 Mbps에서 수 Gbps에 이르기까지 고속 데이터 전송을 위한 통신 선로 시스템에서의 중요성이 커지지만 본 연구에서는 시스템의 기초 부분에서 기본 로직을 구성하여 어떤 오류가 발생하였을 경우 정확히 그 부분을 찾아내고 정상적인 상태로 돌리는 방법을 찾아내고자 하였다.

어느 회로에 오류가 발생하였을 경우 그 부분의 위치를 찾아내고 테스트하는 과정을 거치게 되는데 본 연구에서는 생물학적인 세포의 특성을 모사하여 디지털 회로에 적용하고자 하였으며 간단한 카운터의 동작을 설계를 통해 알아보았다. 어느 특정부분에 대한 회로에 있어서 오류가 발생하는 것을 확인하는 방법은 회로의 이중화 구조를 통해 데이터를 비교하는 DMR(Double Modular Redundancy)가 사용 되거나 출력 데이터의 비교를 Majority를 이용하는 TMR(Triple Modular Redundancy) 방법을 사용한다. 그러나 여기서는 특정 오류가 나는 부붑에 대한 예측 및 이중화 삼중화 구조를 설계를 위한 회로의 복잡성 및 규모가 커질 수 있다. 그래서 기존의 방법을 이용하면서도 효율적으로 복구를 하고 더 나아가 세포기능의 중요한 특성인 자가 복구(Self-Repairing), 분화(Differentiation) 특성을 이용할 수 있는 새로운 알고리즘 방법을 제안 하고자 한다. 구현된 모델은 간단한 카운터의 예제를 가지고 simulation을 통해 특성을 알아보았다.

Ⅱ. 본론

1. Simulation 프로그램 구조

Simulation에서는 ALTERA Quartus II를 이용하여 기본 카운터 예제를 설계하였으며 임의의 시점에 오류가 발생하였을 경우 정확히 그 오류난 부분을 찾아내고 정상적으로 동작하는 회로가 그 기능을 대체 할 수 있는지 확인한다.

2. 오류 검출 및 복구

기존 연구 방향과는 다르게 하나의 회로 내용을 전체적으로 보는 것이 아니라 회로 기능의 최소 단위로 나누었으며 이는 오류 나는 부분에

대한 위치를 정확히 찾아내며 빠르게 복구 할 수 있는 방법을 제시하고자 하는 것이다. 기본 회로에 대한 오류 검출은 DMR 방식을 채택하였는데 하나의 최소 회로는 세 개의 Redundancy 모듈을 가지고 있다. 그러므로 초기 상태는 원래 동작하고 있는 회로와 세 개의 Redundancy에 대한 출력을 비교하는 시스템 구조이며 만일 회로에 오류가 발생한다면 오류 회로를 제외한 정상동작 모듈과 두 개의 Redundancy 모듈이 있게 되며 다시 오류가 발생하면 결국 두 개의 모듈에 대한 출력을 비교하는 회로 구조가 되는 것이다 이는 기존의 TMR 구조와 DMR보다는 더 복구 시간 및 오류 검출에 대한 신뢰성을 높일 수 있다는 장점을 가지게 된다.

III. 데이터 오류 검출 프로그램 구현

1. 전체 시스템 블록도

ALTERA Quartus II를 이용하여 기본 카운터 구조를 설계 하였으며 임의 오류를 발생하는 신호를 삽입하였을 경우 정확히 오류 검출을 하는지 그리고 언제 복구가 시작되는지 타이밍 관계를 확인하는 구조로 설계 되었다.

세부적인 회로 단위로 설계를 하였으며 각 세부 회로 단위는 세 개의 Redundancy를 가지고 있다. 각 회로에 대한 입력은 동시에 받도록 구성 되며 단 출력에 있어서 선택된 회로에서만 출력 하도록 설계하였다. 추후 연구가 되어야 하는 것은 만인 Redundancy회로 모두에서 오류가 발생하였다면 더 이상 동작을 하지 않는 것이 아니라 세포에서처럼 새로운 영역에서 이 기능을 수행하도록 같은 회로를 분화시켜서 정상적으로 동작하도록 하는 방향으로 전개를 해야 한다. 그러나 이 방식은 생물학적인 기능을 모사하였을 지라도 하드웨어적인 의존성이 매우 강하게 되는 현상이 일어날 수 있으므로 해결해야 하는 문제점이 있다고 볼 수 있다.

2. 내부 카운터 구현

기본 사용되는 프로그램은 3비트 카운터를 기준으로 설계하였으며 8비트의 임의의 데이터와 데이터를 비교하기 위한 8비트 레지스터를 가지고 있는 회로로 하였다.

임의 하나의 카운터에서 오류가 발생하였다면 오류에 대한 카운터의 동작이 어떻게 바뀌는 지 알아보기 위한 프로그램 구조로 되어있다.

IV. 카운터 동작 및 오류 발생 시 파형

아래의 파형들은 앞에서 구현된 프로그램에서 카운터의 일반 동작과 오류에 따른 특성을 살펴보았다. 기본 카운터의 동작을 수행하다 오류 비트가 발생하면 오류난 회로는 차단되고 다른 정상적인 회로로 절체되어 동작되는 것을 시뮬레이션을 통해 볼 수 있다.

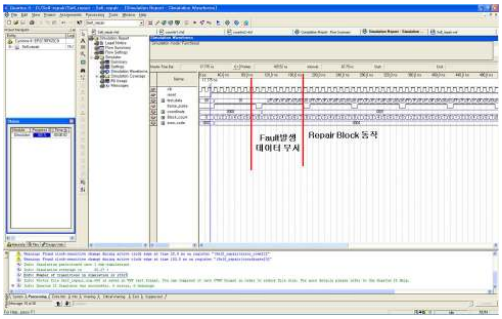

그림 4-1. 데이터 오류에 따른 특성

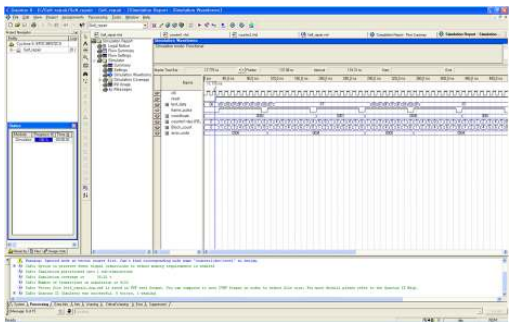

그림 4-2. 오류 회로에 대한 정상회로 절체

3비트 카운터에 대한 동작에서 임의의 비트 오류가 발생을 하면 본 시뮬레이션에서 바로 출력을 차단하는 결과를 얻을 수 있었다. 또한 정상적인 동작을 하는 Redundancy회로로 절체를 하는 것을 보았다. 그러나 Redundancy 회로의 오류에 대한 최종적인 내용은 세포에서 기능을 모사하는 분화기능을 적용하여 보다 더 효율적으로 복구하도록 한다.

Ⅴ. 결론

본 연구에서 실시한 알고리즘 연구는 디지털 회로에서 동작 시 발생하는 오류에 대하여 오류 위치를 정확히 찾아내고 바로 정상적인 상태로 복구 할 수 있도록 하는 방법을 제안하였다. 기존의 연구와의 차이점은 전체적인 회로에 대한 Redundancy 구조가 아니라 세부적인 기능을 나누어서 오류 부분에 대한 정확한 위치를 찾아내고 복구하고자 하는 부분을 최소화 하여 효율적인 시스템의 관리 방향성을 제시하는 것이다.

신경세포가 가지는 오류 복구와 분화에 대한 특성을 시스템에 적용하게 되면 오류에 대한 On-Line Repair 기능이 가능하게 되며 Redundancy 구현에 따른 회로의 면적이 최소화 할 수 있도록 가능케 될 것이다. 본 연구를 통해서 오류에 대하여 정확히 그 위치를 찾아서 출력을 차단 시키며 다른 정상적인 회로로 바로 절체 됨을 알 수 있었다. 추후 필요한 연구는 앞서 설명한 세포 기능을 모사하는 분화방법과 오류가 발생하였을 경우 일시적인 오류인지 영구적인 오류인지 판단 할 수 있는 자가 진단 프로그램을 구현해야 할 것이다.

참고 문헌

[1] 김석환, 이규정, 허창우 “통신 시스템의 데이터 전송 선로에 대한 연구,”한국 해양 정보통신학회 논문지 제 9권 6호, pp. 1277-1281, 2005년 10월

[2] 서정일, 성낙훈, 오택진, 양현모, 최호용, " 자가검출회로 내장의 자가치유시스템 설계, " 전자공학회 논문지 제42권 SD 제5호, pp. 307-313, 2005년 5월.

[3] Bingfeng Mei, Andy Lambrechts, jean-Yves Mignolet, "Architecture Exploration for a Reconfigurable Architecture Template," IEEE Design & Test of Computers, pp. 90-101, 2005.