Journal of Institute of Control, Robotics and Systems (2014) 20(6):625-630
http://dx.doi.org/10.5302/J.ICROS.2014.14.0006 ISSN:1976-5622 eISSN:2233-4335

이중화 구조를 이용한 비동기 디지털 시스템의 방사선 고장 극복

A New Hardening Technique Against Radiation Faults in Asynchronous Digital Circuits Using Double Modular Redundancy

곽 성 우, 양 정 민*
(Seong Woo Kwak[1] and Jung-Min Yang[2,*])
[1]Department of Electronic Engineering, Keimyung University
[2]School of Electronics Engineering, Kyungpook National University

Abstract: Asynchronous digital circuits working in military and space environments are often subject to the adverse effects of radiation faults. In this paper, we propose a new hardening technique against radiation faults. The considered digital system has the structure of DMR (Double Modular Redundancy), in which two sub-systems conduct the same work simultaneously. Based on the output feedback, the proposed scheme diagnoses occurrences of radiation faults and realizes immediate recovery to the normal behavior by overriding parts of memory bits of the faulty sub-system. As a case study, the proposed control scheme is applied to an asynchronous dual ring counter implemented in VHDL code.

Keywords: asynchronous digital circuits, fault diagnosis and tolerance, radiation faults, DMR (Double Modular Redundancy)

I. 서론

비동기 디지털 시스템(asynchronous digital system)이란 전역 클럭(clock) 없이 입력의 변화만으로 상태가 바뀌는 순차 머신(sequential machine)을 통칭한다. 비동기 디지털 시스템은 동기 시스템에 비해 설계하기가 더 어렵다는 단점이 있지만 저전력(low power)이 요구되거나 보다 빠른 과도 상태 천이 속도가 필요한 시스템의 핵심 모듈로서 여전히 많이 사용되고 있다[1].

이번 논문에서는 방사선 고장(radiation fault)에 영향을 받는 비동기 디지털 시스템을 위한 새로운 고장 극복 방법을 제안한다. 군사용, 우주용 등의 용도로 제작된 디지털 시스템은 방사선이 존재하는 환경에서 노출되는 경우가 많이 발생한다. 그런데 실리콘 기반으로 제작된 모든 반도체 제품은 방사선이 시스템에 누적되어 생기는 여러 가지 방사선 고장의 공격을 피하지 못한다[2]. 문제는 그러한 환경 속에서 고장이 발생한 후 인간이 개입하지 않고 해당 시스템을 즉시 정상 상태(normal status)로 복구시키기가 쉽지 않다는 점이다. 이렇듯 방사선 환경 하에서 견실한 고장 탐지 및 복구(fault diagnosis and tolerance)를 구현하는 일은 디지털 시스템의 성공적인 작업 수행을 위해서 반드시 해결되어야 할 주제이다.

비동기 디지털 시스템에 대한 고장 극복 기능을 구현한 과거의 연구들은 내고장성을 고려한 회로 설계에 대한 것들이 주를 이루었다[3,4]. 본 논문에서는 설계보다는 제어의 관점으로 방사선 고장을 극복하는 연구를 다룬다. 고려하는 디지털 시스템에 내고장성을 부여하기 위해서, 시스템은 이중화(DMR: Dual Modular Redundancy) 구조를 가진다고 설정한다. DMR은 동일한 작업을 하는 두 개의 부(副)시스템이 존재하는 하드웨어 여유도(hardware redundancy) 구조의 한 예이다[5]. 시스템이 재설계 없이 내고장성을 가지기 위해서는 이러한 여유도가 반드시 있어야 한다. DMR은 세 개의 부시스템이 존재하는 TMR (Triple Modular Redundancy) [6]보다 시스템 부하(load) 소모 측면에서 더 우수하다.

본 논문에서 다루는 비동기 시스템은 카운터(counter) 형태의 비동기 순차 머신이며, 입력이 시스템 상태와 다른 값을 가지는 입력/출력 형태이다. 또한 내고장성 모듈로서 두 개의 부시스템이 자신의 상태 변수 일부를 다른 부시스템의 해당 상태 변수에 덮어 쓰는(overriding) 로직을 구성한다. 통상 DMR 구조의 시스템에서는 고장이 발생한 부시스템의 모든 상태 변수를 정상적으로 동작하는 다른 부시스템의 모든 변수로 한꺼번에 치환시키는 방법으로 고장 극복 모듈을 꾸민다[7]. 하지만 본 연구에서는 상태 변수 일부만을 치환시키는 기법을 사용하기 때문에 기존 연구보다 더 효율적이다. 또한 고장 극복 과정은 비동기 시스템을 위한 교정 제어(corrective control) 이론[8] 하에서 구현

* Corresponding Author
Manuscript received January 14, 2014 / revised April 14, 2014 / accepted April 14, 2014
곽성우: 계명대학교 전자공학과(ksw@kmu.ac.kr)
양정민: 경북대학교 전자공학부(jmyang@ee.knu.ac.kr)
※ 이 논문은 2012년도 정부(미래창조과학부)의 재원으로 한국연구재단의 기초연구사업 지원을 받아 수행된 것임 (No. NRF-2012R1A2A2A01003419). 이 논문은 2010년도 정부(교육부)의 재원으로 한국연구재단의 기초연구사업 지원을 받아 수행된 것임 (No. NRF-2010-0007271). 이 논문은 2010년도 정부(교육부)의 재원으로 한국연구재단의 기초연구사업 지원을 받아 수행된 것임 (No. NRF-2010-0021366).

되므로 즉각적인 고장 복구가 가능하다.

교정 제어를 이용하여 방사선 고장을 극복하는 주제는 저자의 선행 연구[9]에서 다루어졌다. [9]와 비교하여 본 논문이 보이는 차별성은 다음과 같다.

1) [9]에서 다룬 고장은 총이온화선량(TID: Total Ionization Dose)에 의한 고장으로 디지털 시스템의 상태 천이 특성을 영구적으로 바꾸는 종류였다. 하지만 본 논문에서는 과도 고장(transient fault) [5], 즉 방사선의 영향으로 시스템의 메모리 비트 값이 일시적으로 바뀌는 고장을 연구한다.
2) [9]에서는 제어 대상 디지털 시스템이 하드웨어 여유도를 갖추고 있지 않다고 가정하고 모든 고장 탐지와 극복 과정을 교정 제어기가 구현하였다. 반면 이번 논문에서는 두 개의 부시스템 사이에 각자의 상태 값을 다른 부시스템의 해당 상태에 덮어쓰는 로직이 있다고 설정한다. 즉 교정 제어기는 이러한 내고장성 로직을 운용하는 제어 입력 스트링(string)을 결정하는 역할을 한다.

본 논문에서는 먼저 DMR 구조로 이루어진 비동기 디지털 시스템을 모델링하고 방사선의 영향으로 발생하는 과도 고장을 탐지하고 극복하는 알고리듬을 교정 제어의 틀에서 제안한다. 또한 제안된 고장 극복 모듈의 성능을 검증하기 위해서 DMR 구조로 된 비동기 링(ring) 카운터를 VHDL 코드로 구현하고 주어진 고장 시나리오 하에서 고장 탐지 및 극복 실험을 실시한다.

II. DMR 구조를 가진 비동기 디지털 시스템

본 논문에서는 회로의 출력이 현재 상태와 다른 입력/출력 비동기 디지털 시스템을 다룬다. 입력/출력 비동기 디지털 시스템 Σ를 다음과 같이 모델링하자.

$$\Sigma = (A, Y, X, x_0, f, h)$$

A, Y, X는 각각 입력, 출력, 상태 집합이고 $x_0 \in X$은 초기 상태이다. $f: X \times A \rightarrow X$는 상태 천이 함수이며 $h: X \rightarrow Y$는 출력 함수이다. 전역 클럭이 없는 비동기 디지털 시스템의 상태 천이는 입력의 변화에만 의존한다. 상태/입력 조합 (x,v)가 안정(stable) 조합이면 Σ는 입력이 바뀌지 않는 한 현재 상태 x에 계속 머무른다. (x,v)가 과도(transient) 조합이면 Σ는 $x_1:=f(x,v)$, $x_2=f(x_1,v)$,... 등으로 여러 개의 과도 상태를 거치며 $x'=f(x',v)$를 이루는 '다음 안정 상태(next stable state)' x'로 천이한다[10]. 비동기 시스템에서 과도 상태 천이는 매우 빠르므로 안정 상태 사이의 과도 조합은 무시할 수 있다. 과도 조합을 뺀 안정 상태 사이의 천이만을 표현하기 위해 'stable recursion 함수' s를 다음과 같이 정의한다[10].

$$s: X \times A \rightarrow X,\ s(x,v)=x',\ x,x' \in X,\ v \in A$$

s(x,v)는 (x,v)에 있던 Σ가 이동하여 도달하는 다음 안정 상태 x'를 출력한다. 명확한 이론 기술을 위해 본 논문에서는 Σ가 f(x,v)=s(x,v)인 성질, 즉 모든 상태 천이 내에서 과도 조합이 없는 특성을 가진다고 설정한다. 단위 입력 대신 입력 스트링을 s의 변수로 설정하면 다음과 같이 일반화할 수 있다.

$$s(x,v_1v_2\cdots v_k) = s(s(x,v_1),v_2\cdots v_k),\ x \in X,\ v_1v_2\cdots v_k \in A^+\ (k \geq 2)$$

위 식에서 A^+는 A에 속한 알파벳으로 이루어지는 길이 1 이상의 스트링 집합을 말한다.

DMR 구조를 가지는 Σ는 정상 동작에서 동일 작업을 수행하는 부시스템 두 개로 구성된다. 부시스템을 P와 Q로 명명하고 각 부시스템이 n 비트(bit)의 상태를 가진다고 가정하면 Σ의 상태는 아래와 같이 2n 비트로 표현 가능하다.

$$X = \{(p_1p_2\cdots p_n,\ q_1q_2\cdots q_n) |\ p_i,q_i \in \{0,1\}\}$$

입력 집합 A를 정상 입력(normal input) 집합 A_n과 고장 입력(fault input) 집합 A_d로 나누면

$$A=A_n \cup A_d,\ A_n \cap A_d = \varnothing$$

로 표현된다. A_n과 천이 함수 s의 자세한 명세는 Σ의 역할에 따라 결정된다. 서론에서 밝혔듯이 Σ가 카운터 작업을 한다고 하자. 또한 카운터 종류 중 정상 동작에서 로직 1 비트 값이 하나만 존재하여 순환하는 링 카운터로 문제를 국한시킨다. 링 카운터의 초기 상태 x_0에서 로직 1 값이 첫번째 비트 위치에 있어야하므로 $x_0=(10\cdots0,10\cdots0)$이다.

카운터에서 정상 입력의 구분을 다양하게 할수록 고장 극복 제어의 자유도는 더 커진다. 하지만 입력 구분을 위한 시스템 부하도 함께 증가한다. 이번 연구에서는 로직 1 값이 홀수와 짝수 번째 상태 비트에 각각 위치할 때에만 입력을 구분하는 최소한의 카운팅 입력을 정의한다. 'o'(odd)와 'e'(even)를 해당 카운팅 입력이라고 명하자. x_0과 그 다음 상태에서 o와 e에 대한 천이 함수 s는 아래와 같이 정의된다.

$$s((1000\cdots0,1000\cdots0),o) = (0100\cdots0,0100\cdots0)$$
$$s((0100\cdots0,0100\cdots0),e) = (0010\cdots0,0010\cdots0)$$

위 식에서 보듯이 o는 홀수 번째 위치에 있는 P와 Q의 로직 1 값을 다음 짝수 번째 위치로 전진시키며, e는 짝수 번째 위치에서 다음 홀수 번째 위치로 전진시킨다.

Σ에 내고장성을 부여하기 위해 P와 Q의 일부 상태 비트를 다른 부시스템의 값으로 치환시키는 입력을 정의한다. 카운터의 특성상 치환 입력을 주고받는 상태 비트의 위치를 등(等)간격으로 한정할 수 있다. 예를 들어 간격이 2라면 P와 Q의 모든 짝수 또는 홀수 비트 값을 변경할 수 있으며, 배수가 3이라면 1, 4, 7,...번째(또는 2, 5, 8,...번째) 비트 값을 변경할 수 있다. 간격이 커질수록 고장 복구 능력 및 속도는 더 떨어진다. 이번 연구에서는 P와 Q의 모든 홀수 번째 상태 비트를 바꾸는 입력을 사용한다. c_p와 c_q를 그러한 입력이라고 정의하면 c_p와 c_q에 대해서 다음과 같은 동작이 수행된다(편의상 n이 홀수라고 가정한다).

$$c_p:\ q_i \rightarrow p_i,\ i=1,3,...,n$$
$$c_q:\ p_i \rightarrow q_i,\ i=1,3,...,n$$

c_p는 P의 모든 홀수 번째 상태 비트를 Q의 해당 상태 비트의 값으로 덮어쓰는 명령이며, c_q는 Q의 홀수 비트를 P의 해당 비트 값으로 덮어쓰는 명령이다. 이것은 P와 Q 중 어느 한 부시스템에서 고장이 발생하면 정상적으로 동작하는 다른 부시스템의 값으로 고장 난 부분을 정상 값으로 리셋(reset) 시키는 기능을 구현한다. 하지만 상태 비트의

일부만을 바꾼다는 점에서 고장 난 부시스템의 모든 상태 비트 값을 리셋시키는 기존 방법[7]과 비교하여 차별성을 가진다. 정상 상태 입력 집합 A_n을 종합하면 다음과 같다.

$$A_n = \{o,\ e,\ c_p,\ c_q\}$$

방사선 고장이 시스템에 발생하면 방사선의 고에너지 입자가 반도체 메모리에 저장된 로직 값을 반전시키는 SEU(Single Event Upset) 현상이 일어난다[2]. P와 Q의 모든 상태 비트가 0 또는 1의 값을 가질 때 이러한 고장이 각각 발현될 수 있으므로 방사선 고장 입력 수는 총 4n이다. 비동기 디지털 시스템에서 두 개 이상의 비트에서 동시에 SEU가 일어나는 경우는 극히 드물기 때문에 방사선 고장의 동시 발현은 제외시킨다. 본 연구에서는 방사선 고장 입력을 다음과 같이 표기한다.

$$A_d = \{z_P[i], n_P[i], z_Q[i], n_Q[i] \mid i=1,2,...,n\}$$

$z_P[i]$는 P의 i번째 상태 비트의 값이 방사선에 의해서 0('z'ero)에서 1로 바뀌는 고장을 의미하며, $n_P[i]$는 1(o'n'e)에서 0으로 바뀌는 고장을 의미한다. 또 $z_Q[i]$와 $n_Q[i]$는 Q의 i번째 상태 비트에서 일어나는 고장을 가리킨다. 예를 들어 초기 상태 x_0=(10…0,10…0)에서 P의 첫번째와 두번째 상태 비트 p_1과 p_2에 각각 고장이 발생하는 경우를 A_d의 원소와 천이 함수 s로 표현하면 아래와 같다.

$$s((100\cdots0,100\cdots0), n_P[1]) = (\mathbf{0}00\cdots0,100\cdots0)$$
$$s((100\cdots0,100\cdots0), z_P[2]) = (1\mathbf{1}0\cdots0,100\cdots0)$$

다른 고장 입력이 일으키는 고장 상태 천이도 위와 유사하게 표현된다.

다음으로 Σ의 출력 함수 h를 정의한다. 이번 연구에서는 내고장성 모듈과 마찬가지로 Σ의 출력도 최대한 단순화시켜서 이용하는 것을 목적으로 한다. 등간격으로 정의한 상태 비트 치환 명령 로직에 부합하는 출력을 내기 위해서는 등간격을 가지는 상태 비트들에 대한 패리티(parity) 비트를 하나의 출력 비트로 정의하면 된다. 앞에서 정의한 대로 Σ는 부시스템 P와 Q의 모든 홀수 비트를 리셋시키는 입력 c_p와 c_q를 가지기 때문에 우리는 각 부시스템의 홀수 비트와 짝수 비트에 대한 패리티 비트를 출력으로 정의한다. 만약 리셋 입력이 정의된 상태 간의 등간격이 3이라면 (1, 4, 7, ...) 비트, (2, 5, 8, ...) 비트, (3, 6, 9, ...) 비트에 대한 패리티 비트를 각각 정의해야 한다. 앞의 기술을 바탕으로 Σ의 출력 집합 Y를 다음과 같이 정의한다(n은 홀수).

$$Y = \{(y_1y_2,y_3y_4) \mid y_1,y_2,y_3,y_4 \in \{0,1\}\}$$
$$y_1 = p_1 \oplus p_3 \oplus \cdots \oplus p_n,\ y_2 = p_2 \oplus p_4 \oplus \cdots \oplus p_{n-1}$$
$$y_3 = q_1 \oplus q_3 \oplus \cdots \oplus q_n,\ y_4 = q_2 \oplus q_4 \oplus \cdots \oplus q_{n-1}$$

y_1과 y_2는 P의 홀수 번째와 짝수 번째 상태 비트 값을 각각 exclusive-OR(⊕)한 로직 값이며, y_3과 y_4는 Q에 대해서 동일한 연산을 취한 로직 값이다. 이 정의는 시스템의 상태 비트 수 2n과 무관하다는 점을 주목해야 한다. 다시 말하면 위 설정은 시스템의 상태 차원(dimension)이 증가해도 출력 비트 수와 일정하게 유지시키는 장점을 지닌다.

III. 방사선 고장 극복

1. 고장 탐지

Σ가 방사선 고장에 대한 고장 극복 능력을 가지기 위해서는 먼저 고장 발생을 탐지할 수 있어야 한다. 그런데 앞장에서 정의한 출력 비트의 변화를 관측하면 임의의 상태 비트에서 발생하는 방사선 고장을 모두 탐지할 수 있다. Σ가 정상 동작을 할 때는 P와 Q의 상태 비트 중 한 곳에서만 로직 1을 가지며 나머지는 모두 0을 가진다. 따라서 Σ의 출력 y는 로직 1의 위치가 홀수일 때는 (10,10), 짝수일 때는 (01,01)의 값을 낸다. 또 정상 입력 o와 e가 번갈아 들어와서 상태 천이가 이루어지면 Σ의 출력은 아래와 같이 순환한다.

$$(10,10) \rightarrow o \rightarrow (01,01) \rightarrow e \rightarrow (10,10) \rightarrow o \rightarrow \cdots$$

출력 y는 P와 Q의 패리티 비트로 구성되므로 y만 보고는 링 카운터의 로직 1 값이 어느 위치에 있는지를 알지 못한다. 하지만 정상 동작에서 o,e,o,e,... 입력 스트링의 길이가 n이 될 때마다 로직 1 값이 한 바퀴 순환하여 현재 위치로 돌아온다는 사실은 알 수 있다. 예를 들어 n=5이고 로직 1 값이 차지하는 현재 상태 비트의 위치가 짝수라면 정상 입력 스트링 e,o,e,o,e이 부가된 후에는 로직 1 값이 다시 원래의 짝수 위치로 복귀한다. 이 정보는 나중에 고장 극복 알고리듬을 작성할 때 활용될 것이다.

부시스템 P에서 방사선 고장이 발생한다고 가정하자. Q에 대한 경우는 대칭성을 이용하여 P의 결과로부터 쉽게 유도할 수 있다. 고장 직전 정상 출력이 (10,10)이었다고 하자. 이것은 홀수 번째 상태 비트에 로직 1이 머물렀다는 사실을 의미한다. 로직 1을 가진 상태 비트의 위치를 k라고 하면(p_k=1) k는 홀수이고 $1 \le k \le n$이다. 우선 $z_P[i]$ 고장부터 고려한다. 로직 값이 0에서 1로 반전되는 고장이므로 p_k를 제외한 P의 임의의 상태 비트 i에서 $z_P[i]$가 일어날 수 있다. i가 홀수라면 $z_P[i]$이 일어난 순간 P의 홀수 번째 상태 비트에 로직 1 값이 두 개 존재한다. 따라서 y_1이 0에서 1로 바뀌고(다른 출력 비트는 불변) Σ의 출력 값은

$$y: (10,10) \rightarrow (00,10)\ (k: \text{홀수},\ i: \text{홀수},\ k \neq i)$$

으로 바뀐다. 만약 i가 짝수라면 정의에 의해서 y_2 값이 0에서 1로 변경되므로

$$y: (10,10) \rightarrow (11,10)\ (k: \text{홀수},\ i: \text{짝수})$$

가 된다.

이번에는 고장 직전 정상 출력이 (10,10)이었을 때 $n_P[i]$ 고장이 일어났다고 하자. $n_P[i]$는 p_i 비트가 1에서 0으로 반전되는 방사선 고장이다. 그런데 정상 상태에서 로직 1 값이 k번째 비트 p_k에 있었으므로 오직 i=k이다. 다시 말하면 이것은 로직 1의 값을 가진 현재 카운팅 비트가 방사선의 영향을 받아 그 값이 반전되는 사건이다. P의 홀수 번째 비트가 모두 0이 되므로 y_1=0이며 Σ의 출력 값은 다음과 같이 변한다.

$$y: (10,10) \rightarrow (00,10)\ (k: \text{홀수},\ i=k)$$

고장 직전 정상 출력이 (01,01), 즉 로직 1 값을 가진

상태 비트 위치 k가 짝수일 때도 위의 세 경우와 유사하게 고장 탐지가 가능하다.

입력/출력 비동기 회로의 고장 탐지 및 제어에 대한 기존 연구에서는 상태 관측기(observer)를 이용하여 시스템의 현재 상태를 추적해야 했다[11]. 하지만 본 논문에서 제안한 고장 탐지 기법은 관측기나 다른 고장 탐지 하드웨어를 사용하지 않는다는 뚜렷한 장점이 있다. 물론 고장이 발생한 정확한 상태 비트의 위치를 알기는 불가능하며, 그 위치가 홀수인지 짝수인지만을 파악할 수 있다. 예를 들어서 Σ의 출력 값이 고장 전후로 (10,10)에서 (11,10)으로 변경된다면 우리는 부시스템 P의 어떤 짝수 비트에서 방사선 고장이 일어나서 로직 값이 0에서 1로 반전되었다는 사실을 추론할 수 있다.

2. 고장 극복

현재까지 설정한 Σ의 내고장성 능력과 고장 탐지 기법을 정리하면 다음과 같다. i) Σ는 DMR을 구성하는 부시스템 P와 Q의 홀수 번째 비트를 다른 부시스템의 해당 비트 값으로 치환하는 입력을 가진다(c_p와 c_q). ii) Σ의 출력 y의 변화를 보고 방사선 고장의 발생을 즉시 감지하며, 고장이 발생한 비트가 속한 부시스템과 비트 위치의 홀짝 여부를 안다. 이러한 설정과 비동기 머신에 대한 교정 제어 이론을 결합하여 방사선 고장을 즉시 극복하는 새로운 알고리듬을 제안한다. 고장 극복 교정 제어기를 C라고 하면 C는 Σ로 들어가는 정상 입력 $v\in\{o,e\}$와 Σ의 출력 피드백 y를 받아서 제어 입력 $u\in A_n$를 생성하는 비동기 순차 머신이다. C는 정상 입력 v가 변경되지 않는데도 불구하고 출력 피드백 y가 정상 값 (10,10) 또는 (01,01)을 벗어난 값으로 변하는 순간 방사선 고장의 발생을 감지한다. 앞 절과 마찬가지로 고장은 부시스템 P에서 일어난다고 가정한다.

먼저 방사선 고장이 홀수 번째 비트에서 발생했다고 하자. 홀수 번째 비트 값을 Q의 해당 비트 값으로 리셋시키는 제어 입력이 존재하므로 이 고장은 쉽게 복구할 수 있다. y의 변화로 고장을 감지한 순간 C는 제어 입력 c_p를 Σ에 전달한다. 앞 절에서 논한 예를 계속 사용하여 Σ가 초기 상태 x_0=(10⋯0,10⋯0)에 있었다고 하자. 이때의 출력 y는 y=(10,10)이다. P의 홀수 번째 비트 i에서 방사선 고장이 발생하여 출력이 y=(00,10)으로 바뀐다고 하자. 앞 절에서 기술한 바대로 i=1이면, 즉 로직 1 값을 보유한 비트에서 고장이 발생했다면, 고장 입력 $n_P[1]$이 일어났다는 것을 의미하며, $i\neq1$이면 고장 입력 $z_P[i]$가 발생했다는 것을 의미한다. 두 경우 모두 C가 비트 위치 i의 정확한 값을 알지 못하지만 변경된 출력은 (00,10)으로 동일하다.

고장을 감지한 순간 C는 제어 입력 c_p를 생성하여 Q의 모든 홀수 번째 비트 값을 P의 홀수 번째 비트에 쓴다(write). Q는 정상 동작을 하고 있었기 때문에 c_p의 명령이 실행되면 고장 복구가 완성된다. 출력 변화로 이 과정을 표시하면 아래와 같다.

$$y:\ (00,10)\ \rightarrow\ c_P\ \rightarrow\ (10,10)$$

다음으로 Σ가 x_0에 있을 때 P의 짝수 번째 상태 비트에서 방사선 고장이 발생하여 y가 (10,10)에서 (11,10)로 바뀌었다고 하자. Σ가 보유한 내고장성 능력이 부시스템의 모든 홀수 번째 비트만을 변경 가능하다는 사실에 유의해야 한다. 즉 고장이 발생한 위치는 P의 짝수 번째 비트이며, 값을 변경 가능한 위치는 P의 홀수 번째 비트이다. 이때의 해결책은 Σ에 정상 카운팅 입력을 넣어 상태 천이를 한 번 시킨 다음 리셋 입력을 부과하는 방법이다. 편의상 고장이 발생한 상태 비트가 p_2라고 가정하고 이 기법을 설명한다(다른 위치의 경우도 유사하게 해석된다). 고장이 일어나면 Σ의 상태는 다음과 같이 바뀐다.

$$x:\ (1000\cdots0,1000\cdots0)\ \rightarrow\ z_P[2]\ \rightarrow\ (1100\cdots0,1000\cdots0)$$

카운팅 로직 1 값이 첫번째 비트에 있었으므로 제어기 C는 고장을 탐지한 순간 정상 입력 o를 Σ에 전달한다. Σ의 모든 상태 비트가 한 스텝씩 전진하므로 상태 천이는

$$x:\ (1100\cdots0,1000\cdots0)\ \rightarrow\ o\ \rightarrow\ (0110\cdots0,0100\cdots0)$$

와 같이 된다. 물론 C는 x의 상태 비트를 관측하지 못하며 출력 y의 변화를 통해 상태 천이가 됨을 감지한다. 위 상태 천이와 일치하는 출력 변화는 다음과 같다.

$$y:\ (11,10)\ \rightarrow\ o\ \rightarrow\ (11,01)$$

고장으로 생긴 로직 1 값이 p_2에서 p_3으로 이동하였기 때문에 C는 제어 입력 c_p를 생성하여 P의 홀수 번째 상태 비트 값을 Q의 값으로 리셋시킨다. 출력 변화로 이 과정을 나타내면 아래와 같다.

$$y:\ (11,01)\ \rightarrow\ c_p\ \rightarrow\ (01,01)$$

방사선 고장으로 유발된 로직 값의 반전 현상은 해결되었지만, Σ의 카운팅 로직 1 값이 원래 있었던 위치 (p_1,q_1)보다 한 스텝 전진(p_2,q_2)하였다는 문제가 남아 있다. 이 문제를 해결하기 위해서 C는 교정 제어의 방법으로 제어 입력 스트링을 생성한다. 먼저 Σ에 짝수 번째 상태 비트에 대한 카운팅 입력 e를 전달하면 Σ의 카운팅 로직 1 값은 다시 전진하여 p_3, q_3에 위치하게 된다. 이에 대한 출력도 (01,01)에서 (10,10)으로 바뀐다. 출력 피드백의 변화를 확인한 C는 이번에는 홀수 번째 카운팅 입력 o를 Σ에 전달한다. Σ의 로직 1 값은 다시 전진하고 출력 값도 따라서 바뀐다. 이런 식으로 C는 n-1개의 정상 입력을 Σ에 전달하고 최종적으로 로직 1 값은 원래 있던 상태 비트로 옮겨지고 고장 극복 과정은 완료된다. 전역 클럭이 없는 비동기 디지털 시스템에서 C의 교정 제어 동작은 매우 빠르게 진행되므로 Σ가 n번의 상태 천이를 거침에도 불구하고 외부 사용자에는 방사선 고장의 발생이 감지되지 않는다(교정 제어에 대한 자세한 이론적 분석은 [8,11] 참조). 이러한 일련의 교정 제어 동작을 출력 변화로 기술하면 다음과 같다.

$$y:(01,01)\xrightarrow{e}(10,10)\xrightarrow{\overbrace{o\cdots o}^{n-1}}(01,01)$$

이상과 같은 알고리듬을 적용하면 DMR 구조로 구성된 비동기 디지털 시스템 Σ에서 생기는 방사선에 의한 SEU

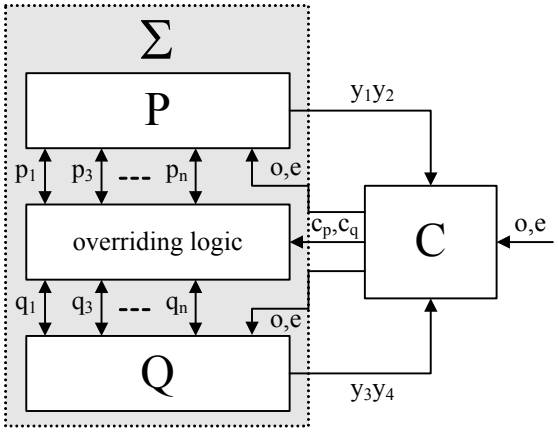

그림 1. 이중화 구조를 가진 비동기 디지털 시스템 Σ에 대한 고장 극복 제어 시스템.

Fig. 1. Fault tolerant control system for an asynchronous digital system Σ with DMR structure.

고장을 극복할 수 있다. 앞에서 기술했듯이 이 방법은 고장이 발생하는 상태 비트 위치를 알려주는 관측기를 사용하지 않고도 최소한의 내고장성 모듈과 교정 제어 이론을 이용하여 모든 고장을 효율적으로 극복한다는 장점을 지닌다. 그림 1은 본 논문에서 제안한 고장 극복 기법을 도시한 그림이다.

IV. VHDL 실험

본 연구에서 제안한 고장 진단 및 극복 기법의 우수성과 응용가능성을 보이기 위해서 DMR 링 카운터와 교정 제어기를 VHDL로 구현하고 고장 주입기(fault injector)를 설계하여 방사선 고장을 실험적으로 발생시킨 다음 검증 실험을 실시하였다. 사례 연구로 쓰인 DMR 링 카운터는 n=5 비트를 가진다고 설정하였다.

먼저 5-비트 DMR 링 카운터의 홀수 번째 비트에서 방사선 고장이 발생한 경우의 고장 극복과정을 모의실험 하였다. 실험에서 세번째 비트에서 고장이 발생한다고 하였다. 즉 고장 주입기를 이용하여 x: (00100,00100) → n_P[3] → (00000,00100) 방사선 고장을 발생시켰다. 고장에 의해 부시스템 카운터 P의 세번째 비트가 1→0으로 변화되므로 출력은 y: (10,10) → (00,10)로 바뀐다. 정상 동작에서의 출력은 y=(10,10) 또는 y=(01,01)이므로 출력 변화로부터 홀수 번째 비트에서 고장이 발생하였다는 것이 탐지된다. 앞 장에서 설명한 바와 같이 홀수 번째 비트에서 고장이 탐지된 경우 교정 제어기 C는 제어 입력 c_p를 전달한다. 이때 출력은 y: (00,10) → c_P → (10,10)로 변하면서 고장 복구가 실현된다.

그림 2는 위에서 설명한 홀수 번째 방사선 고장에 대하여 VHDL로 모의실험한 결과이다. 출력 신호 y=(y_1y_2,y_3y_4), 제어 신호 c_p, o, e와 P, Q의 상태 값 $p_1p_2p_3p_4p_5$, $q_1q_2q_3q_4q_5$ 비트를 차례대로 나타내었다. 시간 t_1=59nsec에 외부 입력에 의해 카운터의 상태가 x: (01000,01000)→(00100,00100)로 천이한다. 시간 t_2=95nsec에 방사선 고장이 발생하여 p_3값이 1→0로 변하고 y_1y_2 값은 01→00으로 변한다. 제어기 C는 고장을 인지하고 홀수 번째 비트에서의 고장이라는 사실과

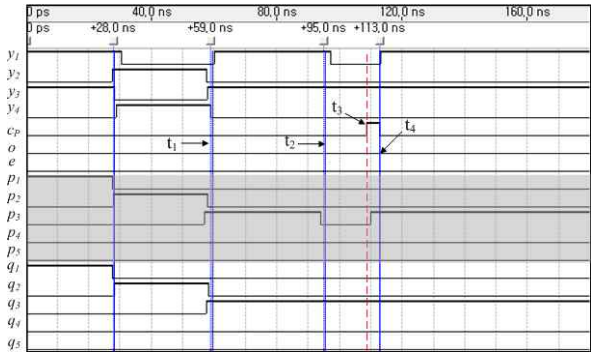

그림 2. DMR 링 카운터의 홀수 번째 비트에서 1→0 방사선 고장에 대한 고장 극복 실험 결과.

Fig. 2. Experiment results of fault tolerance for 1→0 radiation fault at an odd bit of the DMR ring counter.

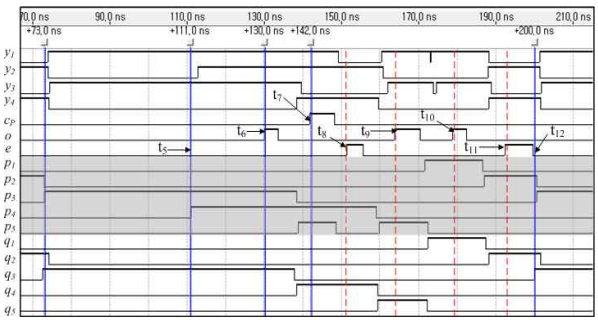

그림 3. DMR 링 카운터의 짝수 번째 비트에서 0→1 방사선 고장에 대한 고장극복 실험 결과.

Fig. 3. Experiment results of fault tolerance for 1→0 radiation fault at an even bit of the DMR ring counter.

고장이 일어난 카운터를 판별한 후 t_3=108nsec에 제어 입력 c_P를 전달한다. 이후 t_4=113nsec에 출력 y_1y_2 값은 00→10로 복귀하므로 제어 과정을 종료한다. 고장 극복에 사용된 제어 스트링은 t=c_P이며, 고장 발생 후 정상 상태 복귀까지 걸린 전체 시간은 t_4-t_2=113-95=18nsec로 매우 짧다. 따라서 이 결과는 본 논문에서 제안한 비동기 제어기가 방사선 고장 발생 후 매우 빠른 시간(이론적으로 0) 내에 DMR 카운터를 원래의 상태로 복귀시킬 수 있음을 보여준다.

그림 3은 P의 짝수 번째 상태 비트에서 방사선 고장이 발생하여 y가 (10,10)에서 (11,10)로 바뀐 경우의 실험 결과이다. t_5=111nsec에서 p_4에 0→1의 방사선 고장이 발생하여 $p_1p_2p_3p_4p_5$=00110로 되고, y_1y_2 값이 10→11로 변한다. 제어기 C는 고장이 짝수 번째 비트라는 것과 고장이 발생한 카운터 P를 판별하고 t_6=130nsec에서 첫번째 제어 입력 o를 전달한다. 이때 DMR 카운터의 상태는 x: (00110,00100)→(00011,00010)로 되고 출력은 y=(11,01)로 된다. 제어 입력 o에 의해 고장이 발생한 비트는 5번째(홀수) 비트로 이동한다. 시각 t_7=142nsec에서 P의 상태를 Q의 상태로 덮어 쓰는 제어 명령 c_p가 전달되어 카운터는 x: (00011,00010) → (00010,00010)로 천이하고 출력 y=(01,01)로 변화되어 P에서 발생한 방사선 고장은 사라진다. 하지만 아직 고장 직전 원래 상태 (00100,00100)로 복귀하는 과정이 남아 있다. 이

후 t_8=152nsec, t_9=164nsec, t_{10}=178nsec, t_{11}=194nsec에서 각각 제어 입력 e, o, o, e를 차례로 전달하면, y값은

(01,01)→e→(10,10)→o→(10,10)→o→(01,01)→e→(10,10)

으로 변한다. n=5이므로 5번째 한 스텝 전진 제어 입력 e 발생(t_{11}) 후에 제어 과정을 종료한다. t_{12}=200nsec에서 카운터는 원래 상태 x=(00100,00100)로 복귀한다. 고장 극복에 사용된 제어 스트링은 t=o,c_P,e,o,o,e이며, 전체 고장 극복 시간은 t_{12}-t_5=200-111=89nsec이다. 실험 결과에서 알 수 있듯이 고장 탐지에 약 20nsec, 제어 입력 한 개를 발생시키는 데 약 10nsec, 그리고 제어 입력 전달 후 카운터 상태가 천이하기까지 약 10nsec가 소요되었다.

V. 결론

방사선 고장에 강인한 디지털 시스템을 구현하는 일은 고신뢰도 군사용 및 우주용 디지털 시스템을 제작하기 위해 반드시 해결해야 하는 기반 기술이다. 이번 연구에서는 이중화 구조를 가지는 비동기 디지털 시스템에 대한 새로운 고장 극복 방법을 제안하였다. 제안된 기법의 핵심 아이디어는 고장 난 메모리 비트의 정확한 위치 정보를 모르고도 시스템이 보유한 내고장성 모듈과 비동기 교정 제어를 이용하여 즉각적인 고장 탐지 및 복구 기능을 실현하였다는 것이다. 이중화 구조를 가지는 링 카운터를 VHDL 코드로 제작하고 주어진 고장 시나리오 내에서 모의실험을 실시하여 고장 극복 알고리듬의 성능을 입증하였다. 본 연구에서 제안된 이중화 구조가 관련 분야 디지털 시스템의 신뢰성 증대에 응용되기를 기대한다.

REFERENCES

[1] J. Sparsø and S. Furber, *Principles of Asynchronous Circuit Design: A Systems Perspective*, Kluwer Academic Publishers, 2001.

[2] S.-M. Ryu, "An optimal scrubbing scheme for auto error detection & correction logic," *Journal of Institute of Control, Robotics and Systems (in Korean)*, vol. 17, no. 11, pp. 1101-1105, Nov. 2011.

[3] Y. Monnet, M. Renaudin, and R. Leveugle, "Designing resistant circuits against malicious faults injection using asynchronous logic," *IEEE Transactions on Computers*, vol. 55 no. 9, pp. 1104-1115, Sep. 2006.

[4] T. Panhofer, W. Friesenbichler, and M. Delvai, "Optimization concepts for self-healing asynchronous circuits," *in Proc. of the 12th International Symposium on Design and Diagnostics of Electronic Circuits & Systems (DDECS '09)*, pp. 62-67, 2009.

[5] C. M. Krishina and K. G. Shin, *Real-Time Systems*, New York: McGraw-Hill, 1997.

[6] S. W. Kwak and K. H. You, "Reliability analysis and fault tolerance strategy of TMR real-time control systems," *Journal of Institute of Control, Robotics and Systems (in Korean)*, vol. 10, no. 8, pp. 748-754, Aug. 2004.

[7] A. Ziv and J. Bruck, Performance optimization of checkpointing schemes with task duplication, *IEEE Transactions on Computers*, vol. 46, no. 12, pp. 1381-1386, Dec. 1997.

[8] T. E. Murphy, X. Geng, and J. Hammer, "On the control of asynchronous machines with races," *IEEE Transactions on Automatic Control*, vol. 48, no. 6, pp. 1073-1081, Jun. 2003.

[9] J.-M. Yang and S. W. Kwak, "Corrective control of asynchronous sequential circuits with faults from total ionizing dose effects in space," *Journal of Institute of Control, Robotics and Systems (in Korean)*, vol. 17, no. 11, pp. 1125-1131, Nov. 2011.

[10] Z. Kohavi and N. Jha, *Switching and Finite Automata Theory,* 3rd ed., New York: Cambridge University Press, 2010.

[11] J. Peng and J. Hammer, "Input/output control of asynchronous sequential machines with races," *International Journal of Control*, vol. 83, no. 1, pp. 124-144, Jan. 2010.

곽 성 우

1993년 한국과학기술원 전기및전자공학과 졸업(공학사). 1995년 한국과학기술원 전기및전자공학과 졸업(공학석사). 2000년 한국과학기술원 전기및전자공학과 졸업(공학박사). 2000년~2002년 인공위성연구센터 선임연구원, 연구교수. 2003년~현재 계명대 전자공학과 부교수. 관심분야는 위성 탑재 컴퓨터, 실시간 시스템, 비동기 시스템 설계, 내고장성 시스템 설계.

양 정 민

1993년 한국과학기술원 전기및전자공학과 졸업(공학사). 1995년 한국과학기술원 전기및전자공학과 졸업(공학석사). 1999년 한국과학기술원 전기및전자공학과 졸업(공학박사). 1999년~2001년 한국전자통신연구원 컴퓨터 · 소프트웨어 기술연구소 선임연구원. 2001년~2013년 대구가톨릭대학교 전자공학과 교수. 2013년~현재 경북대학교 전자공학부 부교수. 관심분야는 비동기 순차 머신 제어, 실시간 시스템, 걸음새 연구.