1999년도 대한전기학회 정기총회 및 추계학술대회 논문집 1999. 11. 20

고장포용시스템에서의 다중 모듈 하드웨어 여분의 신뢰도 분석

홍 태 화 김 학 배
연세대학교 전기컴퓨터공학과

Analysis on the Reliability of the Multi-Module Hardware Redundancy in the Fault Tolerant System

Hong Taehwa and Kim Hagbae
Dept. of Electircal & Computer Engineering Yonsei University

Abstract - 제어 컴퓨터의 고장으로 인해 인명이나 재산에 치명적 영향을 미치는 safety-critical 실시간 시스템을 제어하고 모니터링하기 위해 디지털 컴퓨터의 사용은 점점 일반화되고 있다. 특히, VLSI 기술의 급격한 발달로 인해 하드웨어가 초소형화 되고 대량생산이 가능해진 현실에서 이러한 제어 컴퓨터의 극대화된 신뢰도 요구를 만족시키기 위해 막중한 하드웨어 여분(hardware redundancy)이 널리 사용되고 있는 실정이다. 본 논문에서는 N개의 다중 모듈(multi-module)로 이루어진 하드웨어 여분의 운영 모드를 분석하고 각 운영 모드에서 고장이 발생할 경우 모드의 전환과 그로 인한 신뢰도의 변화를 계산할 것이다. 그리고 간단한 시뮬레이션을 통해 전환된 여러 모드 중 가장 우수한 신뢰도를 갖는 모드를 평가하게 된다.

keywords: 하드웨어 여분, 다중 모듈, 운영 모드

1. 서 론

멀티프로세서 시스템들은 단일 프로세서 시스템들보다 높은 성능과 신뢰도를 구현시킬 수 있다[1]. 일반적으로 고장포용 멀티프로세서 시스템은 하나의 프로세서가 고장난 경우라도 동작이 멈춰서는 안된다[1]. 따라서 시스템은 고장난 전체 시스템의 프로세서를 신속하고 적절하게 새로운 프로세서로 대체하여야 하고 이 때 시스템의 성능은 감소되게 된다.

하드웨어 여분은 고장포용을 수행하기 위해서 여러 다양한 방식으로 이용된다. 가장 잘 알려진 하드웨어 여분 방식은 NMR(N modular redundancy)이다[2]. NMR 시스템은 majority voter와 동일한 작업을 수행하는 N개의 동일한 모듈로 이루어져 있다. 입력이 각 작업 모듈에 분산되어 있고 각 모듈의 출력은 모두 voter에 전달되면서 voter는 각 모듈로부터 전달된 출력값들을 분석하여 다수의 모듈이 나타내는 값을 그의 출력값으로 결정하게 된다. 이러한 NMR 시스템은 고장 모듈의 교체 없이, 고장의 영향을 차폐(masking)하는 능력을 가진다. 비록 NMR 시스템에 스페어 모듈을 둔다면 고장 모듈을 대체하는 데 오버헤드가 걸리지만, (N-1)/2 개 이하의 고장 모듈을 포용하기 위해 필요한 N개의 모듈을 시스템에 갖추기 위한 비용 또한 상당하다. 많은 비용 문제를 해결하기 위한 최선의 방법은 시스템에 스페어 풀(spare pool)을 유지하는 것이다[2]. 일반적으로 스페어 풀은 스위치와 동일한 작업을 수행하는 모듈들로 이루어진다. NMR 시스템 중 하나의 모듈이 고장이 나면, 스페어 풀로부터 하나의 정상 모듈이 선택되어 대체된다. NMR 시스템에 스페어 풀을 이용함으로써 비용 면에서 많은 절감을 이룰 수 있다. 그러나 엄격한 시간 제약을 요구하는 실시간 시스템에서는 스페어 풀의 이용이 반드시 필수적이고 적절한 방법은 아니다.

전통적으로 시스템의 신뢰도는 두 가지 기본적인 전략, 고장 회피(fault-avoidance), 고장 포용(fault-tolerance)[2,4]을 통해 만족되어 왔다. 고장회피는 시스템을 위해 제조과정에서 신뢰성 있는 컴포넌트를 설계하고 보다 강력한 테스팅을 거쳐 제조시에 컴포넌트의 신뢰성을 보장하는 것이다[3]. 한편, 고장포용은 시스템이 동작될 동안 고장이 발생한다는 가정 하에, 여분 기계를 이용하여 이들의 영향을 자동적으로 극복할 수 있도록 하는 것이다. 따라서, 고장포용 시스템은 예측 가능한 고장이 발생하더라도 지속적으로 동작하도록 하는 시스템이다. 이러한 개념들을 바탕으로 본 논문에서는 동일한 작업을 수행하는 N개의 모듈로 구성된 시스템에서 N개의 모듈을 운영하는 여러 모드를 정의하고 고장이 발생할 경우 신뢰도를 계산하여 최선의 운영 모드를 결정하는 방법을 모색한다. 그리고 추후에 성능 계산을 통해 신뢰도와의 관계를 일반화할 계획에 있다.

2. 본 론

2.1 모드의 정의 및 구성

$N(N\geq 4)$개의 하드웨어 여분으로 구성된 고장포용 시스템을 생각하자. 우선 모드는 N개의 여분을 이용하는 방법에 따라 구성된다. 하드웨어 여분으로 가장 인기 있는 방법은 TMR(Triple Modular Redundancy)과 이중 모듈(duplex)이다[2,5]. 본 논문에서는 TMR, 이중 모듈, 그리고 단일 모듈의 세 가지 시스템을 기본단위로 N개의 모듈로 구성된 여분 시스템을 구성하게 된다. N 여분 시스템이 i개의 TMR과 j개의 이중 모듈, k개의 단일 모듈로 이루어 졌다고 가정하자.

$$N=3i+2j+k \quad (i,j,k=0,1,2,\cdots) \tag{1}$$

식 (1)은 이들의 관계를 나타내고 있다. 식 (1)로부터 여분 시스템의 모드 구성을 위해 다음과 같이 가정한다.

i) $i\geq 1$ *or* $j\geq 1$, ii) $k=0$ *or* 1

첫 번째 가정은 N 여분 시스템은 반드시 TMR이나 이중 모듈 중 하나는 소유하고 있어야 한다는 것이다. 두 번째 가정은 단일 모듈의 수가 이중 모듈을 구성하는 단위 여분 시스템은 $i+j+k$개의 작업을 수행하여 운영된다. 우리는 N개의 모듈이 i개의 TMR과 j개의 이중 모듈, k개의 단일 모듈로 구성될 경우, 이들 각각 하나를

표 1. N=4,5,6일 때 운영 가능한 모드

	N=4	N=5	N=6
Mode 1	$M^{4}_{1,0,1}$	$M^{5}_{1,1,0}$	$M^{6}_{2,0,0}$
Mode 2	$M^{4}_{0,2,0}$	$M^{5}_{0,2,1}$	$M^{6}_{1,1,1}$
Mode 3	×	×	$M^{6}_{0,3,0}$

모드라 하고 이는 $M^{N}_{i,j,k}$ 같이 표현하기로 한다. 위의 표 1은 N=4, 5, 6일 때 운영 가능한 모드의 경우를 나타내고 있다.

2.2 이중 모듈 시스템과 TMR

이중 모듈 시스템은 가장 일반적인 고장 검출(fault detection) 여분 시스템이다. 아래의 그림 1에서 보여지듯이, 이중 모듈 시스템은 두 개의 동일한 작업을 수행하는 모듈로 이루어져 있고 이들의 출력값이 비교기(comparator)에 들어가 동일 여부가 비교된다.

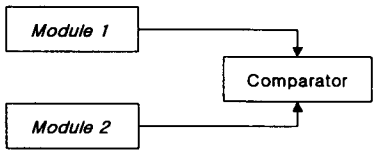

그림 1. 이중 모듈 시스템

만약 값이 다르다면 둘 중 하나의 모듈은 고장이 발생한 것이다. 이와 같이 이중 모듈 시스템은 고장이 검출이 가능하다. 그렇지만 두 개의 모듈 중 실제로 고장난 모듈을 구별할 수 없기 때문에 고장 포용은 불가능하다[2,5]. 이중 모듈 시스템이 신뢰도는 다음의 식 (2)과 같다.

$$R_{duplex} = [R_M{}^2 + 2CR_M(1-R_M)]R_C R_S \qquad (2)$$

여기서 R_M, R_C, R_S는 각각 단일 모듈의 신뢰도, 비교기의 신뢰도, 모듈 선택기(selector)의 신뢰도를 의미하고 C는 고장난 모듈을 구별해 낼 확률을 의미한다.

한편, TMR은 가장 일반적인 고장 차폐(fault masking) 방법이다. TMR은 동일한 작업을 수행하는 세개의 모듈과 이들의 결과를 수합하여 다수의 모듈이 나타내는 결과를 선택하는 voter로 구성되어 있다. 그림 2는 TMR 시스템의 기본 구성도이다. 즉 3개의 모듈 중 하나의 모듈에 고장이 발생하여 잘못된 출력값을 갖더라도 다수 보팅(majority voting)에 의해 시스템 전체에는 고장으로 인한 영향이 차폐되게 된다. 이러한 이유로 TMR을 정적(static) 여분 시스템이라고 한다[2,4,5].

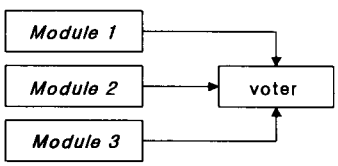

그림 2. TMR 시스템

이러한 개념은 세개의 모듈 대신 일반적인 홀수개의 모듈을 갖는 NMR 시스템으로 확장될 수 있다. 아래의 식 (3)는 TMR 시스템의 신뢰도를 구하는 식이다.

$$\begin{aligned} R_{TMR} &= R_V \sum_{i=0}^{1} \binom{3}{i} (1-R_M)^i (R_M)^{3-i} \\ &= R_V(R_M{}^3 + 3(1-R_M)R_M{}^2) \\ &= R_V(3R_M{}^2 - 2R_M{}^3) \end{aligned} \qquad (3)$$

여기서 R_M와 R_C는 각각 단일 모듈의 신뢰도와 voter의 신뢰도를 의미한다.

2.3 일반적인 모드의 신뢰도

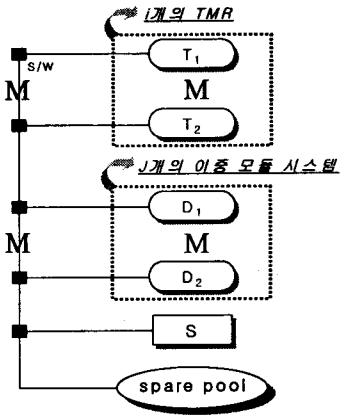

그림 3. i개의 TMR과 j개의 이중 모듈, k개의 단일모듈로 구성된 N 여분 시스템

2.2절에서 설명한 이중 시스템과 TMR, 그리고 단일 모듈의 조합으로 구성된 일반적인 N 여분 시스템의 신뢰도를 구해보자. 그림 3과 같이 i개의 TMR과 j개의 이중 모듈, k개의 단일 모듈로 구성된 $M^{N}_{i,j,k}$ 모드의 신뢰도를 구해보자. 먼저, i개의 TMR에 대해서 생각해보자. 이 때 i개의 TMR 중 k번째 TMR을 $T_k(k=1,2,\cdots,i)$라 하면 이들의 신뢰도는 다음의 식 (4)와 같이 나타낼 수 있다.

$$\begin{aligned} R_i &= 1 - P(T_1 \ faulty)P(T_2 \ faulty)\cdots P(T_i \ faulty) \\ &= 1 - \Pi_{k=1}^{i} P(T_k \ faulty) \end{aligned} \qquad (4)$$

i개의 TMR 각각은 서로 독립적(independent)이기 때문에 $P(T_1 \ faulty) = P(T_2 \ faulty) = \cdots = P(T_i \ faulty)$와 같은 식이 성립한다. 따라서 k번째 TMR의 신뢰도가 식 (3)와 같으므로, 식 (4)는 다음과 같이 나타낼 수 있다.

$$\begin{aligned} R_i &= 1 - [P(T_k \ is \ faulty)]^i \\ &= 1 - [1 - R_{TMR}]^i \end{aligned} \qquad (5)$$

이와 마찬가지로 j개의 독립적인 이중 모듈 시스템에 대해서도 식 (2)을 이용하여 다음의 식 (6)과 같이 신뢰도를 구할 수 있다. k번째 이중 모듈 시스템을 $D_k(k=1,2,\cdots,i)$라 하자.

$$\begin{aligned} R_j &= 1 - P(D_1 \ is \ faulty)P(D_2 \ is \ faulty)\cdots P(D_j \ is \ faulty) \\ &= 1 - [P(D_k \ is \ faulty)]^j \\ &= 1 - [1 - R_{duplex}]^j \end{aligned} \qquad (6)$$

식 (5)과 (6)에 의해서 i개의 TMR과 j개의 이중 모듈, k개의 단일 모듈로 이루어진 N 여분 시스템의 신뢰도를 다음의 식 (7)과 같이 유도할 수 있다.

$$\begin{aligned} R &= 1 - [P(T_1 \ is \ faulty)\cdots P(T_i \ is \ faulty) \\ &\quad P(D_1 \ is \ faulty)\cdots P(D_j \ is \ faulty)(1-R_M)^k] \\ &= 1 - [1 - R_{TMR}]^i [1 - R_{duplex}]^j (1-R_M)^k \end{aligned} \qquad (7)$$

2.4 시뮬레이션 및 수치 결과

2.3 절에서 N 모듈 여분 시스템의 여러 운영 모드를 위한 신뢰도를 식 (7)과 같이 나타내었다. 물론 하나의 운영모드를 구성하고 있는 각각의 단위 시스템들(i개의 TMR과 j개의 이중 모듈, k개의 단일 모듈)은 작업 수행에 있어서 서로간의 독립적 수행을 원칙으로 하고 하나의 단위모드에 고장이 발생했을 경우, 그 모드에 남겨진 작업들은 다른 단위모드에서 적절한 운영 정책에 의해 수행되게 된다.

본 절에서는 특정한 N에 대해 운영될 수 있는 모드들의 신뢰도에 대해 시뮬레이션을 하였다. 2.1절의 가정에 의해 N=4, 5, 6일 때 운영될 수 있는 모드의 수는 각각 2, 2, 3이다.

N=4인 경우, $R_v=0.81$, $R_c=0.9$, $R_s=0.9$, $C=0.8$에 대해서 그림 4와 같은 결과를 얻을 수 있다. 단일 모듈의 신뢰도가 시간에 따라 감소함을 고려할 때 시간이 지날수록 단일 모듈의 신뢰도는 감소되고 $M^4_{1,0,1}$모드가 단일 모듈의 신뢰도가 약 0.7정도 되는 곳에서 $M^4_{0,2,0}$모드보다 신뢰도가 더 떨어진다. 따라서 처음에 $M^4_{1,0,1}$모드로 수행하고 일정한 시간이 흐르면 $M^4_{0,2,0}$모드로 전환하는 것이 최선의 선택이다.

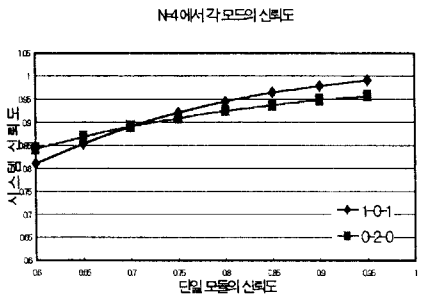

그림 4. N=4에서 각 운영모드의 신뢰도

N=5인 경우, 그림 5에서 볼 수 있듯이, $M^5_{0,2,1}$모드가 시간(시간의 증가와 단일 모듈의 신뢰도는 서로 반비례)에 관계없이, $M^4_{1,1,0}$모드 보다 더 신뢰도가 우수하다.

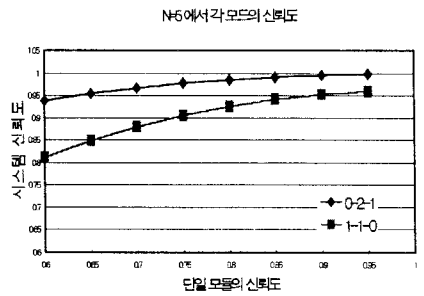

그림 4. N=5에서 각 운영모드의 신뢰도

N=6인 경우, 그림 6에서 볼 수 있듯이, $M^6_{0,3,0}$모드가 $M^6_{2,0,0}$와 $M^6_{1,1,1}$에 비해 시간이 지날수록 신뢰도가 더 떨어짐을 알 수 있고, 시간에 따라 $M^6_{2,0,0}$와 $M^6_{1,1,1}$는 비슷한 신뢰도를 나타냄을 알 수 있다.

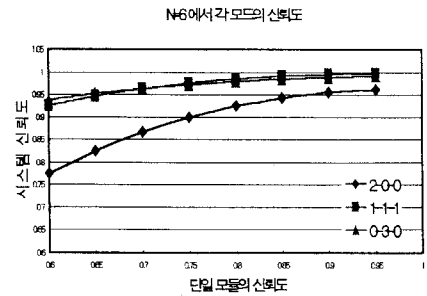

그림 4. N=6에서 각 운영모드의 신뢰도

3. 결 론

본 논문에서는 고장포용을 위한 공간 여분으로서 N개의 모듈을 고려했을 때 이들이 운영모드를 정의하고 단위 모듈간의 독립성이 보장되는 상황만을 고려하여 각 모드에 대한 신뢰도를 계산하였다. 이러한 개념은 고신뢰도를 요구하는 시스템에 적용될 수 있다. 하지만 실시간 시스템과 같이 데드라인 정보를 지닌 작업을 수행해야 하는 시스템에 대해서는 각 모드에 대한 성능의 계산이 필요하다. 향후 제시된 각 운영모드에 대한 성능의 계산을 수행하여 신뢰도와의 상충관계 문제를 통해 최적화된 운영모드를 결정하는 방안을 제시할 필요가 있고 현재의 모듈간의 독립적 관계에 대한 가정 뿐만 아니라 EMI(전자기파 간섭현상)와 같은 외부적 영향에 의해 발생되는 모듈의 연관 고장(correlated fault)까지도 고려하여 보다 현실적인 문제에 접근할 필요가 있다.

[참 고 문 헌]

[1] Chen, C., Asada, H., Kakuda, Y. and Kikuno, T., "comparison of hybrid redundant multiprocessor systems with respect to performabilities", *Fault-Tolerant Computing, 1993. FTCS-23.* Digest of Papers., The Twenty-Third International Symposium on , Aug. 1993 , Page(s): 66

[2] P. K. Lala, *Fault-Tolerant and Fault-Testable Hardware Design.* Englewood Cliffs, NJ: Prentice-Jall, 1985

[3] A. Barbour and A. Wokcik, "A general, constructive approach to fault-tolerant design using redundancy", *IEEE Trans. on computers,* Vol. 38, No.1. pp.15-29, 1989

[4] A. D. Ingle and D. P. Siewiorek, "A reliability model for various switch design in hybrid redundancy", *IEEE Trans. on Computers,* Vol. C-25, No.2, pp.115-133, 1976

[5] Lo, H.-Y., Ju, L.-P. and Su, C.-C., "General version of reconfiguration N modular redundancy system", *Circuits, Devices and Systems, IEE Proceedings* G Volume: 137 1 , Feb. 1990 , pp. 1 -4