論 文

Fault-Tolerance를 위한 시스템의 동작 방식에 대한 비교 연구

正會員 梁 城 鉉* 正會員 李 基 西*

Comparative Study of the System Operational Method for Fault-Tolerance

Sung Hyun Yang* Key Seo Lee* *Regular Members*

要 約

고장 방지 시스템은 하드웨어나 소프트웨어의 여분(Redundancy)을 이용하여 신뢰도(Reliablity) 및 안전도(Safety)를 향상 시킨다.

시스템의 대상 영역(application areas)에 따라 고장 마스크(fault mask), 고장 검출(fault detection), 고장 확인(fault identification)등의 기법을 선택하여 이용한다. 본 연구에서는 최소의 하드웨어와 소프트웨어의 여분을 이용하는 DMR(Double Modular Redundancy) 시스템을 대기 모듈(standby module)과 Fail-safe 모듈로 동작 시킬때 신뢰도와 안전도의 특성을 비교 제시한다.

또한 자기 진단 프로그램의 과도 오류 방지 능력에 대한 시스템의 MTTF를 비교함으로서 과도 오류를 취급하는 효과적인 방법을 제시하였다.

ABSTRACT

Fault-tolerant system is improved the reliability and safety by using hardware and software redundancy.

Fault mask and detection, identification techniques are conditionally used with system's application areas. Here DMR system is operated with standby and fail-safe module method that has minimal hardware and software redundancy, then its reliabilty and safety comparison is presented respecively.

Also this paper propesed an effective methods of dealing with transient faults as compared system's MTTFs to transient faults tolerance capabilities of self-diagnosis program.

I. 서 론

열차의 고속화와 마이크로 일렉트로닉수(Micro Electronics : ME) 기술발전으로 CTC, ATC, 전자연동장치등 신호시스템은 점차 컴퓨터화 되어가고

*光云大學校 電氣工學科
*Dept. of Electrical Engineering, Kwang Woon Univ.
論文番號 : 92-127(接受(接受1992. 7. 28)

있다. 신호시스템에 마이크로일렉트로닉스(ME) 기술을 이용할때, 가장 중요한 문제는 고장방지(fault-tolerant) 기능과 안전도(safety)특성을 병합 시키는 것이다.[1],[2],[3]

위와 같은 목적을 성취하기 위해서는 제어기 설계에 있어서 부가적인 하드웨어와 소프트웨어를 준비하여야 하며 이에 대한 연구는 1960년대의 J.Roth의 조합회로 검사를 위한 알고리즘 개발을 시작으로, 1970년대 후반 항공기 제어(aircrafrt control)를 위해서 A.Hopkins와 John H. Wensley에 의한 FTMP (Fault-Tolerant Multiprocessor)[4]와 SIFT(Software Implemented Fault Tolerance)[5] 시스템등을 설계 하였으며 오늘날 Tandem 시스템 개발 까지 지난 40년동안 연구가 수행되어 왔다. Fault-Tolerance 연구는 대형 컴퓨터의 설계자로 부터 오류 정정 코드를 이용하는 Reliability-Specific칩 뿐만 아니라 Intel 432와 같은 완전한 시스템을 생산하는 반도체 칩의 설계자까지 수행해온 결과 프로세서가 또다른 구조로 계산 성능에서 강력한 칩이 되었기 때문에 시스템내에 다른 위치에 다른 방법으로 적용하여 다중 구조 시스템을 구성 할 수 있게 되었다. 또한 시스템의 설계자는 이미 개발된 기술에 의해 설계된 시스템의 장점과 제약조건을 이해 해야 하며 시스템 응용분야에 따른 이러한 장점과 제약 조건의 영향을 결정하여 시스템의 임무가 성공적으로 완성될수 있게 시스템의 특성을 명세화 해야 한다.

따라서 고장 방지 시스템에 대한 연구는 설계시 제시된 신뢰도, 안전도에 대한 정확한 평가 기술에 관심이 모아지고 있으며[6] 이에 대한 연구로 Barry W.Johnson등은 전기 휠체어(electric wheelchair)의 제어를 위한 시스템의 평가 비교를 연구 하였으며 David B. Turner 와 Roger D. Burns등은 응용 시스템(application system)이 지향 하는 응답에 따라서 NMR 시스템을 Fail-Passive, Fail-Operational로 분류하여 시스템을 해석 하였다.[7]

본 연구에서 신호시스템의 핵심이 되는 DMR (Double Modular Redundancy)구조의 Fail-Safe 방식과 Standby 방식에 대하여 Markov 모델을 이용한 신뢰도와 안전도를 평가하여 그 특성을 비교 검토하였으며, 자기 진단 계산 동안에 발생 할수있는 과도 유류와 보수율에 따른 시스템의 MTTF를 계산 함으로서 과도 오류를 취급하는 효과적인 방법을 제시했다.

Ⅱ. DMR 시스템

시스템에 대한 고신뢰도와 안전도가 요구될때, 시스템은 과도오류와 영구오류를 방지 할수있는 구조가 되어야 하며, 이에 대해 하드웨어와 소프트웨를 통한 여러가지의 신뢰성 있는 구조가 제시되고 있다.[2],[8] DMR 시스템은 의도하는 시스템의 동작에 따라 Hot-Standby 방식과 Cold-Standby 방식, Fail-Safe 방식으로 분류 할수있다.

Ⅱ-1. Hot Standby 방식 [그림 1(a) 참조]

Hot-Standby 시스템은 두개의 모듈에서 프로그램을 수행하는 동안 공통 클럭에 의해서 동기화 되어 동작 한다.

시스템은 같은 기능을 수행하는 두개의 프로세서 M1과 M2를 포함하며, 출력은 두 모듈사이의 상이함(discrepency)을 검출하는 비교기 C 에서 비교된다. 만약 2 개의 프로세서의 출력이 다르면 비교기는 오류신호 Ec를 동작(active)한다. Ec 신호에 의해서 각 프로세서는 고장 프로세서를 찾기 위해 자기 진단 계산(self-diagnostic routine)을 수행 하며, 신호 E1과 E2를 통해서 마이크로프로세서 자신의 오류를 나타낸다. 이러한 세개의 신호(E1,E2,Ec)는 조절기 회로에 입력으로 작용하로 완전한(fault-free)프로세서의 신호를 출력한다. M1이 고장이면 스위치회로는 s=s2이고, M2가 고장이면 s=s1이 된다.

만약, 하나의 모듈이 고장이고, 그 고장이 자기진단에 의해서 검출되지 않으면, 비교기는 출력 Ec를 동작하는 방법으로 비교기 C의 기능은 시스템의 안전도를 증가 시킨다. 이경우 고장의 원인이 되는 모듈을 검출 할수 없기 때문에 스위치는 출력을 변화 시킬수 없게 된다.

Ⅱ-2. Cold- Standby 방식 [그림 1(b)참조]

Cold-Standby 시스템은 시스템이 정상동작을 하는 동안 두개의 프로세서중 하나로 모든 프로그램을 수행하는 On-line 유닛으로 선택된다.

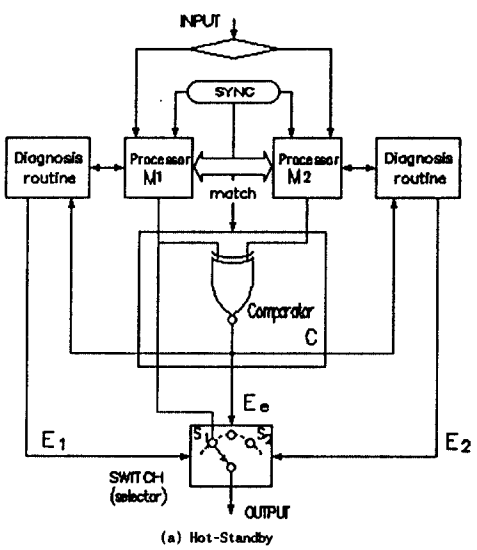
동시에 발생하는 고장을 검출하기 위해 고장 검출 계산(fault detection routine)이 On-line 프로세서에서 수행된다.

On-line 유닛에서 고장이 검출되면 여분 프로세서가 고장난 프로세서의 기능을 수행한다.

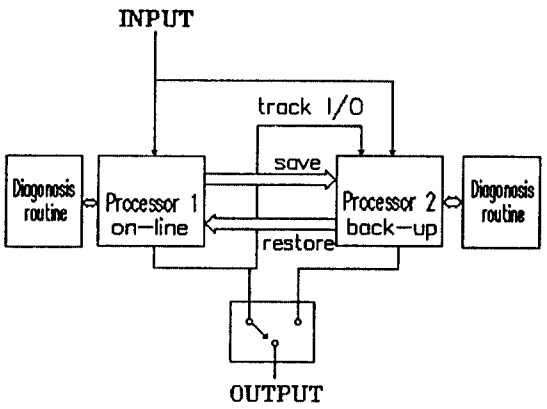
Ⅱ-3. Fail-safe 시스템

Fail-Safe 시스템은 시스템을 구성하고 있는 한 부분에서 고장이 발생한 경우에 안전측으로 동작 하거나, 잘못된 데이터가 출력 되지 않아야 한다. 이러한 완전성(integrity)은 검사중복(check redundancy)에 의해서 성취할 수 있다.

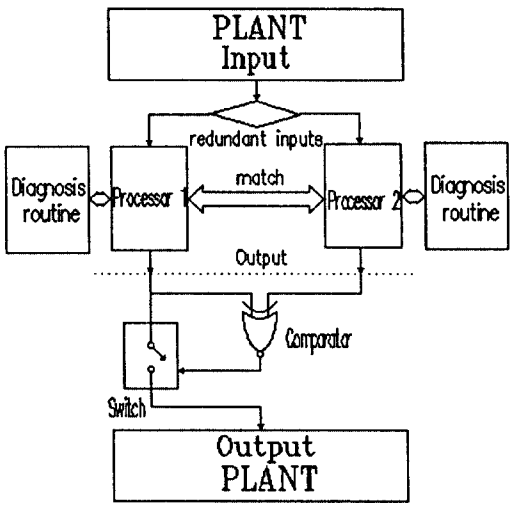
완전성은 오류검출을 요구하며, 이러한 목적으로 도입된 여분이 검사 중복이며, 검사 중복을 이용하여 검사하는 정보의 위치는 오류 잠복(error latency)을 짧게 하고 이중오류를 방지 하기 위해 버스 층(bus level)에서 한다. 일반적으로 복잡한 구조에 대해서는, 기능적으로 동일한 두개의 컴퓨터가 병렬로 같은 프로그램을 수행하고, 출력을 비교함으로서 오류를 검출한다. 또한 자기 진단 계산을 이용한 오류 검출은 완전성을 제공할 수 있다. 특히 신호 시스템에서는 안전측(safe-side)이 정의 되어 있음으로 시스템에 대해서 안전성은 매우 중요하며, 소프트웨어나 하드웨어중 어느 한 방식으로는 불충분하다. 지속성(persistency)을 위한 방법으로 앞에서 언급한 대기 모듈(standby module)의 이용이 일반화 되고 있지만, 잘못된 정보(wrong data)출력을 피하고, 완전성 유지가 우선하는 신호시스템에서는 첫번째 검출되는 오류에서 시스템이 안전측으로 동작 함으로서 안전도를 향상 시킬수 있다.



(a) Hot-Standby



(b) Cold-Standby



(c) Fail-safe Methods

그림 1. DMR 시스템의 블록선1도 (a)Hpt-Standby, (b)Cold-Standby, (c)Fail-safe 방법.

Fig 1. Block diagram of DMR (a)Hot-Standby, (b)Cold-Standby, (c)Fail-safe Methods.

Ⅲ. DMR 시스템에 대한 MARKOV 모델

비교기를 제외한 시스템을 구성 하는 전체 부품위 고장율(failure rate) λ는 Bathtub 곡선을 따르며, 지수 분포(exponential distribution)에 의한 Hazard 함수 Z(t)=λ이다. 두 모듈에 대한 비교기의 고장 검출의 확률은 C_c이며, 자기진단 계산에 의한 고장 확인 범위(fault location coverage)는 C_l, 재구성 과정(reconfiguration pross)에 대한 재구성 범위(recon-

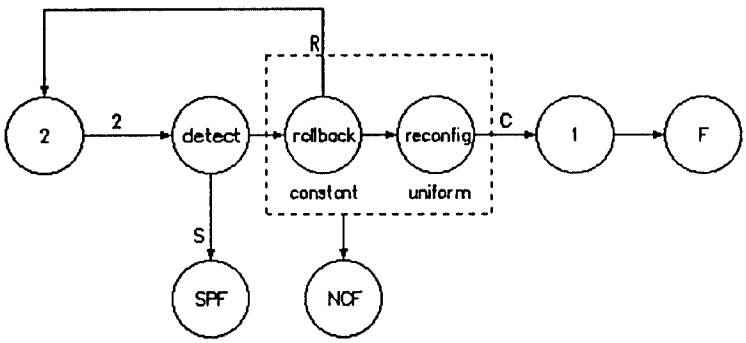
figuration coverage)C_2로 표시 한다.

DMR 시스템에 대해 위와 같은 요소를 고려한 준 Markov(semi-Markov)모델은 그림 2와 같다.

그림에서 상태2는 두개의 프로세서가 정상적으로 동작 하는 상태, 즉 초기에 시스템이 완전한 상태를 의미하며, 어느 하나의 프로세서에서 고장이 발생하고 성공적으로 그 고장이 검출되면서 동시에 상황에 알맞게 취급된 경우가 상태 1이다. 이상태에서 시스템은 하나의 고장 프로세서와 시스템의 기능을 수행하는 완전한 프로세서로 구성된다. 또한 시스템에 고장이 발생해서 시스템의 동작을 보장 할수없는 상태로서 결과에 대한 신뢰성이 없는 시스템 상황을 상태 F로 정의한다.

따라서 초기에 완전한 프로세서중 하나에서 고장이 발생했을 경우, 고장이 검출되고 자기 진단에 의해서 고장 확인이 되었을때 상태 2에서 상태 1로 전이가 발생한다. 따라서 전이가 가능한 확률은 $2\lambda C_c C_1 C_2 \Delta_t$가 되며, 비교기에서의 오류와 자기진단이나 재구성 과정에서의 오류를 고려할 수 있는데 각각 SPF(Single Point Failure)와 NCF(Near-Coincident)상태로 전이하게된다.

상태 2에서 상태 F로의 전이는 비교 과정에서 또한 자기진단 과정에서도 고장을 검출하지 못한 경우 발생한다.



범례. 1)범위(coverage)(C) : 오류 검출과 고장 모듈의 고립(isolation), 그리고 재구성이 성취 될수있는 확률

2)회복(restoration)(R) : 과도 오류(transient error)에 대해 시스템이 정해진 상태로 회복 될 수 있는 상태

3)SFP(single point failure)(S) : 고장이 임계 영역(critical region)으로 전파되어 시스템이 고장날 확률

그림 2. 2DMR에 대한 고장 방지 과정
Fig 2. Fault-Tolerant process of DMR

상태 1에서의 전이는 완전한 프로세서의 고장이 발생하는 경우인데 이 경우, 자기 진단만이 고장을 검출할 수 있는 유일한 장치이다.

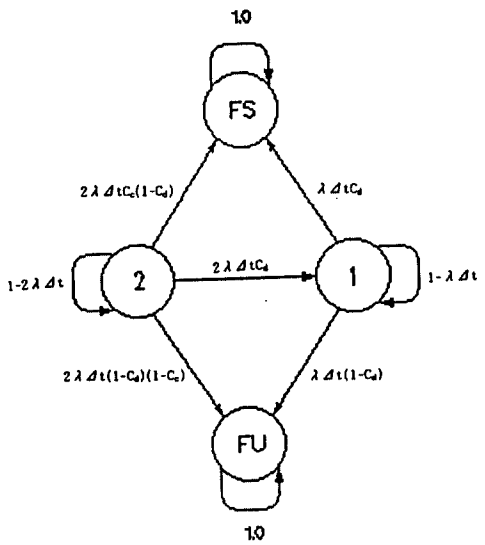
따라서 고장을 검출하지 못하면 상태 F로 전이하게 된다.

신호 시스템은 제어기 신호의 출력에 의한 시스템의 결과에 대해회복이 불가능한 순차적 시스템으로 제어기에 대한 완전성과 안전도가 무엇보다도 중요하다. 따라서 시스템을 구성 하고 있는 어느 부분에서 고장이 발생해서 기능을 중단하는 경우에도 효과적인 방법으로 고장을 취급하여, 전체적 시스템에 대해 이미 정의된 안전측으로 시스템이 동작할 수 있도록 하는 상태는 시스템이 수행하는 동작을 중단하지만 고장이 검출되고 안전하게 취급되거나, 비교기 자체에서의 고장 발생시에도 시스템이 안전측으로 동작할수 있게 하는 경우로 이러한 상태를 FS로 정의

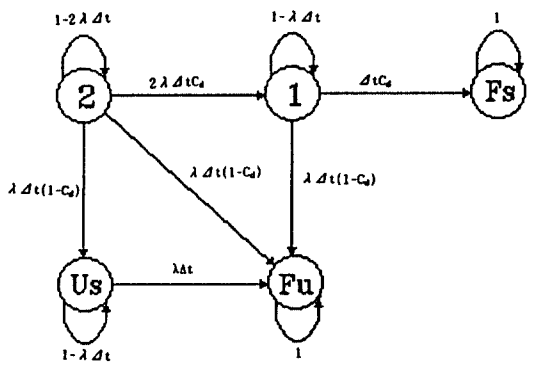
한다.

따라서 FSC(Fail-Safe Comparator)[6]를 고려한다면 그림 2의 SPF 상태는 FS 상태로 대체될 수 있으며 자기진단 과정에서의 두번째 고장은 언제나 시스템 고장으로 분류할 수 있으므로 NCF 상태는 특별히 고려하지 않는다.

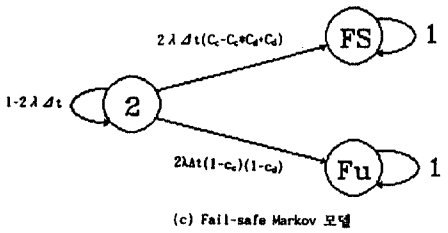
위의 정의에 의해서 Hot-Standby와 Cold-Standby 시스템의 Markov 모델은 그림 2의 Markov Model을 변형하여 그림 3의 (a),(b)와 같으며, 초기 상태에서 첫번째 고장이 발생하면 제어 신호 출력을 중단하는 Fail-Safe 시스템의 모델은 그림 3의 (c)와 같다.



(a) Hot-Standby Markov 모델



(b) Cold-Standby Markov 모델



(c) Fail-safe Markov 모델

그림 3. DMR 시스템의 이산시간 MARKOV 모델
(a)Hot-Standby, (b)Cold-Standby, (c)Fail-safe.

Fig 3. Discrete-time model of DMR system
(a)Hot-Standby, (b)Cold-Standby, (c)Fail-safe
(a)Hot-Standby Markov 모델
(b)Cold-Standby Markov 모델
(c)Fail-safe Markov 모델

Ⅲ-1. 신뢰도의 평가

1) Hot-standby 시스템

그림 3의 (a)에서 시스템의 Markov모델은 4개의 시스템 상태를 포함한다. 시스템이 시간 t+Δt에서 임의의 주어진 상태 S에 존재할 확률은 시스템이 어떠한 상태에서 주어진 상태 S로 전이(transition)할 수 있는 확률과 상태 S로 전이가 발생할 확률로 정의되므로 보(repair)를 고려하지 않을 경우 모델의 방정식은 다음과 같다. 먼저 수학적 항은 정의에 의해서 다음과 같다.

$$
\begin{aligned}
P_1(t+\Delta t) &= (1-2\lambda\Delta t)P_1(t)\\
P_2(t+\Delta t) &= 2\lambda\Delta t C_d P_1(t) + (1-\lambda\Delta t)P_2(t)\\
P_{fs}(t+\Delta t) &= 2\lambda\Delta t Cc(1-Cd)P_1(t)\\
&\quad +\lambda\Delta t CdP_2(t) + P_{fs}(t) \qquad (1)\\
P_f(t+\Delta t) &= 2\lambda\Delta t(1-C_d)(1-Cc)P_1(t)\\
&\quad +\lambda\Delta t(1-Cd)P_2(t) + P_f(t)
\end{aligned}
$$

위의 항들을 행렬 형태로 표현하면, 다음 식(2) 와 같이 동차 방정식(homogeneous equation)이 된다.

$$
\begin{bmatrix} P_1(t+\Delta t)\\ P_2(t+\Delta t)\\ P_{fs}(t+\Delta t)\\ P_f(t+\Delta t)\end{bmatrix} =
$$

$$
\begin{bmatrix} 1-2\lambda\Delta t & 0 & 0 & 0\\ 2\lambda\Delta t Cd & 1-\lambda\Delta t & 0 & 0\\ 2\lambda_t Cc(1-Cd) & \lambda\Delta t Cd & 1 & 0\\ 2\lambda\Delta t(1-Cd)(1-Cc) & \lambda\Delta t(1-Cd)P1(t) & 0 & 1\end{bmatrix}
\begin{bmatrix} P_1(t)\\ P_2(t)\\ P_{fs}(t)\\ P_f(t)\end{bmatrix}
$$

$$P_h(t+\Delta t) = T_h P_h(t) \quad (2)$$

여기서 C는 자기 진단과 재구성능력에 대한 결합 인자($C_d = C_1 * C_2$)이고, Δt는 시간 증가 이다.

Fail-Passive 시스템의 신뢰도는 시스템이 완전하게 동작가능한 상태로서 상태(1)이나 상태(2)에 존재할 수 있는 확률로 나타낼수 있으며 다음 식(3)과 같다.

$$R_h(t) = P_1(t) + P_2(t) \quad (3.1)$$
$$S_h(t) = P_1(t) + P_2(t) + P_f(t) \quad (3.2)$$

2) Cold-standby 시스템

그림 3의 (b)에서 Cold-standby Spare시스템의 Markov 모델을 보여준다.

Cold-Standby Spare 시스템은 5개의 상태를 포함한다.

상태 2는 두개의 프로세서가 완전한 초기 상태를 나타내며, 상태 1은 On-line 프로세서가 고장이고 Spare가 성공적으로 On-line으로 실행된 경우 또는 Spare에 고장이 발생했을경우 고장이 검출되어 Spare의 보수를 끝낸 상태이다. 이때 시스템은 동작가능하다.

상태 FS는 On-line 프로세서와 Spare 프로세서가 고장나고, 양쪽의 고장이 검출되어 시스템이 안전측으로 동작한 상태이다.

상태 FU는 불안전측으로 시스템이 동작한 경우이며, 1)On-line unit의 고장 검출 실패 2)Spare 프로세서의 고장 감출 실패와 On-line unit이 고장일때 spare의 사용이 실패한 경우이다.

이때의 시스템은 검출되지 않은 고장 프로세서로 동작하며 시스템의 상태가 2나 1일때 On-line unit의 Spare가 고장 검출 불능인 상황으로 고장난 경우이기 때문에 On-line unit이 고장나고 Spare로 대체될 때 고장 Spare로 대체하는 경우가 발생한다.

따라서 그림 3의 (b)로 부터 식(4)가 유도 된다.

$$\begin{aligned}
P_1(t+\Delta t) &= (1-2\lambda\Delta t)P_2(t)\\
P_2(t+\Delta t) &= 2\lambda\Delta t C_d P_1(t) + (1-\lambda\Delta t)P_2(t)\\
P_{us}(t+\Delta t) &= \lambda\Delta t(1-C_d)P_1(t) + (1-\lambda\Delta t)P_{us}(t+\Delta t)\\
P_{fs}(t+\Delta t) &= \lambda\Delta t C_d P_2(t) + P_{fs}(t) \quad (4)\\
P_{fu}(t+\Delta t) &= \lambda\Delta t(1-C_d)P_1(t) + \lambda\Delta t(1-C_d)P_2(t)\\
&\quad + P_{fu}(t) + \lambda\Delta t P_{us}(t)
\end{aligned}$$

또는

$$\begin{bmatrix} P_1(t+\Delta t)\\ P_2(t+\Delta t)\\ P_{us}(t+\Delta t)\\ P_{fs}(t+\Delta t)\\ P_{fu}(t+\Delta t)\end{bmatrix} =$$

$$\begin{bmatrix} 1-2\lambda\Delta t & 0 & 0 & 0 & 0\\ 2\lambda\Delta t C_d & 1-\lambda\Delta t & 0 & 0 & 0\\ \lambda\Delta t(1-C_d) & 0 & 1-\lambda\Delta t & 0 & 0\\ 0 & \lambda\Delta t C_d & 0 & 1 & 0\\ \lambda\Delta t(1-C_d) & \lambda\Delta t(1-C_d) & \lambda\Delta t & 0 & 1\end{bmatrix} \begin{bmatrix} P_1(t)\\ P_2(t)\\ P_{us}(t)\\ P_{fs}(t)\\ P_{fu}(t)\end{bmatrix}$$

여기서는 C_d는 자기 진단 계산과 재구성 능력에 대한 결합인자이며, Cold-Standby시스템의 신뢰도는 상태 2, 상태 1, 상태 Fs, 상태 Us에 있을 확률로 식(7)과 같고 안전도는 식 (8)과 같다.

$$R_{ss}(t) = P_1(t) + P_2(t) + P_{us}(t) \quad (7)$$
$$S_{ss}(t) = P_1(t) + P_2(t) + P_{us}(t) + P_{fs}(t) \quad (8)$$

3) Fail-safe 시스템

시스템을 구성하고 있는 모듈에서 첫번째 고장이 검출되었을때 제어 신호의 출력을 차단하게 되어 있는 Fail-Safe 시스템의 상태는(그림3) 3개의 상태로 표현되며 이때 상태식은 같은 정의에 의해서 식(9), (10)로 나타낼수 있다.

$$\begin{aligned}
P_1(t+\Delta t) &= (1-2\lambda\Delta t)P_1(t)\\
P_{fs}(t+\Delta t) &= 2\lambda\Delta t(C_c - Cc * C_d + C_d)P_1(t) + P_{fs}(t) \quad (9)\\
P_{fu}(t+\Delta t) &= 2\lambda\Delta t(1-C)(1-C_c) + P_{fu}(t)
\end{aligned}$$

또는

$$\begin{bmatrix} P_1(t+\Delta t)\\ P_2(t+\Delta t)\\ P_3(t+\Delta t)\end{bmatrix} =$$

$$\begin{bmatrix} 1-2\lambda\Delta t & 0 & 0\\ 2\lambda\Delta t(C_c - C_c * Cd + C) & 1 & 0\\ 2\lambda\Delta t(1-C)(1-Cc) & 0 & 1\end{bmatrix} \begin{bmatrix} P_1(t)\\ P_2(t)\\ P_3(t)\end{bmatrix} \quad (10)$$

이때 시스템 신뢰도는 식(11.1)이며 안전도는 두개의 프로세서가 Fault-Free한 초기 시스템 상태와 어느 프로세서에서 고장이 발생했지만 고장이 알맞게 취급되어 이미 정해진 안전측으로 전체적인 시스템

이 동작한 상태인 FS에 존재할 확률의 합으로 식(11.2)와 같다.

$$R(t) = P_1(t) \quad (11.1)$$
$$S(t) = P_1(t) + P_2(t) \quad (11.2)$$

Ⅲ-2. 시스템의 과도 오류 해석

오류가 검출되면, 오류로 부터 시스템을 회복(recover)하기 위해 롤백(rollback)이나 재시작(restart) 과정이 수행되며, 오류발생의 경우 자기진단 프로그램에 의해서 시스템 재구성이 이루어지도록 설계된 DMR 시스템은 모듈이 기능적으로 완전한 경우에도 과도오류로 인하여 모듈을 제거 하게되는 결과가 될 수있다. 또한 일반적으로 전체오류의 90%가 과도오류인 경우로 알려져 있으며 이러한 과도오류가 연속적이거나 길어지는 경우 자기 진단 로그램은 오류를 검출할수 없게된다.

비교회로는 영구오류인지를 검출할수 없음으로 인해서 과도오류 회복절차(transient fault recovery process)는 오류가 과도오류에 의한 영향으로 확인되면 시스템이 원래상태로 초기화되는 보수과정을 포함해야한다. 따라서 과도오류에 대한 DMR 시스템에 대한 영향을 해석하기 위해 Markov 모델을 그림 4와 같이 제시하며, 과도오류에 따라 변화되는 MTTF의 값을 제시 한다

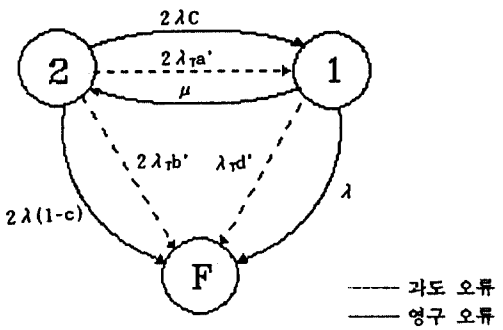


그림 4. 과도오류와 보수에 따른 시스템의 상태전이 선도
Fig 4. The state transition diagram of system from transient error and repair

여기에서

$a = a'\frac{\lambda_T}{\lambda}$, $b = b'\frac{\lambda_T}{\lambda}$, $c = d'\frac{\lambda_T}{\lambda}$ 이며, λ와 λ_T는 영구오류

(permanent error)와 과도오류(transient error)에 대한 고장율(failure rate)이다.

또한 μ는 보수율(repair rate)을 나타내며

a' : 영구오류로 검출되는 과도오류의 비율

b' : 상태 2에서 검출되지 않는 과도오류의 비율

d' : 상태 1에서 검출되지 않는 과도오류 비율을 나타낸다.

이때 시간 t에서 i번째 상태의 확률을 Pi(t)라 하면 다음과 같다.

$$P_1(t+\Delta t) = (1-2\lambda(1+a+b)\Delta t)P_1(t) + \mu\Delta t P_2(t)$$
$$P_2(t+\Delta t) = 2\lambda(c+a)\Delta t C P_1(t) + (1-\lambda(1+d+u)\Delta t)P_2(t) \quad (12)$$
$$P_f(t+\Delta t) = 2\lambda(1-c+b)\Delta t P_1(t) + \lambda(1+d)\Delta t P_2(t) + P_f(t)$$

다시 행렬로 나타내면,

$$\begin{bmatrix} P_1(t+\Delta t) \\ P_2(t+\Delta t) \\ P_f(t+\Delta t) \end{bmatrix} = \begin{bmatrix} 1-\lambda 2(1+a+b)\Delta t & \mu\Delta t & 0 \\ 2\lambda(c+a)\Delta t & 1-\lambda(1+d+\mu)\Delta t & 0 \\ 2\lambda(1-c+b) & \lambda(1+d) & 1 \end{bmatrix} \begin{bmatrix} P_1(t) \\ P_2(t) \\ P_s(t) \end{bmatrix} \quad (13)$$

과 같다.

상태 F는 시스템이 고장난 상태를 나타내며, 상태2와 상태1은 완전한 각각 모듈로 동작하는 상태와 하나의 모듈로 동작하는 상태를 표시한다.

확률 이론으로 부터 MTTF는 고장(failure)의 기대값(expected value)을 처음으로서 계산된다.[9], [10]

난 변수(random variable)x에 대한 기대값은

$$E[x] = \int_0^\infty f(x)dx \quad (14)$$

f(x) : 확률 밀도 함수(probability density function)

따라서 신뢰도 해석에 이용되는 MTTF는 다음과 같다.

$$MTTF = E[X] = \int_0^\infty tf(t)dt \quad (15)$$

f(t) : 고장 밀도 함수(failure density function)
본 시스템의 초기 상태를 다음과 같이 정의 할때

$$P2(0) = 1$$
$$p1(0) = pf(0) = 0 \quad (16)$$

시스템의 비신뢰성(unreliability)을 $P_f(t)$라 하면, 고장 밀도 함수는 다음과 같다.

$$f(t) = \frac{dP_f(t)}{dt} \quad (17)$$

윗식 (17)에 대해 Laplace 변환을 이용하면

$$F(s) = sP_f(s)$$

식 (13)에 의해서

$$P_f(s) = \frac{2\lambda}{s} \cdot \frac{s+\lambda(1+d)+\mu-(c-b)(s+\mu)+\lambda(1+d)(a+d)}{(s+\lambda(1+d)+\mu)(s+2\lambda(1+a+b))-\mu 2\lambda(c+a)} \quad (18)$$

이다.
따라서 식 (15)와 식 (17)에 의해서 MTTF는 다음의 식과 같다.

$$E(x) = -\frac{dF(s)}{ds}\Big|_{s=0} = \frac{\lambda(1+d+2(c+a))+\mu}{2\lambda[\lambda(1+d)(1+a+b)+\mu(1-c+b)]} \quad (19)$$

Ⅳ. 시뮬레이션 및 결과고찰

시스템이 고장을 방지하기 위해서는 하드웨어나 소프트웨어적인 기술 또는 두가지를 결합한 기술을 필요로 하며 이 두개의 매개변수에 의해서 시스템의 신뢰성은 결정될수 있다. 더욱 소프트웨어적인 기술에 대해서는 진단 프로그램 자체의 오류방지 능력이 결정될때 전체 시스템의 평가가 결정 될수 있다.
될수 있다.

따라서 본 절에서는 더욱 복잡한 NMR(N-Modular Redundancy)시스템에 응용을 목적으로 Ⅱ. 절에서 비교적 간단하게 제안된 모델에 대해 하드웨어와 소프트웨어적인 고장 방지 능력이 시스템의 신뢰도와 안전도에 미치는 영향을 컴퓨터 시뮬레이션을 통해 그 특성을 비교 분석 한다.

Hot Stadby 시스템에서는 두개의 Coverage Factor가 고려되어야한다.

1. 비교기가 고장을 검출할 수 있는 확률 C_c
2. 자기진단 계산이 고장을 검출하고, 재구성할 수 있는 확률 C_d

여기서 재구성의 의미는 자기진단 계산이 고장을 검출한후 시스템이 정상 동작상태에 있게 하거나 Shut Down하는 과정이다.

Cold Standby 시스템에서는 고장 검출과 재구성 절차에 있어서 단지 자기진단 계산에 의존하는 반면 Fail-Safe 시스템은 비교기의 고장 검출 확률에 의존한다.

컴퓨터 시뮬레이션에 이용한 비교기의 고장 검출률 및 고장율, 자기진단 계산의 범위 및 이 이외의 총 고장율(λ)은 다음과 같이 설정한다.

$$C_c = 0.9$$
$$C_d = 0.85$$
$$\lambda = 1.71*10^{-5}[h^{-1}]$$

시스템의 신뢰도는 구성부품의 고장율, 범위인자와 시간에 종속된다.

여기에서 전자적인 고장율은 Gate Level에서 입증되는 것으로, 본 연구에서는 고려하지 않으며, 그림 3의 Markov 모델의 시스템은 부품 고장율 λ, 비교기의 고장 검출 범위 Cc, 자기진단 계산의 고장검출 범위 C_d를 갖는 하드웨어 모듈로 구성되어 있으므로, 신뢰도에 대한 모델식은 식(3),(7)과 식(11)과 같다.

결과에 대한 유연성(flexibility)을 증가시키기 위해서 λt의 함수로서 신뢰도와 안전도를 실험함으로서 자기진단 프로그램의 기능(C_d)은 0.85로 가정하여 비교기의 고장범위에 따른 신뢰도와 안전도의 영향을 실험했다.

세 방식에 대한 시스템의 신뢰성 비교는 식(3.1),(7),(11.1)에 의한 결과로 그림 5에서 나타내며, 이때 시스템 안전도는 식(3.2),(8),(11.2)의한 결과 그림 6

과 같다.

결과에서 알 수 있듯이 시스템의 신뢰도는 Cold-Standby가 우수하지만 안전성에 대해서는 Fail-Safe 시스템이 더 높다. 따라서 제어기의 출력이 인명이나 환경의 안전에 직결된다면 Fail-Safe 시스템으로 제어기를 구현 해야 하며 시스템의 신뢰도는 Cold-Standby가 높지만 이 모델은 고장 검출및 시스템의 재구성을 자기진단 계산에 의존하기 때문에 다음 결과에서 알수있듯이 자기진단 프로그래밍의 과도 오류 방지기능이 더욱 요구된다.

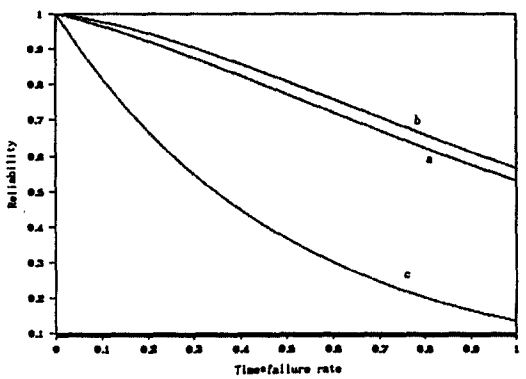


그림 5. DMR 시스템에 대한 신뢰도 비교
(a)Hot-Standby, (b)Cold-Standby, (c)Fail-safe
Fig 5. Comparision of reliability for DMR system
(a)Hot-Standby, (b)Cold-Standby, (c)Fail-safe

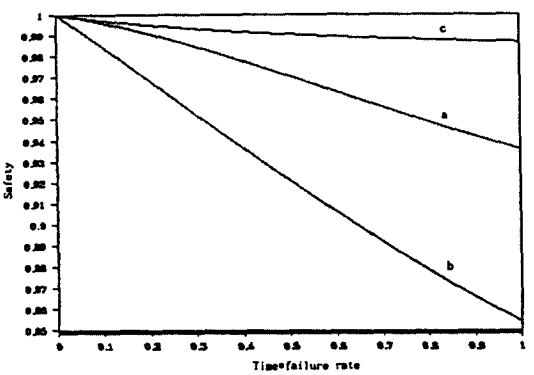


그림 6. DMR 시스템에 대한 안전도 비교
(a)Hot-Standby, (b)Cold-Standby, (c)Fail-safe
Fig 6. Comparision of safety for DMR system
(a)Hot-Standby, (b)Cold-Standby, (c)Fail-safe

또한 오류의 종류에 대해서 과도오류인 경우 비교기의 검출 능력에 의해서 고장 모듈을 제거하기 보다는 시스템이 원래 상태로 초기화 될수있는 오류 회복 절차가 이루어져야 한다. 오류에 대한 판단을 자기진단 프로그램에서 수행한다고 할때 과도 오류의 지속이 길어지거나 연속적일때 불가능 하다. 따라서 과도 오류에 의한 시스템의 영향을 해석 하기위해 그림 4에의한 모델로 다음관계에 따른 MTTF를 식(19)에 의해서 실험 하였다.

$$\mu/\lambda = M \tag{20.1}$$

$$\lambda_t/\lambda = N \tag{20.2}$$

식(20.1)은 고장율에 대한 회복율을 나타내며 식(20.2)는 영구오류 발생율에 대한 과도 오류의 발생율을 표시한다.

실질적으로 M>>1이며, N은 시스템 환경에 의해서 10-100의 범위에 존재 한다.

또한 a', b', d'가 그림 4의 상태 2에서 영구 오류로 검출되는 과도오류의 비율, 검출되지 않는 과도 오류의 비율, 상태 1에서 검출되지 않는 과도오류의 비유로서 식(21)을 유도 할수있다.

$$a = a'\frac{\lambda_T}{\lambda},\ b = b'\frac{\lambda_T}{\lambda},\ c = d'\frac{\lambda_T}{\lambda} \tag{21}$$

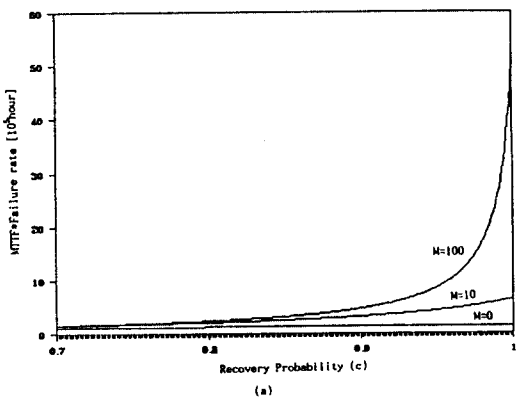
따라서 a,b,d는 전체 영구오류에 대한 과도 오류의 고장율중 각 상태에서 영구오류로 취급되거나 검출되지 않는 과도 오류를 표시함으로 상수(constant) 또는 100[%]로 나타 낼수있다.

그림 7의 (a)는 a=b=d=0일때 M=0, M=100 경우 영구오류를 검출할수 있는 C의 변화에 따른 MTTF의 결과를 나타내며, (b)는 과도오류가 영구를 취급되어진 경우 MTTF의 변화 이다. 이때 =0, C=0.85일때 M=0일때 M=0,M=10이다.

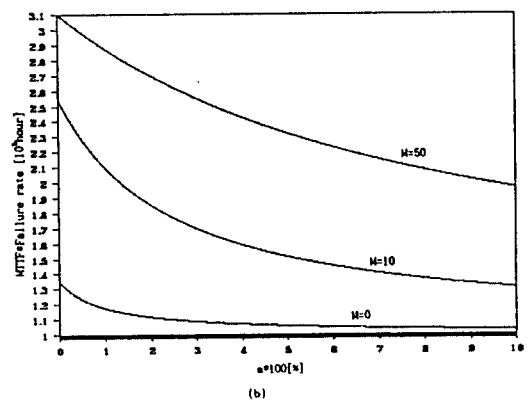
역시 C=0.85 이며 a=d=0, a=b=0 일때, 검출되지 않는 과도오류를 진단 프로그램만으로 오류를 검출하는 성능 저하된 시스템의 오류 검출 능력을 실험한 결과는 그림 7의 (c),(d)이다. 실험 결과에서 진단 프로그램의 재시행(retry)이나 복구 작업(recovery process)의 기능이 시스템의 성능에 미치는 영향을 알수 있었으며 이와 같은 자기진단 프로그램의 테스팅 시간은 시스메의 MTTF에 또다른 변수가 된다는 것을 알수 있었다.

이에 대해서 Built-in-test 능력이 있는 마이크로 프

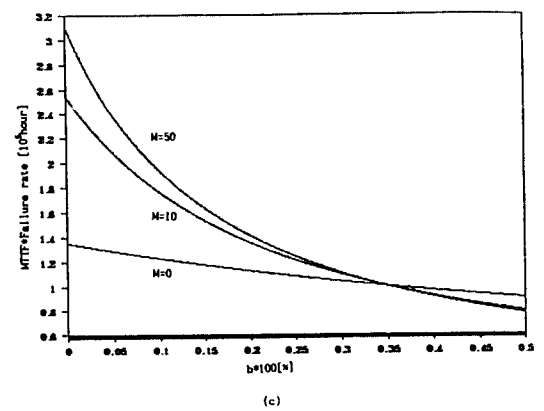
로세서를 이용하는 방법이 제시되고 있으나 이에대한 신뢰성이나 안전성에 대한 데이타가 보장되지 않았기 때문에 실제 이용하기에는 문제 발생의 소지가 있다. 따라서 소프트웨어적인 방법으로 고장을 검출하는 경우에도 과도오류를 Masking 할수있는 하드웨어구조와 함께 On line으로 수행 할수있는 자기진단 프로그램이 설계 되어야 할것이다.



(a)



(b)



(c)

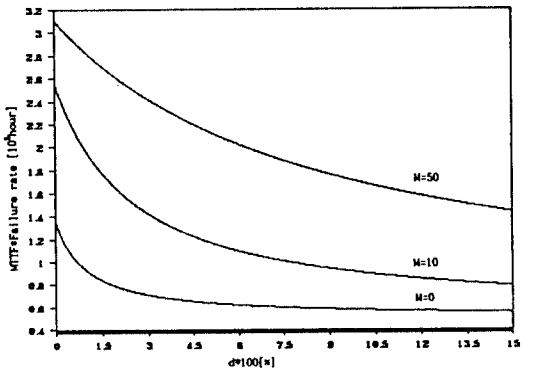


그림 7. DMR 시스템에 대한 오류검출 확률에 따른 MTTF(E[x])
(a)영구오류에 대한 고장검출(c)의 결과
(b)과도오류를 영구오류로 판단한 오류에 의한 결과
(c,d)각 상태에서 검출되지 않은 임계적 과도오류에 의한 결과

Fig 7. The effect of error detection capabilities on MTTF(E[x])
(a)The effect of detection coverage(c) for permanent faults
(b)The effect of false qualitying transient faults as permenent faults
(c,d)The effect of undetected critical transient faults

Ⅵ. 결 론

본 논문에서는 신호 시스템의 제어를 목적으로 DMR 구조에 대해 안전도와 신뢰도를 향상 시킬수 있는 시스템 구성 문제를 고찰 하였다.

Hot-standby 방식으로 제어 신호가 출력 될때 시스템의 지속성은 향상 할수 있었으나, 시스템의 모듈중 첫번째 고장 발생후 하나의 모듈로 동작 하는 동안 시스템의 완전성이 보장 될수 없으며, Cold-standby 방식은 임계적인 시스템에 대해서 진단 프로그램만으로 고장 검출 기능 수행함으로서 오류의 90% 가능성을 포함 하는 고도오류 발생시 오류 잠복(Errorlatency) 시간에 따라 시스템의 MTTF 계산으로 알수 있었다.

이에 비해 Fail-Safe 시스템은 시스템이 정상동작을 하는동안 언제나 비교기나 자기 진단 프로그램에 의해서 고장 검출이 이루어지고 첫번째 고장과 동시

에 제어 신호가 차단 될수 있음으로 시스템의 완전성을 보장할수 있었다.

따라서 시스템의 지속성은 Standby 방식이 높지만 Fail-Safe 방식이 시스템의 완전성을 더 보장 할수 있다는것을 알수 있었다. 그러나 이상적인 신호 시스템은 지속성과 안전성이 보장 될수있는 구조 이여야 함으로 Standby방식의 지속성과 Fail-Safe 방식의 안전성을 결합한 시스템에 대한 연구가 수행 되어야 할것이다.

참 고 문 헌

1. B. W. Johnson, "Fault-Toleranct Microprocessor-Based System"IEEE Micro, vol.4, No.6, dec. 1984 pp.6-21.
2. B. W. Johnson, "Design and analysis of fault tolerant digital system" Addison-Wesley Publishing Company, 1989.
3. Daniel P. siewiorek, "fault Tolerance in Commerical Computer"IEEE Computer 1990. pp. 26-37.
4. ALBERT L. HOPKINS, T. BASIL SMITH, JAYNARAYAN H. LALA. "FTMP-A high reliable fault-tolerant multiprocessor for aircraft" Proc. IEEE, VOL. 66. pp.1221-1239, Oct. 1978
5. J. Wensley, "SIFT: The design and analysis of a fault-tolerant computer for aircraft control," Proc. IEEE, VOL. 66, pp.1240-1255, Oct. 1978
6. Katsuji Akita, Hideo Nakamura, "Safety and Fault-Tolerance in computer-controlled signalling System" QR. of RTRI, vol.31, No.2. 1990, MAY, pp.95-103
7. Daniel p. siewiorek, "Architecture of Fault-Toleant Computers" IEEE Computer.1984. oo. 9018
8. Hubert K. Kirrmann, "Fault-Tolerance in process control" IEEE Micro, 1987,pp. 27-50
9. J. Sonsnowski, "Fault-Tolerant DNR Microprocessor system" IFAC SAFECOMP '89,1989, pp.144-146.
10. H. S. Trived, "Probability and statistic with reliability, queuing, and computer science application." Prentice Hall, 1982.

梁 城 鉉(Sung Hyun Yang) 正會員
1958年 2月 1日生
1985年 2月 : 光云大學 電氣科 卒業(工學士)
1988年 2月 : 光云大學校 大學院 電氣工學科 卒業(工學碩士)
1988年 3月 ~ 現在 : 光云大學校 大學院 電氣工學科 博士課程
※主關心分野 : Fault Tolerant System, Digital System Design & Testing 等임

李 基 西(Key Seo Lee) 正會員
1951年 1月 18日生
1977年 2月 : 延世大學校 電氣工學科 卒業(工學士)
1979年 2月 : 延世大學校 大學院 電氣工學科 卒業(工學碩士)
1986年 8月 : 延世大學校 大學院 電氣工學科 卒業(工學博士)
1988年 : Yale University 交換 教授
1981年 ~ 現在 : 光云大學校 制御計測工學科 教授
※主關心分野 : Neural Network學習알고리듬, Fault Tolerant Computer Control, 鐵道신호 等임