'2000년 춘계학술발표 논문집
한국원자력학회

디지털 시스템의 고장 내구성 기법에 대한 확률론적 안전성 평가

Probabilistic Safety Assessment on the Fault-Tolerant Mechanism of Digital I&C Systems

강현국, 성태용, 이기영, 박종균

한국원자력연구소
대전광역시 유성구 덕진동 150

요 약

디지털 기기가 단순화, 표준화, 운영, 유지보수 측면에서 많은 장점이 있음에도 불구하고, 소프트웨어를 포함하는 디지털 기기에 대한 정량적인 신뢰도 평가 방법이 확보되어 있지 않아 원전에의 적용에 어려움을 겪어 왔다. 디지털 기기의 다양한 계층(소프트웨어, 하드웨어, 시스템)에 다양한 방법으로 적용되고 있는 고장내구성 기법들을 확률론적 안전성 평가에 반영하기 위한 연구가 필수적인데, 본 논문에서는 이러한 기법들을 분류, 정리하였다. 또한 비교적 단순한 알고리즘을 가지는 고장내구성 기법이 원전 안전계통에 적용될 것으로 판단되므로, 이에 대한 확률론적 안전성 평가 방법론을 기술하였다. 평가 과정에서 적용되는 고장 검출률 값이 결과에 큰 영향을 미치게 되므로 각 고장내구성 기법 적용시의 고장 검출률에 대한 정량적 평가가 시급한 것으로 파악되었다.

Abstract

There are various problems in applying the digital equipment including software to the safety-related system of a nuclear power plant because no standard on quantitative safety assessment is well-accepted. Especially, the fault-tolerant features which is one of the most beneficial aspects of a microprocessor-based system should be evaluated quantitatively in order to assess the safety of a digital system. This paper describes the fault-tolerant features of digital systems which can be applied to software, hardware or system. For the case of watchdog timer which is expected to be the most competitive fault-tolerant mechanism for nuclear power plant's safety systems, this paper show an example of the process of probabilistic safety assessment. The estimation of the coverage factor value of applied lerant mechanism is found to be very important.

제 1 장 서론

1-1. 디지털 시스템의 원자력 발전소 안전계통에의 적용

아날로그 기기에 비해 데이터의 전송과 처리 능력이 뛰어나며, 보다 정확하고 신뢰성있게 신호를 처리할 수 있다는 장점 때문에 디지털 기기들의 사용이 급속히 확산되어 기존의 아날로그 기기를 거의 완전히 대체하고 있다. 원자력 분야에서도 디지털 계측제어계통의 전면 채택이 전세계적으로 연구·설계중인 차세대 원자력 발전소의 중요한 특징 중의 하나가 되고 있다. 그러나 여러 가지 장점에도 불구하고 디지털 계측제어계통이 가지는 불확실성 때문에 세계 각국의 안전규제기관의 견해가 신중하고 유보적인 입장을 견지하고 있어 이러한 설계의 실제 채택에 어려움으로 작용하고 있다.

원자력 발전소의 경우와 같이 안전성이 중요시되는 시스템에서는 안전 관련 계통을 다중으로 처리하게 된다. 안전 관련 디지털 계측 제어 시스템의 경우에는 전체 시스템에서 다중성을 가질 뿐만 아니라 내부의 소프트웨어나 하드웨어에서도 다중성을 가지거나 자기 감시 등 여러 가지 고장내구성 기법들이 적용된다. 이러한 기법들이 적절히 적용될 경우, fail-safe 개념에서 큰 신뢰도 향상을 기대할 수 있다. 전체 시스템의 신뢰도 분석에 있어서 이러한 다중·자기감시 기능의 신뢰도는 대단히 큰 영향을 미친다. 일반적으로 디지털 계측제어 시스템에 고장내구성 기능을 추가함으로 인해 시스템의 복잡도가 높아지더라도, 그것은 자기검사로 인한 신뢰도 향상의 장점을 상쇄할 만한 것은 아니기 때문에 이러한 자기감시 기법의 적극적인 도입이 강조되고 있다. 그러기 위해서는 먼저 정확한 평가 방법론의 개발이 선행되어야 한다.

1-2. 고장 내구성 기법과 안전성 평가

디지털 시스템에 고장내구성 기법의 적용할 경우, 시스템의 구조가 복잡하게 되고, 내부의 논리 흐름이 불연속적이 되는 경우가 많아 확률론적 안전성 평가(probabilistic safety assessment; PSA)에 난점으로 작용하게 된다. 소프트웨어로 구현된 고장내구성 기법들은 매우 다양한 알고리즘을 적용할 수 있기 때문에 분석의 난이도가 더욱 높아진다. 그러므로 이러한 기법들에 대한 체계적인 분석 및 확률론적 안전 분석 적용성 검토가 필수적이다. '디지털 계통내의 고장내구성을 확보하기 위해 적용된 다양한 기법들을 어떻게 안전성 분석에 반영할 것인가'하는 문제는 디지털 계측제어 계통 적용의 장점을 정량적으로 분석할 수 있다는 측면에서 반드시 고려되어야 할 것이다.

고장 내구성이 있다는 개념은 결함의 존재 가능성을 인정하고 그에 대해 보다 강인한 특성을 갖는 것을 말한다. 결함에 대해 강인한 특성을 갖는 방법은 크게 두 가지로 분류될 수 있는데, 첫번째는 소재나 알고리즘의 특성상 어느 정도 수준의 외란(disturbance)에는 근본적으로 영향을 받지 않는 경우이고, 두번째는 능동적으로 결함의 발생여부를 감시하고 대처할 수 있는 기능을 갖춘 경우이다.

능동적인 고장 감시 기법들을 다시 두가지로 분류하여 파악할 수 있는데, 회로 수준(circuit-level)의 기법과 시스템 수준(system-level)의 기법으로 구분할 수 있다. Error detecting codes for memories, parity bits for data buses, self-checking circuits 등이 회로 수준의 기법들의 예이며, Capability-based addressing, watchdog timers, fault-tolerant data structures, use of replication (N-version programming 등)이 시스템 수준의 기법들의 예이다. 회로 수준의 기법들은 하드웨어의 신뢰도 계산에서 반영된 것으로 보기 때문에 안전성 평가 모델에서 반영해야 할 부분은 시스템 수준의 기법들이다.

일반적인 고장내구성 시스템에서 능동적으로 그 기능을 구현하는 부분을 watchdog이라 한다. Watchdog은 결함에 대한 감시 및 대처의 역할을 수행하도록 설계된다. 감시 및 고장에 대한 대처의 방법은 적용의 대상에 따라서 다양하게 구현될 수 있는데, programmable logic controller (PLC) 등의 경우에서는 watchdog은 시간 초과(time-over)를 감시하여 시스템 정지(halt)를 감지하는 루틴을 일컫는다. 이 경우에는 감시 타이머(watchdog timer)라고 부른다. 소프트웨어 분야에서는 출력값을 감시하여 기준치를 초과할 시에는 다른 알고리즘과 프로그램 언어로 짜여진 모듈로 넘기는 역할을 담당하는 모듈을 watchdog이라고 한다 [1]. 이러한 복구 블록(recovery block)을 따로 두는 방법이외에도 여러 가지 버전의 소프트웨어를 설치하는 N-version 소프트웨어 기법 등의 방법으로 고장내구성 기능을 갖추기도 한다. 또 대기 시스템을 가지는 경우에는 시스템 정상 동작 여부를 감시하면서 비정상 동작으로 판단될 때에는 대기 시스템으로 기능을 넘기는 역할을 하는 시스템을 watchdog이라고 부른다. 다중 시스템의 경우에는 watchdog의 역할을 하는 시스템을 따로 두기보다는 대기 시스템이 감시 신호(heart bit 등)를 이용하여 watchdog의 기능을 수행하는 형태가 많다. 기존의 아날로그 시스템에서는 surveillance test를 포함한 주기적인 검사만이 가능하였으나 디지털 시스템의 경우에는 이러한 자기 검사 기능이 추가됨으로써, 연속적인 검사

(continuous testing)를 구현할 수 있다. 자기검사 기능을 가진 디지털 시스템을 안전 관련 기기에 이용하는 것에 대한 미국의 NRC의 입장은 ‘디지털 계측제어 시스템에 자기검사 기능을 추가함으로 인해 시스템의 복잡도가 높아지더라도 그것은 자기검사로 인한 신뢰도 향상의 장점을 상쇄할 만한 것이 아니다’라는 것이다 [2]. 즉, 이러한 검사 기능의 적극적인 도입을 권장하고 있는 것이다.

원자력 발전소의 안전 계통은 그 처리 내용이 단순하기 때문에 간단한 고장내구성 기법의 도입으로도 큰 신뢰도 향상 효과를 도모할 수 있다. 따라서 현재의 보호 계통 구조에서는 감시 타이머(watchdog timer)와 같은 기법의 적용이 가장 효율적인 불가용도 저감 방법이 될 수 있을 것으로 판단된다. 감시 타이머를 이용하므로써 현재까지 논란이 계속되고 있는 디지털 기기내의 소프트웨어 등으로 인한 공통원인 고장에도 효과적으로 대처할 수 있을 것으로 판단된다. 즉, 단순한 구조를 가지며 소프트웨어를 포함하지 않는 고신뢰도 감시 타이머를 도입하므로써 복잡한 구조의 디지털 기기의 사용에 따른 불가용도 증가를 억제하고 공통원인 고장에 효율적으로 대처할 수 있다. 그러나 감시 타이머를 고장내구성 향상을 위해 적용하였을 경우, 감시 타이머 자체의 신뢰도와 고장 감지 확률이 전체 시스템의 신뢰도에 결정적인 영향을 미치므로 이에 대한 상세한 분석이 요구된다.

이 논문의 2장에서는 다양한 고장내구성 기법들을 하드웨어와 소프트웨어 그리고 시스템 차원으로 분류하여 설명하였다. 3장에서는 원자력 발전소의 안전계통에 간단한 감시 타이머를 적용한 예를 보이고 그에 대한 안전성 분석 방법을 제시하였다.

2. 디지털 계측제어 기기의 고장내구성 기법

2-1. 디지털 시스템의 고장내구성 기법 분류

디지털 시스템의 신뢰도 향상을 위해 소프트웨어와 하드웨어에 적용될 수 있는 여러 단계에서의 기법들을 정리하면 <표 1>과 같다 [3]. <표 1>에서는 결함에 대한 대응을 4단계로 나누어 생각하였는데, 첫번째 단계는 결함을 회피하도록 설계하는 방법이며 두번째 단계는 결함을 검출할 수 있도록 하는 방법, 세번째와 네번째 단계는 결함의 영향을 제거하기 위한 조치에 해당한다.

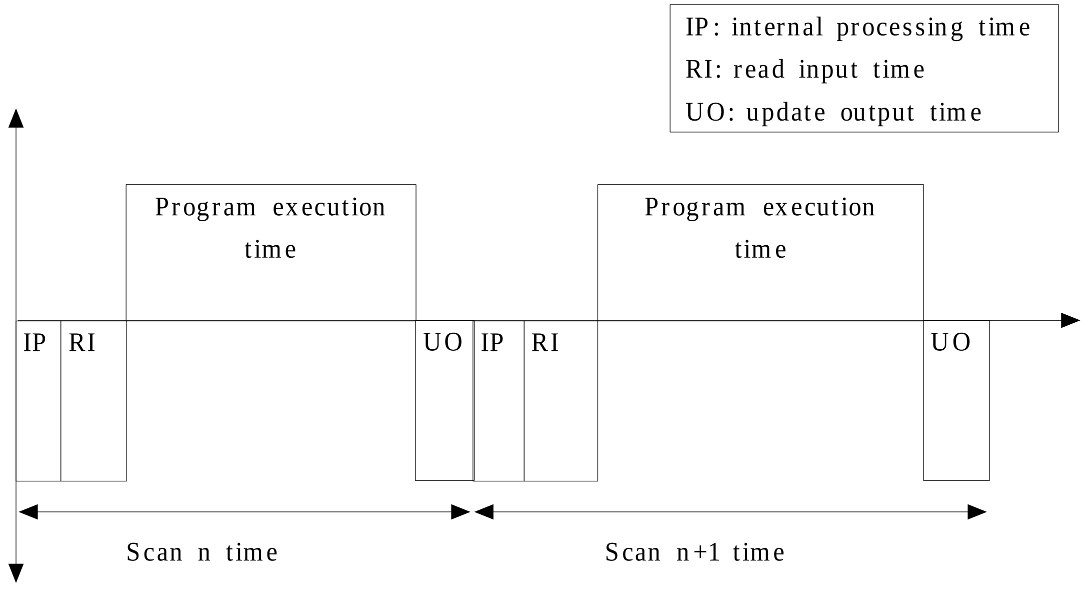
<표 1> 디지털 시스템의 하드웨어와 소프트웨어에서의 신뢰성 향상 기법

분류	하드웨어	소프트웨어
Fault avoidance	Quality changes Component integration level	Software engineering -modularity
Fault detection	Duplication Error detection codes Self-checking and fail-safe logic Watchdog timers and timeouts Consistency and capability checks Processor monitoring	Program monitoring Watchdog timers and timeouts
Masking redundancy	Error correcting codes Masking logic	Algorithm construction
Dynamic redundancy	Reconfigurable duplication Backup sparing Graceful degradation Reconfiguration Recovery	Forward error recovery Backward recovery -retry -checkpointing -journaling -recovery blocks

2-2. 감시 타이머 및 자기검사 기법들

일반적으로 산업 현장에서 많이 이용되는 대표적인 컴퓨터 기반 시스템(computer-based system)인 PLC의 경우에는 감시 타이머를 내장하고 있다. 감시 타이머는 디지털 시스템에 이상이 생겼을 경우, 이를 감지해 내기 위한 가장 기초적인 장치인데, 그 적용 범위는 PLC에 국한되는 것이 아니라 전반적인 컴퓨터 시스템 모두에 해당한다. 그러나 PLC와 같이 주기적인 반복실행(cyclic operation)을 위주로 하는 시스템에 특히 효과적으로 적용될 수 있다. 감시 타이머는 시간 설정치(time set-point)가 고정된 것과 가변적인 것의 두가지 종류로 구분할 수 있다 [4]. 고정 감시 타이머는 하드웨어 interrupt를 발생시키는 일종의 interval timer로서, 대개의 경우 이 고정 감시 타이머의 시간 설정치는 제조과정에서 고정되어 생산되므로 사용자가 임의로 변경하는 것은 불가능하다. <그림 1>은 일반적인 PLC의 작업 수행 과정을 도시한 것이다. 내부적인 처리 시간, 입력을 읽어 들이는 시간, 실행 코드를 수행하는 시간, 출력값을 갱신하는 시간을 합쳐서 하나의 scan time이라고 하는데, 고정 감시 타이머에서는 이 scan time이 미리 정해진 시간 설정치 보다 길어지게 되면 PLC를 정지시키고 고장 신호를 출력하게 된다. 소프트웨어를 이용하여 감시 타이머의 시간 설정치를 조정할 수 있는 경우를 가변 감시 타이머(software-based watchdog timer)라고 하는데, 이것은 scan time이 지정된 이상으로 길어지는 것을 방지하기 위한 목적으로 사용된다. 이렇게 내부에서 처리하는 방법이외에도 PLC 외부에 second line of defense의 성격을 가진 감시 타이머를 설치하는 방법도 있다.

이러한 감시 타이머들의 사용은 “정해진 시간을 초과하는 코드 실행은 시스템의 오류를 의미한다”는 가정하에 이루어진다. 그러나 모든 시스템의 오류가 시간지연을 유발하는 것은 아니므로, 감시 타이머의 유효범위는 제한되어 있다는 점에 유의하여야 한다.



<그림 1> PLC의 작업 수행 시간 개념도

감시 타이머는 오류를 방지하는 것이 아니라 오류 발생후에 감지해 내는 역할을 수행하기 때문에 단일 시스템의 신뢰성 관점에서는 도움이 되지 못한다. 그러나 시스템의 오류를 조기에 인지하고 대처할 수 있으므로 전체 시스템 차원에서는 신뢰성을 높이는 효과와 수리 시간을 줄여 가용도를 향상시키는 효과가 있다. 원자력 발전소의 보호 시스템 등 dangerous failure만이 관심 대상이 되는 경우에는 감시 타이머의 유효범위에 해당하는 만큼의 시스템 신뢰도 향상을 가져오기도 한다.

시스템 내부에 기존의 주프로세서(main processor) 외에 감시 프로세서(watchdog processor)를 별도로 설치하는 방법도 있는데, 이러한 감시 프로세서 시스템은 초기의 설정시에 감시해야 할 정보를 입력해 주고 실제 시스템 운용에 들어가면 온라인 상태에서 연속적으로 시스템의 감시를 수행하는 역할을 하게 된다. 이것은 감시 타이머와 완전 다중 시스템 사이의 중간 형태라고 볼 수 있다. 비교적 저렴한 비용으로 단순한 감시 타이머보다는 넓은 범위의 결함을 감시할 수 있다는 장점이 있다. 감시 프로세서를 별도로 둠으로써 공통원인 고장의 확률을 낮출 수 있다는 장점도 있다 [5].

2-3. 고장내구성 소프트웨어

소프트웨어 분야의 고장 내구성 구현 방법에는 대표적으로 복구 블럭을 두는 경우(backward recovery)와 N개 버전의 소프트웨어를 다중으로 두는 방법(forward error recovery)이 있다. 복구 블럭 방법은 acceptance test를 통해 결함 발생을 감지하면, 결함이 발생하기 전의 상태로 시스템을 되돌려 다시 실행을 시킨다. 이때 다중성이 없는 경우에는 같은 코드를 반복 실행하게 되는데, 이것은 소프트웨어 자체의 문제 해결보다는 하드웨어에서 발생할 수 있는 무작위성 결함에 대응하기 위한 목적이 더 크다. 일반적인 경우에는 acceptance test에서 결함이 발견되면, 다중 모듈을 실행시킨다. 즉, 출력값을 감시하여 기준치를 초과할 시에는 다른 알고리즘과 프로그램 언어로 짜여진 모듈로 넘기게 된다 [2], [6]. 동일한 목적을 가진 여러 가지 버전의 소프트웨어 코드를 설치하여 다중성을 주는 방법을 N-version 소프트웨어 기법이라고 한다 [7]. N개의 코드를 동시에 실행시킨 후, 최종단에서 voting을 하여 결과값을 출력하게 되므로 결함이 있는 코드의 실행 결과는 최종 결과에 반영되지 못하게 된다. 결함 주입 기법을 이용하여 고장내구성 시스템의 고장률을 시험한 연구의 결과가 Hocenski 등에 의해 발표되었는데, 이 시험의 결과중 소프트웨어의 다중성이 있는 경우와 없는 경우를 비교하면, 다중성이 있는 소프트웨어의 경우에는 그렇지 않은 소프트웨어에 비해 전체 고장 수가 수배정도 감소한 것으로 나타났다 [8]. Gokhale등은 distributed recovery block (DRB), N-version programming (NVP), N self-checking programming (NSCP)의 3가지 고장내구성 기법에 대한 적용예제를 제시하였는데, 결과는 대체적으로 NVP, DRB, NSCP의 순으로 오류를 감소시킬 수 있는 것으로 나타났다 [9]. 복구율이 높을수록 (연계 결함이

적을수록), 고장내구성 기법들이 단일 모듈의 결과보다 신뢰성이 높은 것으로 나타났다.

2-4. 시스템 다중성

위의 2-2 절과 2-3 절에 기술한 내용은 하드웨어의 고장 오류에 대한 감시와 소프트웨어의 고장 오류에 대한 감시 기법들에 관한 것이다. 이러한 방법들은 전 시스템에 대한 다중성을 확보하는 비용과 노력을 절감하면서도 고장 내구성을 확보하기 위한 기법들이다. 본질적으로 다중성이 확보되지 않은 시스템은 고장내구성을 가지지 않는다 [3]. 그러므로 이러한 다중성의 확보를 저렴한 비용으로 소프트웨어 또는 하드웨어 각각의 차원에서 추구하는 것이 위의 2-2 절과 2-3 절에서 설명한 내용이다. 시스템 차원의 다중성 확보의 전형적인 형태가 병렬(parallel) 배치나 2/3, 2/4 등의 voting (auction) 방법인데, 이런 경우들에 대한 해석 방법은 이미 잘 설정되어 있으므로 설명을 생략한다.

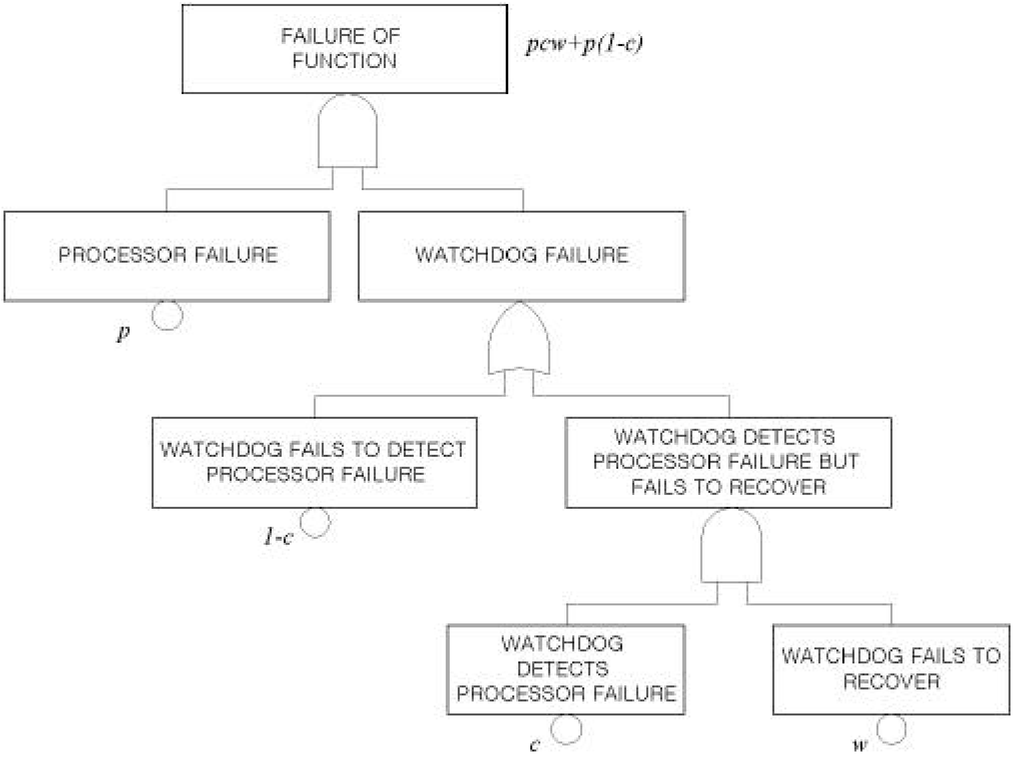
경우에 따라서는 이러한 병렬 배치나 voting 방법을 이용하기 힘든 때가 있는데, 전형적인 경우가 데이터 처리 및 저장(data processing and storage) 시스템이다. 디지털 데이터처리 시스템의 경우, 각 신호의 처리 때마다 voting을 거쳐 결과를 저장하고, 저장한 결과값을 읽을 때마다 여러 database로부터의 데이터를 voting한다는 것은 현실적으로 불가능하거나 매우 비효율적인 일이다. 이러한 경우에는 hot-standby 백업을 두어, 주 시스템(primary system)이 모든 결과를 처리하되 만약 주 시스템의 동작이 비정상적이라고 판단될 때에는, 백업 시스템으로 그 기능을 전환(switch over)하는 것이다. 이런 경우에는 감시 역할을 하는 시스템을 따로 두기보다는 백업 시스템이 작동 신호(heart-bit 등) 감시 기법을 이용하여 watchdog 기능까지 함께 수행하는 형태가 많다. 즉, 백업 시스템이 고장 감시와 고장시의 처리를 함께 수행하는 것이다.

실제 구현에 있어서는 전술한 여러 가지 하드웨어와 소프트웨어, 그리고 시스템의 고장내구성 기법들이 복합적으로 적용되는 경우가 많다는 점과, 각 고장내구성 기법의 고장 감지 확률을 추정하기 어렵다는 점이 디지털 시스템의 신뢰성 해석을 어렵게 하고 있는 것으로 판단된다.

3. 고장내구성 기법에 대한 PSA 방법론의 원전에의 적용

3-1. 고장 수목을 이용한 표현

고장내구성 기법을 PSA에 반영하는 것은 기본적으로 기기의 backup이 하나 추가된 것과 같은 효과를 가진다. 즉, 주어진 기기의 고장과 고장내구성 기법의 실패가 동시에 일어나는 경우에만 기기가 기능 수행에 실패하는 것으로 간주할 수 있다. 고장내구성 기법으로 감시 타이머(watchdog timer)를 이용하는 경우를 고장 수목을 이용하여 표현하면 <그림 2>와 같다. 프로세서의 고장확률을 p, 감시 타이머의 고장으로 회복에 실패하는 확률을 w, 감시 타이머가 프로세서의 고장을 감지할 확률(coverage factor)을 c라고 표시하였다. 감시 타이머의 실패는 두가지의 요인에 기인한다. 첫번째 경우는 처음부터 프로세서의 고장을 감지하지 못한 것($p(1-c)$)이고 두번째 경우는 고장을 감지하였지만 회복(recover)하는데는 실패한 것(pcw)이다.


FAILURE OF FUNCTION
pcw+p(1-c)
PROCESSOR FAILURE
p
WATCHDOG FAILURE
WATCHDOG FAILS TO DETECT PROCESSOR FAILURE
1-c
WATCHDOG DETECTS PROCESSOR FAILURE BUT FAILS TO RECOVER
WATCHDOG DETECTS PROCESSOR FAILURE
c
WATCHDOG FAILS TO RECOVER
w

<그림 2> 고장내구성 기법이 적용된 경우의 고장 수목

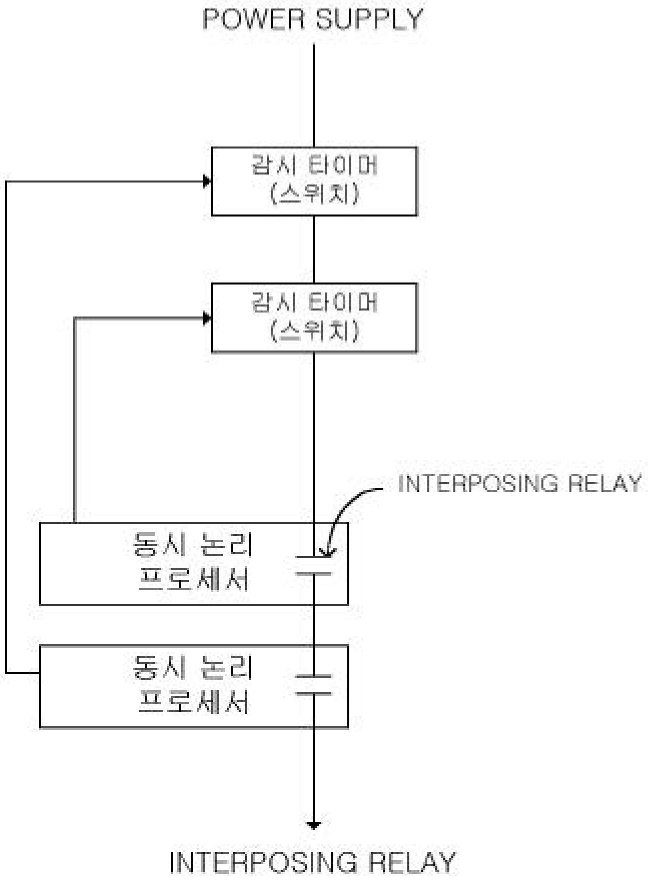
이것을 다음과 같이 해석할 수도 있다. 처음부터 프로세서의 고장에는 두가지가 있어서 감시 타이머가 감지할 수 있는 고장과 그렇지 못한 고장으로 구분하여 생각할 수 있는데, 그 확률은 각각 pc와 $p(1-c)$이다. 감지할 수 없는 고장의 경우($p(1-c)$)에 대해서는 감시 타이머가 아무런 역할을 할 수 없지만, 감지할 수 있는 고장의 경우(pc)에는 감시 타이머에 문제가 있지 않는 이상에는 회복이 가능하다. 그러므로 주어진 임무를 수행하지 못할 확률은 $p(1-c) + pcw$ 로 표현되는 것이다.

<그림 2>에서와 같이 감시 타이머를 이용하는 경우에 고장 검출 확률 c는 디지털 기기가 정지(halt)되는 고장인지, 그렇지 않고 단순히 잘못된 연산결과를 출력하는 경우인지에 따라 결정된다. 즉, c는 다음과 같이 정의될 수 있다. 감시 타이머의 경우에는 출력값을 직접 검증하는 것이 아니고 단순히 기기가 작동하고 있는지를 점검하는 것이기 때문이다.

$$c = \frac{\text{시스템이정지되는고장의횟수}}{\text{전체고장의횟수}}$$

3-2. 원전 안전 계통에의 적용

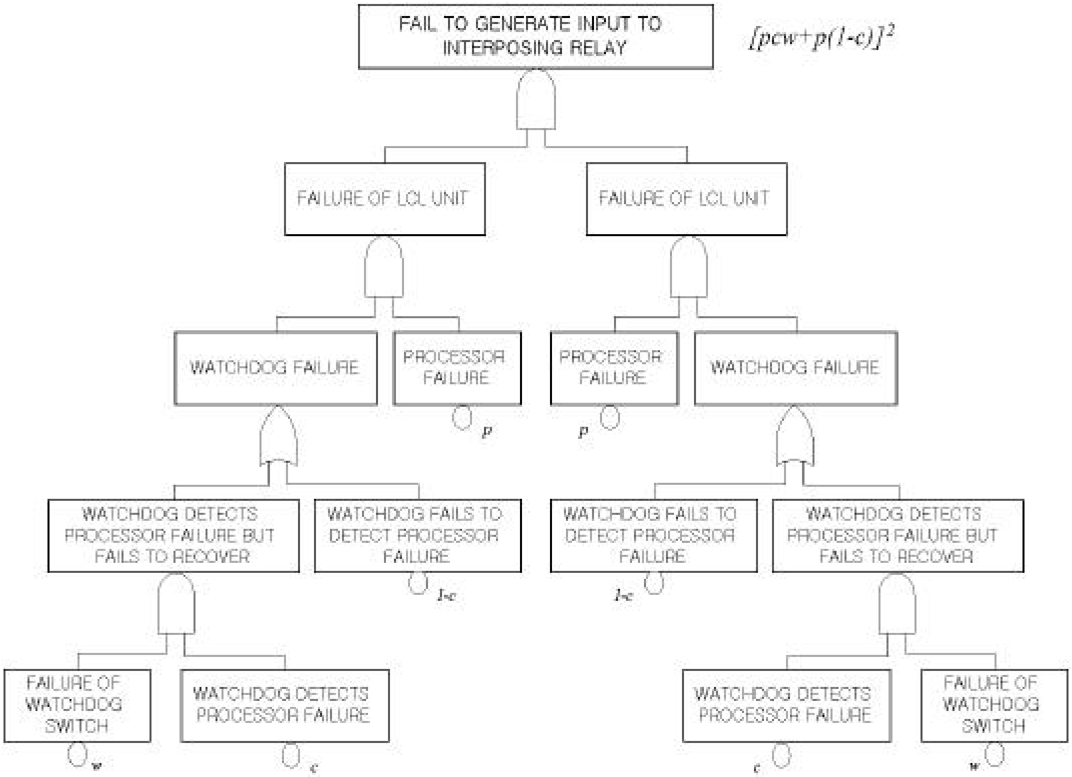
원자력 발전소의 안전계통에 디지털 시스템을 적용할 경우 가장 간결하고 적용이 용이한 감시 타이머를 이용하여 안전 기능의 정확한 수행 확률을 높일 수 있다. 일반적인 플랜트의 경우 시스템의 고장을 감시 타이머가 검출하면 재시도 등을 통해 복구를 시도하는 것에 비해 원자력 발전소의 경우 원자로 정지 신호의 발생을 통해 안전성의 향상을 도모하게 된다. <그림 3>은 원자력 발전소의 보호계통에 감시 타이머를 적용한 경우에 대한 예제이다. 이러한 구성을 실제로 구현하는 기술적인 방법은 여러 가지가 있겠으나 동시 논리 프로세서의 외부에 전원을 차단하는 스위치를 내장한 감시 타이머를 두어 일정 시간내에 감시 타이머로 입력이 들어가지 않으면 전원을 차단하는 구조로 되어 있다. 또한 하나의 채널내에 시스템 전체 중복을 두어 안전성을 높일 수 있도록 설계되어 있다.



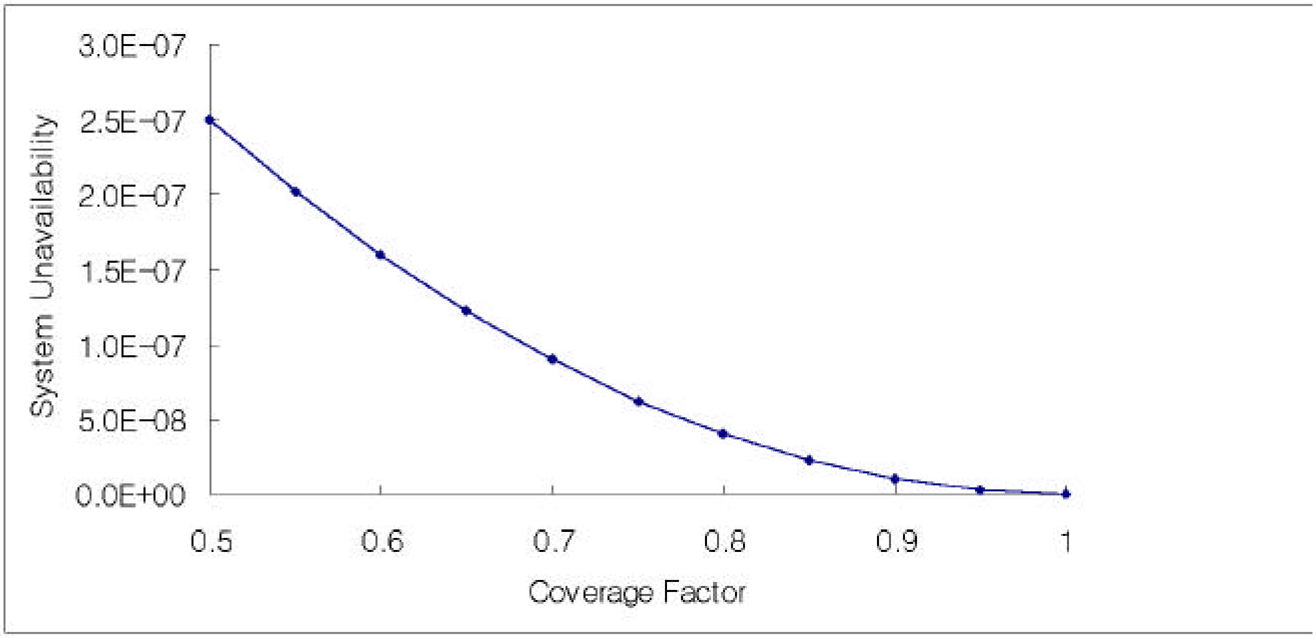
<그림 3> 원전 보호계통에 감시 타이머를 적용한 예제

<그림 4>는 <그림 3>의 예제에 대해 고장 수목을 구성한 것이다. 편의를 위해 공통원인 고장 등의 요인은 생략하였다. 같은 채널내의 두 계통중에서 하나만이라도 정상 작동을 할 경우에는 interposing relay를 구동시킬 수 있으므로 <그림 2>가 논리곱으로 연결된 모양을 갖는다.

대개의 경우 interposing relay를 구동하는 동시논리 프로세서의 고장확률은 감시 타이머의 고장확률보다 수백에서 수만배 정도 높다. 동시 논리 프로세서는 마이크로 프로세서, 메모리 등을 포함한 복잡한 구조로 이루어져 있는데 비해, 감시 타이머는 신호를 검출하여 제한시간 이내인지를 검사하는 간단한 회로와 전원을 차단하는 스위치로만 이루어져 있기 때문이다. $p=10^{-3}$, $w=10^{-7}$으로 가정하고 c에 따른 시스템의 불가용도 변화를 그래프로 나타내면 <그림 5>와 같다.


FAIL TO GENERATE INPUT TO INTERPOSING RELAY
[pcw+p(1-c)]²
FAILURE OF LCL UNIT
FAILURE OF LCL UNIT
WATCHDOG FAILURE
PROCESSOR FAILURE
p
PROCESSOR FAILURE
p
WATCHDOG FAILURE
WATCHDOG DETECTS PROCESSOR FAILURE BUT FAILS TO RECOVER
WATCHDOG FAILS TO DETECT PROCESSOR FAILURE
1-c
WATCHDOG FAILS TO DETECT PROCESSOR FAILURE
1-c
WATCHDOG DETECTS PROCESSOR FAILURE BUT FAILS TO RECOVER
FAILURE OF WATCHDOG SWITCH
w
WATCHDOG DETECTS PROCESSOR FAILURE
c
WATCHDOG DETECTS PROCESSOR FAILURE
c
FAILURE OF WATCHDOG SWITCH
w

<그림 4> 그림 3의 예제에 대한 고장 수목



<그림 5> 그림 4의 예제에서 $p=10^{-3}$, $w=10^{-7}$의 가정하에 c에 따른 시스템의 불가용도

c=0일 경우, 즉 감시 타이머가 전혀 역할을 할 수 없는 경우에는 전체 시스템의 불가용도는 10^{-6}이다. 이것은 단순히 동시 논리 프로세서를 이중으로 구성한 경우와 동일하다. 한편, c=1일 경우, 즉 완벽한 고장의 검출이 가능한 경우에는 전체 시스템의 불가용도는 10^{-20}이다. 이와 같이 c 값에 따른 편차가 심하므로, 고장을 검출할 확률을 정확히 추정하는 것이 전체 시스템의 불가용도를 평가하는데 대단히 중요한 역할을 하게 된다.

그러나 이러한 coverage factor (c)는 디지털 기기의 소프트웨어, 하드웨어, 고장의 형태 등 많은 변수가 작용하게 되므로 일괄적으로 이론적인 추정을 하기는 어렵다. 많은 실험적 연구를 통해 각 개별 경우(application)에 대한 coverage factor를 도출하여야 할 것으로 판단된다.

4. 결론

디지털 시스템의 정량적 안전성 평가에는 기존의 아날로그 시스템과 크게 다른 방법론을 적용하여야 할 것으로 판단된다. 디지털 기기의 특성상 병렬 배치나 voting 이외에도 다양한 고장내구성 기능의 구현이 가능하며, 원자력 발전소 안전 계통에서는 신뢰성 향상을 위해 광범위하게 사용될 것으로 예상된다. 따라서 이들에 대한 적절한 평가가 매우 중요하다.

이 논문에서는 디지털 계통내의 결함에 대한 내구성을 확보하기 위한 메카니즘을 하드웨어, 소프트웨어, 시스템 차원에서 정리하였다. 여러 가지 고장내구성 기법중 원전에 적용될 가능성이 가장 높은 감시 타이머의 경우에 대해 예제를 만들어 검토하여 보았다. 감시 타이머의 경우 소프트웨어를 내장하고 있지 않으므로 소프트웨어의 신뢰도와 관련된 논란을 피할 수 있으며, 간단한 구조로 인해 높은 신뢰성을 확보할 수 있으므로 원전 안전 계통에 적용될 경우 많은 장점이 있을 것으로 판단된다. 그러나, 예제에서 보인 바와 같이 고장 검출 확률인 c가 시스템 전체의 신뢰도에 미치는 영향이 매우 크기 때문에 이 값의 적절한 추정이 중요하다. 이와 관련한 최근의 연구의 현황을 분석한 결과, 다양한 계층에 다양한 방법으로 적용되고 있는 고장내구성 기법들을 일반적으로 평가할 수 있는 방법론은 아직 개발되지 않은 것으로 파악되었으며 실험등 다양한 방법의 연구를 통해 이를 평가해야 할 것으로 판단된다.

5. 참고 문헌

[1] NUREG-0800:HICB-BTP17, “Guidance on Self-Test and Surveillance Test Provisions.”

[2] H. Choi, W. Wang & K.S. Trivedi, "Analysis of conditional MTTF of fault-tolerant systems," Microelectronics & Reliability, Vol. 38, No. 3, 1998.

[3] M.R. Lyu & V.B. Mendiratta, "Software fault tolerance in a clustered architecture: Techniques and reliability modeling," Proceedings of the IEEE Aerospace conference, p. 141-150, 1999.

[4] NUREG/CR-6463, Rev. 1, "Review Guidelines on Software Languages for Use in Nuclear Power Plant Safety systems."

[5] A. Mahmood & E.J. MacCluskey, "Concurrent error detection using watchdog processors -A survey," IEEE Tr. on Computers, p. 160-174, Vol. 37, No. 2, February 1988.

[6] J. Wu, E.B. Fernandez & M. Zhang, "Design and modeling of hybrid fault-tolerant software with cost constraints," J. of systems Software, vol. 35, p. 141-149, 1996.

[7] R.K. Scott & D.F. McAllister, "Cost modeling of N-version fault-tolerant software systems for large N," IEEE Transactions on Reliability, Vol. 45, No. 2, p. 297-302, 1996.

[8] Z. Hocenski & G. Martinovic, “Influence of Software on fault - Tolerant Microprocessor Control system Dependability,” Proceedings of the IEEE International Symposium on Industrial Electronics, Vol. 3, p. 1193-1197, 1999.

[9] S.S. Gokhale, M.R. Lyu & K.S. Trivedi, "Reliability Simulation of fault-Tolerant Software and systems," Proceedings of Pacific Rim International Symposium on fault-tolerant systems, p. 167-173, 1997.