논문 2010-47SC-5-2

교정 제어를 이용한 비동기 순차 머신의 영구 고장 극복

(Corrective Control of Asynchronous Sequential Machines for Tolerating Permanent Faults)

양 정 민*

(Jung-Min Yang)

요 약

교정 제어는 비동기 순차 머신의 안정 상태 동작을 원하는 목적에 맞게 바꾸어주는 역할을 한다. 본 논문에서는 교정 제어를 이용하여 영구 고장이 존재하는 비동기 순차 머신의 고장 극복 기법을 제안한다. 비동기 순차 머신에서 영구 고장이 발생하면 머신은 원래 상태로 영원히 회복되지 못하고 제한된 상태 집합 내에서만 동작하게 된다. 하지만 비동기 순차 머신이 출어든 작동 범위 안에서도 정상적인 동작을 계속할 수 있는 여유도를 가지고 있다면 교정 제어 기법을 이용하여 고장 극복 문제를 해결할 수 있다. 본 논문에서는 영구 고장을 탐지할 수 있는 조건과 함께 고장 극복 문제를 푸는 교정 제어기가 존재할 필요충분조건을 밝힌다. 또한 사례 연구를 통해서 제안된 제어기의 설계 과정을 예시한다.

Abstract

Corrective control compensates the stable-state behavior of asynchronous sequential machines so that the closed-loop system can be changed in a desirable way. Using corrective control, we present a novel fault tolerance scheme that overcomes permanent faults for asynchronous sequential machines. When a permanent fault occurs to an asynchronous machine, the fault is not recovered forever while the machine is irreversibly stuck in a set of failure states. But, if the machine has control redundancy in the limited behavior range, corrective control can be applied to solve the fault tolerance problem against permanent faults. We present the condition on detecting permanent faults and the existence condition of an appropriate corrective controller. The design procedure for the proposed controller is described in a case study.

Keywords : Asynchronous Sequential Machines, Corrective Control, Fault Tolerance, Permanent Faults

I. 서 론

전역 클럭(clock)을 사용하지 않고 입출력의 즉각적인 변화에 의해서 동작하는 회로인 비동기 순차 머신은 디지털 공학의 역사와 더불어 발전하였다[1]. 클럭 속도가 매우 빠른 동기(synchronous) CPU가 발달함에 따라 비동기 순차 머신의 의미는 퇴색되고 있으나 고속 컴퓨터 개발, 병렬 연산[2], 시스템 생물학 모델링[3] 등에서 비동기 순차 머신은 여전히 활발하게 쓰이고 있다.

'교정 제어(corrective control)'는 전통적인 피드백 제어의 원리를 이용하여 비동기 순차 머신의 안정 상태(stable-state) 동작을 보정해주는 새로운 제어 기법이다[4~6]. 교정 제어는 클럭 없이 동작하는 비동기 머신이 과도 상태(unstable state)에서 아주 빠른(이론적으로 0 시간) 반응 속도로 움직이는 성질을 이용한다. 비동기 머신이 원하지 않는 상태 천이를 한 순간 교정 제어기가 원하는 동작을 이끌어내는 제어 입력을 비동기 머신에 넣어줌으로써 머신의 상태를 보정한다. 비동기 머신의 과도 응답 시간이 극히 짧기 때문에 이러한 교정 동작은 순식간에 이루어지며 따라서 폐루프 시스템의 동작은 원하는 입력-출력 조합 특성을 보인다.

교정 제어는 비동기 순차 머신에서 발생하는 여러 가지 고장을 복구하는 데에도 쓰일 수 있다. 교정 제어를

* 정회원, 대구가톨릭대학교 전자공학과
(Department of Electrical Engineering, Catholic University of Daegu)
접수일자: 2010년7월19일, 수정완료일: 2010년9월10일

이용하여 입력/상태 비동기 순차 머신에서 발생하는 상태 고장(state fault)을 탐지하고 복구하는 기법이 제안되었으며[7], 입력/출력 비동기 머신에 대한 고장 복구 제어 기법도 발표되었다[8~9].

본 논문의 목적은 교정 제어를 이용하여 비동기 순차 머신에서 발생하는 영구 고장(permanent fault)을 탐지하고 복구하는 방법을 제안하는 일이다. 이전 연구 결과[7~9]와 비교하여 이번 연구가 가지는 핵심적인 차별성은 다루는 고장의 종류이다. 이전 연구에서 고려했던 고장은 모두 과도 고장(transient fault)[10]이다. 과도 고장은 발생 즉시 고장 극복 메커니즘에 의해서 머신이 오류 상태에서 정상적인 상태로 복귀 가능한 종류로서 우주용 메모리에서 주로 발생하는 Single Event Upset(SEU)[11]이 대표적인 예이다.

이번 연구에서 다루는 고장은 영구 고장(permanent fault)[10]이다. 영구 고장이 발생하면 고장이 발생한 부분은 영원히 복구되지 못하며 고장 발생으로 인해서 머신은 제한된 상태 집합(failure state) 내에서만 동작하게 된다. 영구 고장의 예로는 머신의 하드웨어 한 부분이 손상되는 사건을 들 수 있다.

영구 고장은 과도 고장보다 머신의 동작과 도달가능성을 더 심각하게 제한한다. 하지만 영구 고장이 발생한 후에도 머신이 부분 동작만으로 정상적인 목적을 달성할 수 있는 하드웨어 여유도(hardware redundancy)를 보유하는 경우가 있을 수 있다. 예를 들어 삼중 여유도(Triple Modular Redundancy: TMR)[11]를 가지는 디지털 모듈에서 한 모듈이 파괴되어도 나머지 두 모듈이 정상적으로 동작하면 시스템이 원하는 출력 값을 그대로 낼 수 있다. 본 논문에서는 이러한 하드웨어 여유도를 활용하여 폐루프 시스템의 고장후(post-failure) 동작을 정상적으로 유지시키는 고장 극복(fault tolerance) 문제를 해결하는 교정 제어기를 제안한다.

본 논문에서는 먼저 영구 고장이 존재하는 비동기 순차 머신을 유한 상태 머신(finite state machine)으로 모델링한다. 그런 다음 영구 고장 발생 조건을 규정하고 고장 발생을 탐지할 수 있는 관측기(observer)의 존재 조건을 제안한다. 그리고 영구 고장이 일어난 후에도 폐루프 시스템의 동작을 정상적으로 유지시키는 내고장성 교정 제어기가 존재할 조건을 규명한다. 또한 사례 연구를 통해서 제안된 제어기의 설계 과정을 예시한다.

Ⅱ. 영구 고장이 존재하는 비동기 순차 머신

1. 입력/출력 비동기 머신

본 논문에서 다루는 비동기 순차 머신은 출력이 머신의 현재 상태 값과 다른 입력/출력 머신(input/output machine)이다. 앞서 기술했듯이 영구 고장이 일어나면 머신이 정상적인 상태 집합에서 더 이상 동작할 수 없다. 따라서 현재의 상태 값을 출력으로 가지는 입력/상태 비동기 머신(input/state machine)은 영구 고장이 발생하면 정상적인 동작으로 영원히 복구되지 못한다.

유한 상태 머신으로 입력/출력 비동기 머신을 표현하면 다음과 같다.

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

A는 입력 집합, Y는 출력 집합, X는 상태 집합, x_0은 초기 상태이며 $f:X\times A\rightarrow X$와 $h:X\times A\rightarrow Y$는 각각 상태 천이 함수와 출력 함수이다(Σ가 Moore 머신으로 모델링된다고 가정한다).

전역 클럭이 없는 비동기 머신의 성질에 따라서 Σ의 과도 상태(transient state) 천이 시간은 극히 짧다. 예를 들어 $(x,u)\in X\times A$가 '안정 상태(stable state) 조합'이라고 하면 $f(x,u)=x$이다. 이때 외부 입력이 u'로 바뀐다고 하고 (x,u')가 '과도 상태 조합'이라고 한다면 $f(x,u')=x_1$, $x_2=f(x_1,u')$,...등으로 머신 Σ가 과도 상태 x_1,x_2,...를 순식간에(이론적으로 0 시간) 거쳐서 '다음 안정 상태(next stable state)' x'에 도달한다($f(x',u)=x'$). 외부 사용자에게는 머신이 안정 조합 (x,u)에서 다음 안정 조합 (x,u')으로 즉시 이동하는 모습만 관측된다.

Σ의 안정 상태 동작만을 따로 표현하기 위해서 'stable-state 머신 $\Sigma_{|s}$'를 아래와 같이 정의한다[5].

$$\Sigma_{|s} = (A, Y, X, x_0, s, h),\ s(x,u):=x'$$

위 식에서 f 대신 사용되는 'stable recursion 함수' s는 상태-입력 조합 (x,u)의 다음 안정 상태 x'를 반환하는 함수이다. 단위 입력 대신 입력 스트링(string)을 s의 변수로 설정하면 다음과 같이 일반화할 수 있다.

$$s(x,ut) := s(s(x,u),t),\ x\in X,\ u\in A,\ t\in A^+$$

위 식에서 A^+는 A에 속한 단위 입력들로 이루어지는 길이 1 이상의 스트링 집합을 말한다. 또 $s(x,t)=x'$인 입력 스트링 $t\in A^+$가 존재하면 상태 x'는 상태 x로부터 '도달가능하다고(stably reachable)'[5] 말한다.

2. 영구 고장 모델링

Σ에 p개의 영구 고장 모드(mode) F_1, F_2, ..., F_p가 있다고 설정하고 A와 X를 서로소인 정상(normal) 부분 집합과 영구 고장 부분 집합으로 나눈다.

$$A = A_N \cup A_{F1} \cup \cdots \cup A_{Fp}$$

$$X = X_N \cup X_{F1} \cup \cdots \cup X_{Fp}$$

위 식에서 A_N과 X_N은 정상 입력 집합과 정상 상태 집합을 가리키며 A_{Fi}와 X_{Fi}는$(0\le i\le p)$ i번째 영구 고장 모드 F_i를 일으키는 고장 입력 집합과 F_i가 발생했을 때 머신 Σ가 천이하는 상태 집합을 각각 가리킨다. 영구 고장에 대한 이러한 정의는 이산 사건 시스템(discrete-event systems) 분야의 고장 진단(fault diagnosis) 이론에서 사용되는 모델링과 유사하다[12].

고장 사건의 특성상 A_{Fi}에 속하는 고장 입력들은 외부에서 관측되지 않는다. 또한 영구 고장의 정의에 따라서 머신 Σ에 영구 고장 F_i가 한 번 발생하면 Σ는 X_N의 상태에서 X_{Fi}의 상태로 천이한 후 다시는 X_{Fi} 상태를 벗어나지 못한다. 이 성질을 stable recursion 함수 s를 이용하여 표시하면 아래와 같다.

$$\forall x\in X_{Fi},\ \forall v\in A,\ s(x,v)\in X_{Fi} \text{ or } s(x,v)=\varnothing,\ i=1,...,p$$

이와 함께 이번 논문에서는 이론 전개를 명확하게 하기 위해서 A_{Fi}를 단조 집합(monotone set), 즉 원소가 하나인 집합으로 설정하고

$$A_{Fi} = \{f_i\},\ i=1,...,p$$

라고 한다. f_i는 Σ가 정상 상태 집합 X_N의 원소에 있을 때 발생하며 그 결과로 Σ는 영구 고장 집합 X_{Fi}로 천이한다. f_i가 발생하는 정상 상태를 z_i라 하고 Σ가 천이하는 고장 상태를 z'_i라 하면 다음과 같은 관계가 성립한다.

$$s(z_i, f_i) = z'_i,\ i=1,...,p$$

즉 $(z_i, f_i, z'_i)\in X_N\times A_{Fi}\times X_{Fi}$ 한 조합이 영구 고장 F_i의 발생을 나타낸다.

예제 1: 그림 1은 영구 고장이 존재하는 입력/출력 비동기 순차 머신 $\Sigma=(A,Y,X,x_0,f,h)$의 한 예이다. 편의상 $\Sigma = \Sigma_{ls}$, 즉 현재의 상태흐름도(state flow diagram)가 stable-state 머신과 동일한 시스템을 선택하였다. 그림에서 표기된 대로 초기 상태는 $x_0=x_1$이며 p=2, 즉 두 개의 영구 고장 모드가 존재한다. Σ가 가지는 입력과 상태 집합은 아래와 같다.

$$A_N = \{a, b, c, d, e, g, h, i\}$$

$$A_{F1} = \{f_1\},\ A_{F2} = \{f_2\}$$

$$X_N = \{x_1, x_2, x_3, x_4\}$$

$$X_{F1} = \{x_5, x_6\},\ X_{F2} = \{x_7, x_8\}$$

영구 고장 발생을 의미하는 상태 천이는

$$s(x_3, f_1) = x_5,\ s(x_3, f_2) = x_7$$

이다. 또 Σ의 출력 집합은 Y={0, 1, 2}이며 출력 함수 h(x)는 아래와 같이 정의되었다.

$$h(x_1) = 0,\ h(x_2) = 1,\ h(x_3) = 0,\ h(x_4) = 1$$

$$h(x_5) = 1,\ h(x_6) = 0,\ h(x7) = 2,\ h(x_8) = 0$$

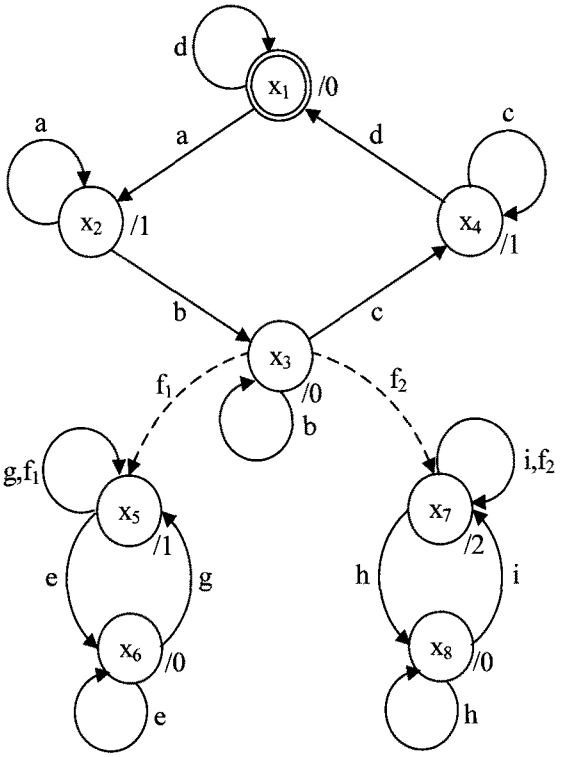


그림 1. 영구 고장이 존재하는 입력/출력 비동기 순차 머신 Σ

Fig. 1. Input/output asynchronous sequential machine Σ with permanent faults.

Ⅲ. 내고장성 교정 제어기 구조

그림 2는 기존 연구[9, 13]를 바탕으로 구성한 입력/출력 비동기 머신을 위한 내고장성 교정 제어 시스템이다. 그림에서 C는 교정 제어기이며 B는 관측기이다. 입력/출력 머신은 머신의 상태 값을 바로 알 수 없기 때문에 현재의 출력 값을 받아서 상태 값을 구해 제어기 C에 전달해주는 관측기 B가 필요하다. 제어기 C와 관측기 B 모두 비동기 순차 머신으로 구현된다. $v \in A_N$은 외부에서 들어오는 정상 입력이며 $u \in A_N$은 제어 입력(control input)이다. 또 $f \in A_{F1} \cup \cdots \cup A_{Fp}$는 영구 고장 입력으로 그림에서 볼 수 있듯이 제어기 C나 관측기 B를 거치지 않고 비동기 머신 Σ로 직접 들어간다. f는 또한 머신 Σ 내부에서 발생하는 영구 고장 사건으로도 해석될 수 있다. f의 이러한 모델링은 고장 사건이 원칙적으로 제어 불가능하며 관측 불가능하다는 일반적인 이론과 부합된다.

$y \in Y$는 Σ의 출력 값으로 Σ의 현재 상태를 알아내기 위해서 관측기 B로 피드백 된다. 그림 2에서 표시된 y^*는 피드백 되는 출력이 현재의 단일(unit) 값이 아니라 직전 안정 상태에서 현재 안정 상태(next stable state)까지 천이했을 때 나오는 출력 신호의 연속 값, 즉 burst로서[12] 사용된다는 것을 의미한다. 관측기 B는 제어 입력 u와 출력 피드백 y^*를 이용하여 머신 Σ의 현재 상태 x를 찾아내 제어기 C에 전달한다. B는 상태 관측 작업 이외에도 영구 고장 사건을 탐지하는 역할도 수행한다.

제안하는 고장 탐지 및 복구 과정을 개략적으로 설명하면 다음과 같다. 그림 2에서 머신 Σ가 정상 동작을 보일 때에는 제어기 C가 아무 일을 하지 않고 외부 입력 v를 그대로 머신 Σ에 전달해준다(u=v). 영구 고장 입력 $f=f_i$가 발생한 순간 X_N에 속한 정상 상태에 머물러 있던 Σ는 X_{Fi}에 속한 상태로 원하지 않는 천이를 한다. 이때 관측기 B가 Σ의 상태 천이를 감지하여 고장 탐지를 하고 이 정보를 받은 C가 고장 극복을 위한 제어 입력 u를 생성하여 Σ에 전달한다.

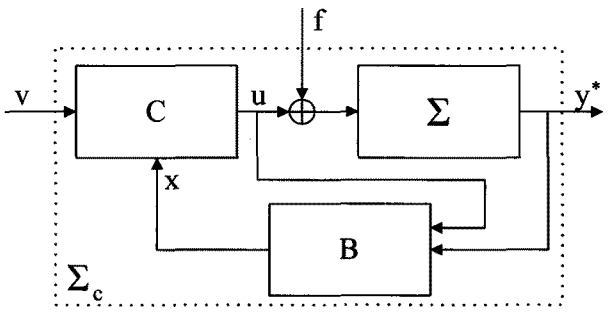


그림 2. 입력/출력 비동기 순차 머신을 위한 내고장성 교정 제어 시스템

Fig. 2. Fault tolerant corrective control system for input/output asynchronous sequential machines.

Ⅳ. 고장 탐지

1. 관측 가능성

앞 장에서 기술한 바대로 이번 논문에서 제안하는 내고장성 시스템에서 관측기 B는 상태 관측 이외에도 영구 고장을 탐지하는 역할을 한다. 먼저 머신 Σ에 그림 2의 영구 고장 사건 f가 아직 발생하지 않은 정상적인 동작에서 관측기 B가 출력 burst y^*와 제어 입력 u를 받아서 현재 상태 x를 찾는 과정을 설명한다. 비동기 머신은 안정 상태에서만 관측 가능하므로 B가 찾는 상태 x는 모두 안정 상태이다.

머신 Σ가 어떤 안정 상태 조합 $(x,u) \in (X_N, A_N)$에 머물러 있다가($f(x,u)=x$) 제어 입력 $u' \in A_N$를 받고 다음 안정 상태 x'로 천이한다고 하자($s(x,u')=x'$). Σ는 x에서 x'까지 이동할 때 중간에 여러 개의 과도 상태 $x_1, x_2, \ldots, x_{q-1}$를 거칠 수 있다. 따라서 Σ가 내는 출력 burst는 일반적으로

$$y^* = y_0 y_1 y_2 \cdots y_q,$$

$$y_0 = h(x),\ y_1 = h(x_1),\ \ldots,\ y_q = h(x')$$

으로 표현된다. 그런데 클럭 없는 비동기 머신에서는 오직 변수 값의 변화에 의해서만 의미 있는 동작이 시작되므로(event-driven) 위 식의 y^* 대신에 'burst 함수' $\beta(y^*)$를 정의해서 쓴다[13]. $\delta_1, \delta_2, \ldots, \delta_r \in Y$이며 $\delta_{i+1} \neq \delta_i$, $i=1,\ldots,r-1$이라고 할 때 출력 burst $y^* = \delta_1\delta_1\cdots\delta_1\delta_2\delta_2\cdots\delta_2\cdots\delta_r\delta_r\cdots\delta_r$의 burst 함수 값 $\beta(y^*)$는 다음과 같이 정의된다.

$$\beta(y^*) := \delta_1\delta_2\cdots\delta_r,\ y^* \in Y^*$$

$\beta(y^*)$는 y^*에서 연속적으로 나타나는 알파벳 스트링을 한 번씩만 쓴 것이다. 예를 들어 $\beta(aabb)=ab$이다. 이러한 burst의 정의는 그림 2의 관측기 B가 Σ로부터 실제 받는 출력 피드백의 물리적 의미와 일치한다고 말할 수 있다. Σ가 과도 상태 조합 (x,u)에서 과도 상태

$x_1,x_2,...,x_{q-1}$를 거친 다음 안정 상태 $x_q=s(x,u)$로 천이된 후 관측기 B가 받는 출력 burst 함수 값을 $\beta(x,u)$라고 다시 표기하면 $\beta(x,u)$는 다음과 같다.

$$\beta(x,u) = \beta(h(x)h(x_1)\cdots h(x_{q-1})h(x_q))$$

또 마지막 출력 $h(x_q)$를 제외한 출력 burst에 대한 burst 함수를 $\beta_{-1}(x,u)$이라고 하면

$$\beta_{-1}(x,u) = \beta(h(x)h(x_1)h(x_2)\cdots h(x_{q-1}))$$

이다.

비동기 머신 Σ를 교정 제어하려면 Σ가 옮겨 가는 모든 안정 상태를 정확하게 관측해야 한다. 또한 전역 클럭이 없기 때문에 그림 2의 폐루프 시스템이 비결정적(non-deterministic) 동작을 보이지 않게 하기 위해서는 Σ가 '언제' 다음 안정 상태에 도달하는지도 정확하게 알아야 한다. 이 정보를 정확하게 알지 못한 채 제어기 C가 제어 입력 u를 바꾸어주면 Σ의 현재 상태가 C가 예측한 상태가 아닐 수가 있으므로 시스템의 다음 동작은 비결정적이 될 수도 있다.

Σ가 어떤 과도 상태 조합 (x,u)에서 다음 안정 상태로 진입을 완료했는지 정확하게 판별할 수 있으면 Σ는 (x,u)에서 '관측 가능하다'라고 말한다. 입력/출력 비동기 머신 Σ가 (x,u)에서 관측 가능하기 위해서는 Σ가 다음 안정 상태 바로 직전에 통과하는 상태의 출력 값이 다음 안정 상태의 출력 값과 달라야 한다[13]. burst 함수 $\beta(x,u)$로 이 조건을 표현하면 아래와 같다.

$$\beta(x,u) \neq \beta{-}1(x,u) \qquad (1)$$

예를 들어 $x_q=s(x,u)$이고 $h(x_q)=h(x_{q-1})$, 즉 $\beta(x,u) = \beta_{-1}(x,u)$이라고 가정하자. Σ가 안정 상태 x에 있다가 입력 u를 받아서 상태 천이를 시작하면 Σ가 거치는 상태 값이 바뀜에 따라 출력 burst y^*도 변한다. Σ가 다음 안정 상태 x_q의 직전 상태 x_{q-1}에 도달하는 순간 관측기 B가 가지는 출력 피드백은 정의에 의해서 $\beta_{-1}(x,u)$이다. 그런데 $\beta(x,u) = \beta_{-1}(x,u)$이므로 B는 Σ가 현재 과도 상태 x_{q-1}을 거치는 과정에 있는지 아니면 다음 안정 상태 x_q에 도달했는지를 구분하지 못한다. 따라서 식 (1)은 입력/출력 비동기 머신의 관측 가능성에 대한 필요충분 조건이 된다.

2. 영구 고장 탐지

다음으로 관측기 B가 영구 고장 사건을 탐지하는 방법을 제안한다. 앞에서 영구 고장 사건은 A_{Fi}의 입력 f_i에 의한 상태 천이 $s(z_i, f_i)=z'_i$로 모델링되었다($z_i \in X_N$, $z'_i \in X_{Fi}$). 따라서 영구 고장 탐지는 Σ가 안정 상태 z_i에 있을 때 입력 f_i가 들어와서 다음 안정 상태 z'_i로 천이하는 과정을 관측기 B가 알아내는 과정으로 정상적인 상태 천이 때와 크게 다르지 않다.

하지만 영구 고장의 발생과 정상적인 상태 천이와의 가장 큰 차이점은 영구 고장 입력 f_i가 관측 불가능하다는 사실이다. 이 문제는 관측기 B가 C로부터 받는 제어 입력 u의 변화를 주시함으로써 해결될 수 있다. 비동기 머신의 기본 모드 동작(fundamental mode operation) 원리[1]에 따라서 모든 입력 값의 변화는 머신이 안정 상태에 있을 때에만 이루어진다. 즉 f_i는 Σ가 안정 상태 z_i에 있을 때에만 발생한다. 경우에 따라서 z_i가 과도 상태가 될 수도 있는데 이 경우에는 f_i가 발생할 수 없다. 또한 영구 고장은 제어기 C의 작동과 상관없이 일어나므로 f_i가 발생할 때 제어 입력 u는 변화하지 않는다(u가 변화한다는 것은 정상적인 교정 동작이 실행된다는 의미이므로 Σ는 z_i에서 X_N에 속한 다른 상태로 천이하고 따라서 f_i는 역시 일어나지 않는다.).

정리하면 관측기 B는 아래와 같은 일련의 상황에서 영구 고장 입력 f_i의 발생을 감지할 수 있다.

- Σ가 안정 상태 z_i에 머물러 있다.
- 제어 입력 u의 값은 바뀌지 않는다.
- 출력 burst y^*가 바뀐다.

제어 입력 u가 일정한 상황에서 출력 y^*가 바뀌었다면 그림 2에서 추론할 수 있듯이 영구 고장 입력 f가 발생했다는 뜻이 된다.

하지만 위의 조건들은 영구 고장 사건이 일어날 수 있는 필요조건일 뿐이지 관측기 B가 고장 사건을 정확하게 탐지할 수 있는 조건은 아니다. B가 고장 사건을 정확하게 탐지한다는 의미는 1) 발생한 고장의 모드가 $F_1,...,F_p$ 중 어느 것인가를 알고, 2) 고장 발생 후 Σ가 다음 안정 상태로 천이를 완료했는지를 안다는 뜻이다.

Σ가 안정 상태 z_i에 있을 때 영구 고장 사건 f_i가 발생했다고 다시 가정하자. p개의 고장 모드가 있으므로 영구 고장이 발생할 수 있는 정상 상태는 $z_1,...,z_p$이다. 그런데 이 상태들 중 서로 동일한 값이 있을 수 있다. 즉 어떤 상태 z_i에서 복수 개의 영구 고장 사건들이 (따로따로) 일어날 수 있다. 본 논문에서는 z_i에서 일어날

수 있는 영구 고장 모드의 종류 수를 m(i)라고 정의하고 그러한 영구 고장 모드의 index를 (i,1),...,(i,m(i))라고 표기한다. 즉 이제부터 상태 z_i에서 일어날 수 있는 영구 고장 입력은 $f_{i,1},...,f_{i,m(i)}$으로 표시된다(이 입력 중 한 개는 반드시 f_i이다). m(i)=1이면 $f_{i,1}=f_i$가 된다.

Σ가 안정 상태 z_i에 있을 때 영구 고장 사건이 발생한 후 관측기 B가 이를 정확하게 탐지할 수 있는 조건은 다음과 같다.

1) 먼저 발생한 영구 고장 모드가 어느 것인가를 알기 위해서 z_i에서 발생할 수 있는 영구 고장 입력이 유도하는 출력 burst가 모두 달라야 한다. 그렇지 않으면 서로 다른 고장 모드의 발생을 관측기가 구분할 수 없다. 이 조건을 식으로 쓰면 다음과 같다.

$$\beta(z_i,f_{i,j}) \neq \beta(z_i,f_{i,k}),\ \forall j,k\in\{1,2,...,m(i)\} \qquad (2)$$

2) 정상적인 상태 천이 동작과 마찬가지로 관측기 B는 영구 고장의 발생이 언제 끝나는지를 알아야 한다. 즉 Σ가 고장 발생 후 다음 안정 상태로 도달했는지 여부를 판별해야 한다. 이 과정을 실현할 수 있는 조건은 식 (1)과 유사하게 아래와 같이 유도된다.

$$\beta(z_i,f_{i,j}) \neq \beta^{-1}(z_i,f_{i,j}),\ \forall j\in\{1,2,...,m(i)\} \qquad (3)$$

모든 F_i에 대해서 조건 (2)와 (3)이 만족된다면 머신 Σ에서 발생하는 영구 고장을 탐지할 수 있는 관측기 B를 꾸밀 수 있다. 관측기 B의 구조와 작동에 대한 상세한 설명은 이전 연구 결과[9, 13]에 나와 있다.

예제 2: 그림 1의 입력/출력 비동기 머신 Σ에 대한 영구 고장 탐지 가능 여부를 조사한다. 그림에서 볼 수 있듯이 고장 입력 f_1과 f_2는 모두 상태 x_3에서 발생한다. $z_1=x_3$라 하면 m(1)=2이며 (1,1)=1, (1,2)=2이다. 그림 1에서 burst 함수 값을 구하면

$$\beta(x_3,f_1) = 01,\ \beta(x_3,f_2) = 02$$

이다. $\beta(x_3,f_1)\neq\beta(x_3,f_2)$이므로 조건 (2)가 성립한다. 또한 두 과도 상태 조합 (x_3,f_1)과 (x_3,f_2)의 출력 burst는

$$\beta(x_3,f_1) \neq \beta^{-1}(x_3,f_1),\ \beta(x_3,f_2) \neq \beta^{-1}(x_3,f_2)$$

이므로 조건 (3)도 만족된다. 따라서 그림 1의 비동기 머신 Σ에서 일어나는 영구 고장 사건들은 모두 탐지 가능하다.

V. 고장 극복

1. 교정 제어기 원리 및 존재 조건

비동기 머신 Σ가 임의의 영구 고장 사건에 대해서 관측 가능할 때 발생한 고장을 극복하는 교정 제어기의 존재 조건과 설계 과정을 제안한다. 비동기 머신이 고장을 극복한다는 것은 고장 발생 후에도 폐루프 시스템이 정상적인 입력-출력 특성을 그대로 유지할 수 있다는 의미이다.

먼저 상태 집합 X_N이 아래와 같은 원소들을 가진다고 설정한다.

$$X_N = \{x_1, x_2, ..., x_n\}\ (|X_N|=n)$$

Σ는 정상 상태 집합 X_N에서 동작하다가 모드 F_i의 영구 고장이 발생하면 고장 상태 집합 X_{Fi}로 천이한 후 X_{Fi} 내에서 계속 머무르게 된다. Σ가 고장 모드 F_i에 대해서 내고장성을 가지기 위해 필요한 첫번째 조건은 정상 상태 집합 X_N에 속하는 임의의 상태와 동일한 출력 값을 가지는 상태들이 고장 상태 집합 X_{Fi}에서 적어도 하나씩 존재해야 한다는 것이다. 이 조건을 수학적으로 표시하기 위해서 X_N에 대한 X_{Fi}의 '출력 등가 리스트 (output equivalent list)' $E(X_{Fi},X_N)$를 아래와 같이 정의한다.

$$E(X_{Fi},X_N) := \{E_1, E_2, ..., E_n\},$$

$$E_i := \{\pi\in X_{Fi} |\ h(\pi) = h(x_i)\},\ i=1,...,n$$

$E(X_{Fi},X_N)$의 원소 E_i는 X_{Fi}의 부분 집합이며($E_i\subset X_{Fi}$) E_i에 속한 임의의 고장 상태 π는 정상 상태 x_i와 동일한 출력을 낸다. 만약 x_i와 동일한 출력을 내는 상태가 X_{Fi} 안에 존재하지 않는다면 $E_i=\varnothing$이다.

출력 등가 리스트 $E(X_{Fi},X_N)$의 모든 원소가 공집합이 아니라고 가정하고 그림 2의 교정 제어기 C가 고장 모드 F_i를 극복하는 과정을 기술한다. Σ가 관측 가능하다고 했으므로 영구 고장 모드 F_i가 발생한 순간 교정 제어기 C는 관측기 B로부터 고장 발생 정보를 얻는다. 앞에서 고장 입력 f_i가 가지는 상태 천이는 $s(z_i, f_i)=z'_i$라고 정의하였다($z_i\in X_N$, $z'_i\in X_{Fi}$). 따라서 제어기 C는 관측기 B로부터 받는 상태 값 x가(그림 2 참조) z_i에서 z'_i로 바뀌는 순간 고장 발생을 감지하고 극복 동작을 시작한다.

조건 (3)으로부터 $h(z_i)\neq h(z'_i)$이다. 즉 z'_i는 z_i의 출

력 등가 리스트 원소 E_i에 속하지 않는다. 제어기 C는 이러한 입력-출력 모델 부정합을 없애기 위해서 Σ가 z'_i로 천이하는 순간 제어 입력 u를 이용하여 Σ를 $\pi_i \in E_i$인 고장 상태 π_i로 천이시킨다. 이 교정 동작은 비동기적으로 매우 짧은 순간 벌어지므로 외부 사용자에게는 관찰되지 않으며, $h(\pi_i)=h(z_i)$이므로 출력 변화도 없는 것으로 인식된다.

Σ가 안정 상태 π_i에 있을 때 외부 입력 v가 바뀌어 $v=\Theta$가 되었다고 가정하자. 영구 고장이 발생하지 않았다면 Σ는 상태 z_i의 다음 안정 상태 $x_k := s(z_i,\Theta)$로 천이했을 것이다. Σ의 입력-출력 특성을 그대로 유지하기 위해서는 앞서와 마찬가지로 Σ가 x_k의 출력과 동일한 출력 값을 내는 고장 상태로 이동하도록 교정해주어야 한다. $\pi_k \in E_k$라 하면 제어기 C는 Σ를 π_i에서 π_k로 천이시키는 제어 입력을 찾아서 u에 넣어준다. 이런 식으로 교정 제어기가 계속 일을 하면 폐루프 시스템은 고장이 발생하지 않은 상태와 똑같은 입력-출력 특성을 보이게 되며 내고장성을 확보한다.

상기한 고장 극복 동작을 구현하는 교정 제어기가 존재할 조건을 구해보자. 우선 정상적인 교정 동작을 위해서는 모든 안정 상태 간의 천이 과정에서 Σ가 관측 가능해야 한다. 그 다음으로 필요한 조건은 안정 상태 간의 도달가능성(stable reachability)이다. 먼저 고장이 발생한 직후 상태 z'_i로 천이한 Σ는 고장이 일어나기 직전 안정 상태인 z_i와 동일한 출력 값을 가지는 고장 상태로 다시 이동해야 한다. $X_N=\{x_1,x_2,...,x_n\}$일 때 $z_i=x_j$라 하고 이 조건을 명시적으로 표현하면 다음과 같다.

조건 1: $z_i=x_j$일 때 출력 등가 리스트 $E(X_{Fi},X_N)$의 원소 E_j 안에 z'_i로부터 도달가능한 상태가 적어도 한 개 존재해야 한다.

또 Σ가 $x_j(=z_i)$에서 x_k로 옮겨가도록 하는 외부 입력 값이 들어왔을 때에는 Σ를 출력 등가 리스트 E_j의 원소에서 E_k의 원소로 천이시킴으로써 내고장성을 이룩한다고 하였다. 이것은 E_j의 모든 원소가 E_k 안의 한 원소까지 도달가능한 경로가 적어도 한 개 이상 존재해야 한다는 사실을 의미한다. 이 조건을 일반화하면 아래와 같이 표현된다.

조건 2: i) $E(X_{Fi},X_N)$은 공집합 원소를 하나도 포함하지 않는다. ii) Σ의 정상 상태 x_k가 x_j로부터 도달가능하다면 $E(X_{Fi},X_N)$의 E_j의 모든 원소는 E_k 안의 어떤 원소까지 도달가능해야 한다. ($\forall j,k \in \{1,...,n\}$)

영구 고장 모드 F_i에 대해서 조건 1과 2가 만족되면 F_i에 내고장성을 가지는 폐루프 시스템을 구성할 수 있다.

예제 3: 그림 1의 입력/출력 비동기 머신 Σ는 모든 안정 상태 간의 천이 과정에서 관측 가능하다(증명은 생략). 또 Σ에서 $X_N=\{x_1,x_2,x_3,x_4\}$이고 $h(x_1)=h(x_3)=0$, $h(x_2)=h(x_4)=1$이므로 $E(X_{F1},X_N)$과 $E(X_{F2},X_N)$은 아래와 같이 유도된다.

$$E(X_{F1},X_N) = \{\{x_6\}, \{x_5\}, \{x_6\}, \{x_5\}\}$$

$$E(X_{F2},X_N) = \{\{x_8\}, \varnothing, \{x_8\}, \varnothing\}$$

$E(X_{F2},X_N)$가 공집합 원소를 가지고 있으므로 영구 고장 모드 F_2는 조건 2.i)를 만족시키지 못하고 따라서 F_2에 대해서 내고장성을 가지는 폐루프 시스템을 구성하는 교정 제어기 C가 존재하지 않는다. 반면 $E(X_{F1},X_N)$은 공집합 원소를 하나도 가지지 않는다. 또 고장 사건 입력 f_1은 정상 상태 x_3에서 발생하여 고장 상태 x_5로 천이한다. $h(x_3)=0$이고 $h(x_5)=1$이므로 Σ는 고장 발생 즉시 E_3의 원소로 다시 상태 천이해야 한다. 그런데 $E_3=\{x_6\}$이고 그림 1에서 알 수 있듯이 x_6은 x_5에서 도달가능하므로($s(x_5,e)=x_6$) 고장 모드 F_1은 조건 1을 만족시킨다. 그렇다면 검증해야 할 나머지 조건은 조건 2.ii)이다. 그림 1로부터 X_N의 임의의 상태는 다른 임의의 상태로부터 도달가능하다는 사실을 알 수 있다. x_2로 예를 든다면

$$s(x_1,a)=x_2,\ s(x_3,cda)=x_2,\ s(x_4,da)=x_2$$

와 같이 X_N에 속한 다른 상태에서 x_2로 가는 정상 입력 스트링을 항상 찾을 수 있다. 또 x_1과 x_3의 출력 값은 0이고 x_2와 x_4의 출력 값은 1이므로 내고장성 교정 제어기가 존재하려면 조건 2.ii)에 따라서 E_1과 E_3 안의 임의의 상태가 E_2 또는 E_4의 한 원소로 도달가능해야 하고 그 역도 성립해야 한다. $E(X_{F1},X_N)$에서 $E_1=E_3=\{x_6\}$, $E_2=E_4=\{x_5\}$이고 그림 1에서 알 수 있듯이 고장 상태 x_5와 x_6은 서로 도달가능하므로 조건 2.ii)에 부합한다. 따라서 영구 고장 모드 F_1을 극복하는 교정 제어기를 꾸밀 수 있다.

2. 제어기 설계 예제

이번 절에서는 예제 3에서 존재를 입증한 영구 고장 모드 F_1에 대한 내고장성 교정 제어기를 설계한다. 그림 2에서 제어기 C는 외부 입력 v와 관측기 B가 제공하는 상태 x를 입력으로 받아서 제어 입력 u를 출력으로 낸다. 따라서 C를 유한 상태 머신으로 표현하면 아래와 같은 입력/출력 머신 형태가 된다.

$$C = (X \times A_N,\ A_N,\ \Xi,\ \xi_0,\ \phi,\ \eta)$$

$X \times A_N$과 A_N은 각각 C의 입력과 출력 집합이며 Ξ은 C의 상태 집합, $\xi_0 \in \Xi$는 초기 상태, $\phi: \Xi \times X \times A_N \rightarrow \Xi$는 상태 천이 함수, 그리고 $\eta: \Xi \times X \times A_N \rightarrow A_N$는 출력 함수이다. 정상 상태 집합 X_N의 원소들은 0과 1 두 개의 서로 다른 출력 값을 내므로 C가 가지는 상태 원소들은

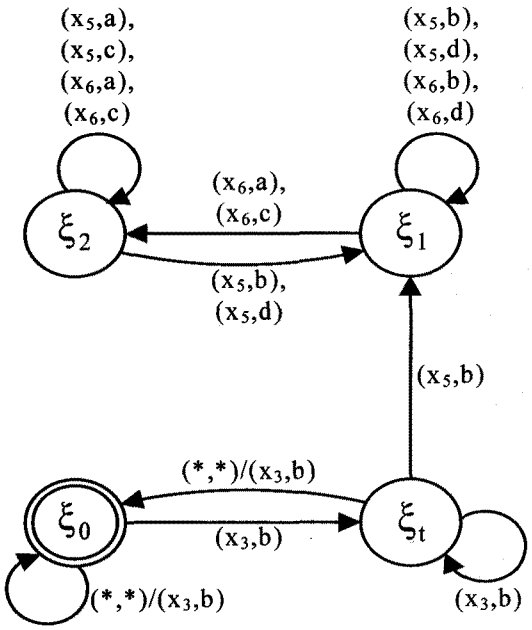
$$\Xi = \{\xi_0,\ \xi_t,\ \xi_1,\ \xi_2\}$$

로 총 4개이다(ξ_t, ξ_1과 ξ_2의 의미는 아래에 설명한다).

초기 상태 ξ_0에 있던 C는 Σ가 x_3과 안정 조합을 이룰 때, 즉 (x_3,b)일 때 'transition 상태' ξ_t로 천이한다. transition 상태 ξ_t는 비동기 머신 Σ가 모델 부정합이나 고장이 일어날 수 있는 상태로 진입했다는 사실을 제어기에 알려주는 역할을 한다[5]. C는 ξ_t에서 f_1의 발생을 기다린다. 고장이 일어나는 대신 다른 정상 입력이 들어오면 C는 다시 초기 상태 ξ_0으로 되돌아간다.

ξ_t에서 관측기 B가 주는 상태 측정값이 x_3에서 x_5로 바뀌는 순간 C는 영구 고장 모드 F_1의 발생을 인지한다. 예제 3에서 기술했듯이 f_1의 발생이 야기하는 모델 부정합을 없애기 위해서 C는 우선 Σ를 x_5에서 E_3의 한 원소로 보내주는 제어 입력을 출력해야 한다. $E_3=\{x_6\}$이며 $s(x_5,e)=x_6$이므로 C는 ξ_t에서 상태 x_5를 받는 순간 새로운 상태 $\xi_1 \in \Xi$로 천이하고 제어 입력 e를 발생시킨다.

Σ가 x_6으로 이동한 이후 C는 외부 입력의 변화에 따라서 내고장성을 유지하는 제어 입력을 Σ에 계속 넣어준다. 예를 들어 C가 ξ_1로 이동한 후 (또 Σ는 x_6으로 이동한 후) 외부 입력 v가 v=c로 바뀌었다고 하자. $s(x_3,c)=x_4$이므로 폐루프 시스템이 고장이 일어나지 않은 상태와 같이 행동하려면 Σ가 E_4의 한 원소로 다시 천이해야 한다. $E_4=\{x_5\}$이고 x_5는 x_6으로부터 입력 g에 의해 도달 가능하므로 제어기 C는 외부 입력이 c로 바뀌는 순간 상태 ξ_2로 이동하고 제어 입력을 u=g로 출력한다. 이런 식으로 교정 동작을 반복하면 그림 2의 폐루프 시스템은 영구 고장 모드 F_1을 극복하게 된다.



state	output
ξ_0	v (valid)
ξ_t	v
ξ_1	e
ξ_2	g

*: any valid value

그림 3. 내고장성 교정 제어기 C의 상태흐름도
Fig. 3. Signal flow diagram of the fault tolerant corrective controller C.

그림 3은 제어기 C의 상태흐름도로서 앞에서 설명한 고작 극복 과정과 부합된다. 초기 상태 ξ_0에서 C가 입력 (x_3,b)를 받으면 transition 상태 ξ_t로 천이한 후 고장이 발생하지 않으면 다시 초기 상태로 복귀하는 원칙이 그림에서 구현되어 있다. (*,*)는 Σ가 가질 수 있는 임의의 유효한 상태-입력 조합을 말하며 $(*,*)/(x_3,b)$는 유효한 상태-입력 조합 중 (x_3,b)를 제외한 집합을 말한다. 또 ξ_t에서 입력 (x_5,b)를 받으면 제어기는 ξ_1로 천이하는데 이것은 위에서 설명했듯이 영구 고장 모드 F_1이 발생하였다는 뜻이다. C가 ξ_1로 천이한 순간 제어 입력 u=e를 출력으로 내기 때문에(표 참조) 비동기 머신 Σ는 상태 x_6으로 즉시 이동한다. 그림 3에서 볼 수 있듯이 C의 상태는 바뀌지 않고 ξ_1을 유지한다. 또 ξ_1에서 외부 입력이 a 또는 c로 바뀌면 제어기 C는 상태 ξ_2로 천이하여 제어 입력 u=g를 출력함으로써 Σ를 x_5로 옮긴다. 반대로 상태 ξ_2에서 외부 입력이 b 또는 d로 바뀌면 C는 상태 ξ_1로 천이하여 제어 입력 u=e를 출력함으로써 Σ를 x_6으로 옮긴다. 외부 관리자에 의해 영구 고

장이 수리되는 시간까지 C는 상태 ξ_1과 ξ_2 사이를 반복하여 천이하면서 내고장성을 실현한다.

VI. 결 론

본 논문에서는 영구 고장이 존재하는 입력/출력 비동기 순차 머신의 내고장성 극복 문제를 교정 제어를 이용하여 해결하였다. 이번 논문의 주요 기여도는 영구 고장 발생 후에도 정상적인 동작을 실현할 수 있는 도달가능성이 비동기 머신에 잔존하면 교정 제어를 이용하여 내고장성을 구현할 수 있다는 사실을 입증한 일이다. 과도 고장과는 달리 영구 고장 발생 후 머신이 옮겨 갈 수 있는 상태가 한정되므로, 폐루프 시스템이 내고장성을 가질 수 있는 가능성은 과도 고장 경우보다 더 낮다. 본 논문에서는 우선 복수 개의 영구 고장 모드가 존재할 때 관측기가 영구 고장의 종류와 발생 시기를 정확하게 탐지할 수 있는 조건을 규명하였다. 그런 다음 그림 2의 폐루프 시스템이 고장 발생 후에도 고장이 일어나기 전의 정상적인 입력-출력 동작 특성을 그대로 유지하도록 하는 교정 제어기가 존재할 필요충분조건을 밝혔다. 일련의 예제들을 통하여 관측기와 제어기의 존재조건 확인 과정, 그리고 내고장성 교정 제어기의 설계 과정을 보였다.

이번 논문에서 제안한 내고장성 교정 제어기는 우주공간, 원자력 발전소 등 시스템의 즉각적인 고장 수리가 불가능한 환경에서 작동해야 하는 디지털 시스템에 적용하여 내고장성을 높이는 데 기여할 수 있다. 후속 연구로서 제안된 교정 제어기를 FPGA와 ASIC 기반으로 구현하는 작업이 추후 진행될 예정이다.

참 고 문 헌

[1] Z. Kohavi, Switching and Finite Automata Theory, 2nd ed. New York: McGraw-Hill, 1978.

[2] C. H. (Kees) Van Berkel, M. B. Josephs, and S. M. Nowick, "Scanning the technology: applications of asynchronous circuits," Proceedings of the IEEE, vol. 87, no. 2, pp. 223-233, 1999.

[3] J. Hammer, "On the modeling and control of biological signal chains," Proceedings of IEEE International Conference on Decision and Control, pp. 3747-3752, 1995.

[4] J. Hammer, "On the corrective control of sequential machines," International Journal of Control, vol. 65, no. 2, pp. 249-276, 1996.

[5] 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, 2003.

[6] 양정민, "비결정 모델에 대한 비동기 순차 회로의 교정 제어 II: 제어기 설계", 전자공학회논문지, 제 45권 SC 4호, pp. 11-19, 2008.

[7] J.-M. Yang and J. Hammer, "State feedback control of asynchronous sequential machines with adversarial inputs," International Journal of Control, vol. 81, no. 12, pp. 1910-1929, 2008.

[8] J.-M. Yang and S. W. Kwak, "State recovery for input/output asynchronous machines using output feedback," Proceedings of the 7th Asian Control Conference, pp. 1193-1198, 2009.

[9] 양정민, 곽성우, "외란 입력을 극복하기 위한 입력/출력 비동기 머신의 교정 제어," 전기학회논문지, 제58권 3호, pp. 591-597, 2009.

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

[11] P. K. Samudrala, J. Ramos, and S. Katkoori, "Selective triple modular redundancy (STMR) based single-event upset (SEU) tolerant synthesis for FPGAs," IEEE Transactions on Nuclear Science, vol. 51, no. 5, pp. 2957-2969, 2004.

[12] S. H. Zad, R. H. Kwong, and W. M. Wonham, "Fault diagnosis in discrete-event systems: framework and model reduction," IEEE Transactions on Automatic Control, vol. 48, no. 7, pp. 1199-1212, 2003.

[13] X. Geng and J. Hammer, "Input/output control of asynchronous sequential machines," IEEE Transactions on Automatic Control, vol. 50, no. 12, pp. 1956-1970, 2005.

저 자 소 개

양 정 민(정회원)

2010년 7월 대한전자공학회 논문지
제 47 권 SC편 제 4 호 참조