TMR시스템의 고장안전제어를 위한 FPGA 개발
(A FPGA Development for the Fail Safe Control of TMR System)

강민수* 이정석* 김현기* 유광균** 이기서***
Kang Min-Soo Lee Jung-Suk Kim. Hyun-Key Ryu Kwang-Kyun Lee Key-Seo

ABSTRACT

This paper proposes the failsafe control logic which has applied to the voting on the TMR system by using FPGA

The self-detection circuit is also designed for detecting a characteristic of fault at TMR system. The fault producing in the self-detection system is largely classified among an intermittent fault, a transient fault and a permanent fault.

If it is happened to the permanent fault, the system can be failed. Therefore, it is designed the logic circuit which is not transferred the permanent fault to the system after shut off output. The control logic of the Fail Safe proposed in the paper is required for a circuit integrate of device to minimize the failure happened. Therefore, it makes to design FPGA with modeling of VHDL. The circuit of the Fail Safe of TMR system is able to apply to nuclear system, rail-way system, aerospace and aircraft system which is required for high reliability.

1. 서론

기존의 제어 시스템에서 고장의 발생은 예측하지 못한 치명적인 결과를 초래할 수 있다. 이에 대한 대책으로 신뢰성이 높은 부품을 선택하지만 외부의 교란, 구현과정의 오류, 환경 등에 의해서 고장의 확률은 높다. 따라서 하드웨어 여분을 이용한 TMR[1](Triple Modular Redundancy)로 다수결 출력이 가능한 보터[3](Voter)를 출력 단에 설계하여 결함허용[6][7] 시스템을 구현하였다. 보터는 입력 수에 따라 다수결법칙(Majority rule)에 의해 출력됨으로써 결함을 마스킹(Masking) 할 수 있다. 그러나 영구결함(Permanent fault)의 영향이 시스템에 미치게 되면 예측할 수 없는 결과를 초래하게 된다. 특히 안전 측이 정의된 철도 시스템에서는 안전출력을 보장할 수 있는 안전성(Safety) 확보는 필수적이다. 따라서 고장안전 제어는 반드시 필요하게 된다. 고장안전제어[5]를 위해서는 정확한 자기검출[6](Self Detection)이 요구되며 결함이 발생할 경우 결함의 특성을 판단하기 위하여 결함특성 진단 회로가 필요하다. 이러한 검사에 의해 고장 신호가 발생되어야 한다. 이미 Wang과 Chang은 정보(Information)여분을 이용하여 k-out-of-n coding을 이용한 순차적인(Sequential) 고장안전 시스템을 실현하기 위하여 결함 또는 오류를 정확히 검출한 후 결함의 영향을 무시하는 방법으로 대체 모듈로 스위칭(Switching)하거나 재시도(Restart)하는 방법을 택하였다. 이때 완전한 고장안전회로를 구현하기 위해서 정확한 자체 검사회로를 이용하여 고장안전제어 가능하게 하였다.

본 연구에서는 TMR 시스템의 고장안전제어 로직을 시스템 레벨에서 결함의 발생을 최소화하기 위하여 VHDL을 이용한 FPGA로 집적화 하였으며, 비트 단위에서 로직의 확장으로 바이트 단위까지 결함을 마스킹하는 결함 허용 시스템을 구현하였다.

*광운대학교 제어계측공학과 박사과정.
**한국철도대학 철도신호과 교수, 정회원
***광운대학교 제어계측공학과 정교수, 정회원

또한 결함이 오류로 발전할 가능성이 있는 결함 즉, 영구결함을 판단하기 위해서 결함의 특성을 진단하는 결함 진단회로를 구현하였다. 만약 결함이 오류로 전이되어 시스템에 고장이 발생할 경우 고장신호를 발생하여 시스템을 안전한 상태로 유지하여 시스템을 중단하게 설계하였다.

2. 결함허용 시스템

결함허용 시스템은 시스템 내의 결함이 발생하더라도 명시된 알고리즘을 정확하게 수행할 수 있는 시스템이다. 여기서 허용의 의미는 결함을 통과(Passive) 하는 의미로 볼 수 있다.

2.1. 결함 분석

결함은 부품의 물리적 결점, 운영자의 실수 또는 부정확한 설계에서 발생하는 하드웨어나 소프트웨어의 잘못된 상태를 말하며, 오류는 프로그램이나 데이터 구조의 결함으로부터 발생하는 결과로써 고장으로 연결되는 단계이다. 고장은 시스템의 동작이 요구하는 시스템의 명세로부터 벗어날 때 발생하는 상태가 되어 그 기능을 수행할 수 없는 불량상태와 같다. 이러한 결함의 특성으로 발생하는 영역에 따라 하드웨어 또는 소프트웨어적인 결함을 구분하며, 결함의 영향이 전체적인가, 국부적인가에 따른 결함의 영향범위를 회로에 따라서 분류한다. 본 연구에서는 결함을 취급하는데 일반적인 기준이 되는 지속시간에 따라 분류하였다.

- 간헐(Intermittent) 결함: 결함이 발생하는 최초에 결함이 나타났다 사라지고, 다시 나타나는 동작을 되풀이하는 결함으로 하드웨어의 불안전성, 접촉불량에 기인하여 발생
- 과도(Transient)결함: 매우 짧은 시간에 결함의 영향이 나타났다 사라지는 현상으로 외부의 요인에 의한 순간적인 조건에 의해 발생
- 영구(Permanent) 결함: 교정동작이 취해지지 않는 한 무한정 존재하는 결함

2.2. TMR 시스템 구성

일반적으로 NMR(N Modular Redundancy)시스템은 임계적인 응용분야에 이용되며 시스템의 신뢰도를 향상시키기 위하여 시스템을 몇 개의 동일한 모듈로 분할하여 모듈을 N중화하여 결함을 마스킹한다. 그림 1은 3개의 모듈을 사용하여 결함을 허용하는 로직을 블럭도로 나타내었다.

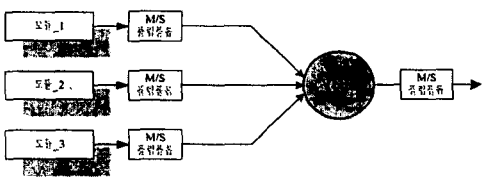


그림 1 TMR 시스템과 보터

그림 1은 수동적인 기법에 대표적으로 사용되는 TMR 구조로써 보터가 각 모듈의 값을 비교하여 과반수 이상 (2-out-of-3)의 값이 동일할 경우 그 값을 출력한다. 만약 결함이 발생하면 그 모듈의 값을 마스킹 하여 결함의 영향이 다음 단계로 전이되는 것을 방지할 수 있다. 이 기법은 재구성(Reconfiguration)이나, 복구(Recovery)등의 기능 없이도 결함을 허용하며, 동기를 맞추기 위하여 마스터 슬레이브(Master/Slave)플립플롭을 입력단과 출력단에 각각 연결하였다.

3. 고장안전(Fail Safe)회로

안전도(Safety) 설계를 목표로 하는 고장안전제어의 한 방법으로는 결함이 발생한 후부터 시스템에 고장의 영향을 주기 전까지 미리 지정된 안전한 값을 출력하는 것이다. 미리 정의된 안전한 값이 없다면 즉 TMR 시스템의 경우 "0"이나"1"의 값 중 어떠한 값이 안전한지 알 수 없다. 따라서 고장안전 제어 시스템이 고장의 영향이 있을 경우 그 효과가 최소한으로 축소되기 위해서 안전한 상태로

출력 값이 생성되어야만 한다. 안전한 값이 출력되기 위해서 결함의 특성 및 결함 검출이 자동적으로 증명되어야 하기 때문에 본 연구에서는 결함의 특성을 지연되는 시간으로 분류하였으나 기준이 되는 시간이 없으므로 카운터 회로를 응용하여 연속되는 3클럭 이상이 계속해서 결함으로 발생할 경우에 영구 결함으로 판단하였다. 그러나 결함 신호가 4클럭 미만일 경우 연속되지 않는 한 영구결함이라고 인정하지 않았다. 그러나 영구결함이 발생했지만 TMR 시스템에 의해 결함이 허용되는 경우에는 시스템에 고장의 영향을 주지 않으므로 고장신호만 연속적으로 유지하도록 했고 나머지 한 시스템에서 고장신호가 발생할 경우 이를 고장안전 제어하도록 했다. 그러나 안전한 기준 값을 정하기가 곤란하기 때문에 고장 신호를 "1"로 출력할 수 있게 구현하였다. 그림 2는 결함허용 시스템에 고장안전 제어 가능한 회로를 블록도로 나타내었다.

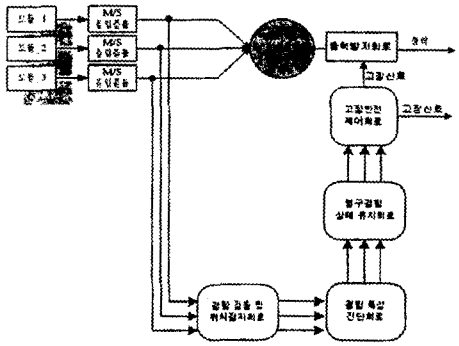


그림 2 고장안전제어를 위한 블럭도

그림 2에서는 3개의 동일한 모듈이 동기화를 위하여 마스터 슬레이브 플립플롭을 통하여 보터를 거쳐 다수결 출력을 하게 된다. 만약 3개의 모듈중 어느 한 모듈에서 결함이 발생하면 결함검출 및 위치검지회로에서 결함을 검출한 후 결함특성 진단회로에서 결함의 특성을 지연 시간에 따라 간헐결함, 과도결함, 그리고 영구결함을 판단하게 된다. 만약 영구결함일 경우 영구결함 상태유지회로에서 영구결함 발생신호를 유지하고 있으면서 시스템은 계속해서 정상동작을 하게된다. 그리고 또 다른 모듈에서 영구결함이 발생하면, 이미 영구결함임을 인식하고 있는 고장신호와 현재 발생한 고장신호를 적용하여 시스템이 고장임을 알릴 수 있게 고장신호를 출력하고 출력방지 회로에서는 출력이 더 이상 발생되지 않게 출력을 차단하게 된다.

4. 결함 모델

시스템 내에서 또는 회로 내에서 결함은 신호선의 단락이나 시간의 지연과 같은 결함이 일반적으로 기대되는 결함이다. 그렇지만 제조상의 결함이나 제작자의 실수에 의해서도 발생할 수 있다. 따라서 입력되는 신호에 의해 기대하지 않았던 결과가 발생함으로써 결함을 모델해야 한다. 회로에서의 결함은 신호선(Signal Line)의 단선(Open), 전원이나 다른 신호선과 단락(Short)이 되었거나, 신호의 지연(Delay) 등에 의해 발생한다. 이러한 것을 수학적으로 모델링하여 표현한 것을 결함 모델(Fault Model)이라 한다. 결함 모델을 구성하고 표현하기 위해서는 일반적으로 논리 함수를 이용한다. 결함 모델이 회로내의 결함을 어느 정도까지는 표현할 수 있지만, 완벽하게 모든 결함을 모델링할 수는 없다. 다음의 그림3을 이용하여 일반적인 결함 모델에서의 결함을 표현하였으며, 결함 모델의 종류는 (a) Stuck-at 결함, (b) Stuck-open 결함, (c) Stuck-on 결함, (d) Open line 결함, (e) Bridging 결함, (f) Delay 결함 등으로 나눌 수 있다.

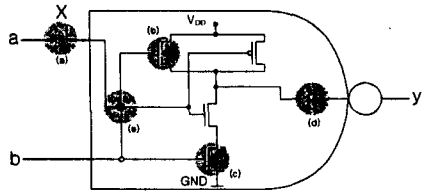


그림 3 CMOS Nand 게이트에서의

여러가지결함

4.1. 고착결함(Stuck-Fault)

결함을 모델하는데는 여러 가지가 있지만, 본 논문에서는 다수결 보터의 입력과 출력에 대해서 발생할 수 있는 결함에 대하여 고착 결함(Stack-at-X) 결함을 모델링 하였다. 고착결함은 가장 일반적인 결함 모델(Fault Model)이 고착결함[4] (stuck-fault) 모델이다. stuck-fault 모델은 stuck-at-0, stuck-at-1로 표현한다. 모듈내의 결함의 결과가 입력 또는 출력이 물리적으로 stuck-at 1, 0으로 응답하고 회로의 기본적인 결함의 결과가 바뀌지 않으며, 결함이 영구 결함일 경우 입력이나 출력 중 어느 하나가 물리적으로 논리"1" 이나 "0"에 물려 생기는 고장을 말한다. 아래의 그림은 보터의 각 도선에 발생할 수 있는 Stack-at-0 과 Stack-at-1 결함을 나타내었다.

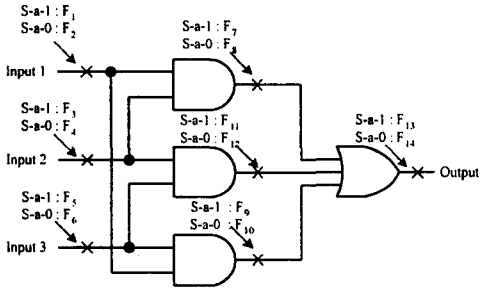


그림 4 다수결 보터에서 발생 가능한 결함

각각의 Stuck-at-1결함이 다수결 보터에서 발생했을때의 출력을 $O(F_1)$, $O(F_3)$, $O(F_5)$, $O(F_7)$, $O(F_9)$ $O(F_{11})$, $O(F_{13})$으로 하고 그에따른 출력의 논리를 표 1에 나타내었다. 또한 Stuck-at-0 결함이 다수결 보터에서 발생했을 경우 $O(F_2)$, $O(F_4)$, $O(F_6)$, $O(F_8)$, $O(F_{10})$, $O(F_{12})$, $O(F_{14})$로 하고 그에 따른 출력의 논리를 표 2에 나타내었다.

표 1. Stuck-at-1 결함 발생시 출력표

In1	In2	In3	O_{Normal}	$O(F_1)$	$O(F_3)$	$O(F_5)$	$O(F_7)$	$O(F_9)$	$O(F_{11})$	$O(F_{13})$
0	0	0	0	0	0	0	1	1	1	1
0	0	1	0	1	1	0	1	1	1	1
0	1	0	0	1	0	1	1	1	1	1
0	1	1	1	1	1	1	1	1	1	1
1	0	0	0	0	1	1	1	1	1	1
1	0	1	0	1	1	1	1	1	1	1
1	1	0	0	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1

표 2. Stuck-at-0 결함 발생시 출력표

In1	In2	In3	O_{Normal}	$O(F_2)$	$O(F_4)$	$O(F_6)$	$O(F_8)$	$O(F_{10})$	$O(F_{12})$	$O(F_{14})$
0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	0
0	1	1	1	1	0	0	1	1	0	0
1	0	0	0	0	0	0	0	0	0	0
1	0	1	1	0	1	0	1	0	1	0
1	1	0	1	0	0	1	0	1	1	0
1	1	1	1	1	1	1	1	1	1	0

표 1과 2를 이용하여 Stuck-fault 가 발생했을 때 결함을 검출할 수 있는 입력 패턴을 얻기위한 결함 검출표를 표 3과 표 4에 나타내었다.

표 3. Stuck-at-0 결함 발생시 출력표

In1	In2	In3	(F_1)	(F_3)	(F_5)	(F_7)	(F_9)	(F_{11})	(F_{13})
0	0	0	0	0	0	1	1	1	1
0	0	1	1	1	0	1	1	1	1
0	1	0	1	0	1	1	1	1	1
0	1	1	0	0	0	0	0	0	0
1	0	0	0	1	1	1	1	1	1
1	0	1	0	0	0	0	0	0	0
1	1	0	0	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0	0

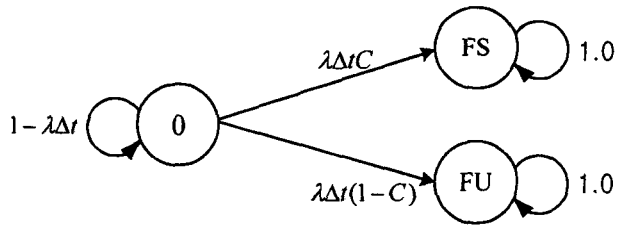
표 4. Stuck-at-1 결함 발생시 출력표

In1	In2	In3	(F_2)	(F_4)	(F_6)	(F_8)	(F_{10})	(F_{12})	(F_{14})
0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0
0	1	1	0	1	1	0	0	1	1
1	0	0	0	0	0	0	0	0	0
1	0	1	1	0	1	0	1	0	1
1	1	0	1	1	0	1	0	0	1
1	1	1	0	0	0	0	0	0	1

5. 시스템 평가

5.1 고장안전에 대한 안전도(Safety)

시스템의 신뢰성 평가를 마코브(Markov) 모델을 이용하였다. 마코브 모델은 시스템이 가질 수 있는 상태에 따라 표현될 수 있는 확률적인 시스템 평가 모델을 제공한다. 즉, 설계된 시스템의 신뢰도 표현을 위해 각각의 상태는 동작 가능한 모듈로 표현된다. 시스템에서 각 모듈은 동작 상태와 결함 상태에 한 조건이 되고 N 개의 모듈로 이루어진 시스템에 대한 모델은 2^N개를 가지게 된다. 이러한 상태 변화를 상태 전이(State Transition)라고 한다. 마코브 모델을 이용하여 신뢰도를 표현하기 위해서 시스템의 고장률 과 시스템이 시간간격 $\triangle t$에서 발생할 확률을 $\lambda\triangle t$로 표현한다. 그림 5에서 시스템이 정상동작하는 상태와 고장이 났지만 안전한 상태 고장이 났지만 안전하지 못한 상태를 나타내었다. 안전한 고장(FS:Failed Safe)상태는 자기진단에서 결함이 발견된 상태라야 한다. 결과적으로 비안전 고장(FU:Failed Unsafe)는 결함이 있음에도 불구하고 검출되지 않은 상태가 된다.



C: fault detection Coverage

그림 5. 안전도 계산을 위한 마코브 모델

마코브 모델에서 표현된 시스템의 안전도는 다음과 같이 정의할 수 있다.

$$(S) = P_0(t) + P_{FS}(t) \tag{1}$$

$P_0(t)$:어떤시간 t 에서 정상동작할 확률

$P_{FS}(t)$:어떤시간 t 에서 안전한 고장상태의 확률

이산시간에서 마코브모델에 의해서 표현된 시스템의 안전도는

$$\begin{aligned} P_0(t+\Delta t) &= (1-\lambda\Delta t)\ P_0(t) \\ P_{FS}(t+\Delta t) &= \lambda\Delta t C P_0(t) + P_{FS}(t) \\ P_{FU}(t+\Delta t) &= \lambda\Delta t(1-C)P_0(t) + P_{FU}(t) \end{aligned} \tag{2}$$

연속 시간계에서 마코브 모델의 미분방정식은

$$\begin{aligned} \frac{dP_0(t)}{dt} &= -\lambda P_0(t) \\ \frac{dP_{FS}(t)}{dt} &= -\lambda P_0(t) \\ \frac{dP_{FU}(t)}{dt} &= -\lambda(1-C)P_0(t) \end{aligned} \tag{3}$$

식(3)을 라플라스 변환하면

$$\begin{aligned} P_0(S) &= \frac{P_0(0)}{S+\lambda} \\ P_{FS}(S) &= \frac{\lambda C P_0(0)}{S(S+\lambda)} + \frac{P_{FS}(0)}{S} \\ P_{FU}(S) &= \frac{\lambda(1-C)P_0(0)}{S(S+\lambda)} + \frac{P_{FU}(0)}{S} \end{aligned} \tag{4}$$

여기서 $P_0(0)$, $P_{FS}(0)$, $P_{FU}(0)$는 각 초기상태의 기대되는 확률 값이다. 만약에 시스템이 초기상태 "0"에서 시작하고 초기값을 각각 $P_0(0)=1$, $P_{FS}(0)=0$, $P_{FU}(0)=0$이라고 가정했을 때 다음의 식으로 정리할 수 있다.

$$P_0(S)=\frac{1}{S+1}$$

$$P_{FS}(S)=\frac{\lambda C}{S(S+\lambda)}=\frac{C}{S}-\frac{C}{S+\lambda} \tag{5}$$

$$P_{FU}(S)=\frac{\lambda(1-C)}{S(S+\lambda)}=\frac{1-C}{S}-\frac{1-C}{S+\lambda}$$

식(5)를 시간영역으로 정리하면

$$P_0(t)=e^{-\lambda t}$$

$$P_{FS}(t)=C-Ce^{-\lambda t} \tag{6}$$

$$P_{FU}(t)=(1-C)-(1-C)e^{-\lambda t}$$

따라서 시스템의 안전도는

$$S(t)=P_0(t)+P_{FS}(t)=C+(1-C)e^{-\lambda t} \tag{7}$$

$$=C+(1-C)e^{-0.12\times10^{-8}}$$

ALTERA사의 FPGA 고장률: -0.12×10^{-8}

가 된다.

6. 실험 및 결과

TMR 시스템에서 3개의 동일한 모듈을 이용하여 보터에 의해 다수결 값을 출력할 수 있었다. 이는 시스템에 결함이 발생하더라도 시스템의 동작이 중단 없이 정해진 입력이나 출력의 정보가 계속해서 수행될 수 있는 기능을 갖게 구현하였으나, 만약 TMR 시스템에서 영구결함이 발생할 경우 시스템을 고장안전제어 하게 시스템을 다운시키는 방법으로 설계하였다. 아래의 그림에 시뮬레이션 결과를 나타내었다. 그림6은 아무런 결함이 없으며 보터를 통하여 정상적인 동작을 하고 있는 그림이다.

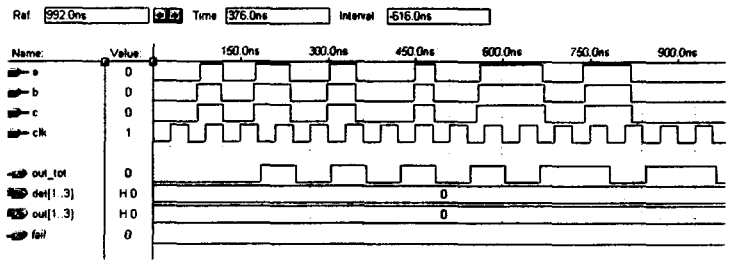


그림 6 정상적인 동작을 수행하는 상태

그림 7은 모듈 b에 결함이 발생하였지만 하나의 모듈이 단순히 결함이 아니라 고장이 발생했다하더라도 보터의 특성상 시스템에 영향을 주지 않으므로 시스템을 중단하지는 않는다.

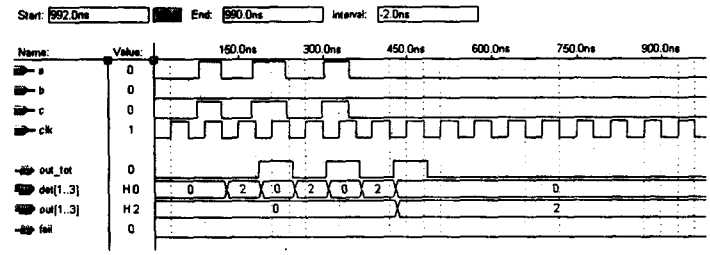


그림 7 모듈 b에 영구결함이 발생한 상태

그림 8은 이미 모듈 b에서 결함이 발생했으며, 발생한 신호를 영구결함 상태유지회로에서 그 값을 유지하고 있는 상태에서 모듈 a에서 영구결함이 발생하였다. 따라서 결함이 최소 2개의 모듈에서 발생했고 발생시간이 3클럭을 초과했기 때문에 시스템이 고장이라고 인식한다. 따라서 시스템을 안전측으로 동작할 수 있게 신호를 출력하고 시스템의 출력은 발생하지 않게 하고 시스템을 중단시킨다.

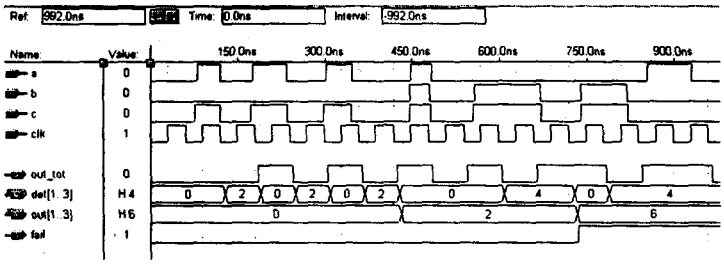


그림 8. 시스템 고장상태

7. 결론

인간의 생명과 관계되는 고 신뢰성을 갖는 시스템을 구현하기 위하여 TMR 시스템을 구현하였으며 발생된 결함을 검출하여 고장의 영향을 최소화하도록 하였다. 만약 결함이 고장의 영향을 가질 경우 결함의 특성을 파악하여 고장안전제어 가능하게 로직을 구성하여 시스템의 안정성을 확보할 수 있었다. 그러나 정확한 결함의 특성에 대해서는 아직도 명확하게 언급된 보고나 연구 자료가 없다. 따라서 전체 시스템의 상태를 완벽하게 파악해서 잘못된 결함 특성의 진단을 미연에 방지할 수 있는 연구가 앞으로 계속 되어야 하겠다. 이러한 방법으로 개발된 결함허용의 고장안전제어 시스템은 고신뢰성을 요구하고 인간의 안전에 관계되는 철도, 항공, 원자로, 우주항공 등의 시스템에 적용 할 수 있다.

참고문헌

1.Barry W. johnson,"Design and Analysis of Fault Tolerant Digital System,"Addison Wesley,1989.
2.Danel P. Siewior,"Fault-Tolerant Computer", IEEE Computer, pp26-37,1990.
3.Behrooz parhami,"Voting networks",IEEE Trans.on Reliability, Vol.40,No.3,pp.380-393,Aug.,1991.
4.Abramovici,Breuer,Friedman,"Digital Systems Testing and Testable Design",IEEE PRESS,1990.
5. A.Saeed, T.Anderson and M.Koutny,"A Formal Model for Safety-Critical computing systems",IFAC SAFECOMP'90,pp 1-6 1990.
6. Mark G.Karpovsky and Saeed M.chaudhry,"Design of Self-Diagnostic Boards By Multiple Signature Analysis",IEEE Trans.on computer,Vol.42, pp1035-1044,sep.1993.
7.양성현, 이기서,"Fault Tolerance를 위한 시스템의 동작방식에 대한 연구 한국 통신학회 논문지 Vol.17,No.11,November,1992.