

인공 위성 탑재 소프트웨어의 소프트 에러 고장 주입시험

배지훈¹, 김형신^{2*}

¹만도, ²충남대학교 컴퓨터공학과

Soft Error Fault Injection Test for Spacecraft Flight Software

Ji-Hoon Bae¹, Hyungshin Kim^{2*}

¹Mando

²Department of Computer Science and Engineering, Chungnam National University

요약 인공위성은 태양과 심우주에서 발생한 우주 방사선 환경인 지구 주위의 궤도에서 동작해야 한다. 우주방사선은 위성의 탑재 컴퓨터의 하드웨어에 SEU(Single Event Upset)라는 에러를 유발할 수 있다. SEU는 비 파괴성 오류로 일시적으로 메모리나 레지스터에 저장된 비트를 반전시키는 소프트 에러다. 인공위성 탑재 컴퓨터는 이러한 소프트 에러를 감내해야 한다. 위성의 탑재 컴퓨터는 소프트 에러가 발생하더라도 오동작을 하지 않도록 결함을 검출하고 대응할 수 있어야 하며, 개발과정에서 이러한 소프트 에러에 대해 오류를 검출하고 대응할 수 있는지를 검증해야 한다. 본 논문에서는 NASA의 표준 탑재 소프트웨어인 cFS(Core Flight Software)가 소프트 에러에 대해 강인한지 고장 주입시험을 시행한 결과를 정리하였다. 기존 테스트 연구에서 적용한 것과 유사하게 소프트 에러를 메모리 전역 변수 영역, 스택 영역, 프로세서 레지스터에 주입하였으며, 탑재 컴퓨터 시뮬레이터에 cFS를 실행하면서 고장을 주입하였다. 고장 주입시험 결과 cFS에 소프트 에러 주입 시 탑재 소프트웨어의 주요 기능과 명령처리 기능에 문제가 생길 수 있는 것을 확인하였다. 이 시험결과를 통해 실제 위성의 탑재 소프트웨어에 다양한 소프트 에러 주입 방법을 시도하여, 향후 자동 고장 주입 도구 구현에 적용할 수 있을 것으로 기대된다.

Abstract Satellites operate in an orbital environment where radiation particles from the sun and deep space. When radiation particles hit a flight computer, it may induce an SEU(Single Event Upset) on the computer. An SEU is a non-destructive, transient soft error that flips a single bit of memory and registers on the processor. Flight computers should be fault-tolerant when this event occurs and detect and respond to them. During the development, flight software should be resilient against those soft errors. This paper reports the fault-injection testing result with NASA's cFS(Core Flight Software) flight software. Soft errors were injected into memory global variable area, stack area, and the processor registers as a previous testing study while executing cFS on a simulator of the flight computer. The test showed that soft errors might affect flight application behavior and the telecommand interface. The proposed soft error injection testing will contribute to the development of fault injection testing tools.

Keywords : Satellite Software, Soft Error Testing, Fault Injection Testing, Flight Computer, Single Event Upset

본 연구는 국방과학연구소의 "저궤도 고기동 정찰용 위성체 탑재 소프트웨어 신뢰성 검증방안 연구"(관리번호:UD190017FD)의 지원을 받아 수행되었음.

*Corresponding Author : Hyungshin Kim(Chungnam National Univ.)

email: hyungshin@cnu.ac.kr

Received May 16, 2022

Revised June 17, 2022

Accepted July 7, 2022

Published July 31, 2022

1. 서론

인공위성의 운용 환경에는 태양이나 심우주에서 발생한 우주 방사선과 같은 고에너지 입자들이 존재한다. 이러한 고에너지 입자들에 의해 위성 탑재체에 전기적인 장애를 일으키는 SEE(Single Event Effect)가 발생할 수 있다[1]. SEE에는 크게 두 가지 종류가 있는데, 시스템에 영향을 주지만 회로가 다시 정상 작동을 할 수 있는 비 파괴형 SEE와 회로에 심각한 손상이 발생하여 제 기능을 하지 못하게 되는 파괴형 SEE가 있다. 비 파괴형 SEE와 같이 일시적으로 영향을 주는 오류를 소프트 에러(Soft Error)라고 하며, 소프트 에러 중 메모리나 레지스터에 저장된 비트가 반전되는 현상을 SEU(Single Event Upset)라고 한다. 인공위성 탑재 컴퓨터에 SEU가 발생하면 프로그램에서 사용되는 변수의 값이 변경되는 데이터 오류 또는 프로그램이 정상적인 실행 흐름에서 벗어나는 제어 흐름 오류가 발생할 수 있다[2].

고신뢰성이 요구되는 위성 시스템에서 소프트 에러를 감내하지 못한다면 위성의 임무 수행에 치명적일 수 있다. 소프트 에러가 일어나는 것을 방지하기 위해 상용 위성에서는 내방사성을 갖는 특수한 부품을 사용하지만, 소프트 에러의 발생 가능성이 여전히 존재하게 된다. 따라서, 위성 탑재 컴퓨터에는 소프트 에러가 발생하더라도 위성의 운영이 안정적으로 지속할 수 있도록 하는 고장 검출 기능과 복원 기능을 가지고 있으며, 탑재 컴퓨터의 개발과정에서는 소프트 에러가 발생하더라도 위성이 감내할 수 있을지 검증하는 작업이 필요하다.

인공위성 탑재 소프트웨어에 대한 소프트 에러 시험은 실제로 방사선 빔을 하드웨어에 조사하여 수행할 수도 있으나, 시간과 비용이 많이 들고, 반도체에 손상을 줄 수도 있는 등의 단점이 있다. 따라서 본 연구에서는 실제 방사선 시험을 하는 대신, 프로그램을 이용한 고장 주입 방법을 사용하여 인공위성 탑재 소프트웨어의 소프트 에러에 대한 오류 검출 여부, 대응 기능의 작동여부를 분석하였다. 소프트 에러 결함 주입시험의 대상으로 NASA의 cFS(Core Flight Software)[3]를 사용하였다. 일반적으로 인공위성의 탑재 소프트웨어는 공개하기 어려운 특성이 있지만, cFS는 NASA가 표준화한 공개 소프트웨어형 탑재 소프트웨어이기 때문에 소스 코드가 공개되어 고장 주입시험이 가능하였다. cFS 탑재 소프트웨어를 NOS3[7] 시뮬레이터에서 실행하면서 탑재 컴퓨터의 메모리와 레지스터에 소프트 에러 결함을 주입해, 위성 소프트웨어가 소프트 에러 발생 시 어떠한 영향을 받는지 분석하였다.

인공위성 탑재 소프트웨어의 개발과정에서 적용한 다양한 신뢰성 확보방안에 대한 연구 결과들은 발표된 경우가 있으나[4-6], 탑재 소프트웨어에 소프트 에러 결함을 주입하여 결함의 영향을 분석한 연구는 발표된 것이 없다.

소프트 에러 결함 주입은 소스 코드 수정을 통해, 소프트 에러의 효과를 모사하는 방법으로 진행하였다. 메모리의 전역 변수 영역과 스택 영역에 결함을 주입하여 전역 변수, 지역 변수의 값을 변경시켜 데이터 오류를 발생시켰고, 프로그램 카운터 레지스터(PC)의 비트 값을 변경시켜 소프트웨어가 정상적인 실행 흐름에서 벗어나도록 제어 흐름 오류를 발생시켜 소프트 에러 결함 주입 시험을 진행하였다.

소프트 에러 고장 주입시험의 결과를 네 가지로 분류하고, 주입한 오류의 유형에 따른 탑재 소프트웨어의 반응 비율을 분석하였다. 고장 주입시험을 통하여 NASA cFS에 소프트 에러 발생 시 탑재 소프트웨어들의 주요 기능 수행 및 원격명령 처리에 문제가 생길 수 있는 것을 확인하였다.

이 논문의 구성은 다음과 같다. 2장에서는 본 연구의 대상인 cFS 탑재 소프트웨어와 소프트 에러 결함에 의한 영향 분석 연구를 소개하였다. 3장에서는 소프트 에러 결함 주입과정을 설명하고, 4장에서는 결함 주입시험 결과를 분석한다. 5장에서는 결론을 맺는다.

2. 이론적 배경 및 선행연구

2.1 cFS 고장 주입 대상 태스크

cFS는 NASA가 과거에 개발한 위성 소프트웨어들을 기반으로 하여, 위성 임무 수행 시 공통으로 필요한 기능들로 구성되어 있으며 이것은 오픈소스로 웹사이트에서 다운로드가 가능하다. cFS는 현재까지 NASA의 LRO, Morpheus, LADEE 프로젝트와 JHU/APL(Johns Hopkins University Applied Physical Laboratory)의 Parker Solar Probe 프로젝트 등에 사용되었으며, 이외에도 큐브셋 위성에도 사용되고 있다[8,9].

본 연구에서 소프트 에러 발생 시 NASA cFS가 받는 영향을 분석하기 위해 cFS 태스크 중에서 HS(Health and Safety)태스크와 TO(Telemetry Output) 태스크를 선정하였다. HS 태스크는 cFS 애플리케이션들에 문제가 생겼는지 감시하는 기능을 수행하고, Watch dog 타이머 서비스를 수행한다. 이러한 기능으로 볼 때, HS

태스크는 안전한 위성 소프트웨어의 동작을 위해 필수적인 태스크이다. 따라서, HS 태스크 실행 중 소프트 에러 결함이 발생했을 때 감내할 수 있는지 확인하기 위해 결함을 주입하기로 하였다. TO 태스크는 외부와의 통신을 담당하기 때문에 그 중요성을 고려하여 고장 주입 대상으로 선정하였다.

2.2 소프트 에러 결함 주입 연구

소프트 에러 결함을 소프트웨어에 주입하고, 주입한 결함이 시스템에 미치는 영향을 분석하는 연구가 진행되었다. Ignat[10]은 실시간 운영체제에서 실행되는 애플리케이션의 소프트 에러에 대한 민감성을 연구하였으며, 실시간 운영체제에 소프트 에러가 발생하였을 때 그 영향이 어느 수준까지 미치는 지를 분석하였다. 이 연구에서는 MicroC 커널의 주요 서비스 실행 중 Braekpoint를 이용하여 원하는 시기에 레지스터에 단일 비트 플립 결함을 주입하고, 주입된 결함의 영향을 분석하였다.

Amarnath[11]는 응용 프로그램 실행 중 메모리의 각 영역에 존재하는 데이터에 소프트 에러 결함을 주입하여 데이터 값의 단일 비트를 무작위로 반전시키고, 소프트 오류에 대한 취약성을 분석하였다. 해당 연구에서는 결함 주입 및 결과분석 도구인 BIFIT(Binary Instrumented Fault Injection Tool)을 만들고 응용 프로그램 실행 중 결함을 주입하였다.

3. 결함주입 방법

본 연구에서는 사용한 소프트 에러 결함 주입 방법은 기존의 소프트 에러 결함 주입 방법에 관련된 연구들에서 사용한 방법을 주입 방법으로 구현하였다. 결함 주입 위치를 결정하는 방법은 [11]을, 결함 주입 시점을 결정하는 방법은 [11,12]을 적용하였다. 결함 주입 위치는 함수별 명령어 스트림을 n개의 세그먼트로 나누고, 세그먼트의 앞부분에서 결함이 주입되도록 하였다. 결함 주입 시점으로는 결함 주입 지점을 일정 횟수 이상 실행 시 결함이 주입되도록 하였는데, 실행되는 횟수는 랜덤 함수를 이용해 무작위로 선택된 값만큼 결함 주입 지점을 실행 후 결함이 주입된다. 본 연구에서는 n의 값은 2이고, 실행되는 무작위 값은 1~30의 값 사이에서 결정되도록 하였다. 이러한 시험 값의 선택은 기본적인 시험 결과를 얻기 위해 작은 값을 선택한 것이다.

Table 1. Fault injection strategy for data error generation

Injection Target	Global Variables	Local Variables
Injection Position	Inject at the main loop of cFS Apps	every functions in the App
	HS_ProcessMain, TO_RcvMsg	
Injection Time	Inject after a delay	Inject after a delay
	Delay values in random	Delay values in random
Injection Info	Number of bit flips	Number of bit flips
	multiple bit flips	multiple bit flips

결함 주입 대상은 메모리의 전역 변수 영역, 스택 영역에 존재하는 전역 변수 및 지역 변수 그리고 프로그램 카운터 레지스터다. 소프트 에러로 인해 발생할 수 있는 데이터 오류 상황을 재현하기 위해 전역 변수와 지역 변수의 값을 대상으로 비트 플립 결함을 주입하였고, 제어 흐름 오류 상황을 재현하기 위해 시뮬레이터의 EIP 레지스터의 값에 비트 플립 결함을 주입하였다.

비트를 플립시키는 범위는 각 변수의 비트 길이대로 결함을 주입하였고, EIP 레지스터는 매 k 비트 마다 구간을 나누어 결함 주입을 진행하였는데, 본 테스트에서는 k는 4를 사용하였다. 플립되는 비트는 랜덤 함수를 이용하여 다중 비트가 플립되도록 결함을 주입하였다. 데이터 오류를 생성하는 결함주입 정보는 Table 1과 같다.

제어 흐름 오류를 발생시키는 결함 주입을 위해, 앞에서 데이터 오류를 발생시키는 결함을 주입하기 위해 사용한 결함 주입 위치와 결함 주입 시점에 대한 정보를 그대로 사용한다. 여기에 추가로 매 k번째 비트 마다 결함을 주입하도록 하였고, k의 값은 4이다. 매 k번째 비트 마다 나누어서 결함을 주입하는 방식은 [13]의 연구에서 적용한 결함 주입 방법을 참고하였다. 결함 주입을 위해 필요한 정보인 주입 위치, 시점, 비트 플립의 범위, 주입 대상(변수 또는 레지스터)을 결정하고 소스 코드를 수정하고 컴파일을 진행한다. 컴파일이 완료되면 프로그램을 실행한다.

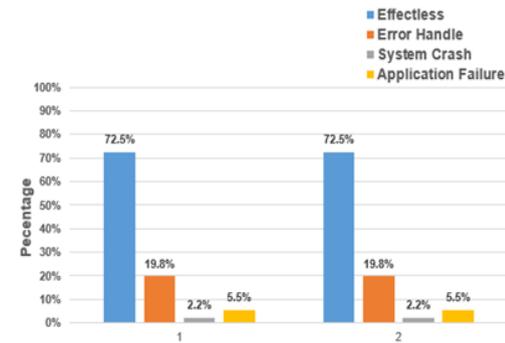
프로그램 실행 중 결함 주입 시점에 도달되면 결함이 주입된다. 결함이 주입되면 주입 결과를 감시한다. 모니터링 단계에서는 결함 주입으로 인해 실행에 어떤 문제가 있는지 지상국 소프트웨어의 콘솔 창을 통해 직접 확인한다. 이후 고장 주입으로 인해 태스크의 주요 기능 실행에 문제가 있는지 확인한다.

Table 2. Number of soft error injection tests

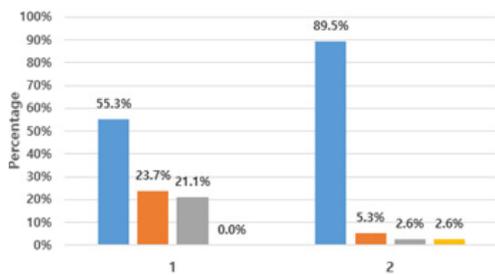
Injection Type	Memory		Register	Tests
	Global data	Local data	Number of target function	
Data Error	91	38	-	258
Control Error	-	-	12	1921
Total	91	38	12	450

4. 결함주입 시험 결과

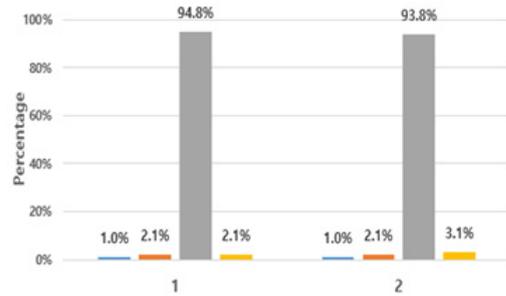
이 장에서는 NASA cFS에 소프트 에러 결함을 주입하여 소프트웨어가 받는 영향을 분석한 결과를 설명한다. 결함주입 시험은 리눅스 운영체제에서 앞에서 설명한 것 같이 cFS의 HS와 TO 태스크에 대해 Table 2와 같이 결함 주입을 한 후에, NOS3 시뮬레이터를 실행하여 동작을 확인하는 방법으로 진행되었다. 결함 주입은 3장에서 설명한 결함 주입 방법을 적용하였다. Table 2를 보면 고장을 주입한 전역 변수와 지역 변수의 수는 129개이고, n은 2이므로 변수들에 258번의 결함 주입 테스트가 수행되었다. 또한, EIP 레지스터 결함 주입은 12개 함수에서 n은 2이고, k는 4이므로 (레지스터의 비트 길이는 32) 레지스터에 192번의 결함 주입 테스트가 수행되었다.



(a) Global variables



(b) Local variables



(c) EIP Register

Fig. 1. cFS responses from different injection types

소프트 에러 결함을 주입했을 때 실행 중인 프로그램에 미치는 영향은 다음과 같이 분류할 수 있다.

- 영향 없음 : 태스크의 주요 기능을 수행하는 데 문제가 없다고 판단되나, 아직 분석되지 않은 소프트 에러의 영향이 존재할 가능성이 있음
- 오류 검출 및 복구 : 주입된 결함으로 인해 프로그램에 오류가 발생하여 프로그램이 발생한 오류를 처리함
- 프로그램 종료 : 세그멘테이션 폴트(Segmentation Fault)등으로 인해 프로그램이 종료됨
- 앱 오동작 : 결함 주입의 영향으로 태스크 주요 기능 수행 불가

위의 결함 주입 시 프로그램에 미치는 영향 분류를 기반으로, cFS의 고장주입 시험 결과를 분석하였다. Fig. 1은 고장 주입 결과 cFS의 반응 비율을 시험의 유형별로 비교한 것이다. x축은 고장 주입 위치로 선정된 1과 2 위치를 의미한다.

Fig. 1(a)는 전역변수에 대한 결함 주입결과를 보여준다. 전역변수에 결함 주입 시, 2개의 주입 위치에 대해 반응 비율은 동일하였다. 전역변수에 대한 소프트 에러 주입에 대해 72.5%의 비율로 영향 없음(Effectless)을 보였다. 전역 변수에 결함을 주입해 데이터 오류가 발생한 경우 애플리케이션 실패가 발생한 경우가 있었고, HS 태스크의 주요 기능인 애플리케이션, 이벤트 모니터링, CPU 독점 모니터링 기능이 수행 불가능했다.

Fig. 1(b)는 스택 영역의 지역 변수에 결함을 주입해 데이터 오류 발생이 프로그램에 미치는 영향을 나타낸다. 지역변수에 대한 소프트 에러 주입은 주입 위치에 따라 태스크의 반응이 달랐다. 1번 주입 위치에서 고장 주입 시에 소프트웨어가 고장을 검출하는 비율(Error Handle)이

23.7%로 전역변수에서 더욱 높았지만, 2번 위치에서는 89.5%가 검출을 하지 못했다. 그 이유로는 지역변수는 함수가 리턴할 때 소멸되기 때문에 2번 위치에서 검출 비율이 낮아진 것으로 판단된다. 전역 변수에 결함을 주입해 데이터 오류가 발생한 경우 애플리케이션 실패가 발생한 경우가 있다. 지상국의 원격명령 소프트웨어로 위성의 비행 소프트웨어에 명령을 송신하였을 때, 위성 비행 소프트웨어가 명령처리에 실패하는 경우가 발생하였다.

Fig. 1(c)는 EIP 레지스터에 결함을 주입해 제어 흐름 오류 발생이 프로그램에 미치는 영향을 보여준다. EIP 레지스터에 결함 주입 시, 주입 위치별 영향은 거의 비슷한 비율을 보였다. 프로그램 카운터 레지스터에 결함을 주입하는 경우 대부분 프로그램종료(System Crash)가 발생한 점이 데이터 오류와 큰 대비를 보인다. 그 이유는 프로그램이 비정상적인 메모리 위치로 이동할 때, 하드웨어에 의한 예외처리 기능이 작동하기 때문이다. 프로그램 카운터 레지스터에 결함을 주입해 제어 흐름 오류가 발생한 경우 애플리케이션 실패가 발생한 경우가 있었고, 지상국으로부터 수신한 원격명령을 처리하는 데 실패하거나 수신한 명령이 아닌 다른 명령을 실행하는 경우가 발생하였다.

5. 결론

인공위성의 탑재 컴퓨터는 우주 방사선에 의한 소프트 에러 발생의 위험에 노출되어 있다. 본 논문에서는 소프트 에러를 탑재 소프트웨어에 주입하여 소프트 에러가 발생했을 때, 어떤 동상을 보이는지를 분석하였다. 인공 위성 탑재 컴퓨터에 사용되고 있는 NASA의 cFS 탑재 소프트웨어를 대상으로 사용하여, 소프트 에러 발생 시에 탑재 소프트웨어가 오류를 검출할 수 있는지, 복구 불가능한 오류가 발생하였는지 등을 분석하였다. 메모리의 전역 변수 영역, 스택 영역과 프로그램 카운터 레지스터에 결함을 주입하여 소프트웨어가 받는 영향을 분석하였다.

결함 주입시험 결과, 메모리와 프로그램 카운터 레지스터에 소프트 에러 발생 시 데이터 오류와 제어 흐름 오류가 발생해 애플리케이션 동작에 문제가 생겨 위성의 임무 수행에 영향을 받을 수 있다는 것을 확인하였다. 소프트 에러 결함의 영향을 분석한 결과, 소프트 에러 발생 시 1) HS 태스크의 주요 기능인 cFS App 모니터링 기능 수행 불가, 2) 지상국으로부터 수신한 원격명령 처리 실패, 3) 수신한 원격명령과 다른 명령의 실행, 4) 비정상적

제어 흐름으로 인한 프로그램 종료 등의 오류가 발생하는 것을 확인할 수 있었다. 본 연구에서 확인한 소프트 에러에 의한 탑재 소프트웨어의 영향 분석 결과는 향후 인공위성 탑재 소프트웨어의 국산화 과정에서도 적용이 가능할 것으로 기대된다. 향후에는 소프트 에러를 자동으로 주입해 주는 고장 주입 시험도구를 개발하여 테스트 자동화를 진행할 계획이다.

References

- [1] Jungkyu, Y., Hyungun Yoo, "SEU characteristic analysis and mitigation method for LEO satellite payload", IEIE Fall Workshop, pp. 692-695, 2020
- [2] Seyyed Amir Asghari, Atena Abdi, Hassan Taheri, Hossein Pedram, Saadat Pourmozaffari, "SEDSR: Soft Error Detection Using Software Redundancy," Journal of Software Engineering and Applications, 5, pp. 664-670, 2012.
DOI: <https://doi.org/10.4236/jsea.2012.59078>
- [3] MCCOMAS, David. NASA/GSFC's Flight Software Core Flight System. In: Flight Software Workshop Flight Workshop. 2012.
- [4] Holzmann, Gerard J. "Mars code." Communications of the ACM 57.2, pp. 64-73. 2014.
DOI: <https://doi.org/10.1145/2560217.2560218>
- [5] ALONSO, Javier, et al. "The nature of the times to flight software failure during space missions", In: 2012 IEEE 23rd International Symposium on Software Reliability Engineering. IEEE, 2012. p. 331-340.
DOI: <https://doi.org/10.1109/ISSRE.2012.32>
- [6] GROCE, Alex, et al. "Establishing flight software reliability: Testing, model checking, constraint-solving, monitoring and learning", Annals of Mathematics and Artificial Intelligence, 2014, 70.4: 315-349.
DOI: <https://doi.org/10.1007/s10472-014-9408-8>
- [7] GRUBB, Matthew, et al. "NASA operational simulator for small satellites (nos3): Tools for software-based validation and verification of small satellites", 2016.
- [8] Choi, Won-sub, Jin-Hyoung Kim, and Hae-dong Kim. "A study on developing flight software for nano-satellite based on NASA CFS," Journal of The Korean Society for Aeronautical & Space Sciences, vol.44, issue.11, pp.997-1005. 2016,
DOI: <https://doi.org/10.5139/JKSAS.2016.44.11.997>
- [9] MORRIS, Justin, et al. "Simulation-to-Flight 1 (STF-1): A mission to enable Cubesat software-based verification and validation", In: 54th AIAA Aerospace Sciences Meeting. 2016. p. 1464.
DOI: <https://doi.org/10.2514/6.2016-1464>
- [10] N. Ignat, B. Nicolescu, Y. Savaria and G. Nicolescu, "Soft-error classification and impact analysis on real-time

operating systems," Proceedings of the Design Automation & Test in Europe Conference, pp. 1-6, 2006.
DOI: <https://doi.org/10.1109/DATE.2006.244063>

[11] Li, Dong, Jeffrey S. Vetter, and Weikuan Yu. "Classifying soft error vulnerabilities in extreme-scale scientific applications using a binary instrumentation tool." SC'12: Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis. IEEE, 2012.
DOI: <https://doi.org/doi: 10.1109/SC.2012.29>

[12] F. Rosa, F. Kastensmidt, R. Reis and L. Ost, "A fast and scalable fault injection framework to evaluate multi/many-core soft error reliability," 2015 IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFTS), pp. 211-214, 2015.
DOI: <https://doi.org/10.1109/DFT.2015.7315164>

[13] R. Amarnath, S. N. Bhat, P. Munk and E. Thaden, "A Fault Injection Approach to Evaluate Soft-Error Dependability of System Calls," 2018 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW), pp. 71-76, 2018.
DOI: <https://doi.org/10.1109/ISSREW.2018.00-28>

배 지 훈(Ji-Hoon Bae)

[정회원]



- 2020년 8월 : 충남대학교 컴퓨터 공학과 (학사)
- 2022년 2월 : 충남대학교 컴퓨터 공학과 (석사)
- 2021년 12월 ~ 현재 : (주)만도 연구원

<관심분야>

SW신뢰성, 기능안전, 차량 보안

김 형 신(Hyungshin Kim)

[정회원]



- 1990년 12월 : Univ. of Surrey, 위성통신공학과 (공학 석사)
- 2003년 2월 : 한국과학기술원 전산학과 (전산학 박사)
- 2004년 2월 ~ 현재 : 충남대학교 컴퓨터공학과 교수

<관심분야>

임베디드 시스템, 모바일 시스템, 에지 컴퓨팅