

논문 2008-45TC-10-2

과학기술위성 1호 탑재 컴퓨터(On-board Computer)에서의 SEUs(Single Event Upsets) 극복 알고리즘

(Algorithm to cope with SEUs(Single Event Upsets) on STSAT-1
OBC(On-board Computer))

정 성 인*, 박 흥 영*, 이 흥 호**

(Sung In Chung, Hong Young Park, and Heung Ho Lee)

요 약

보통 저궤도를 선회하는 위성은 자기장으로 연결된 반알렌대를 통과하며, 이 안에 간혀 주기적인 운동으로 남극과 북극을 이동하는 하전입자에 의해 부품이 손상되고 수명이 단축되는 악 영향을 받고 있다. 그중 방사선에 의한 SEU (Single Event Upset) 등은 우주선에 탑재된 반도체 소자의 오동작 유발의 원인이 되고 있다. 본 연구에서는 우주환경 방사선에서 고려해야 할 점들 중에서 특히 과학기술위성 1호 탑재 컴퓨터(On-board Computer, OBC)에서의 싱글이벤트업셋(Single Event Upset, SEU)의 영향을 고찰해 보고 거기에서 극복할 수 있는 알고리즘을 제시하고 있다. SEU 누적을 방지하기 위하여 매 일정한 시간마다 전체 메모리를 읽고/쓰는 과정(memory wash)이 필요하며 워쉬 주기 설정에 대해서도 고찰했다. 이러한 실험은 과학기술위성 시리즈 및 저궤도 위성용 탑재 컴퓨터의 성능 저하를 이해하는데 도움을 줄 수 있을 뿐만 아니라, 다목적 실용위성 시리즈의 각 모듈 개발에도 적극 활용 할 수 있을 것으로 기대된다.

Abstract

Generally, the satellite circling round in a low orbit goes through Van Allen belt connecting with the magnetic field, in which electronic components are easily damaged and shortened by charged particles moving in a cycle between the South Pole and the North Pole. In particular, Single Event Upset(SEU) by radiation could cause electronic device on satellite to malfunction. Based on the idea mentioned above, this study considers about SEU effect on the On-board Computer(OBC) of STSAT-1 in the space environment radiation, and shows algorithm to cope with SEUs. In this experiment, it also is shown that the repetitive memory read/write operation called memory wash is needed to prevent the accumulation of SEUs and the choice for the period of memory wash is examined. In conclusion, it is expected that this research not only contributes to understand low capacity of On-board Computer(OBC) on Low Earth Orbit satellite(LEOS) and SaTReC Technology satellite(STSAT) series, but also makes good use of each module development of Korea Multi-Purpose Satellite(COMPSAT) series.

Keywords : Single Event Upset(SEU), On-board Computer(OBC), STSAT-1, Memory wash

I. 서 론

위성에 사용되는 탑재 컴퓨터(On-board Computer, OBC)는 위성의 원격 명령, 원격 검침 정보 처리 등 다양한 기능을 수행하는 위성의 두뇌에 해당하는 중요한

시스템이다. 특히 과학기술위성과 같은 소형위성의 경우 탑재 컴퓨터는 명령 및 데이터 처리 뿐 아니라 자세 제어, 전력계통의 제어에 이르기까지 다양한 기능을 수행하고 있다. 따라서 위성의 탑재 컴퓨터는 고 성능의 계산 능력과 고도의 신뢰성을 요구한다. 또한 위성은 우주라는 열악한 환경에 놓여 있기 때문에 탑재 컴퓨터는 이것을 고려하여 우주 환경에서도 동작 할 수 있도록 설계되어야 한다.

보통 저궤도를 선회하는 위성은 자기장으로 연결된

* 정희원, 한국과학기술원 인공위성연구센터
(Satellite Technology Research Center, KAIST)

** 정희원, 충남대학교

(Chung Nam National University)

접수일자: 2008년7월24일, 수정완료일: 2008년10월14일

반알렌대를 통과하며, 이 안에 갇혀 주기적인 운동으로 남극과 북극을 이동하는 하전입자에 의해 부품이 손상되고 수명이 단축되는 악 영향을 받고 있다. 그중 방사선에 의한 SEU (Single Event Upset) 등은 우주선에 탑재된 반도체 소자의 오동작 유발의 원인이 되고 있으며, 또한 항공기 제어용 반도체의 경우도 우주 방사선에 의한 오동작 유발로 국제적으로 항공우주 소자의 내방사선 규격이 점점 강화되고 있다.^[1]

두 개의 띠 모양을 하고 있는 반알렌대 중 특히 마이크로프로세서(micro-processor)와 메모리 소자 등의 전자 부품에 SEU(Single Event Upset)와 같은 SEEs (Single Event Effects)를 일으킬 수 있다. 특히 과학기술위성 1호, UoSAT 등과 같은 저궤도 위성에는 이러한 고에너지 양성자가 중요한 영향을 미친다. 따라서 위성 설계에 있어서는 SEU 현상을 극복하기 위한 여러 방법들을 고려하여야 한다.

본 연구에서는 우주환경 방사선에서 고려해야 할 점들 중에서 특히 과학기술위성 1호 탑재 컴퓨터(On-board Computer, OBC)에서의 싱글이벤트업셋(Single Event Upset, SEU)의 영향을 고찰해 보고 거기에서 극복할 수 있는 알고리즘을 제시하고자 한다. SEU 누적을 방지하기 위하여 매 일정한 시간마다 전체 메모리를 읽고/쓰는 과정(memory wash)이 필요하며 위와 주기 설정에 대해서도 고찰했다. 이러한 실험은 과학기술위성 시리즈 및 저궤도 위성용 탑재 컴퓨터의 성능 저하를 이해하는데 도움을 줄 수 있을 뿐만 아니라, 다목적 실용위성 시리즈의 각 모듈 개발에도 적극 활용 할 수 있을 것으로 기대된다.

II. 저궤도 인공위성의 동작 환경

1. 우주 환경요소들에 의해 전자 장비들에 끼치는 영향

위성 탑재용 전자 장비는 우주 환경과 궤도 진입에 요구되는 발사 환경에 만족하여야 한다. 전자장비의 요구항목은 고 진공 상태의 공간에 주변 입자의 밀도가 적은 관계로 태양에서 나오는 복사열에 의해 빛이 비춰는 부분과 그늘진 부분의 온도 차이가 아주 크다. 또한 지상처럼 대기의 대류 현상에 의한 열전달이 이루어지지 않기 때문에 전자 장비 중 열이 많이 나는 소자는 전도에 의하여 열전달이 이루어 질 수 있도록 설계하여야 한다. 따라서 사용될 전자 소자의 동작 온도는 지상에서 보편적으로 사용되는 상용 소자의 동작 범위인

0°C에서 70°C보다 큰 -55°C에서 125°C까지의 범위를 갖는 소자들을 택하여 전자 장비의 동작 신뢰도를 높이도록 한다.

우주 환경 중 전자 장비가 고려해야 될 또 하나의 요인은 우주 방사선이다. 지구 자기권에 의해 지구 주변에 고립된 입자들은 그 분포 밀도가 크지는 않으나, 전자소자의 수명을 결정하는 중요한 요인이다. 위성의 수명 기간 동안 우주 방사선이 전자 소자들에 계속해서 누적되는데, 이 누적되는 총 방사선량(Total Ionizing Dose, TID)에 대하여 견딜 수 있도록 설계된 전자 소자들을 사용하여야 한다. 그러나 이러한 내방사선 소자들에 대한 기술은 모든 전자 소자들에게 적용할 수 있는 단순한 문제가 아니라, 사용되는 물질과 소자 내부의 구조에 따라 그 특성이 매우 다르게 나타난다. 특히 최근의 반도체 기술의 개발 동향은 새로운 기술들이 매우 빠르게 개발되고 있기 때문에 새로운 기술이 적용된 소자들에 대하여 내 방사선 설계를 적용하기가 어려운 상황이다. 이러한 경우 부분적으로 전자 장비 주변에 국부적으로 보호벽을 둘으로써 방사선의 투과량을 줄이는 방법을 쓰기도 하지만, 방사선을 차폐하기 위한 물질들은 위성의 무게를 증가시키는 요인이 되므로 한없이 두껍게 하기는 못한다.

우주 방사선에 대한 고려 중 다른 중요한 한 가지는 고에너지 입자들에 의한 순간 이상 동작 현상들이다. 싱글이벤트영향(Single Event Effects)이라고 불리어지는 이 현상은 저궤도 위성의 경우 남대서양 상공에 낮게 분포하고 있는 에너지를 띤 양성자 입자에 의해 집중적으로 발생하며(South Atlantic Anomaly, SAA), 우주에서 날아오는 고에너지 입자들에 의해서 정지 궤도 위성 등 우주에서 동작하는 모든 전자 소자들에 나타나는 현상이다. 이중에서 대표적인 현상 중의 하나인 싱글 이벤트 업셋(Single Event Upset, SEU)은 전자 장비의 소자에 기록된 정보를 '1'에서 '0'으로 '0'에서 '1'으로 바뀌는 현상으로 위성에 탑재된 컴퓨터의 프로그램 메모리에 이러한 현상이 나타나면, 위성이 오동작하게 되는 치명적인 현상이다. 따라서 위성에 탑재되는 모든 장비는 이러한 현상에 대비하여 보호 회로들을 사용한다.

2. 반 알렌 방사선 밸트

(The Van Allen radiation belts)

지구를 도우넛 모양으로 감싸고 있으며 반 알렌 방사선 밸트는 에너지를 가진 양성자(Proton)와 전자

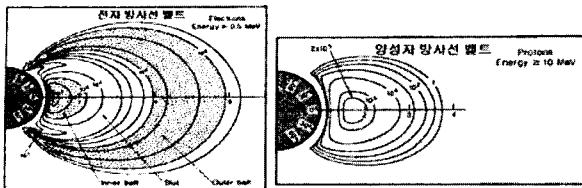
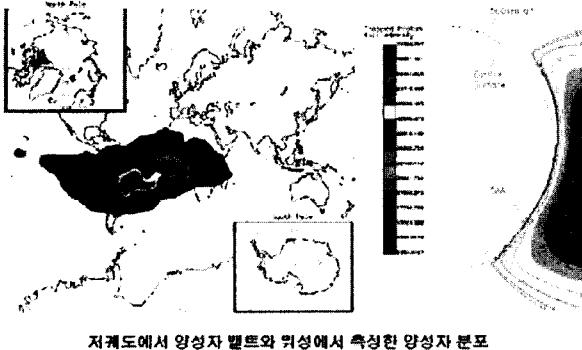


그림 1. 전자(왼쪽) 및 양성자(오른쪽) 방사선 벨트
Fig. 1. Electron(left) and proton(right) radiation belt.



저궤도에서 양성자 벨트와 위성에서 측정한 양성자 분포

그림 2. 저궤도에서 양성자 벨트와 위성에서 측정한 양성자 분포
Fig. 2. Proton belt at LEO and measured proton distribution at a satellite.

(Electron)가 지구 자기장에 의해 갇혀서 형성되었다.

그림1은 전자와 양성자 벨트의 모양을 고도에 따라 에너지 분포로 나타낸 것이다. 그림에서 알 수 있듯이 정지 궤도 위성이 위치하고 있는 고도 36,000km 상공에는 강한 전자 방사선 대역이 있음을 알 수 있으며, 주로 태양 동기 궤도를 갖고 있는 저궤도 위성은 양성자 방사선의 영향 아래 있다.

그림 2는 저궤도에서 양성자 벨트와 위성에서 측정한 양성자 분포를 나타낸 것이다. 그림에서 보듯이 양성자 방사선 벨트가 지구 자기장의 영향에 의해 남 대서양 지역에 낮게 분포하고 있는 것을 보여주고 있다. 이러한 현상은 저궤도 인공위성이 이 지역을 지날 때 찾은 고장을 일으키는 요인으로서 남대서양 이상지역 (South Atlantic Anomaly, SAA)으로 불려지고 있다. 이 지역은 입자에 의하여 위성의 컴퓨터, 메모리 등이 오동작을 일으키는 지역이다.

III. SEU 영향 및 메커니즘

1. SEU 영향

위성이 동작하는 우주 궤도에는 우주 방사선들이 반도체 소자에 영향을 끼치는데 이러한 방사선 중에는 고에너지 입자들이 존재한다. 고 에너지 입자의 밀도는

표 1. 싱글 이벤트 영향

Table 1. Single Event Effects.

Acronym	Definition
SEU	Single Event Upset
SED	Single Event Disturb
SET	Single Event Transient
SEDR	Single Event Dielectric Rupture
SEGR	Single Event Gate Rupture
SEL	Single Event Latchup
SES	Single Event Snapback
MBU	Multiple Bit Upset
SEFI	Single Event Functional Interrupt

높지 않지만 하나의 고 에너지 입자가 반도체 소자에 부딪치게 될 경우 반도체 소자에는 다음과 같은 현상을 일으키게 된다.

SEU는 저장된 정보가 바뀌는 현상이며, SED는 저장된 정보가 순간적인 왜란에 의해서 잠시 바뀌는 현상이며, SET는 조합회로에서 입력이 변하지 않았음에도 불구하고 출력 값이 순간적으로 변하는 현상이며, SEDR은 아예 끊어져 버린 현상이며, SEGR은 고 전류에 의해서 게이트가 망가진 현상이며, SEL은 래치업(Latch up)에 의해서 고전류가 흐르는 현상이며, MBU는 여러 개의 비트의 정보가 바뀌는 현상이며, SEFI는 업셋(Upset)에 의해서 제어 구조가 바뀌어 기능이 방해 받는 현상이다.

2. SEU 메커니즘

위성체가 우주환경에 직면하게 되면 먼 우주로부터 오는 우주선(cosmic ray)에 의한 고에너지 입자, 태양 활동에 의해 분출되는 입자, 또는 지구 자기장에 붙잡혀 있는 여러 에너지 입자들에 의해 위성체의 전자회로가 영향을 받게 된다. 이중에서 메모리에 발생하는 중요한 영향은 이를 고에너지 입자에 의해 메모리에 저장되어 있는 정보가 “1”→“0” 또는 “0”→“1”로 변경되는 SEU(Single Event Upset) 현상이다.

대부분의 메모리 소자는 그림 3과 같이 MOS(Metal Oxide Semiconductor) 트랜지스터를 이용하여 NOT 게이트(gate) 2개를 서로 연결시킨 기본 구조에 바탕을 두고 있다.

이 회로는 2개의 안정된 상태에, 즉 “0” 또는 “1”, 항상 머물러 있으므로 1비트의 2진 정보를 저장할 수 있다. 각각의 상태에서 그림 3의 4개의 MOS 트랜지스터 중 2개는 턴-온(Turn-ON), 2개는 턴-오프(Turn-OFF) 상태에 있게 된다. 이들 MOS 트랜지스터에 그림 4에서

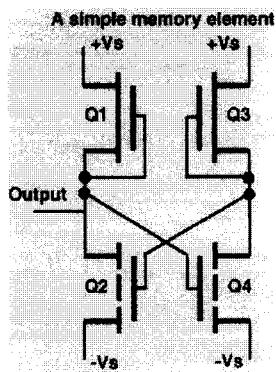


그림 3. 메모리 기본 구조
Fig. 3. Basic memory structure.

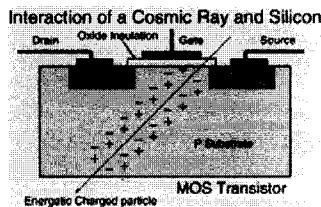


그림 4. 고에너지 입자에 의한 SEU 발생 메카니즘
Fig. 4. SEU mechanism by high energy particle.

와 같이 고에너지 입자가 입사 되면 실리콘 반도체의 P-Substrate에는 이 입자가 지나가는 패스를 따라 반도체 내부에 있던 원자가 이온화 되어 전자(electron) 정공(hole) 쌍이 생성되고, 이때 발생된 전자, 정공 쌍의 일부는 일정 시간 후 재결합하여 없어지지만 일부는 트랜지스터의 소스(source)와 드레인(drain) 쪽으로 이동하여 소스에서 드레인 쪽으로 전류 펄스를 야기 시킨다. 이 전류 펄스의 크기가 일정 이상이 되면 오프되어 있던 그림 3의 트랜지스터를 터-온 시켜 이 회로의 상태가 다른 상태로 천이하게 된다("1"→"0" 또는 "0"→"1"). 이와 같이 회로의 상태가 천이함으로 인해 원래 저장되어 있던 1비트 정보가 바뀌는 것이 SEU 현상이다.

IV. SEU의 극복 알고리즘

1. SEU 극복

과학기술위성 1호의 경우 프로그램 메모리에서 발생하는 SEU를 극복하기 위하여 해밍코드(Hamming code)에 기반을 둔 에러 복구 알고리즘을 채택하고 있다. 이 알고리즘은 7비트 중 4비트의 정보 비트와 3비트의 여분 비트(Redundancy bit)를 가지며, 1비트 에러를 복구하고 2비트 에러는 탐지할 수 있다. 따라서 매 4비트의 데이터마다 1비트 에러를 복구할 수 있는 능력

을 가지고 있다. 하지만 하나의 SEU 발생 후 또 다른 SEU가 발생하면 2비트의 오류를 발생시켜 에러 복구 알고리즘으로는 복구할 수 없는 상황이 발생한다.

에러의 복구는 메모리를 읽고/쓰는 과정에서 처리되고 있으므로 SEU 누적을 방지하기 위해 매 일정한 시간마다 전체 메모리를 읽고/쓰는 과정(memory wash)을 필요로 한다. 메모리 위쉬 주기는 SEU 발생율 및 메모리 위쉬에 따른 CPU와 오버헤드(overhead)를 고려하여 적절히 설정하여야 한다. 워쉬 주기를 길게 하면 메모리의 SEU의 누적이 증가하는 대신 CPU의 오버헤드는 감소하고, 주기를 짧게 하면 SEU의 누적은 감소하지만 CPU의 오버헤드는 증가한다. 이러한 이徼 알고리즘은 다음의 수식으로 4비트의 정보 비트를 이용하여 3비트의 여분 비트를 만든다.

$$R(2) = I(3) + I(2) + I(1)$$

$$R(1) = I(2) + I(1) + I(0)$$

$$R(0) = I(3) + I(2) + I(0)$$

에러의 복구는 먼저 정보 비트와 여분 비트를 이용하여 3비트의 신드로미(Syndrome)을 구한 다음 테이블을 이용하여 에러가 발생한 비트를 반전 시킨다. 또한 에러 복구는 여분 비트들에서 발생한 에러에 대해서는 하지 않고, 정보 비트들에서 에러가 발생하였을 때만 하여 EDAC 알고리즘을 FPGA로 구현했을 때 사용되는 로직 수를 최대한 줄일 수 있도록 하였다.

$$S(2) = (I(3) + I(2) + I(1)) + R(2)$$

$$S(2) = (I(2) + I(1) + I(0)) + R(1)$$

$$S(2) = (I(3) + I(2) + I(0)) + R(0)$$

프로그램 메모리는 데이터를 저장하기 위한 데이터 메모리와 여분 비트들을 저장하기 위한 EDAC 메모리로 구성되어 있으며 CPU에서 프로그램 메모리에 데이

표 2. (7,4) 헤밍코드를 이용한 에러 복구
Table 2. Error recovery using (7,4) Hamming code.

Syndrom(2:1:0)	ErrorVector I (3:2:1:0)	R(2:1:0)	Digit In Error
000	0000	000	NONE
001	0000	001	7
010	0000	010	6
011	0001	000	4
100	0000	100	5
101	1000	000	1
110	0010	000	3
111	0100	000	2

터를 기록할 때 EDAC용 FPGA에서 여분 비트들을 만들어 데이터는 데이터 메모리에 여분 비트들은 EDAC 메모리의 똑 같은 번지(Address)에 저장한다.

CPU가 프로그램 메모리로부터 데이터를 읽어 올 때 EDAC FPGA에서 데이터 메모리와 EDAC 메모리에서 데이터와 여분 비트들을 각각 읽어와 위에서 설명한 알고리즘에 따라 에러가 있는지 확인하고 에러가 있으면 복구하여 CPU로 전송한다.

메모리에 저장된 데이터의 에러 복구는 항상 CPU가 데이터를 읽어올 때 이루어지고 또한 메모리의 오류는 데이터 4비트 당 1비트만 복구가 가능하기 때문에, 메모리 전체의 데이터가 항상 오류가 없도록 하기 위해서는 정기적으로 전 메모리를 읽어 오류를 복구하고 다시 저장해야 할 필요가 있다(Memory Scrubbing). 그렇지 않으면 하나 이상의 SEU에 의해 데이터 4비트당 2비트 이상의 오류가 발생할 수 있어 오류를 복구할 수 없는 상황이 발생 할 수 있기 때문이다. 따라서 탑재 컴퓨터의 시스템 소프트웨어는 매 1초마다 전 메모리에 걸쳐 1 킬로바이트 씩 프로그램 메모리를 읽고 다시 쓰는 작업을 하도록 구현되어 있다.

2. EDAC 알고리즘

대부분의 상용 위성들은 SEU를 방지하기 위해 SOS(Silicon On Sapphire)기법 등을 이용한 Radiation Hardened 또는 Radiation Tolerant 급의 메모리 소자를 사용한다. 하지만 이러한 메모리를 사용하면 SEU를 소자 차원에서 극복할 수 있으나 너무 고가라서 과학기술 위성 시리즈와 같은 저비용 소형 위성에서는 사용하기에 무리가 있다. 과학기술위성 시리즈에서는 훨씬 저렴한 밀리터리 급(Military grade)의 메모리 소자에 EDAC(Error Detection and Correction) 로직을 추가하여 SEU를 극복함으로써 비용을 줄이도록 설계되었다.

TMR을 이용한 EDAC 알고리즘은 그 특성상 3비트마다 1비트의 SEU를 극복할 수 있지만, 2비트 이상의 SEU 복구 할 수 없으므로, SEU에 의해 메모리에 저장된 1비트의 값이 변경된 후 EDAC 로직에 의해 복구되기 전에 또다시 SEU에 의해 이들 3비트 중 어느 1비트가 변경되면 전체적으로 2비트의 오류가 발생하여 EDAC 알고리즘으로는 오류를 복구할 수 없는 경우가 발생한다. 즉 메모리에서 발생한 SEU를 적당한 시간 내에 복구하지 않으면 오류가 누적되어 복구할 수 없는 경우가 발생할 수 있다. 이와 같이 EDAC 알고리즘의 단점 때문에 탑재 컴퓨터의 시스템 소프트웨어는 메모

리 전체 데이터가 항상 오류가 없도록 하기 위하여 정기적으로 메모리를 읽어 오류를 복구하고 다시 저장하는 작업(memory wash)을 수행하도록 하여야 한다.

따라서 전체 메모리를 워쉬(wash)하는 주기를 어떻게 설정하느냐에 따라 오류 복구 가능성이 달라진다. 즉 주기를 길게 하면 2비트 이상의 SEU가 발생하여 오류를 복구할 수 없는 가능성이 커지고, 주기를 짧게 하면 오류를 복구할 수 없는 가능성은 줄어들지만 시스템 소프트웨어의 부담이 커져(메모리를 워쉬하는데 일정 시간을 소모하므로) 탑재컴퓨터의 다른 소프트웨어가 수행될 시간이 줄어드는 단점이 있다. 그러므로 메모리 워쉬 주기는 이들을 고려하여 적절하게 설정되어야 한다.

3. 메모리 워쉬 주기 선정

탑재 컴퓨터의 시스템 소프트웨어가 t 시간 주기마다 전체 메모리를 워쉬하는 것은 시간적인 오버헤드(overhead)를 요구하며 이것은 다른 소프트웨어가 수행될 시간을 빼앗기 때문에 시스템의 성능을 떨어뜨린다. 이 오버헤드는 다음의 식으로 나타낼 수 있다.

$$\Theta(t) = \frac{\sigma}{t} \quad (\sigma \leq t) \quad (1)$$

여기서 σ 는 전체 메모리를 워쉬하는데 걸리는 시간이며, t 는 워쉬 주기이다.

오버헤드는 워쉬 주기가 길수록 적어지고, 짧을수록 커져 주기가 σ 가 됐을 때 최대가 된다. 워쉬 주기가 σ 보다 같거나 작게 되면 CPU의 모든 시간을 메모리 워쉬에만 사용하게 되므로 다른 소프트웨어를 수행할 시간이 없어지게 된다.

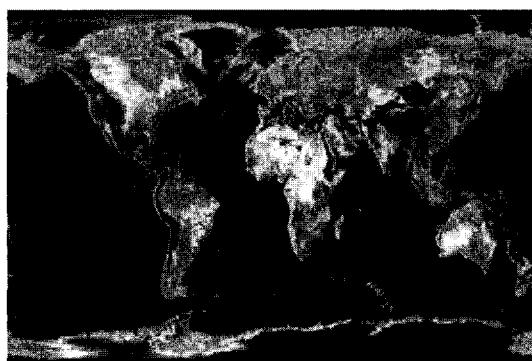


그림 5. 과학기술위성 1호의 SEU 발생 장소(2004년 5월 20일까지 3달간)
Fig. 5. SEU occurrence place of STSAT-1.(May/2004 for 3 months)

궤도상에서 SEU가 발생하는 궤도상의 위치와 발생빈도사이의 관계를 살펴 보면, 그림 5에서 보듯이 전궤도상에서 고르게 분포하는 것이 아니라 SAA(South Atlantic Anomaly) 부근과 극 지방에서 집중되는 것을 알 수 있다. 특별히 과학기술위성 1호의 위상 주기는 30분 정도이며 그림 5에서 나타난 점들은 30분이라는 시간 차가 있을 수 있다. 이것을 감안하면 SAA 지방에 집중적으로 나타나는 것을 알 수 있다. 따라서 위성이 SAA와 극지방을 지나는 시간이 전체 궤도상의 시간의 대략 1/100이라고 가정하고 여기서 대부분의 SEU가 발생한다고 하면 위성이 궤도상에서 경험하는 최대 SEU는 다음과 같다.

$$4.06901 \times 10^7 / 100 = 4.06901 \times 10^5 \text{ SEUs/bit min}$$

메모리 각 셀에서의 발생하는 SEU 발생율은 대략적으로 추정할 수 있다.

$$4.06901 \times 10^5 \text{ SEUs/bit min}$$

위에서 추정한 SEU 발생율은 상한 값을 상당히 보수적으로 계산한 값이므로 근사적으로 적용할 수 있을 것으로 판단된다.

반면 위성 궤도의 차이, 사용된 메모리의 종류, 위성 내에서의 탑재 컴퓨터의 위치, 위성 외부 프레임의 두께 등의 조건이 과학기술위성 1호의 경우 운용 결과 하루 평균 3.745763번의 SEU가 발생하며, 한 비트당 분당 발생률은

$$3.745763 / 4 \times 10^6 / 24 / 60 = 6.050306 \times 10^{-10} \text{ SEUs/bit min}$$

다음으로 메모리 위상이 CPU에 주는 부담을 계산하기 위하여 탑재 컴퓨터의 CPU가 메모리를 엑세스하는데 필요한 클럭 사이클을 계산한다. 탑재 컴퓨터의

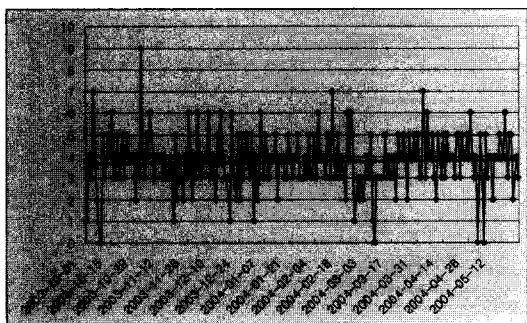


그림 6. 과학기술위성 1호에서의 일별 SEU 발생 횟수 (평균 3.745763)

Fig. 6. SEU occurrence frequency of STSAT-1.(average 3.745763)

CPU는 PowerPC603로 32비트 데이터 버스를 가지고 있다. 그리고 PowerPC603이 주 메모리에 읽기/쓰기를 하는데에는 각각 6클럭 사이클을 요구하며, 메모리를 워싱하는 것은 메모리의 특정 부분을 읽고 쓰는 과정을 요구하므로 4바이트(32비트)의 메모리를 엑세스하는데는 최소 $6 \times 2 = 12$ 개의 클럭 사이클을 필요로 한다. 뿐만 아니라 전체 메모리를 워싱하기 위해서는 매 4바이트마다 어드레스를 증가시키는 과정과 주메모리를 읽는 Instruction Fetch 등의 과정이 필요하다. 따라서 4바이트의 메모리를 워싱하기 위한 클럭 사이클은 충분한 여분의 시간까지 고려하면 다음과 같이 나타낼 수 있다.

$$\begin{aligned} 4\text{bytes wash} &= \text{read/write} + (2*\text{instruction fetch}) + \\ &\quad \text{address increasing} + \text{etc} \\ &= 12 \text{ cycles} + (2*\text{instruction fetch}) + \\ &\quad \text{address increasing} + \text{etc} \\ &= 40 \text{ clock cycles} \end{aligned}$$

프로그램 메모리 크기가 1메가 바이트(M bytes)이므로 전체 메모리를 워싱하는데는 다음의 클럭 사이클을 필요로 한다.

$$40 \times 10^6 / 4 \text{ cycles} = 10^7 \text{ cycles}$$

탑재 컴퓨터의 주 클럭 주파수는 25 MHz이고 이것의 5배인 150 MHz가 실제 CPU의 클럭으로 사용된다. 그러므로 매 분당 오버헤드의 상한 값을 다음과 같이 보수적으로 추정할 수 있다.

$$\sigma = 10^7 / 25 \times 10^6 / 60 = 0.006667/\text{min}$$

그림 7은 SEU 발생율 $\lambda = 6.50306 \times 10^{-8}$ SEUs/bit min와 오버헤드 파라메터 $\sigma = 0.00667$ 을 사용했을 때 워싱 주기 t의 변화에 따른 $\Gamma(\lambda, t)$ 와 $\Theta(t)$ 를 나타낸 것이다.

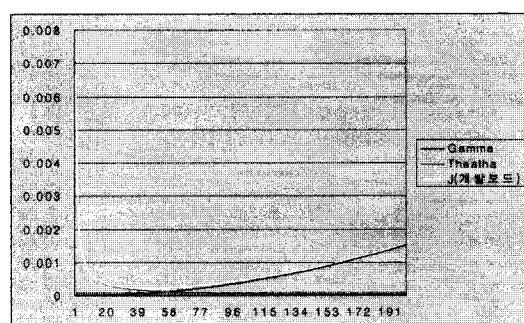


그림 7. 워싱 주기에 대한 $\Gamma(\lambda, t)$ 와 $\Theta(t)$ 의 변화

Fig. 7. Variation of $\Gamma(\lambda, t)$ and $\Theta(t)$ for wash period.

이 변화를 살펴보면 위주 주기가 짧은 왼쪽에서는 개발 보드의 성능 함수는 오버헤드 $\Theta(t)$ 의 영향을 주로 받으며, 위주 주기가 길어지는 오른쪽으로 갈수록 개발 보드 성능 함수는 메모리 복구 불가능 확률인 $\Gamma(\lambda, t)$ 에 주로 영향을 받는 것을 알 수 있다. 여기서는 위주 주기가 50분 근처에서 두 곡선이 서로 교차하므로 이 부근의 값을 주기로 선택하는 것이 $\Gamma(\lambda, t)$ 와 $\Theta(t)$ 을 동시에 적게 할 수 있다. 따라서 λ 와 σ 의 추정 오류를 고려하더라도 대략 4~25 미터(m) 사이의 값을 위주 주기로 설정하는 것이 가장 타당하다고 할 수 있다.

VI. 결 론

본 연구에서는 과학기술위성 1호 탑재 컴퓨터(On-board Computer, OBC)에서의 싱글이벤트업셋(Single Event Upset, SEU)를 극복할 수 있는 알고리즘을 연구했으며, 그 중 특히 EDAC(Error Detection and Correction) 알고리즘에 대해 주로 고찰함으로써 SEU의 영향에 대한 극복할 수 있는 알고리즘을 제안했다.

또한 과학기술위성 1호에서의 SEU 발생률을 바탕으로 개발된 보드의 위주 주기에 대한 분석도 고찰하였다. 과학기술위성 1호가 가지는 약 30분의 위주 주기는 좀 더 빠른 CPU를 사용하고 헤밍코드(Hamming code) 대신 TMR의 기능을 사용하면 값싼 메모리를 가지고 저궤도 위성과 같은 곳에서 사용할 수 있다는 것을 의미하며 테스트를 통하여 위성 개발 비용을 줄일 수 있는 결과를 가져오게 된다.

강한 입자에 의해 SEU가 발생 시 주변 메모리 소자가 같은 영향을 받을 수 있는데, 이 경우 주변의 값을 이용하는 과학기술위성에서 사용하는 방법보다 TMR이 안전하다고 할 수 있다. 또한 TMRM의 경우 판단하고 복구하는 로직 자체가 간단한 로직에 의해 구현되므로 빠른 동작이 가능하다.

참 고 문 헌

- [1] 정성인, 이재진, 이홍호, “고에너지 전자빔을 이용하여 저궤도 인공위성의 실리콘 태양센서의 내방사선 특성 연구” 대한전자공학회 제45권 SD편 제3호, 2008.
- [2] 박홍영, 오대수, 정성인 외, “FPGA를 이용한 우주용 On-board Computer의 소형화/경량화 기술 개발”, 과학기술부, 우주기술개발사업, 2004.
- [3] 곽성우, 류상문, 박홍영 외, “과학기술위성1호 탑재 컴퓨터의 설계 및 구현”, 한국항공우주학회지, vol.31 (4), pp. 105-111, 2003.
- [4] 곽성우, 박홍영, “과학기술위성 1호 탑재 컴퓨터(On-board Computer)에서의 SEUs(Single Event Upsets) 극복을 위한 메모리 운용 및 해석”, 한국항공우주학회지, 제32권 1호, 2003.
- [5] H.S. Kim, H.K. Lee, and S.D. Choi, “On-board Computer System for KITSAT-1 and KITSAT-2”, J. of The Astr. and Space Sci., VI.13(2), pp. 41-51, 1996.
- [6] H.K. Lee, and H.S. Kim, J.H. Park, K.H. Park, and S.D. Choi, “Design and Implementation of the small Satellite On-board Computer System:KASCOM”, J. of The Astr. and Space Sci., VI.13(2), pp. 52-66, 1996.

저 자 소 개



정 성 인(정회원)
2006년~2008년 충남대학교
박사과정.
1992년~2008년 한국과학기술원
인공위성연구센터
선임 연구원.
<주관심분야 : 통신, 컴퓨터, 신호
처리, 반도체>



이 흥 호(정회원)
1973년 서울대학교
전기공학과 졸업.
1979년~현재 충남대학교 교수
<주관심분야 : 전력 및 자동화>

박 홍 영(정회원)

2008년 현재 한국과학기술원 인공위성연구센터
선임 연구원.