

技術論文

과학기술위성 3호 대용량 메모리에 대한 오류복구 코드 및 SEU 시험 결과 분석

서인호*, 유광선*, 오대수*, 김병준**

Error Correction Code and SEU Test Analysis of Mass Memory for STSAT-3

In-Ho Seo*, Kwangsun Ryu*, Dae-Soo Oh* and Byung-jun Kim**

ABSTRACT

RS(10,8) Code by 4-bit symbol was developed to protect the mass memory of STSAT-3 from SEU in orbit. Therefore, one symbol can be corrected for 32-bit data with 8-bit parity configuration. Moreover, scrubbing period and SEU occurrence rate was calculated based on the KITSAT-3 result. A prediction of SEU rates was performed based on the ground experiment results with a proton accelerator in the KIRAMS(Korea Institute of Radiological Medical Sciences).

초 록

과학기술위성 3호 대용량 메모리를 SEU로부터 보호하기 위해서 4비트 심볼을 이용하는 RS(10,8) 코드를 개발 하였다. 따라서 32비트 데이터에 대해서 8비트의 패리티를 추가 하였으며 1 심볼에 대해서 에러를 복구할 수 있다. 또한 우리별 3호의 결과를 이용하여 예상되는 SEU 발생률과 스크립 주기를 계산하였다. 이 결과를 바탕으로 한국원자력 의학 원에 있는 Cyclotron 양성자 가속기를 이용하여 SEU 시험을 수행 하였다.

Key Words : RS Code, 대용량 메모리, SEU

1. 서 론

위성이 우주환경에 노출되면 그 궤도에서 운용되고 있는 위성체를 구성하는 물질과 전자부품은 지구 자기장 주변의 이온화 방사선에 의해서 여러 가지 비정상적인 현상에 직면하게 된다[1]. 특히 우주에 존재하는 고에너지 양성자는 마이크로프로세서와 메모리 소자들의 전자 부품에 SEU(Single Event Upset)를 일으킬 수 있다.

대부분의 상용 위성들은 SEU를 방지하기 위해

SOS(Silicon On Sapphire) 기법 등을 이용한 Radiation Hardened 또는 Radiation Tolerant 급의 메모리 소자를 사용한다. 그러나 이러한 메모리를 사용하면 SEU를 소자 차원에서 극복할 수 있으나 고가 이므로 과학기술위성에서는 비용을 줄이기 위해서 Military급의 메모리 소자에 EDAC(Error Detection and Correction) 로직을 추가하여 SEU를 극복 하도록 설계되어 있다. 가장 대표적인 EDAC 알고리즘 중의 하나는 Hamming 코드와 Reed-Solomon 코드를 이용한 오류 복구 방식이다[1,2].

본 논문에서는 과학기술위성 2호의 대용량 메모리에 사용된 EDAC 알고리즘을 보완하여 3호용으로 개발된 RS(10,8)[4] 코드에 대한 설계 결과를 나타내었으며 SEU 오류 극복 확률과 스크

† 2009년 10월 26일 접수 ~ 2009년 12월 28일 심사완료

* 정희원, 한국과학기술원 인공위성연구센터
교신저자, E-mail : inho@satrec.kaist.ac.kr
대전시 유성구 구성동 373-1

** 정희원, 대구경북과학기술연구원

럽 주기를 계산 하였다. 또한 한국 원자력 의학 원의 양성자 가속기에서의 시험을 통해서 임무 수행 기간 동안 발생할 SEU 발생 빈도를 예측 하였다.

II. 본 론

2.1 과학기술위성 2호와 3호의 EDAC 구조 비교

과학기술위성 2호의 대용량 메모리 유닛은 탑재체의 데이터를 수신하면 먼저 대용량 메모리에 저장하고 추후에 데이터를 수신하지 않을 때 RS(207,187)로 변환 하는 형태로 대용량 메모리를 SEU로부터 보호 하였다. 탑재체에서 데이터를 수신하는 속도와 메모리에 저장하는 속도를 고려하여 이러한 구조를 사용 하였지만 메모리에 저장된 데이터를 다시 읽어서 RS 코드로 변환해야 하는 번거로움과 RS 코드로 변환하는데 시간이 소요 된다는 단점이 있다. 따라서 과학기술위성 3호에서는 RS(10,8) 코드를 사용하여 메모리에서 데이터를 다시 읽어서 RS 코드로 변환해야 하는 번거로움을 없애고 변환 시간 또한 단축 시켰으며 지상으로 데이터를 전송하는 시점에 RS(255,223) 코드로 변환하는 형태로 구조로 변경 하였다. EDAC 알고리즘을 선택할 때는 (7,4)Hamming 코드, RS(16,8) 코드, RS(10,8) 코드를 비교하여 복구 불가능한 SEU가 존재할 확률이 비교적 낮고, 패리티 메모리가 적게 필요한 RS(10,8) 코드를 채택 하였다[3]. 표 1에 지금까지 설명한 과학기술위성 2호와 3호의 대용량 메모리 유닛의 사양을 비교 하였다. 데이터 수신 속도와 대용량 메모리가 각각 500배와 16배로 크게 증가 하였으므로 과학기술위성 2호 때와는 달리 FPGA가 직접 메모리를 관리하는 형태로 구조를 변경 하였다.

2.2 RS(10,8) Encoder 설계

심볼(Symbol)의 크기가 4비트인 RS(10,8) 코드를 사용 하였다. 따라서 32비트 데이터에 대해서

표 1. 대용량 메모리 유닛의 사양 비교

ITEM	STSAT-2	STSAT-3	비고
Processor	PowerPC 860	FPGA Processing	
FPGA	Actel RT54SX72S	Xilinx V4LX100	
Mass Memory Size	2Gb	32Gb	16배 증가
Payload Data Rx Rate	200Kbps	100Mbps (max)	500배 증가
Payload Data Downlink Rate	10Mbps	10Mbps	
Mass Memory EDAC	Formemory Scrubbing	RS(207,187)	H/WRS(10,8)
	For downlink	RS(207,187)	RS(255,223)

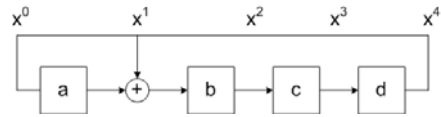


그림 1. GF(16)을 위한 원시 다항식

표 2. GF(16) 원소의 벡터 테이블

	d	c	b	a
a^0	0	0	0	1
a^1	0	0	1	0
a^2	0	1	0	0
a^3	1	0	0	0
a^4	0	0	1	1
a^5	0	1	1	0
a^6	1	1	0	0
a^7	1	0	1	1
a^8	0	1	0	1
a^9	1	0	1	0
a^{10}	0	1	1	1
a^{11}	1	1	1	0
a^{12}	1	1	1	1
a^{13}	1	1	0	1
a^{14}	1	0	0	1
a^{15}	0	0	0	0

8비트의 패리티를 추가 하였으며 1 심볼에 대해서 에러를 복구할 수 있다. 심볼의 크기가 4비트 이므로 유한 필드(Galois Field)는 $GF(2^4)$ 이며 유한 필드를 생성을 위한 원시 다항식(Primitive Polynomial)은 다음과 같이 구현하기 쉽고 회로 구성 요소가 적은 것으로 선택 하였다.

$$P(X) = 1 + X + X^4 \tag{1}$$

따라서 GF(16)을 생성하기 위한 원시 다항식은 그림 1과 같이 나타낼 수 있으며 $\alpha^0 = "0001"$ 부터 시작해서 한번 쉬프트하면 $\alpha^1 = "0010"$ 이 된다. 이것을 반복 쉬프트해서 만들어지는 값을 각각의 원소로 하는 집합이 GF(16)이며 표 2는 Encoder에서 사용하기 위한 GF(16) 원소를 벡터로 표현한 테이블이다.

또한 오류 검출 및 정정을 위한 패리티 심벌이 2개 이므로 생성 다항식(Generation Polynomial)은 다음과 같이 나타낼 수 있다.

$$g(X) = (X - \alpha)(X - \alpha^2) = X^2 - \alpha^5 X + \alpha^3 \tag{2}$$

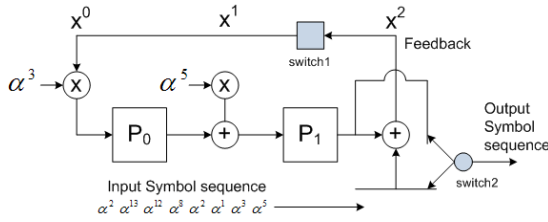


그림 2. RS(10,8)을 위한 LFSR Encoder

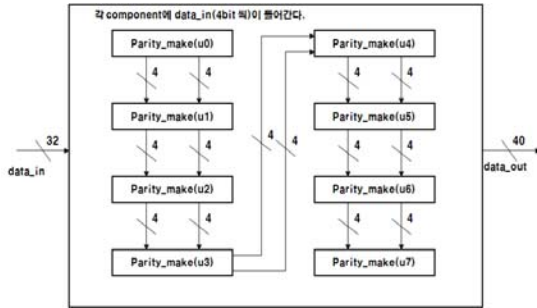


그림 3. 설계된 RS(10,8) Encoder

표 3. 패리티 생성 과정

Input Queue	Clock	Contents		Feedback
		P0	P1	
... a ¹ a ³ a ⁵	0	0	0	a ⁵
... a ² a ¹ a ³	1	a ⁸	a ¹⁰	a ¹²
... a ⁸ a ² a ¹	2	0	a ²	a ⁵
... a ¹² a ⁸ a ²	3	a ⁸	a ¹⁰	a ¹

그림 2에 위의 생성 다항식을 바탕으로 만들어진 LFSR(Linear Feedback Shift Register) Encoder를 나타내었다.

따라서 예를 들어 입력 데이터(Input Symbol)가 그림 2와 같다면 표 2를 이용한 결과는 표 3과 같이 나타낼 수 있으며 8개의 심볼 데이터가 다 입력된 후 2개의 패리티가 생성된다.

따라서 위의 결과를 바탕으로 설계된 RS(10,8) Encoder를 그림 3에 나타내었다. 32비트의 입력 데이터에 대해서 그림 2의 Encoder를 8번 반복 수행하면 8비트의 패리티와 32비트의 데이터를 포함하여 40비트를 출력한다.

2.3 RS(10,8) Decoder 설계

에러가 발생한 경우 신드롬(S, Syndrome)을 계산하여 에러심볼(e)과 에러위치(β)를 식 (3)을 이용하여 계산할 수 있으며 에러가 발생하지 않았을 때의 신드롬 값은 제로이다.

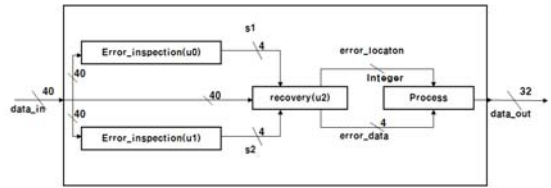


그림 4. 설계된 RS(10,8) Decoder

$$S_1 = r(\alpha) = e_{j_1}\beta_1 + e_{j_2}\beta_2 + \dots + e_{j_v}\beta_v \quad (3)$$

$$S_2 = r(\alpha^2) = e_{j_1}\beta_1^2 + e_{j_2}\beta_2^2 + \dots + e_{j_v}\beta_v^2$$

$$S_{2t} = r(\alpha^{2t}) = e_{j_1}\beta_1^{2t} + e_{j_2}\beta_2^{2t} + \dots + e_{j_v}\beta_v^{2t}$$

RS(10,8)의 경우 심볼에러복구능력(t)이 1 이므로 신드롬이 2개이며 식 (3)을 식 (4)와 같이 나타낼 수 있다.

$$S_1 = r(\alpha) = e\beta \quad (4)$$

$$S_2 = r(\alpha^2) = e\beta^2$$

식 (4)을 이용해서 연립 방정식으로 풀면 에러 위치(β)와 에러값(e)을 구할 수 있다. 그림 4에 설계된 RS(10,8) Decoder를 나타내었다. Error_inspection 프로세스에서 계산된 신드롬 값이 제로가 아닌 경우에는 recovery 프로세스에서 에러값과 에러 위치를 계산하게 되며 이 결과를 이용하여 데이터 메모리에서 에러 위치와 에러 데이터 값을 XOR 하면 정보 데이터가 복구 된다. 위와 같은 과정을 VHDL로 구현하여 FPGA내에서 처리 가능하도록 하였으며, 실제 실험을 통하여 검증하였다.

2.4 RS(10,8) 시뮬레이션 결과

그림 5에 RS(10,8) 코드의 시뮬레이션 결과를 나타내었다. 입력 데이터 32비트에 대해서 40비트가 출력되고 그 반대의 경우도 정상적으로 동작함을 알 수 있다. 또한 1 심볼에 에러가 있는 경우에는 정상적으로 에러를 복구 하지만 2 심볼에 에러가 있는 경우에는 정상적으로 복구하지 못함을 알 수 있다.

		HEX		
Encoder	d.in	12345678		Data(32bit)
	d.out	5112345678		Parity(8bit) + Data(32bit)
Decoder - No Error	d.in	5112345678		Parity(8bit) + Data(32bit)
	d.out	12345678		Data(32bit)
Decoder - Error(4bit)	d.in	51E2345678		=> Recovery
	d.out	12345678		
Decoder - Error(8bit)	d.in	51EA345678		=> No Recovery
	d.out	E4345678		

그림 5. RS(10,8) 시뮬레이션 결과

2.5 스크립 주기 분석

대용량 메모리 유닛의 메모리는 주탑재체인 적외선영상시스템과 부탑재체인 소형영상분광기에서 수신한 데이터를 저장한다. 저장된 데이터는 지상으로 전송되기 전에 우주환경에 노출되므로 우주의 고에너지 입자에 의한 SEU가 발생할 수 있다. 따라서 SEU로부터 데이터를 보호하기 위해서는 메모리를 주기적으로 스크립 해줘야 한다.

메모리의 SEU 확률 $\Gamma(\lambda, t)$ 을 구하기 위해서는 과학기술위성 3호 대용량 메모리 유닛의 λ 가 필요하다. 따라서 우리별 3호의 궤도가 730 Km의 태양동기궤도이며 과학기술위성 3호도 700 Km의 태양동기궤도로써 서로 비슷하므로 메모리 각 셀에서의 SEU 발생률 λ 는 우리별 3호 탑재 컴퓨터에서의 SEU 발생률을 조사하여 대략적으로 그 값을 추정하였다. 우리별 3호 탑재 컴퓨터의 SEU 검출수는 90 SEUs/Day 이내였다. 따라서 1 궤도상에서의 SEU 개수는 다음과 같다.

$$90 \text{ SEUs}/14\text{orbits} = 6.4 \text{ SEUs/orbit} \quad (5)$$

과학기술위성3호 MMU의 총 메모리는 40Gbit 이고 2개의 Bank로 나누어져 있으므로 20Gbit에서의 메모리 각 셀의 SEU는

$$\begin{aligned} 6.4 \text{ SEUs}/20\text{Gbit/orbit} = \\ 3.2 \times 10^{-10} \text{ SEUs/bit/orbit} \end{aligned} \quad (6)$$

이때 1궤도상에서 SEU가 발생하는 궤도상의 위치와 발생 빈도사이의 관계는 전 궤도상에서 고르게 분포하는 것이 아니라 SAA(South Atlantic Anomaly) 부근과 극지방에 집중되어 있다. 따라서 위성이 SAA와 극지방을 지난 시간은 궤도에서 따라 다르지만 대략 1궤도 시간의 1/20정도가 된다. 따라서 대부분의 SEU가 이 시간에 발생한다고 가정하고 위성이 궤도상에서 경험하는 최대 SEU를 다시 계산하면 다음과 같다.

$$3.2 \times 10^{-10} \times 20 = 6.4 \times 10^{-9} \text{ SEUs/bit/orbit} \quad (7)$$

따라서 메모리 각 셀에서의 SEU 발생률은 다음과 같이 대략적으로 추정할 수 있다.

$$\begin{aligned} \lambda &= 6.4 \times 10^{-9} \text{ SEUs/bit}/98.5 \text{ minute} \\ &= 6.5 \times 10^{-11} \text{ SEUs/bit/minute} \end{aligned} \quad (8)$$

위의 계산 결과를 보면 위성 궤도의 차이, 사용된 메모리의 종류, 메모리 위치 등의 조건이 우리별 3호와 과학기술위성 3호가 서로 다르므로 우리별 3호에서 추정된 SEU 발생률과 과학기술

위성 3호가 우주에서 경험하게 될 실제 SEU 발생률은 차이가 있을 수 있으나 근사적으로 과학기술위성 3호에 적용하여 대략적인 SEU 발생률을 확인하는데는 적용할 수 있을 것으로 판단된다. 본 연구에서 다룬 EDAC 알고리즘에 대한 확률 분석을 위하여 다음과 같은 기본 확률식을 유도한다. 메모리의 각 셀(cell)에서 발생하는 SEU는 Poisson 과정을 따른다고 가정한다. 메모리의 각 셀에서 t 시간 내에 n 개의 SEU가 발생할 확률은 다음과 같다.

$$a_n(\lambda, t) = (\lambda t)^n \frac{e^{-(\lambda t)}}{n!} \quad (9)$$

여기서 λ 는 메모리 각 셀의 SEU 발생률이다. (9)식을 이용하면 메모리 각 셀에 저장되어 있는 2진 정보가 SEU에 의해 변경되었을 확률은 다음과 같이 유도할 수 있다.

$$\begin{aligned} p(\lambda, t) &= \sum_{k=0}^{\infty} a_{2k+1}(\lambda, t) \\ &= \sum_{k=0}^{\infty} (\lambda t)^{(2k+1)} \frac{e^{-(\lambda t)}}{(2k+1)!} \\ &= e^{-(\lambda t)} \sum_{k=0}^{\infty} \frac{(\lambda t)^{(2k+1)}}{(2k+1)!} \end{aligned} \quad (10)$$

또한 (9)식으로부터 각 셀의 2진 정보가 변경되지 않을 확률은 아래의 식으로 나타낼 수 있다.

$$q(\lambda, t) = e^{-(\lambda t)} \sum_{k=0}^{\infty} \frac{(\lambda t)^{(2k)}}{(2k)!} \quad (11)$$

식(8)에서 알 수 있듯이 λt 가 굉장히 작으므로 식(10)과 (11)을 Taylor series를 이용하여 간략화하면 메모리 각 셀에 저장된 2진 정보가 변경될 확률($p(\lambda, t)$)과 변경되지 않을 확률($q(\lambda, t)$)은 다음의 수식으로 나타낼 수 있다.

$$p(\lambda, t) = [1 - e^{-2\lambda t}]/2 \quad (12)$$

$$q(\lambda, t) = [1 + e^{-2\lambda t}]/2 \quad (13)$$

각 메모리 블록에서 복구 불가능한 SEU가 존재할 확률을 유도하기 위해서 먼저 식(12),(13)을 이용하여 1심벌에서 1비트의 정보가 SEU에 의해 변경되었을 확률 $pp(\lambda, t)$ 와 변경되지 않을 확률 $qq(\lambda, t)$ 를 아래와 같이 구한다.

$$pp(\lambda t) = 1 - q(\lambda, t)^4 \quad (14)$$

$$qq(\lambda t) = q(\lambda, t)^4 \quad (15)$$

식(14),(15)를 이용하여 2심벌 이상이 SEU에

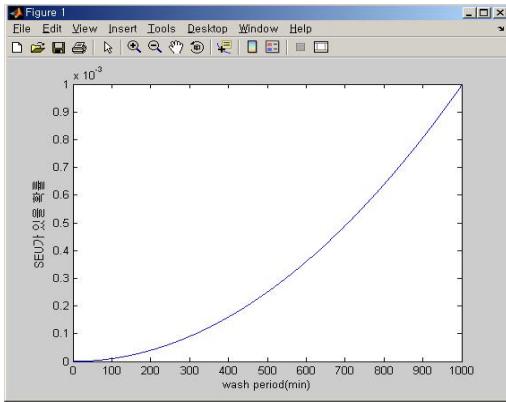


그림 6. 대용량 메모리에 SEU가 발생할 확률

의해 원래의 값과 다르게 될 확률을 다음과 같이 유도 한다.

$$\Theta(\lambda, t) = 1 - \alpha q(\lambda, t)^{10} - 10 \beta p(\lambda, t) \alpha q(\lambda, t)^9 \quad (16)$$

즉 $\Theta(\lambda, t)$ 는 RS(10,8) 코드를 이용한 EDAC 알고리즘을 사용했을 때 복구 불가능한 SEU가 존재할 확률이다.

그리고 복구 불가능한 블록이 1개 이상 있을 확률은 다음과 같다.

$$\Gamma(\lambda, t) = 1 - [1 - \Theta(\lambda, t)]^N \quad (17)$$

N은 블록수이며 하나의 블록은 데이터 심볼(32bit) + 패리티 심볼(8bit)이다. 각각의 탑재체에 대한 대용량 메모리가 20 Gb(16Gb(Data) + 4 Gb(Parity))이므로 총 블록 수는 20 Gb/40bit = 536870912 이다. 따라서 SEU가 있을 확률을 MATLAB를 이용해 계산을 하면 그림 6과 같다. 현재 과학기술위성 3호 대용량 메모리 유닛은 탑재체에서 데이터를 수신하지 않고 지상국으로 데이터를 전송하지 않는 경우에는 반복적으로 스크립 작업을 하도록 프로그램되어 있으며 실험에 의하면 주기는 대략 15분 정도이다. 그림 6의 결과를 보면 대략 50분 이후부터 SEU 발생 확률이 증가하므로 15분은 적당하다는 것을 예측할 수 있다.

2.6 SEU 시험 환경

우주입자 환경에 의한 영향을 평가하기 위해서는 지상에서 입자가속기를 이용한 지상실험을 수행하는 방법을 사용한다. 우주입자의 구성 성분은 크게 양성자와 기타 이온으로 나누어 생각할 수 있으며 각각이 전자소자와 반응하는 방식이 다르지만 나타나는 결과는 유사하다. 원칙적으로 이온과 양성자의 효과를 모두 실험하는 것

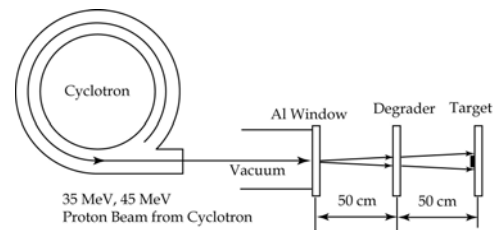


그림 7. 원자력의학원의 양성자빔의 형상

이 좋으나, 이온 가속기를 사용하려면 외국의 시설을 사용해야 하므로 비용과 시간이 많이 소요되며, 양성자의 경우 한국원자력의학원의 45 MeV 빔을 사용하여 시험할 수 있다.

그림 7에 한국원자력의학원에 있는 Cyclotron 양성자 가속기 빔의 형상을 나타내었다. 45 MeV와 35 MeV로 빔의 에너지를 튜닝하여 사용할 수 있으며, 빔 윈도우에서 약 50 cm 떨어진 곳에 알루미늄 판을 두어 빔의 에너지를 낮추어 사용이 가능하다.

2.7 SEU 시험 결과 분석

본 실험에서는 30 MeV와 37 MeV 두가지 에너지에 대해 실험을 하였으며, 실험은 2009년 8월 28일 하루에 걸쳐 수행 되었다. 그림 8에 과학기술위성 3호 대용량 메모리 유닛과 SEU 시험에서 실제로 빔이 조사되는 부분의 메모리를 나타내었다. 실험에 사용된 메모리는 과학기술위성 2호에서도 사용된 경험이 있는 3D-PLUS사의 MMSD16256808S-V-MB이다.

실험은 각각의 에너지에서 30초간 양성자 가속기에 노출시켜 수행이 되었으며, 30 MeV에서는 총 8140회의 SEU가 발생하여 이중 1873회의 SEU에 대해서 데이터가 복구 되었으며 37 MeV 빔에 노출시켰을 때에는 총 8055회의 SEU가 발생하여 이중 1035회의 데이터가 복구되었다. 이

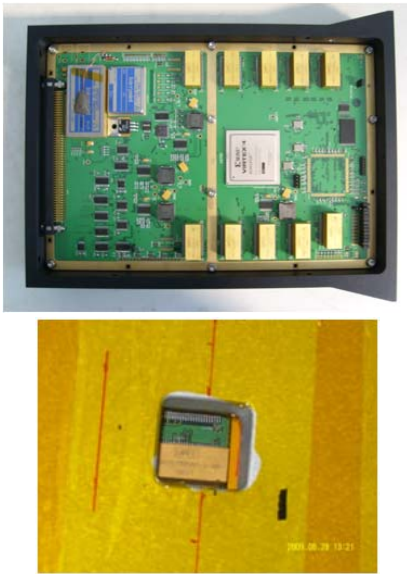


그림 8. 과학기술위성 3호 대용량 메모리 유닛과 SEU에서의 빔 조사 부분

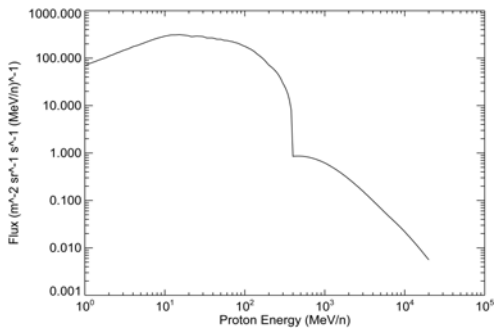


그림 9. 과학기술위성 3호가 임무수명동안 받게되는 양성자 입자의 플럭스

중 복구가 안 된 경우는 최소 2bit 이상에서 에러가 발생한 것이므로 이들에 대해서는 실제 발생회수에 2를 곱하여 SEU Cross Section을 구하였다. SEU의 발생횟수를 N_{SEU} 라 하고, SEU Cross Section을 σ , 양성자 빔의 Fluence를 f 라 하였을 때, 이들 사이에는 $N_{SEU} = \sigma \times f$ 의 관계가 성립하므로 이를 이용하여 각 에너지에 따른 SEU Cross Section 값을 유도하였다. 양성자 빔에 의한 SEU Cross Section은 에너지에 대해 다음과 같이 Bendel 함수로 기술할 수 있다.

$$F(E) = \left(\frac{B}{A}\right) [1 - e^{-0.18\sqrt{Y}}]^4 \quad (18)$$

$$Y = \sqrt{\frac{18}{A}} (E - A) \quad (19)$$

여기서, E는 양성자 빔의 에너지를 의미하며 A와 B의 값은 실험에서 구한 Cross Section을 위의 함수로 조정 하였을 때 유도되는 매개변수이다. 두가지 에너지에 대해 실험을 하였으므로 직접 A와 B의 값을 구하기가 어려웠으며, 대신 과학기술위성 2호에 사용된 메모리에 대하여 구한 SEU Cross Section과[4] 실험에서 구한 cross section을 그 차이만큼의 비율로 곱하여 사용하였다. 이렇게 구한 에너지에 따른 SEU Cross Section의 값을 우주 공간에서 예상되는 입자의 플럭스 값과 곱하여 에너지에 대해 적분을 하면 실제로 인공위성이 우주공간에서 운용될 때 예상되는 SEU 발생빈도(R_{SEU})를 구할 수 있으며 다음과 같이 표현된다.

$$R_{SEU} = \int_0^{\infty} \sigma(E) f'(E) dE \quad (20)$$

여기서 $f'(E)$ 는 위성의 외부가 아닌 실제 사용되는 메모리가 위치한 곳에서의 입자 플럭스로서 구조물에 의한 차폐효과를 고려한 입자 플럭스의 양이다. 본 실험에서는 2 mm 두께의 알루미늄에 의해 차폐되고 있다는 가정 하에 계산이 되었으며, 우주공간 상에서 양성자의 플럭스는 SPENVIS을 이용하여 계산하였다.

이러한 계산을 통해 계산된 값에 SAA를 위성이 지날 때와 태양활동에 의한 양성자 플럭스의 증가 등을 고려하여, Worst Case로서 평균값에 비해 각각 100배 씩 증가할 수 있다고 가정을 하여 최종적으로 위성 운용 시 발생할 수 있는 SEU 발생 빈도를 유도하였다. 이렇게 유도된 SEU 발생빈도는 1초를 기준으로 하였을 때, 전체 메모리에 대해서 대략 3~6 회 발생하는 것으로 나타났다. 따라서 전체 메모리에 대한 SEU 발생 확률은 그림 6의 결과와 비슷하게 매우 작다는 것을 알 수 있다.

본 시험의 결과에 대한 신뢰도가 높다고 가정한다면 스크립 주기 15분 동안에 SEU는 발생할 수 있지만 RS(10,8) 코드로 대부분 복구될 것으로 예상된다. SEU에 대한 분석 결과와 실험 결과를 종합해보면 대용량 메모리에 SEU가 발생하지 않도록 하는 것은 불가능하다. 그러나 대부분의 SEU는 SAA 지역에서 발생하며(스크립에 의해서 복구) 대용량 메모리에 저장된 데이터가 영상 데이터(특정 부분의 에러는 영상을 판독하는데 영향을 주지 않음)인 것을 감안하면 RS(10,8) 코드를 대용량 메모리의 EDAC 알고리즘으로 사용하는 데 문제가 없을 것으로 예상된다.

III. 결 론

본 논문에서는 과학기술위성 3호 대용량 메모리의 EDAC용으로 사용된 RS(10,8) 코드 대한 개발 및 분석 결과를 나타내었다.

또한 우리별 3호의 데이터를 이용하여 적당한 스크립 주기를 계산해 보았으며 원자력 병원에서의 실험을 통해서 SEU 발생률을 예측 하였다.

본 연구를 통해서 과학기술위성 3호 대용량 메모리 유닛에 사용된 RS(10,8) 코드를 자세히 분석 하였으며 실제로 발사 되었을 때 어느정도 SEU가 발생하는지 실험과 분석을 통해서 예측 하였다.

후 기

본 연구논문은 교육과학기술부 과학기술위성 3호 개발사업의 예산지원을 받아 작성되었습니다.

참고문헌

- 1) 곽성우, 박홍영, "과학기술위성 1호 탑재 컴퓨터에서의 SEUs 극복을 위한 메모리 운용 및 해석", 항공우주학회지, Vol. 32 (1), pp. 98-105, 2004.
- 2) 곽성우, "과학기술위성 3호 대용량 메모리에서의 SEU 극복 방법 및 확률해석", 전자공학회 논문지, 제45권 4호, pp. 35~41, 2008.
- 3) 김병준, 곽성우, 유상문, 김형신, "과학기술위성 3호 탑재 컴퓨터와 대용량 메모리에 적용될 EDAC 알고리즘의 비교 및 분석", 2008년도 한국항공우주학회 추계학술발표회, 논문집2, pp. 1586 ~ 1589.
- 4) http://www.informit.com/content/images/art_sklar7_reed-solomon/elementLinks/art_sklar7_reed-solomon.pdf
- 5) Kwangsun Ryu, Goo-Hwan Shin, Hyung-Myung Kim, Heejoon Kim, and Kyungwook Min, "Single Event Upset Measurements of Memory Chips for the Langmuir Probe on STSAT-2", Journal of the Korean Physical Society, Vol. 52, No. 3, pp. 853-857, 2008.