

---

# 다중 비트 소프트 에러 대응 메모리 소자를 위한 스크러빙 방안

류상문\*

\*군산대학교

Scrubbing Scheme for Advanced Computer Memories for Multibit Soft Errors

Sang-moon Ryu\*

\*Kunsan National University

E-mail : smryu@kunsan.ac.kr

## 요 약

컴퓨터 시스템의 신뢰도에 가장 큰 영향을 미치는 것이 메모리 시스템의 신뢰도이며 메모리 시스템에서 발생하는 가장 빈번한 오류는 소자의 물리적 손상 없이 저장 정보가 변경되는 소프트 에러이다. 메모리에서 발생하는 소프트 에러의 영향은 오류 검출 및 정정 회로와 스크러빙 작업을 통하여 극복할 수 있다. 메모리 소자의 집적도가 높아짐에 따라 인접한 메모리 셀에 걸쳐서 발생하는 다중 비트 소프트 에러의 발생 빈도가 증가했으며 이를 해결하기 위한 메모리 구조와 스크러빙 기법이 제안되었다. 본 논문은 다중 비트 소프트 에러 대응 메모리 소자에 대한 이전 연구 결과에 적용할 수 있는 스크러빙 수행 방안을 제안하고, 그에 따른 신뢰도 성능 해석 결과를 보여준다.

## ABSTRACT

The reliability of a computer system largely depends on that of its memory systems, which are vulnerable to soft errors. Soft errors can be coped with a combination of an Error Detection & Correction circuit and scrubbing operation. Smaller geometries and lower voltage of advanced memories makes them more prone to suffer multibit soft errors. A memory structure against multibit soft errors and a suitable scrubbing scheme for it were proposed. This paper introduces a key issue for the scrubbing of the memories with protection against multibit soft errors and the result of the performance analysis from a reliability point of view.

## 키워드

다중 비트 소프트 에러, 신뢰도, 메모리 스크러빙

## 1. 서 론

컴퓨터 시스템에서 정보 저장에 사용되는 메모리 소자의 집적도가 높아지고 동작 전압이 낮아짐에 따라 소프트 에러의 영향은 더욱 커지고 있다. 이에 따라 컴퓨터 시스템의 신뢰도가 크게 영향 받는다. 소프트 에러는 우주 방사선이나 외부 잡음에 의해 저장된 정보의 값이 변경되는 현상으로 물리적인 손상에 기인한 것은 아니다. 따라

서 올바른 정보를 다시 기입하여 주면 소프트 에러는 정정된다. 소프트 에러를 극복할 수 있는 방법으로 오류 검출 및 정정 코드[1]과 스크러빙 기법[2][3][4][5]를 결합한 방식이 가장 널리 쓰이고 있으며, 이를 적용하면 소프트 에러의 발생을 검출하고 일정 범위 이내에서 대응할 수 있다.

메모리의 워드를 구성하는 비트들 중 한 개의 비트에서 발생하는 소프트 에러를 단일 비트 에러라고하고, 두 개 이상의 비트에서 발생하는 소

프트 에러를 다중 비트 에러라고 한다. 다중 비트 에러는 인접한 메모리 셀들에서 발생할 확률이 높으며 하나의 워드에 다중 비트 에러가 많이 발생할수록 메모리 시스템의 신뢰도는 하락한다. 다중 비트 에러의 영향을 줄이기 위하여 아래의 그림과 같은 구조의 메모리가 제안되었다.[6][7]

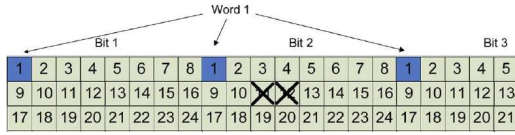


그림 1 다중 비트 소프트 에러 대응 메모리의 구조

그림 1은 메모리의 워드를 구성하는 비트들 중에서 비트 1, 2, 3만을 보여준다. 각 비트 셀에 표시된 숫자는 소속 워드의 논리적 주소를 의미한다. 따라서 물리적으로 인접한 비트 셀들은 서로 다른 주소를 할당 받은 워드에 사용되어 인접한 셀들에 다중 비트 에러가 발생하더라도 그 효과가 하나의 워드에 집중되지 않도록 하는 구조이다. 그림 1에서는 주소가 11과 12인 워드의 비트 2 역할을 맡는 두 개의 비트에 에러가 발생(X 기호로 표현)했다. 이것은 워드 11과 워드 12에 단일 비트 에러가 발생한 것과 같으므로 쉽게 정정될 수 있으며 이를 통하여 메모리의 신뢰도가 증가하게 된다.

이전 연구 [7]에서는 그림 1과 같은 구조를 갖는 메모리에 적합한 스크러빙 기법을 제안하고, 그에 따른 신뢰도 해석 결과를 보여주었다. 본 논문에서는 [7]에서 제안된 스크러빙 기법을 보완하고, 단일 비트 에러 발생률과 다중 비트 에러 발생률을 함께 고려한 신뢰도 해석 결과를 보여준다.

## II. 스크러빙 기법

이전 연구 [7]에서 제안된 다중 비트 소프트 에러를 대응할 수 있는 스크러빙 기법을 그림 2를 이용하여 설명하면 다음과 같다. 전체 메모리 영역을 청색으로 표시된 워드들과 회색으로 표시된 워드들로 구성되는 2개의 영역으로 구분한다. 그리고 각 영역에 대해 스크러빙을 병행한다. 그림 2처럼 워드 11과 12의 비트 2에 이중 비트(double bit) 에러가 발생한 상황을 가정하자. 청색 영역의 스크러빙이 진행되는 동안 워드 12의 비트 2에 단일 비트 에러가 감지되면 이를 정정하고 청색 영역의 스크러빙 작업을 잠시 보류한다. 그리고 에러가 감지된 비트의 주변 비트들 즉, 회색 영역의 워드 4, 11, 13, 20을 검사하여 워드 11의 비트 2의 에러를 감지할 수 있다. 워드 11의 비트 2의 에러를 정정한 후에는 보류되었던 청색 영역의 스크러빙을 다시 진행한다.

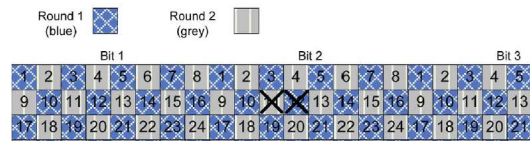


그림 2 이중 비트 소프트 에러 대응 스크러빙 기법

만일 메모리 시스템의 전체 워드를 주소에 따라 순차적으로  $T_s$ 의 주기로 스크러빙을 수행하면 발생된 이중 비트 에러가 감지되고 정정되는데 최대  $T_s$ 가 필요하다. 하지만 [7]이 제안한 것처럼 전체 워드를 2개의 영역으로 구분하고 각각  $T_s$ 의 주기로 스크러빙을 수행하며 제안된 기법을 적용하면 발생된 이중 비트 에러는 최대  $T_s/2$ 이내에 감지되고 정정될 수 있다. 스크러빙 부담은 유지하면서 메모리 시스템의 신뢰도를 개선할 수 있는 것이다.

[7]에서 제안된 기법이 효과를 발휘하기 위해서는 2개의 영역을 스크러빙하는 과정에서 고려해야 할 것이 하나 있다. 이를 설명하기 위하여 그림 2의 워드들을 청색으로 표시된 워드들을 모아 놓은 청색 영역과 회색으로 표시된 워드들을 모아 놓은 회색 영역으로 나누고 이들을 할당된 주소의 오름차순으로 정리하여 그림 3처럼 표현하기로 한다.

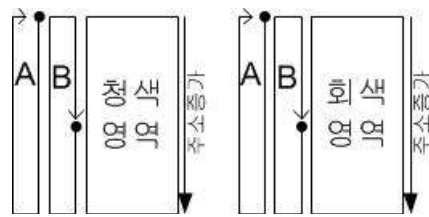


그림 3 2개 영역 스크러빙 병행 방안

각 영역에 대해 스크러빙을 수행하는 방식을 두 가지로 나누어 생각해 볼 수 있다. 각 영역의 최소 주소 워드에서 시작하여 최대 주소 워드까지 스크러빙을 수행하고 이를 반복하는 방식과 각 영역의 중간 주소 워드에서 시작하여 반복적으로 스크러빙을 수행하는 방식이다. 전자를 A 방식이라하고 후자를 B 방식이라 부르기로 한다.

만일 두 영역을 모두 A 또는 B 방식에 따라 동일하게  $T_s$ 의 주기로 스크러빙한다면 각 워드에서 발생하는 단일 비트 에러와 그림 2와 같은 이중 비트 에러가 감지되고 정정되는데 최악의 경우  $T_s$ 가 필요하다. 따라서 [7]에서 제안된 방식에 따른 신뢰도 개선 효과를 얻을 수 없는 것이다.

하지만 두 영역을 서로 다른 방식으로, 즉 청색 영역은 A 방식 그리고 회색 영역은 B 방식, 또는 청색 영역은 B 방식 그리고 회색 영역은 A 방식에 따라  $T_s$ 의 주기로 스크러빙한다면, 각 워

드에서 발생하는 단일 비트 에러는 적어도  $T_s$  이 내에 감지되고 정정될 수 있고 이중 비트 에러는 적어도  $T_s/2$ 이내에 감지되고 정정될 수 있다.

### III. 성능 해석

이전 연구 [7]에서는 워드당 소프트 에러 발생률을 이용하여 신뢰도를 해석하였다. 하지만 2장에서 보였듯이 단일 비트 에러가 검출되는데 필요한 최소 시간과 이중 비트 에러가 검출되는데 필요한 최소 시간이 다르다. 그리고 단일 비트 에러 발생률과 이중 비트 에러 발생률도 같지 않다. 본 논문에서는 단일 비트 에러 발생률과 이중 비트 에러 발생률을 각각 고려하고, 스크러빙 수행 중 단일 비트 에러와 이중 비트 에러가 검출 및 정정되는데 필요한 시간을 따로 고려한 신뢰도 해석을 수행한다.

메모리의 총 워드수를  $N$ , 각 메모리 워드의 크기는 정보 저장을 위한  $w$ 비트와 패리티 저장을 위한  $c$ 비트로 구성되어 있고, 단일 비트 에러와 이중 비트 에러는 각각 평균값  $\lambda_s$ 와  $\lambda_d$ 를 갖는 포아송 분포에 따라 발생한다고 가정[7][8][9][10][11]한다. 그리고 각 워드당 1비트의 소프트 에러가 정정될 수 있는 에러 검출 및 정정 코드 성능을 가정한다.

스크러빙 수행 방식은 2장에 소개된 것처럼 두 영역을 서로 바꾼 방식으로, 즉 청색 영역은 A 방식 그리고 회색 영역은 B 방식, 또는 청색 영역은 B 방식 그리고 회색 영역은 A 방식에 따라  $T_s$ 의 주기로 스크러빙하는 것으로 가정한다. 따라서 각 워드에서 발생하는 단일 비트 에러는 최대  $T_s$  이내에 감지되고 정정될 수 있고 이중 비트 에러는 최대  $T_s/2$ 이내에 감지되고 정정될 수 있다.

우선 단일 비트 에러에 따른 신뢰도 함수를 구하고 이를 이용하여 이중 비트 에러에 따른 신뢰도 함수를 구한다. 그리고 전체 MTTF를 구한다.

$\lambda_s \ll 1$ 인 경우 하나의 워드에 대한 신뢰도 함수는 식(1)과 같다.

$$r_{so}(t) = e^{-\lambda_s(w+c)t} + (w+c)(1 - e^{-\lambda_s t})e^{-\lambda_s(w+c-1)t} \approx 1 - \frac{1}{2}(w+c)(w+c-1)\lambda_s^2 t^2, t \geq 0 \quad (1)$$

편의를 위하여  $\alpha_s = \frac{1}{2}(w+c)(w+c-1)\lambda_s^2$ 를 정의하면 식(1)은 다음과 같이 된다.

$$r_{so}(t) \approx 1 - \alpha_s t^2, t \geq 0 \quad (2)$$

식 (2)로부터  $N$  워드 메모리 영역에 대한 신뢰

도 함수는 다음과 같다.

$$R_{so}(t) = (r_{so}(t))^N \approx 1 - \alpha_s N t^2, t \geq 0 \quad (3)$$

$T_s$ 의 주기로 스크러빙이 적용되면  $T_s$ 마다 단일 비트 에러가 정정되므로 신뢰도 함수는 매 스크러빙 작업이 완료되는 시점에서 모든 에러가 정정되고, 최근 스크러빙 작업 이후에 정정 불가능한 에러가 발생하지 않을 확률과 같다. 따라서 스크러빙 효과를 고려한  $N$  워드 메모리 영역에 대한 신뢰도 함수는 다음과 같다.

$$R_{st}(t) = [R_{so}(T_s)]^l R_{so}(\tau) \approx (1 - \alpha_s N T_s^2)^l (1 - \alpha_s N \tau^2) \approx (1 - \alpha_s N T_s^2)^{\frac{t}{T_s}}, t \geq 0 \quad (4)$$

여기서  $t = T_s l + \tau$  이고  $\tau$ 는  $0 \leq \tau < T_s$  를 만족하는 실수이며  $l$ 은  $l \geq 0$ 를 만족하는 정수이다.

마찬가지의 방법으로 이중 비트 에러에 따른 신뢰도 함수를 구한다. 편의를 위하여

$\alpha_d = \frac{1}{2}(w+c)(w+c-1)\lambda_d^2$ 를 정의하면  $N$  워드 메모리 영역에 대한 신뢰도 함수는 식 (3)에서  $\alpha_s$ 를  $\alpha_d$ 로 대체하여 다음과 같다.

$$R_{do}(t) \approx 1 - \alpha_d N t^2, t \geq 0 \quad (5)$$

이중 비트 에러는  $T_s/2$ 마다 정정되므로 식 (4)을 유도한 것과 마찬가지로 의하여 스크러빙 효과를 고려한  $N$  워드 메모리 영역에 대한 이중 비트 에러에 따른 신뢰도 함수는 다음과 같다.

$$R_{dt}(t) = \left[ R_{do}\left(\frac{T_s}{2}\right) \right]^m R_{do}(\tau) \approx \left( 1 - \alpha_d N \frac{T_s^2}{4} \right)^m (1 - \alpha_d N \tau^2) \approx \left( 1 - \alpha_d N \frac{T_s^2}{4} \right)^{\frac{2t}{T_s}}, t \geq 0 \quad (6)$$

여기서  $t = \frac{T_s}{2} m + \tau$  이고  $\tau$ 는  $0 \leq \tau < \frac{T_s}{2}$  를 만족하는 실수이며  $m$ 은  $m \geq 0$ 를 만족하는 정수이다.

단일 비트 에러와 이중 비트 에러가 발생에 따른  $N$  워드 메모리 영역에 대한  $MTTF_{prop}$ 는 그 정의에 따라 식 (4)와 (6)를 이용하여 구하면 다음과 같다.

$$MTTF_{prop} = \int_0^{\infty} R_{st}(t)R_{dt}(t) dt \quad (7)$$

$$\approx -\frac{1}{\frac{1}{T_s} \ln(1 - \alpha_s N T_s^2) + \frac{2}{T_s} \ln\left(1 - \alpha_d N \frac{T_s^2}{4}\right)}$$

두 영역을 모두 A 또는 B 방식에 따라 동일하게  $T_s$ 의 주기로 스크러빙하는 경우의  $MTTF_{bad}$ 는 식 (7)의 분모 중 우측 항의  $T_s$ 를  $2T_s$ 로 대체하면 얻을 수 있으며 다음과 같다.

$$MTTF_{bad} \approx -\frac{1}{\frac{1}{T_s} \ln(1 - \alpha_s N T_s^2) + \frac{1}{T_s} \ln(1 - \alpha_d N T_s^2)} \quad (8)$$

식 (7)과 (8)로부터 2장에서 제안된 스크러빙 방법의 적용 효과를  $MTTF_{prop}$ 와  $MTTF_{bad}$ 의 비율로 계산하여 정리하면 다음과 같다.

$$\frac{MTTF_{prop}}{MTTF_{bad}} \approx 1 + \frac{\lambda_d^2}{2\lambda_s^2 + \lambda_d^2} \quad (9)$$

식 (9)를 유도하는 과정에서  $0 < x \ll 1$  을 만족하는  $x$ 에 대해  $\ln(1-x) \approx -x$  임을 이용하였다.

식 (9)를 통하여 이중 비트 에러 발생률이 0이 아닌 경우에는 2장에서 제안된 스크러빙 방법으 로 신뢰도가 개선됨을 확인할 수 있다. 그리고 이중 비트 에러 발생률이 커질수록 개선 효과도 커짐을 알 수 있다.

#### IV. 결론

본 논문은 이전 연구 [7]에서 제안한 기법의 효과가 발휘되기 위한 스크러빙 수행 조건을 소개하고 이에 따른 신뢰도 해석을 수행한 결과를 보여주었다. 신뢰도 해석 결과 제안된 스크러빙 방법이 신뢰도를 개선할 수 있으며 이중 비트 에러 발생률이 커질수록 신뢰도 개선 효과도 커짐을 확인할 수 있었다.

#### 참고문헌

[1] R. Morelos-Zaragoza, The Art of Error Correcting Coding, Wiley, 2002.  
 [2] S. Karp and B.K. Gilbert, "Digital system design in the presence of single event upsets," IEEE Trans. Aerospace and Electronic Systems, vol.29, no. 2, pp. 310-316, Apr., 1993.

[3] A.M. Saleh, J.J. Serrano, and J.H. Patel, "Reliability of scrubbing recovery-techniques for RAMs," IEEE Trans. Reliability, vol. 39, no.1, pp. 114-122, Apr., 1990.  
 [4] G.C. Yang, "Reliability of semiconductor RAMs with softerror scrubbing techniques," IEE Proc. in Computers and Digital Techniques, vol.142, pp. 337-344, Sep., 1995.  
 [5] R.M. Goodman and M. Sayano, "The reliability of semiconductor RAM memories with on-chip error-correction coding," IEEE Trans. Information Theory, vol.37, no. 3, pp. 884-896, May., 1991.  
 [6] S. Baeg, S. Wen, and R. Wong, "SRAM interleaving distance selection with a soft error failure model," IEEE Trans. Nucl. Sci., vol. 56, no. 4, pp. 2111-2118, Aug. 2009.  
 [7] P. Reviriego, J. A. Maestro, and Sanghyeon Baeg, "Optimizing Scrubbing Sequences for Advanced Computer Memories," IEEE Trans. Device and Materials Reliability, vol. 10 no. 2, pp. 192-200, Aug. 2009.  
 [8] P. Reviriego, J.A. Maestro, and Sanghyeon Baeg, "Optimizing Scrubbing Sequences for Advanced Computer Memories," IEEE Trans. Device and Materials Reliability, vol.10, no. 2, pp. 192-200, Jun., 2010.  
 [9] Sang-Moon Ryu and Dong-Jo Park, "Transient Bit Error Recovery Scheme for ROM-based Embedded Systems," IEEE Trans. Information and System, vol. EE88-D, no. 9, pp. 2209-2212, Sep., 2005.  
 [10] S. Baeg, S. Wen and R. Wong, "Minimizing Soft Errors in TCAM Devices: A Probabilistic Approach to Determining Scrubbing Intervals," IEEE Trans. Circuits and Systems, vol. 57, no. 4, pp. 814-822, Apr., 2010.  
 [11] Z. Ming, X. L. Yi, L. Chang and Z. J. Wei, "Reliability of Memories Protected by Multibit Error Correction Codes Against MBUs," IEEE Trans. Nuclear Science, vol. 58, no. 1, pp. 289-295, Feb., 2011.