

온칩 메모리 내 다중 비트 이상에 대처하기 위한 오류 정정 부호

Error correction codes to manage multiple bit upset in on-chip memories

Hoyoon Jun*

*Principal engineer, Conextt.Inc, Seoul, 06777 Korea

ABSTRACT

As shrinking the semiconductor process into the deep sub-micron to achieve high-density, low power and high performance integrated circuits, MBU (multiple bit upset) by soft errors is one of the major challenge of on-chip memory systems. To address the MBU, single error correction, double error detection and double adjacent error correction (SEC-DED-DAEC) codes have been recently proposed. But these codes do not resolve mis-correction. We propose the SEC-DED-DAEC-TAED(triple adjacent error detection) code without mis-corrections. The generated H-matrix by the proposed heuristic algorithm to accomplish the proposed code is implemented as hardware and verified. The results show that there is no mis-correction in the proposed codes and the 2-stage pipelined decoder can be employed on-chip memory system.

Keywords : Error Correction Code, multiple bit upset, On-chip Memory, soft errors

I. 서론

최근 시스템 반도체(system large-scale integration) 제조공정이 초미세 공정으로 발전함에 따라, 온칩 메모리(on-chip memory)에서 양성자(proton) 또는 중성자(neutron)에 의한 소프트 에러(soft error)가 이제는 무시할 수 없는 도전과제다. 이러한 소프트 에러는, 일정 이상의 에너지를 갖는 양성자나 중성자가 실리콘에 충돌

했을 때, 큰 운동에너지를 갖는 중이온(heavy ion)이 다수 발생하고, 이 중이온이 온칩 메모리 내 다중 비트 이상(multiple bits upset)을 일으켰을 때 발현된다[1-4].

온칩 메모리에서는 대부분 데이터 보호를 위해 해밍 부호(Hamming codes)를 기반으로 하는 SEC-DED (Single Error Correction and Double Error Detection) 부호가 주로 채택되어 사용된다. 그러나, SEC-DED는 단일 비트는 에러 정정과 2중 비트 에러 검출만 가능하기 때문에, 다중 비트 이상(multiple bits upset) 문제를 해결할 수 없고, 다중 비트 에러 정정과 검출이 가능한 BCH (Bose-Chaudhuri-Hocquenghem)/RS(Reed-Solomon) 부호는 2비트 이상 정정 능력을 제공하지만, 많은 체크비트 요구 및 긴 디코딩 시간은 온칩 메모리 적용에 적합하지 않다 [4-5].

이러한 소프트 에러에 대처하기 위해 인접한 2중 비트 에러 정정 부호 (DAEC, Double Adjacent Error Correction code)가 제안되었다[6-7]. 기제안된 DAEC 부호는 SEC-DED을 수행할 수 있고, 추가로 인접한 2중 비트에 에러가 발생했을 때, 정정 능력을 제공한다. 그러나 인접하지 않는 2중 비트 에러가 발생했을 시, 인접한 2중 비트 에러로 인식하여 정정하는 오정정(mis-correction)이 일어날 수 있는 단점이 존재한다.

본 논문에서는 소프트 에러에 의한 다중 비트 이상으로부터 온칩 메모리를 보호할 수 있는 오류 정정 부호를 제안한다. 제안하는 부호는 SEC-DED-DAEC-TAED (triple adjacent error detection) 기능을 수행하고 또한 최소 체크 비트 증가를 수반하면서 오정정이 발생하지 않도록 설계하였다.

II. 본론

제안된 부호는 최소 해밍거리(minimum hamming distance) 4이상 5이하 사이의 값을 갖는 단축된 해밍 부호(shortened hamming code)다. 제안된 부호는 다음 4가

Received 30 September 2022, Revised 10 October 2022, Accepted 11 October 2022

* Corresponding Author Hoyoon Jun(E-mail:hockma24@yonsei.ac.kr, Tel:+82-02-4211-8996)

Principal engineer, Conextt.Inc, Seoul, 06777 Korea

Open Access <http://doi.org/10.6109/jkiice.2022.26.11.1747>

print ISSN: 2234-4772 online ISSN: 2288-4165

지 기능을 제공한다. 1) 단일 비트 정정, 2) 오정정 없이 모든 인접한 2중 비트 정정, 3) 2중 비트 에러 검출, 4) 인접한 3중 비트 에러 검출. 본 연구에서는 이러한 4가지 기능을 구현하기 위해, 오류정정 부호 생성행렬(generation matrix, H-matrix) 생성 시 반드시 준수해야 하는 6개 조건을 제안한다.

- <1> 생성행렬 내 모든 열벡터는 영벡터 (zero-column vector)를 포함할 수 없다.
- <2> 생성행렬 내 모든 열벡터는 서로 중첩 없이 고유 (unique vector)벡터로 구성한다.
- <3> 생성행렬 내 모든 열벡터는 항상 홀수(odd) 해밍무게 (hamming weight)를 갖는다.
- <4> 생성행렬 내 서로 인접한 2개 열벡터의 XOR 연산값은 고유하도록 열벡터를 배치한다.
- <5> 생성행렬 내 비인접한 2개 열벡터의 XOR 연산값과 인접한 2개 열벡터의 XOR 연산값은 상호 배타적이 (mutually exclusive) 되도록 배치한다.
- <6> 생성행렬 내 서로 인접한 3개 열벡터의 XOR 연산값은 어떤 단일 열벡터의 값과도 상호 배타적이 되도록 배치한다.

각 조건에 의해 제안된 코드가 갖게 되는 특성은 다음과 같다. 첫 번째 조건은 부호어(codeword) 내 각 비트에 발생된 에러를 검출 기능을 제공한다. 두 번째 조건은 부호어 내 발생된 단일 비트 에러를 정정 기능을 제공한다. 세 번째 조건은 제안된 코드의 해밍 거리 값을 최소 4가 되게 하여, 2중 비트 에러 검출 기능을 완벽히 제공한다. 네 번째 조건은 부호어 내 발생된 인접 2중 비트 에러를 정정 기능을 제공한다. 다섯 번째 조건은 부호어 내 발생된 비인접 2중 비트 에러를 인접 2중 비트 에러로 인식하여 발생하는 오정정을 방지한다. 여섯 번째 조건은 오정정 없이 부호어 내 발생된 인접한 3중 비트 에러 검출 기능을 제공한다.

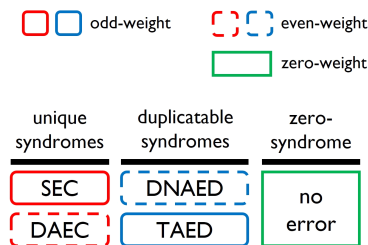


Fig. 1 Syndrome space for the proposed codes

그림 1은 생성행렬 생성 시 적용된 6개 조건 들을 준수할 경우 정의되는 신드롬 공간(syndrome space)을 도식화하였다. 부호어 내 에러가 발생하지 않았다면, 신드롬은 영벡터 값을 갖게 되고, 단일 비트 에러 정정과 인접한 3중 에러 검출을 위한 신드롬은 홀수 해밍 무게를 갖게 되고, 인접한 2중 비트 에러 정정과 비인접 2중 비트 검출을 위한 신드롬은 짝수 해밍 무게를 갖게 된다.

제안된 부호를 만족하는 생성행렬 생성하기 위해, 휴리스틱 알고리즘(heuristic algorithm)을 사용하였다. (n, k) 부호어 생성을 위해 (n : 부호어 비트 수, k : 데이터 비트 수), 홀수 해밍 무게를 갖는 생성행렬을 작성할 때, 총 선택 가능한 홀수 무게 열벡터의 개수(cases)는 수식 1과 같다. 예를 들어, 홀수 해밍 무게를 갖는 (39, 32) 부호어 생성행렬에 사용될 수 있는 총 열벡터 수는 57이고, 데이터 비트 수가 32이므로 열벡터 선택을 위해 선택 가능한 조합의 수는 $9.9e+15$ (${}_{57}C_{32}$)개이다. 또한, 열벡터 선택 조합과 더불어 열벡터의 순열도 고려한 총 경우의 수는 $32! \times {}_{57}C_{32}$ 개가 된다. 즉, 완전탐색 기법으로는 생성행렬을 생성 불가능하므로 휴리스틱 알고리즘이 필요하다.

$$cases = \sum_{w=2i+1}^{w \leq n-k} \binom{n-k}{w}, \text{ for } i \geq 1. \quad (1)$$

그림 2에 제안된 부호의 생성행렬을 생성하기 위해 사용된 휴리스틱 알고리즘을 기술하였다. 제안된 알고리즘의 주요 과정은 크게 4가지로 정의된다(홀수 해밍 무게를 갖는 열벡터 행렬 생성 / 열벡터 후보 선택 / 6개 조건 테스트 / 최소 1의 개수를 갖는 생성행렬 검색). 첫 번째, 홀수 해밍 무게를 갖는 열벡터 행렬 생성을 생성하기 위해 초기 체크 비트 수는 SEC-DED의 수와 동일하게 적용하고, 같은 홀수 해밍 무게를 갖은 열벡터를 사전적 순서(lexicographical order)로 정렬하여 열벡터 행렬을 생성한다. 두 번째, 생성된 열벡터 행렬에서 해밍 무게가 작은 열벡터부터 선택한다. 세 번째, 열벡터가 선택될 때마다, 현재까지 선택된 열벡터들에 대해 제안된 6가지 조건 만족 여부를 테스트한다. 네 번째, 6가지 조건을 만족하는 열벡터가 데이터 비트 수 만큼 선택되었을 때, 생성행렬 후보 내 1의 개수가 가장 작은 경우를 선택한다. 마지막으로 데이터 비트 수 만큼 6가지 조건을 만족하는 열벡터를 찾지 못하면, 체크 비트를 1 증가시켜 주요 과정을 다시 수행한다. 그림 3은 알고리즘에 의해 생성된 (42, 32) 부호이다.

```

Algorithm H-matrix generation algorithm
1: check_bits ← calc_default_check_bits(data_width, SECDED)
2: found ← false
3: while true do
4:   create_column_pools(check_bits)
5:   for seed column in odd weight column pools
6:     hmatrix_initialization(seed column)
7:     num_found_cols ← 1
8:     while num_found_cols < data_width do
9:       sel_col ← select_candidate_column()
10:      is_pass, uinfo ← examine_hmatrix_conditions(sel_col)
11:      if is_pass == true then
12:        update_hmatrix_column_pools(uinfo)
13:        num_found_cols++
14:      else
15:        exit line 8 while loop
16:      end if
17:    end for
18:    if num_found_cols == data_width then
19:      found ← true
20:      h-matrix ← min_ones_hmatrix()
21:    endif
22:  end for
23:  if found == true then
24:    exit line 3 while loop
25:  else
26:    check_bits ++
27:  end if
28: end while
    
```

Fig. 2 H-matrix generation algorithm

III. 제안된 부호 디코딩 하드웨어 구현

제안된 SEC-DED-DAEC-TAED 부호의 디코더를 하드웨어로 구현하고 검증하였다. 구현과 검증에는 하드웨어 언어인 베릴로그(verilog)를 사용하였고, 시놉시스(Synopsys) 32nm 공정으로 합성을 진행하였다. 100,000개 입력 데이터를 이용하여 검증을 진행하였고, 수행된 검증 항목은 단일 비트 오류 정정(SEC), 2중 비트 오류 검출(DED), 인접한 2중 비트 오류정정 (DAEC), 비인접 2중 비트 에러에 의한 오정정 유무, 인접한 3중 비트 오류 검출 (TAED), TAED 신드롬에 의한 단일 비트 오정정 유무, 총 6가지이다.

```

011111111111111100000000000000101000000000
110101010000000111111001000000100100000000
010010101000000010000100000100000010000000
101010000100000101000010100000100001000000
100100000010100000100000101001010000100000
0010000000100000010101111010000000010000
000001000010010010010000101000000000001000
000000100001001001000000010111100000000100
000000010100010000001100001101010000000010
000000001000101100001011000001011000000001
    
```

Fig. 3 H-matrix for the proposed (42,32) codes

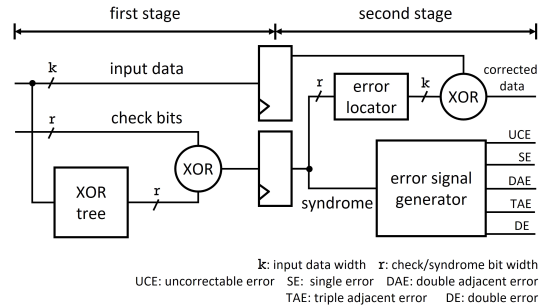


Fig. 4 The proposed SEC-DED-DAEC-TAED decoder

그림 4는 본 연구에서 제안한 부호의 디코더 회로를 도식화하였다. 빠른 동작 클럭(operating clock)으로 동작하기 위해 2-단계 파이프라인으로 설계하였다. 첫 번째 단계에서는 부호어(입력 데이터 + 체크 비트)를 입력으로 받아 XOR tree에 전달된다. 생성행렬을 기반으로 구성된 XOR tree를 통과 후, XOR tree 결과 값과 체크 비트를 XOR 연산하여 최종적으로 신드롬이 생성된다. 두 번째 단계에서는 생성된 신드롬을 이용하여 에러 판별 및 위치를 계산한다. 에러 발생 시 에러 위치 탐색기(error locator)의 값과 입력 데이터 값을 XOR하여 최종적으로 정정된 데이터값을 출력한다.

IV. 실험 결과

표 1을 통해 기제안된 SEC-DED-DAEC 부호와 본 연구에서 제안한 부호 특징을 비교하였다. 수식 2는 오정정비(mis-correction ratio) 산출 방법 나타낸다. 즉, 비인접 2중 비트 에러에 대한 총 신드롬과 인접 2중 비트 에러에 대한 신드롬과의 중첩률을 의미한다.

$$mc\ rate = \frac{sharable\ syndroms}{all\ double\ non\ adjacent\ errors} \quad (2)$$

제안된 부호를 포함하여 비교되는 3개 부호 모두 데이터 비트 길이는 32, 추가된 체크 비트 길이 10으로 동일하다. 제안된 부호를 제외하고 Datta와 neale 부호의 경우 오정정이 존재한다. Datta부호는 neale 부호에 비해 오정정비가 근소하게 낮지만, 생성행렬 내 1의 개수가 너무 많다. 생성행렬 내 1의 개수는 곧 XOR tree에 사용되는 XOR gate 개수를 의미하기 때문에 작을수록 속도/면적/파워 측면에서 이점이 크다. 그리고 제안된 부호는

TAED를 지원하기 때문에 Datta나 neale 부호는 제공하지 못한 인접한 3중 에러에 대한 에러 검출이 가능하다.

표 2에 SEC-DED-DAEC-TAED 디코더 회로의 합성 결과를 정리하였다. SEC-DED 부호를 기준으로 산출하였고 동작 속도는 23% 낮고, area와 power는 각각 39%, 33% 증가 되었다(정규화 후 비교). 면적과 파워의 증가는 신드롬 생성에 사용된 XOR 게이트와 오류 정정시 사용된 AND 게이트 증가에 기인한다. 그러나 2중 에러를 완벽하게 정정할 수 있는 DEC BCH의 증가와 비교하면 본 연구에서 제안하는 부호는 온칩 메모리에 사용 가능한 범위 내 있는 사소한(trivial) 오버헤드(overhead)로 판단된다.

V. 결론

본 논문은 온칩 메모리를 다중 비트 이상으로부터 보호할 수 있는, 단일 비트 에러 정정, 2중 비트 에러 검출, 인접한 2중 비트 에러 정정, 인접한 3중 비트 에러 검출 능력을 갖는 (SEC-DED-DAEC-TAED) 부호를 제안하였다. 제안된 부호를 생성하기 위한 생성행렬은 제안된 6개의 조건을 만족하는 휴리스틱 알고리즘으로 실현 가능했다. 그리고 기존 제안된 SEC-DED-DAEC 부호와 동일한 체크 비트 수를 사용하더라도 오정정 없이 인접한 2중 비트를 정정할 수 있고 추가적으로 인접한 3중 비트 에러 검출도 가능하다. 제안된 부호를 하드웨어로 구현/검증 하였고, 합성을 진행하여 온칩 메모리 영역에 사용될 수 있는 가능성을 보였다.

Table. 1 Comparison between existing codes

Codes	# of check bit	mis-correction rate for DAEC	# of 1s in H	TAED
Datta [6]	10	8.8%	140	NO
Neale [7]	10	9.0%	80	NO
Proposed	10	0%	98	YES

Table. 2 Comparison between SEC-DED, DEC BCH and the proposed SEC-DED-DAEC-TAED

Codes	# of check bit	# of 1s in H	freq.	area	power
SEC-DED[5]	7	96	1	1	1
DEC BCH[5]	12	200	0.52	6.12	5.22
Proposed	10	98	0.77	1.39	1.33

REFERENCES

- [1] C. Constantinescu, "AMD EPYCTM 7002 Series - A Processor with Improved Soft Error Resilience," in *Proceeding of the 51st IEEE/IFIP International Conference on Dependable Systems and Networks - Supplemental Volume*, Taipei, Taiwan, pp. 33 - 36, 2021.
- [2] G. Prasad, B. C. Mandi, and M. Ali, "Soft-Error-Aware SRAM for Terrestrial Applications," *IEEE Transaction on Device and Materials Reliability*, vol. 21, no. 4, pp. 658-660, Dec. 2021.
- [3] M. Rezaei, P. M. Holgado, Y. Morilla, F. J. Franco, J. C. Fabero, H. Mecha, H. Puchner, G. Hubert, and J. A. Clemente, "Evaluation of a COTS 65-nm SRAM Under 15MeV Protons and 14 MeV Neutrons at Low VDD," *IEEE Transaction on Nuclear Science*, vol. 67, no. 10, pp. 2188-2195, Oct. 2020.
- [4] H. Farbeh, L. Delshadtehrani, H. Kim, and S. Kim, "ECC-United Cache: Maximizing Efficiency of Error Detection/Correction Codes in Associative Cache Memories," *IEEE Transaction on Computer*, vol. 70, no. 4, pp. 640-654, Apr. 2021.
- [5] S. Lin and D. Costello, *Error Control Coding*, 2nd ed. Pearson, India, 2004.
- [6] R. Datta and N. A. Touba, "Exploiting Unused Spare Columns to Improve Memory ECC," in *Proceeding of the 27th IEEE VLSI Test Symposium*, Santa Cruz: CA, US, pp. 47 - 52, 2009.
- [7] A. Neale and M. Sachdev, "A New SEC-DED Error Correction Code Subclass for Adjacent MBU Tolerance in Embedded Memory," *IEEE Transaction on Device and Materials Reliability*, vol. 13, no. 1, pp. 223-230, Mar. 2013.