SECDED code and its extended applications in DRAM system

Haochen Zhou

Beijing university of post and telecommunication, No. 10 West Tucheng Road, Haidian District, Beijing, China

zhouhaochen_048@bupt.edu.cn

Abstract. SECDED code is the combinational logic for communication of data between microprocessors and memories which is commonly used in semiconductor systems due to its advantage in implementation. With advances in semiconductor integrated circuit technology, the soft error rate and importance of selecting optimal codes increase. This paper presents SECDED codes and their major expansion theoretical application and gives suggestions for optimization needs.

Keywords: DRAM, error correcting code, DAEC, COMET.

1. Introduction

Circuit size and operation voltage are decreased as a result of the market's insatiable need for bigger electronic gadgets, which greatly increases the semiconductor's radiation sensitivity. It's important to keep in mind that soft errors, often referred to as single-event upsets (SEUs), happen when a radiation incidence introduce enough charge disturbance to turn the data state in a memory cell or other crucial circuitry, dynamic random-access memory which abbreviate as DRAM is an outstanding example [1]. Error-correcting codes (ECC) have become widely apply in recent years to maintain the stability and data integrity of semiconductor systems. The tendency toward semiconductor integration has led to an increase in the use of ECCs. System conductors are desirous of a cost-effective ECC technique as a result [2–3]. The single error correction double error detection codes (SECDED), which is an extended SEC method implement through extending Hamming distance, is frequently used to boost the reliability of computer semiconductor systems due to its low storage and area overheads. Examples of typical applications include the IBM System/360 Model 85 and the IBM 7030 [4].

DRAM is one of the most important components of the computer semiconductor system. The constantly growing disciplines of computer science and data research make faster access DRAM and larger compatible capacities more important. because the recent growth in DRAM density has raised the likelihood of data failures. As a result, the DRAM system depends largely on suitable ECCs.

In DRAM systems, it is essential to choose and employ ECCs in a way that effectively captures their salient characteristics and assesses error performance using a probabilistic simulation. Examples of the collaborative Memory ECC Technique (COMET) and double adjacent error correction (DAEC), two of the most recent applications of SECDED code for unique error types-aiming expansion in DRAM systems, are given in the material that follows.

In this paper, we will present more details about SECDED Hamming code in the next section and introduce two major extension applications of SECDED code also and error correction and detection performance in the second following section.

2. SECDED code

This section provides insight details about SECDED code and its extension algorithm in DRAM.

2.1. Hamming code algorithm

The Hamming code was created in 1950 by Richard W. Hamming to automatically fix mistakes caused by punch cards. In the field of communications, a type of linear error-correcting codes is known as hamming codes. Figure 1 of the flowchart shows the channel process for linear codes. The greatest encoding rate is achieved with a Hamming code with a length of block and a minimum distance of three.
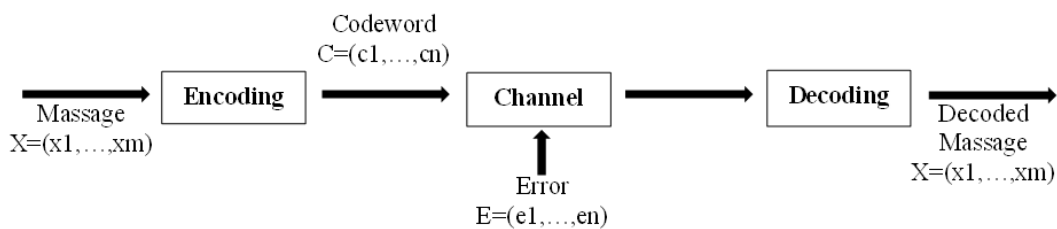

Figure 1. The coding process of the linear code system (Photo Credit: Original).

In mathematics, hamming code fall under the umbrella of binary linear codes (BLC). Parameters of BLC are the number of stored bits in total, $n = k + r$, the number of data bits, k, and the number of processing check bits, r. The n-dimensional bit stream vectors area of n-tuples is translated onto such a k-dimensional subspace using a (n, k) code. When encoding, the G-matrix is utilized, and when decoding, the H-matrix. The code is well described by any matrix. In this investigation, the H-matrices of the codes are described. In order to decipher the codeword X, it was necessary for $X*HT=0$, where HT remains for the notate of H and zero for the r bits zero vector.

E flips the bits of the codeword X, resulting in an m-bit error (x1,..., xm). (x1,...,xm). The error vector E can be used to describe the issue (e1,..., en). There was an error at the jth element of e when $ej = 1$, but there was no error at the jth bit position of e when $ej = 0$. The bitwise XOR operation of the two n bits size matrix to generate the outcome and accepted, vector **C=X+E** may be used to symbolize the wrong vector E being inserted into the codeword X.

The H matrix uses the produced vector X to determine the error syndrome. E. The error E syndrome has the model $\mathbf{S=C*H^T=(X+E)*H^T=E*H^T}$. The syndrome S(E), which is the column-wise XOR outcome in H that correspond to the vector error sites in E, which if H is seen as a collection of n column vectors, such as H=, is given by S(E)=(i1,...,im) for the m bit error E(h1,...,h2).

If the syndrome vector S(E) for an error E is not zero, the error can be found. Additionally, if the set of syndromes S(E1),..., S(Es) is distinct, the collection of mistakes E1,..., Es is correctable. A 2r-1 nonzero error pattern relates to an r-bit syndrome. A pattern of all zeros is used to demonstrate that there hasn't been a mistake. A complete Hamming code is an H-matrix of r-row with entire the 2r-1 unique non-zero columns. The complete Hamming codes are K = 2r-1 and N = 2r-1 [5]. This method can only identify and correct one error. Every other fault type will be represented as different single-bit error because every symptom has a direct connection to a single-bit problem.

Some columns in the H-matrix can be removed to improve the overall Hamming code's error-handling capabilities (enabling the correction and/or detection of various fault kinds). Applying this theory will help you protect against MBUs. It is customary to define k to the size of a simple memory word, such as 8 bits ,16 bits , 32bits , or 64 bits, to calculate how many check bit r are needed to stand the necessary function of error prevention. As the consequence, the code becomes shorter. It is feasible to shift underutilized syndromes to other error types by allowing K2r-1-r.

2.2. SECDED code algorithm

Hamming codes can only detect and correct errors when the fault rate is low since they only give the data a little level of redundancy. This is true in data storage, where errors are rare and Hamming codes are frequently used. Hamming codes with an extra parity bit are frequently employed in this situation. The extended Hamming code employs a Hamming distance of 4, enabling the decoder to discriminate between situations in which there are no more than one error and those in which there are two errors. Extended Hamming codes, often known as SECDED or single error correction and double error detection, are described in this way.

According to hamming code algorithm if a single error occurred the syndrome S will equal the relative parity check matrix H's column vector; if components are upset S will equal the XOR operation result of the parity check matrix H's column vector. Thus, to meet single error correction and double error detection requirements the H matrix should satisfy two conditions.

1)   In the parity check matrix, each pair of columns should be unique.

2)   In parity check matrix, the outcome of the XOR process of two columns ought to be different from the results of the other columns.

Take an (8,4) extended hamming code as an example. The Structures of matrix H is demonstrated below.

$$H = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 \end{pmatrix}$$
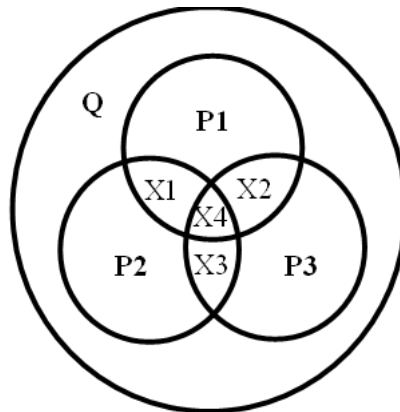
$$X*G = (Q \quad P1 \quad P2 \quad P3 \quad X1 \quad X2 \quad X3 \quad X4$$

Figure2. (8,4) extended Hamming Code codeword construction (Photo Credit: Original).

The minimum distance appears to have increased from d=3 to d=4 with the increase in codework length from 7 to 8, which also increased the maximum number of errors that can be detected by 1, from d-1=2 to d=4.

The construction of the example code word is illustrated in Figure 2. It's obvious that the Q bit is the result of XOR all parity check bits also codeword bits, and each parity check bit (P1,…P3) covered different codeword bits. Due to the bond of check-bit and massage the parity feature of Q and P check-bit represents the same feature of the complete coded massage which provides an ideal error detection method to distinguish the error type between one-bit and two-bit errors.

For a more refined presentation, we use the example of (8,4) extended hamming code here. We can distinguish error type by the following approach:

1)   If the Q and two of P check-bit convert it means that there are one-bit upsets in the codeword.

2)   If only one of P check-bit converts it means there is a high possibility that a two-bit upset occurred.

In this SECDED code example, We add a global parity bit to the (7,4) Hemming code, so that the SECDED code only increases the latency and space occupation caused by the XOR operation compared to the base Hemming code

SECDED is an ECC code frequently utilized for embedded memory applications. Because of the advantage of being easy to implement and having little influence on the memory system in terms of latency and space.

2.3. SECDED-COMET code

On the memory die, common SEC and single error correcting double error SECDED are linked together. Regrettably, the SEC can improperly rectify two bits faults up to 45% of the time, leading to triple-bit mistakes. In a traditional in-controller SECDED, these errors would have been securely discovered but left unfixed. Next, they are improperly corrected by the controller of memory more than 55 percent [6] of the time, which induce silent data destruction.

According to the linear hamming code algorithm's decoding technique, a double bit mistake in bit position 1 and 2 results in the processing outcome stream c, which is the same logic as introducing the error e1, e2 to the initial coding outcome. Look at the SEC H matrix example below, which has 128 message bits in total and 8 of parity check bits, as an instance. The total of the two columns represents the resultant syndrome. If the H matrix's column 4 equals the sum of columns 1 and 2, then what happens next is since it fits the syndromes created, the decoder would interpret this as a one-bit mistake occur at the position of 4 and turn column 4 as portion of its repair method. A double-bit error that was previously existent has changed into a triple-bit error as a result.

Let's think back to the earlier instance. Chip 4's bits 1, 2, and 4 are impacted by a DBE, and bit 4 is wrongly corrected by the SEC decoder. Following the transmission of the data, this causes triple-bit errors in the SECDED codeword 9, 10, and 12 bit positions. Since the SECDED code's H matrix's total of these columns equals another column, this turns into an SDC. When bit 63 flips, the decoder states that the fault repair was successful and transmits the tainted data to the processor.

Table 1. Comparison of COMET code with SECDED code.

	Code Independence	Memory protocol change	Free of SDC induced by DBE	In DRAM SBE ECC	ECC check bits every128b codeword	ECC check bits every 64b codeword
SECDED	Independent	no	no	yes	8	8
COMET	Independent	no	yes	yes	8	8

To prevent this SDC in a circuit with this specific on-die SEC code, an SECDED parity check matrix should be made so that the total of all the groups of columns corresponding to the bit positions mentioned above does not equal any of the columns in the abandoning the rest H matrix. We first fill columns 1, 2, and 4 with any three nonzero eight-bit values to produce such a H matrix. The recalculate of three specific columns is then recorded in a list that cannot be used. For the next set of three columns, we once more choose three randomly generated non-zero values, but these values are different from the ones that are kept in the never-use list. The list of items not to utilise is then increased by the sum of these three columns. We repeat the operation for the other columns.

It is necessary to address the triple-bit errors within the codeword of SECDED caused by the SEC data word individually. The eight bits numbers allotted to the final H-matrix columns then were created at randomly and do not relate to anything on the never-use list once all triplets have now been taken into account. The system memory architecture demands that the computed values of the columns of the H matrix be distinct from those of any other column since each bit or 3 columns in the parity check of SECDED may give SDC for a specific SEC code [6].

This method enables the generation of the SECDED code, which ensures the mitigation of SDCs when double bit mistake occur, with the proper SEC implementation and system design. Table 1 compares COMET systems with SECDED on-die and in-controller techniques.

2.4. SECDED-DAEC code

Due to the increased integration of density, soft mistakes, such as single bit and multiple bit errors, do now and then happen. Solutions for repairing various DRAM mistakes are important to assure the dependability of memory systems since Read-cells' performance may be related. The importance of ECC, which use parity bits to safeguard the data bits, has increased. A Gilbert-Elliot channel model, a two-state Markov process, is employed when real-world data sets are available, and the channel parameter is modified to fit the data. Unaware that double adjacent errors are primarily burst errors [7]. The DAEC code is an extended SECDED code which capable to correct single-bit errors and adjacent two-bit errors. Any two adjacent columns must have sums that are distinct from those of any other two adjacent columns as well as from any single column to execute DAEC [8].

1) In H, every column is non-zero and distinct.

2) In H, each columns get an odd weight larger than one.

3) In H, any two neighbouring columns' sums are entirely separate from one another and equal to zero. As an illustration, consider the SEC-DED-DEAC code in Figure 3 that has the underlying parity check matrix with k = 16 and n = 22. in order to show the decoder [8].

$$
H = \begin{pmatrix}
1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\
1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{pmatrix}
$$

Figure 3. parity check matrix of (22,16) SECDED-DAEC code (Photo Credit: Original).

SEC-DED codes are similarly decoded, but the error location logic is easier because there are fewer mistake patterns to rectify. To interpret SECDED and DAEC codes, three procedures are used: syndrome calculation, error localisation, and error rectification. The parity check equations are essentially recalculated as part of the syndrome computation. The error localization phase compares the symptoms with each of the H matrix's columns for single mistakes. To check for double neighbouring errors, the comparison is done using the add outcome of the adjacent columns. Finally, an XOR could be used for mistake correction.
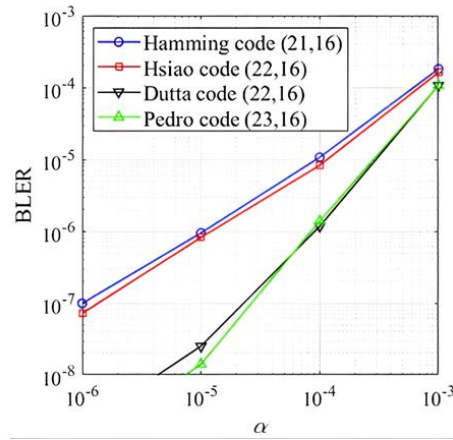
Figure 4. Performance comparison of SECDED code and SECDED-DAEC code implementations (Photo Credit: Original).

Figure 4. shows the performance of SECDED codes and the comparison of SECDED-DAEC codes in the (n, 16) GE channel implementation, which shows that DEAC codes have better error correction and detection performance in GE channels.

3. Code performance evaluation

A crucial parts of digital computers system is DRAM. The success of data science and artificial intelligence is increasing demand for faster, larger-capacity DRAM. The unusually high cell density of modern DRAM chips renders them more prone to errors.

SECDED coding is an ECC that is most frequently utilised for memory system applications. because it is advantageous to have a low impact on the latency and space of the memory system because it is straightforward to implement. However, the multi-bit error COMET and DAEC codes perform more effectively at error detection and correction for a variety of high-integrated circuits.

To prevent SDC when a DBE happens in the DRAM system, the collaborative Memory ECC Technique (COMET) is presented. This technique effectively designs the embedded SEC method and SECDED method that leads the SDC. COMET can mitigate a considerable amount of double-bit mistakes that result in SDC and repair 99.9% of all DBE induced SDC with little impact on performance [6].

Without COMET, more than eighty percent of double bit error are often identified as DUE, but the application only consistently conceals the accompanying SDC in less than 2% of instances. In 12% of circumstances, on average, the erroneous output is produced that noticeably degrades the output quality; in the remaining 80% of scenarios, the application either hangs or crashes. When unacceptable output errors or crashes happen in 18% of cases, SECDED-COMET code structures get rid of SDCs and convert them into more acceptable DUEs [6].

The DAEC codes surpass SECDED codes in the GE channel for all length parameters. When the channel model of GE's parameters are minimal, the DAEC code property performs error correction even successfully. Although the simulation findings appear to be normal, this comparison of error performance is significant. It is verified in the paper that DAEC codes greatly improved the error repairing performances for a variety of length parameters by addressing often occurring double adjacent mistakes [9].

4. Conclusion

The most used ECC for DRAM applications is SECDED codes [10]. Linear error correction codes (SECDED codes), which are based on Hamming codes, can detect and repair two-unit faults as well as one-unit errors per memory word. SECDED code has the benefit of being simple to implement and

having little effect on the storage system's latency and space. Despite the SECDED code's outstanding advantages in the embedded memory environment, the issue of multiple error rate improvement brought on by the ongoing expansion of integrated circuits prevents its performance from meeting the requirements for multiple error correction and detection. Therefore, an extended theory of SECDED codes (COMET and DEAC are the main focus of this paper) has emerged, which can meet the error correction and error detection needs in some specific circumstances in the DRAM environment.

Selecting ECC codes with varying performances for various system fault characteristics is crucial in a DRAM system. This work presents the expanded use of two recently suggested SECDED codes for various error kinds in DRAM systems and summarises and assesses their performance.

SECDED-COMET code and SECDED-DAEC respectively propose a feasible structure optimization and basic theoretical extension based on SECDED code for silent data corruption and burst error channel error correction in a DRAM environment. The resulting problem of insufficient error detection and correction performance provides a basis for theoretical development

References

[1] Baumann R C 2005 Radiation-induced soft errors in advanced semiconductor technologies *IEEE Transactions on Device and Materials Reliability* 10.1109/TDMR.2005.853449.

[2] Levine L and Meyers W 1976 Special Feature: Semiconductor Memory Reliability with Error Detecting and Correcting Codes *Computer* 10.1109/C-M.1976.218410.

[3] Ferris-Prabhu A V 1979 Improving Memory Reliability through Error Correction *Computer design* VOL. 18; NO 7; PP. 137-144; (4 P.); BIBL. 6 REF.

[4] Hsiao M Y A Class of Optimal Minimum Odd-weight-column SEC-DED Codes *IBM Journal of Research and Development* 10.1147/rd.144.0395.

[5] Cui Y, Lou M, Xiao J, Zhang X, Shi S and Lu P 2013 Research and implementation of SEC-DED Hamming code algorithm *IEEE International Conference* 10.1109/TENCON.2013.6718953.

[6] Alam I and Gupta P 2022 COMET: On-die and In-controller Collaborative Memory ECC Technique for Safer and Stronger Correction of DRAM Errors *2022 52nd Annual IEEE/IFIP International Conference on DSN* 10.1109/DSN53405.2022.00024.

[7] Lee D, Cho E and Kim S H 2021On the performance of SEC and SEC-DED-DAEC codes over burst error channels *2021 ICTC.* 10.1109/ICTC52510.2021.9621154.

[8] Reviriego P, Martínez J, Pontarelli S and Maestro J A A Method to Design SEC-DED-DAEC Codes With Optimized Decoding *IEEE Transactions on Device and Materials Reliability* 10.1109/TDMR.2014.2332364.

[9] Dutta A and Touba N A Multiple Bit Upset Tolerant Memory Using a Selective Cycle Avoidance Based SEC-DED-DAEC Code *Proc. 25th IEEE VLSI Test Symposium* pp. 349-354

[10] Hsiao M Y 1970 A Class of Optimal Minimum Odd-weight-column SEC-DED Codes *IBM Journal of Research and Development* 10.1147/rd.144.0395.