Microelectronics Reliability 81 (2018) 167–173

Contents lists available at ScienceDirect

Microelectronics Reliability

journal homepage: www.elsevier.com/locate/microrel

Correspondence

Fault tolerant encoders for Single Error Correction and Double Adjacent Error Correction codes

ARTICLE INFO

Keywords:
Memory
Fault tolerant
Logic sharing
Error correction codes (ECCs)
Single Error Correction-Double Adjacent Error Correction (SEC-DAEC)

ABSTRACT

Error correction codes (ECCs) are commonly used to deal with soft errors in memory applications. Typically, Single Error Correction-Double Error Detection (SEC-DED) codes are widely used due to their simplicity. However, the phenomenon of more than one error in the memory cells has become more serious in advanced technologies. Single Error Correction-Double Adjacent Error Correction (SEC-DAEC) codes are a good choice to protect memories against double adjacent errors that are a major multiple error pattern. An important consideration is that the ECC encoder and decoder circuits can also be affected by soft errors, which will corrupt the memory data. In this paper, a method to design fault tolerant encoders for SEC-DAEC codes is proposed. It is based on the fact that soft errors in the encoder have a similar effect to soft errors in a memory word and achieved by using logic sharing blocks for every two adjacent parity bits. In the proposed scheme, one soft error in the encoder can cause at most two errors on adjacent parity bits, thus the correctness of memory data can be ensured because those errors are correctable by the SEC-DAEC code. The proposed scheme has been implemented and the results show that it requires less circuit area and power than the encoders protected by the existing methods.

1. Introduction

Soft errors caused by radiation particles have become one of the most challenging issues for memory reliability [1]. When a radiation particle, such as an atmospheric neutron or a heavy ion hits the sensitive node of a memory cell, extra electron-hole pairs are generated, and the charges can be collected by depletion regions. When the amount of the collected charge exceeds a critical value, the voltage level of the node could be changed, and a soft error on the cell occurs [2]. For memories, soft errors can lead to data corruption and system failure. As technology scales down, the charge induced by a radiation particle can be shared by several neighboring cells, thus multiple cell upsets (MCUs) occur [3]. MCUs, especially double adjacent errors, are a significant percentage of soft errors in current technology nodes [4–6] (e.g., the percentage of double adjacent errors is more than 70% of all multiple error patterns for 65 nm/45 nm SRAMs in [5]).

Error correction codes (ECCs) are commonly used to protect memories against soft errors [7–8]. Additional redundancy cells that store the parity bits are added to each word in the memory array. Then, the encoder turns the k-bit data into an n-bit code-word and for the errors in the code-word, the decoder can correct them and output the original data if the code has enough correction capability. Single Error Correction-Double Error Detection (SEC-DED) codes are widely used due to their simplicity [7]. However, SEC-DED codes alone are no longer sufficient to protect the memories in the presence of MCUs. One common approach to deal with MCUs is to combine SEC-DED codes with the use of interleaving in the arrangement of the memory cells, so that cells that belong to the same logical word are physically separated. In that case, double adjacent errors as an example, can only affect one bit per word and thus can be corrected by the SEC-DED codes. However, interleaving is not suitable in small memories or register applications as its use may have an impact on floor-planning, access time and power consumption [6,9]. In order to deal with double adjacent errors, which are the major error pattern among the MCUs, Single error correction-Double Adjacent Error Correction (SEC-DAEC) codes have been studied by using the same number of parity bits as the ones of SEC-DED codes [9]. Therefore, in most cases, SEC-DAEC codes are a good choice to provide protection for memories.

When ECCs are used to deal with the soft errors, there is also a reliability problem for the encoders and decoders because the circuits can also be struck by the radiation particles as the memory cells. An error in the encoder during a write operation can corrupt the memory word being written. An error in the decoder during a read operation can cause an incorrect output data. This means that the reliability for encoders and decoders can affect the memory data directly. A well-known technique such as Triple Modular Redundancy (TMR) can protect the logic by tripling the circuits and making a majority vote among the outputs [10]. However, it will result in a large circuit overhead. Some other alternatives with moderate overheads have been presented in recent years. A fault tolerant encoder for SEC-DED codes has been proposed in [11] by considering the error in the encoder has a similar effect to the SEU in memory and then can be corrected by the code. A concurrent error detection technique to detect errors in the encoders of Orthogonal Latin Squares (OLS) codes was proposed in [12]. This scheme takes advantage of the property of OLS codes, which have no logic sharing among the computations of the parity bits, to make the encoder self-checking. It can also be implemented for the syndrome computation part of the decoder. This work was extended to the decoder of OLS codes in [13]. Similarly, fault tolerant encoders and decoders for Euclidean Geometry-Low Density Parity Check (EG-LDPC) codes were proposed in [14]. However, as in the case of OLS codes, the proposed scheme is tailored to the specific features of EG-LDPC codes and cannot be used for SEC-DAEC codes.

In this paper, a method to design fault tolerant encoders for SEC-DAEC codes is studied based on the method presented in [11] but by using logic

https://doi.org/10.1016/j.microrel.2017.12.017
Received 11 April 2017; Received in revised form 9 November 2017; Accepted 10 December 2017

$$\begin{bmatrix} 1&1&0&0&1&1&0&0&0&1&1&1&0&1&0&0&1&0&0&1&0&0&0&0&0 \\ 0&0&0&1&0&1&0&0&1&1&0&1&1&0&0&1&0&1&0&1&0&0&0&0 \\ 1&0&1&0&1&0&0&1&1&1&1&0&0&0&1&1&0&0&1&0&0&0 \\ 1&0&0&1&0&0&1&1&1&0&0&1&0&1&1&0&0&0&0&1&0&0 \\ 0&1&1&0&1&0&1&1&0&0&0&0&1&1&0&1&0&0&0&0&1&0 \\ 0&1&1&1&0&1&1&0&0&0&1&0&1&0&1&0&0&0&0&0&0&1 \end{bmatrix}$$

Fig. 1. Parity check matrix H for the (22, 16) SEC-DAEC code in [9].

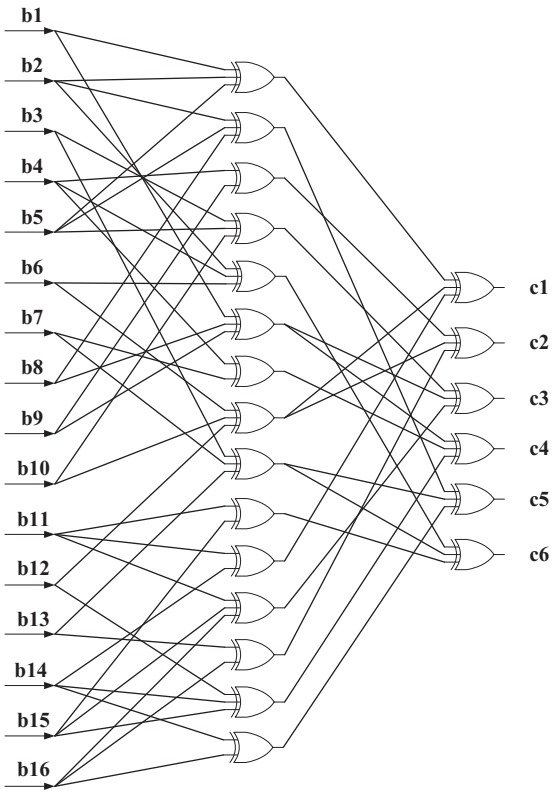

Fig. 2. Illustration of the encoder for the (22, 16) SEC-DAEC code in [9].

sharing blocks. The idea will be explored in detail in the following sections showing that it can be used to further reduce the circuit overheads comparing to the existing methods.

The rest of the paper is organized as follows. Section 2 includes a description of fault tolerant encoders for SEC-DED codes in [11]. Based on that, the proposed fault tolerant encoders for SEC-DAEC codes by using logic sharing blocks are presented in Section 3. Section 4 gives the comparison for the implementation of different methods to protect encoders, which shows the benefit of the proposed scheme. Finally, the conclusions are presented in Section 5.

2. Related work

SEC-DAEC encoders can be implemented simply by computing the parity bits for the input data bits. According to the parity check matrix H of the codes, each parity bit can be obtained by performing multiple *xor* operations among the data bits. For example, the H matrix and the encoder implementation for (22, 16) SEC-DAEC code (i.e., $n = 22$, $k = 16$) in [9] are shown in Fig. 1 and Fig. 2 respectively, in which b1-b16 are the data bits and c1-c6 are the parity check bits. As mentioned before, the parity bit c1 can be obtained by performing multiple *xor* operations among the data bits b1, b2, b5, b6, b10, b11, b12, and b14 based on H matrix, and so on for other parity bits.

In Fig. 2, it can be observed that errors on many of the first stage gates would corrupt more than one of the parity bits. For example, if the gate that has inputs b6, b10, and b12 is affected by a soft error during a write operation, the output of these two gates used to obtain c1 and c2 will be corrupted. Then there will be a double error on c1 and c2 stored in the memory word and it is uncorrectable by the SEC-DED code.

Soft errors in the encoder have a similar effect to soft errors in a memory word, having this into account, the fault tolerant encoders for SEC-DED codes have been designed in [11] by adding some additional *xor* gates to avoid any logic sharing among the parity bits. The method can also be implemented for SEC-DAEC codes. The modified encoder for (22, 16) SEC-DAEC code using the scheme in [11] is shown in Fig. 3, in which the grey gates are the added ones. It can be observed that if an error affects any gate of the encoder circuit, at most one bit error will be created among the parity bits, and then the error can be corrected by regarding it as an error in the memory word. With this scheme, the encoders for SEC-DAEC codes can have an adequate protection level at the cost of some overheads in the encoder implementation.