Implementation of Error Detection and Correction Codes using VLSI

K. Kaviya Priya, Mr. M. Palaniappan M. E.,
PG Scholar, Assistant Professor
Shanmuganathan Engineering College, Arasampatti

Abstract
Errors that affect memories are a major issue in advanced electronic circuits. As technology scales, multiple bit errors become more likely. This limits the applicability of traditional protection techniques like Matrix code or single error correction codes that can correct only one error. Multiple errors tend to affect adjacent bits, and therefore it is interesting to use error correction codes that can correct adjacent errors. The issue with these codes is that they require a large area and delay that limits their use to protect flip-flops in circuits. This project presents the implementation and evaluation of the encoder reuse technique algorithm for the detection and correction of multiple transient faults in volatile memories with low cost implementation. The logic reusable technique is proposed to minimize the area overhead of extra circuits without disturbing the whole encoding and decoding process. The proposed system has been coded in Verilog HDL and simulated using Xilinx 12.1.

I. INTRODUCTION
A parity is involved in older method of error detection. It processed by adding an extra bit to each character. The bit is obtain by a number of factors such as the type of parity and the data character has the number of logic-one bits.

The another structure in code is alliteration code is that discloses to detect error. The coding doodle to attain error-free communication to reveals bit across channel. In data streame data are divided into data bits in data blocks. Every block is transmitted a scheduled number of times. The errors in the same place leads to more problems so it is not effective as parity. They are simple, then used in the transmission of number stations. Error correction is the revelation of errors and mordenization of the original data as error-free data.

Fault tolerance is the property that assist a system to extend contriving properly in event of the failure of one or more faults of its constituents. Hamming code is a process that has a set of ECC that can be used for detect and correct bit errors while in data transmission and storage.

II. RELATED WORKS
1) Punctured Difference Set (PDS) code is a process to identify the multiple cell upsets in memories.
2) The bits by the same logical word into different physical words which has been used in restrain multiple cell upsets in interleaving technique.
3) Built-in current sensors (BICS) are scheduled for reinforcement on correction single error detection and double error detection for granted fortification against multiple cell upsets.
4) 2-Dimmensional matrix codes are prospective to conducive correct multiple cell upsets per word with a low decoding delay in which one word has been divided into multiple rows and column.

Drawbacks:
1) PDS codes require more area, power, and delay overheads since the encoding and decoding circuits are more complex in these complicated codes.
2) Interleaving technique may not be practically used in content-addressable memory (CAM), because of the tight coupling of hardware structures from both cells and comparison circuit structures.
3) BICS technique can only correct two errors in a word.
4) 2D MC is capable of correcting only two errors in all cases.

In the recent technique names as FUEC–triple adjacent error correction (TAEC), is able to correct an error in a single bit, or an error in two adjacent bits (2-bit burst errors) or a 3-bit burst error, or it can detect a 4-bit burst error. This is possible by adding one more code bit. In this case, for a 16-bit data word, the FUEC–TAEC code needs eight code bits. The parity-check matrix H for this code is presented. As in the case of the FUEC–DAEC, Ci are the code bits and Xi are the primary data bits. Similarly, from H it is very easy to design the encoder/decoder circuitry. But this technique will be considered as less precision which could not correct the large number of datas.
III. PROPOSED SYSTEM

In this paper, we propose a new algorithm named Data Segmentation Section Code (DSSC) based on divide-symbol is proposed to provide enhanced memory reliability. This algorithm for the detection and correction of multiple transient faults in volatile memories with low cost implementation.

Fig.1 Proposed architecture

Data Segmentation section code is an Error Correction code based on two-dimensional code. The code in this codifies 16 data bits in 32 bits. Thus only parity bits are used for encoding data bits to reduce the area and time conception.

A, Data Segmentation Section Code Encoding Process

Fig.2 shows the structure of 32 bits of data encoded by Data Segmentation Section Code. The cells in gray were data bits, they were divided into four groups (A, B, C, D).

<table>
<thead>
<tr>
<th>A1</th>
<th>A2</th>
<th>A3</th>
<th>A4</th>
<th>Di1</th>
<th>Di2</th>
<th>Di3</th>
<th>Di4</th>
<th>CbA13</th>
<th>CbA24</th>
</tr>
</thead>
<tbody>
<tr>
<td>B1</td>
<td>B2</td>
<td>B3</td>
<td>B4</td>
<td>Di2</td>
<td>Di4</td>
<td>B1B13</td>
<td>B1B24</td>
<td></td>
<td></td>
</tr>
<tr>
<td>C1</td>
<td>C2</td>
<td>C3</td>
<td>C4</td>
<td>P1</td>
<td>P3</td>
<td>C1C13</td>
<td>C1C24</td>
<td></td>
<td></td>
</tr>
<tr>
<td>D1</td>
<td>D2</td>
<td>D3</td>
<td>D4</td>
<td>P2</td>
<td>P4</td>
<td>CbD13</td>
<td>CbD24</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Fig.2 DSSC Encoded data model.

The cells in green are the Diagonal bits (Di) analyzed with XOR operations in specific data bits:

\[
\begin{align*}
Di_1 &= A_1 \oplus B_2 \oplus C_1 \oplus D_2 \\
Di_2 &= A_2 \oplus B_1 \oplus C_2 \oplus D_1 \\
Di_3 &= A_3 \oplus B_4 \oplus C_3 \oplus D_4 \\
Di_4 &= A_4 \oplus B_3 \oplus C_4 \oplus D_3
\end{align*}
\]

The cells in blue were Parity bits (P) analyzed by XOR operations in the data bits columns:

\[
\begin{align*}
P_1 &= A_1 \oplus B_1 \oplus C_1 \oplus D_1 \\
P_2 &= A_2 \oplus B_2 \oplus C_2 \oplus D_2 \\
P_3 &= A_3 \oplus B_3 \oplus C_3 \oplus D_3 \\
P_4 &= A_4 \oplus B_4 \oplus C_4 \oplus D_4
\end{align*}
\]

The cells orange is a Check bits (Cb) analyzed by XOR operations in interleaved bits of each group:

\[
\begin{align*}
CbA_{13} &= A_1 \oplus A_3 \\
CbA_{24} &= A_2 \oplus A_4 \\
CbB_{13} &= B_1 \oplus B_3 \\
CbB_{24} &= B_2 \oplus B_4 \\
CbC_{13} &= C_1 \oplus C_3 \\
CbC_{24} &= C_2 \oplus C_4 \\
CbD_{13} &= D_1 \oplus D_3 \\
CbD_{24} &= D_2 \oplus D_4
\end{align*}
\]

The redundancy bit was analyzed and the encoding process ends and the 32 bits was stored. The Dibits and Cbbits are arranged between the data bits and Cbbits, in order to develop the efficiency of Data Segmentation Section Code against Multiple Cell Upsets characterized by adjacent error patterns. Figure describes the mains elements of the parity operation of the Data Segmentation Section Code encoder.

The decoding process of DSSC is divided into three steps:

1. Syndrome appraisal of the redundancy bits - The syndrome appraisal consists of a XOR operation between the redundancy data stored and the recalculated redundancy bits (RDi, RP, and Rcb). So the values for the Syndrome of Diagonal, Parity and Check bits are estimated by:
Verification of error decoding conditions - After the analysing of the Syndromes, one of these two conditions need to be satisfied before the error correction execution: (i) SDi and SP vectors have at least one value similar to one; (ii) more than one SCb value was similar to one. The conditions permite the algoritihm for identify the error for data bits region.

Selection and correcting the wrong data region and correction processes - In this decoding process a distinct region is selected in the data bit and corrected. The region are divided into regions and it is shown below. They split the data bits in three regions and it was explained so as to select a definitvegroup of bit for the correction process. This reduce the area and the time conception.

The fig 3(a), (b) and (c) show that region 1, 2 and 3 are formed by data bits distributed in columns (1 and 2), (3 and 4) and (2 and 3), respectively. The selection of which region will be corrected is defined by the integer sum (+) of specific bits of SDi and SP. Table presents a group of equations which describes the criterion for region selection of DSSC, where the region with more syndrome bits equals to 1 is be declared as the wrong one (Region 1 or Region 2). If the sum of the equations presents equal value, then the Region 3 is selected.

<table>
<thead>
<tr>
<th>Region selected</th>
<th>Criterion to selection</th>
</tr>
</thead>
<tbody>
<tr>
<td>Region 1</td>
<td>(SDi1 + SDi2 + P1 + P2) &gt; (SDi3 + SDi4 + P3 + P4)</td>
</tr>
<tr>
<td>Region 2</td>
<td>(SDi1 + SDi2 + P1 + P2) &lt; (SDi3 + SDi4 + P3 + P4)</td>
</tr>
<tr>
<td>Region 3</td>
<td>(SDi1 + SDi2 + P1 + P2) = (SDi3 + SDi4 + P3 + P4)</td>
</tr>
</tbody>
</table>

Table1 : Region selection criterion.

For regions 1 and 2, the correction procedure consists in a XOR operation between the region selected and the SCb matrix. Region 3 is a special case where it is strictly necessary that neither of all SDi and SP bits are null, even if the condition II of step 2 is satisfied. Note that Region 3 has its first column formed by values with the even index (2), meaning that the correction performed has to be different from the other regions. If region 3 is selected, the correction procedure must be performed by SDi with shifted positions, to align the indexes of SCbs with the matrix of Region 3.

IV. SIMULATION RESULTS

The proposed circuit are simulated and synthesized by using modelsim and xilinx12.1 which occurs low area than the existing. The experimental results are given in Table2 and the simulation results of layout and the waveforms are shown in the fig.4 and 5. Then the synthesis result of the proposed are shown in fig.6.
<table>
<thead>
<tr>
<th>S.No</th>
<th>Parameter</th>
<th>Existing</th>
<th>Proposed</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Slice</td>
<td>56</td>
<td>29</td>
</tr>
<tr>
<td>2</td>
<td>LUT</td>
<td>96</td>
<td>50</td>
</tr>
<tr>
<td>3</td>
<td>IOB</td>
<td>39</td>
<td>47</td>
</tr>
</tbody>
</table>

Table 2: comparison of existing and proposed results

![Image of time conception for existing and proposed.]

Table 3: Comparison of time

<table>
<thead>
<tr>
<th>S.No</th>
<th>Time for existing</th>
<th>Time for proposed</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>14.027</td>
<td>13.616</td>
</tr>
</tbody>
</table>

V. PERFORMANCE ANALYSIS

The Figure given below is shown that there is a considerable reduction based on no of transistors and the performance chart has been shown below in fig.7

![Image of performance analysis graph]

VI. CONCLUSION

This project proposes with Data Segmentation Section Code this is an error detection and correction code to memory devices exposed to MCUs. By using this code on parity code and enclosed to handle with more Multiple Cell Upsets. Data Segmentation Section Code exhibited the lowest cost of coding, low area and improve time conception. Though, hamming and Extended Hamming codes in the ECCs Matrix brought advantages in error coverage it increased heavily the cost of both codes, when compared with Data Segment Code. In regard to the expounded results, Data Segmentation Section Code offering the dominant results for all fault summaries, where that Data Segmentation Section Code has the high performed compromise between error coverage and in lower area and in time conception by the analyse.

REFERENCES