International Journal of Science and Research (IJSR)
ISSN: 2319-7064
SJIF (2022): 7.942

Enhanced Hamming Codes on SRAM based FPGA for Space Applications

Shahana. P. N

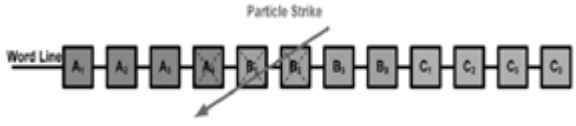
E-mail: *pnshahana[at]yahoo.com*

Abstract: ***Field Programmable Gate Arrays (FPGAs) are increasingly demanded by spacecraft electronic designers due to its improved performance. SRAM-based FPGAs are uniquely suited for remote missions' applications. But due to high radiation on the sensitive part of the circuits introduces soft errors, also called Single Event Upset (SEU). Radiation-induced soft error rate (SER) degrades the reliability of static random-access memory (SRAM)-based field programmable gate arrays (FPGAs). Error Correction Codes (ECCs) are used to prevent soft errors from causing data corruption in memories and registers. Highly preferred error detection technique is Triple Modular Redundancy (TMR). TMR comes with high area and power dissipation penalties. Most of the error detection and correction technique can correct errors by the entire reprogramming techniques. Hence affects the entire system functionality. This paper presents Enhanced Hamming codes on SRAM-based FPGAs in hostile operating environments such as space. This technique provides multibit error correction and adjacent error detection capability can improve the reliability, and hence, system availability, by orders of magnitude. Here we have 2D Hamming encoder section with selective bitplacement strategy and decoder section with lexicographic check matrix. Simulation results show that the multibit error correction capability of Enhanced hamming codes can recover configuration bits without depending on an external memory preserving a golden copy of the configuration bits. This multibit error correction technique provides higher FPGA system throughput and high error tolerance.***

Keywords: FPGA (Field Programmable Gate Arrays), SEU (Single Event Upset), SER (Soft Error Rate), ECC (Error Correction Code), TMR (Triple Modular Redundancy), SRAM (Static Random Access Memory)

1. Introduction

Field Programmable Gate Array (FPGA) devices have been used in space for more than a decade with a mixed level of success. Reprogrammable devices have been used in spacecraft due to their sensitivity to involuntary reconfiguration due to Single Event Upsets (SEU) induced by radiation. But with the advent of reprogrammable devices featuring a million system gates or more, it is not longer feasible to disregard these technologies. The FPGA vendors have already begun to develop SEU mitigation techniques in order to make their devices usable in space applications. The capacity and performance of FPGAs suitable for space flight have been increasing steadily for more than a decade. For reprogrammable devices the increase has been from tens of thousands to millions system gates. The application of FPGAs has moved from simple glue logic to complete subsystem platforms that combine several real time system functions on a single chip, even including microprocessors and memories. The potential for FPGA use in space is steadily increasing, continuously opening up new application areas. The FPGAs are more commonly being used in critical applications and are replacing ASICs on a regular basis. SRAM-based FPGAs, its widespread use in mission-critical space applications is limited due to its susceptibility to single-event upsets (SEUs). Impact of highly energized particles like protons, neutrons or alpha particles on sensitive locations of circuits results in SEUs.



a) Design of System with Hamming Codes

During the data transmission probability for occurring malfunction due to errors are very high. Different coding techniques were introduced and used. Codes generated will have message bits and check bits, each check bit will be the function of message bits. Techniques for detecting and correcting errors proposed by Richard W Hamming known as the Hamming codes. 2-D Hamming product code (2-D HPC) which can be described in such that an array of data processed, for soft error protection on SRAM based FPGA. 2-D Hamming product code (2-D HPC) scheme which provides large performance improvement in multibit error detection and correction technique. Based on the survey of existing soft error mitigation techniques of SRAM arrays and their limitations in the state-of the-art FPGAs, we address the need of a new 2-D error correction (EC) scheme for FPGA configuration data SRAM-based FPGAs, its widespread use in mission-critical space applications is limited due to its susceptibility to single-event upsets (SEUs).For instance, a single-bit soft error in configuration memory can result in permanent malfunction of a system. Data stored in the SRAM cells in FPGA consist of user data as well as configuration bits. User data is as important as the configuration bits to ensure the correct functionality of a system. The number of configuration bits is more than 98% of all memory elements in an FPGA. Since the portion of user data is very small compared with that of configuration data and several effective protection techniques are available for user data, 2D hamming product code focuses on protection of configuration memory. To ensure the integrity of configuration memory data in FPGAs, some recent FPGAs provide background read-back function that performs continuous read of the memory contents without disturbing the function of FPGAs. In a satellite system, FPGA performs programmed functions while it interacts with other system components. Due to the susceptibility of SRAM-based FPGA to soft errors, frequent repair processes have to be performed to ensure reliable operation. Thus, repairing FPGAs in such architecture involves interruptions of multiple system components. Therefore, the actual cost of correcting errors in FPGA and restarting the entire system is

much higher than that of repairing FPGA itself. To prevent frequent system interruptions due to SEUs in FPGA we use this 2D hamming product code. The figure1 below shows the error caused when the particles strike.

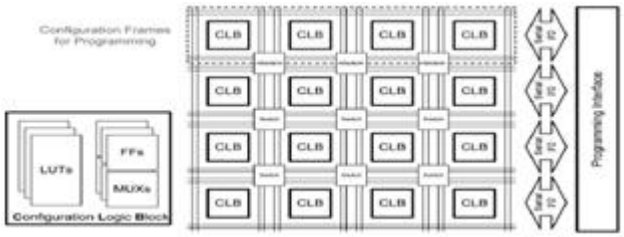


Figure1: Error occurred when particle strike

On the figure we can see that particle strike on bit B1 but the damage cause to the neighboring bit also in the case of 1D Hamming code only that error is detected. But the 2D Hamming code that we used on which column and row wise access will take place and results multibit error correction technique. Here we discuss about Enhanced Hamming codes on mission based critical space applications. Enhanced Hamming code application technique which includes 2D Hamming encoder and decoder section together with different new strategies. Hamming codes which can be described as for any positive integer m ≥ 3 they have the following parameters:

$$n = 2m - 1$$
$$k = n - m$$
$$d\ min = 3$$

where n is the block size, m the parity check bits, k the number of information bits and dmin the minimum distance of the code. Minimum distance which can be described as,

x 0000
y 0101

$$\delta(x,y) = 2$$

b) Soft Errors in Scaled Technologies

Scaling of devices has been driven by demands for higher functionality, higher density, lower cost and lower power. Aggressive feature size and supply voltage scaling has resulted in reduction of critical charge (Qcrit) in memory cells. Qcrit is defined as the minimum charge capable of flipping the stored bit in a memory cell. Intuitively, smaller Qcrit would result in higher SER. However, it has been observed that SRAM bit SER has started to saturate and is expected to decrease in-deep submicrometer regimes. Saturation in the supply voltage scaling and decrease in junction collection efficiency has resulted in the saturation of SRAM bit SER. Notwithstanding the saturation in SRAM bit SER, the SRAM system SER has increased dramatically with each technology generation. The increase in system SER can be attributed to the exponential growth in SRAM integration density with device scaling and has become a great concern for future technology nodes. In addition to the increase in SER due to aggressive technology scaling, cosmic radiation is another significant factor in accelerating SER in high-altitude and space applications.

c) Multibit Upsets in SRAM-Based FPGAS

As scaling of devices enables higher density, higher performance, low cost, and low power in ASICs, the same trend has been observed in FPGAs. Moreover, higher integration density is desirable in FPGAs due to the presence of excessively large programmable circuitries (e.g., configuration logic blocks (CLBs) and routing logic). Since the contents of configuration memory define the functionality, susceptibility to soft error is even more important issue in high-density SRAM- based FPGAs. For instance, a single-bit soft error in configuration memory can result in permanent malfunction of a system. While prevention of SEUs in configuration bits of FPGAs is of paramount importance, the SEU mitigation techniques commonly used in conventional SRAM arrays are not easily applicable to SRAM-based FPGAs.

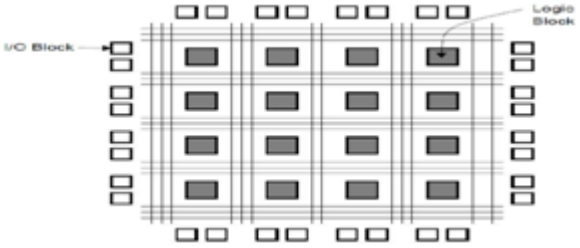


Figure 2: Basic FPGA Architecture

FPGAs are programmable semiconductor devices that arebased around a matrix of Configurable Logic Blocks (CLBs) connected through programmable interconnects. FPGAs allow designers to change their designs very late in the design cycle– even after the end product has been manufactured and deployed in the field. In addition, Xilinx FPGAs allow for field upgrades to be completed remotely, eliminating the costs associated with re-designing.

d) Hamming Encoder

Basic hamming encoder converts 4bit data to 8bit codeword by inserting parity bits. Parity bit (P) is placed as power of two positions. Remaining bit positions are placed with data bits (D)

Table 1: Hamming encoder

1	2	3	4	5	6	7	8
2^0	2^1	-	2^2	-	-		2^3
P1	P2	D1	P3	D2	D3	D4	P4

Parity p1 = data [3] ^ data [5] ^ data [7]
Parity p2 = data [3] ^ data [6] ^ data [7] Parity p3 = data [3] ^ data
[2] ^ data [1]
Parity p4 = data [3] ^ data [5] ^ data [6] ^ data [7] ^ p1 ^ p2 ^ p3
Here is an example for the data 1101 that are encoded with Hamming Code by inserting parity bits on the powers of two positions as per the XOR operations of the data bits.

Table 2: Example of basic Hamming encoder

D3	D5	D6	D7
1	1	0	1

→

P1	P2	D3	P3	D5	D6	D7	P4
1	0	1	0	1	0	1	0

The table below describes the steps for the development of hamming code word for data 1101.

Table 3: Steps of basic Hamming encoder

Bit position	2^0	2^1		2^2				2^3
Data packet	1	2	3	4	5	6	7	8
Input data			1		1	0	1	
P1	1		1		1	0	1	
P2		0	1		1	0	1	
P4			1	0	1	0	1	
P8			1		1	0	1	0
Code word	1	0	1	0	1	0	1	0

e) Hamming Decoder

Decode section on which data bits are decoded from the codeword generated on encoder section. Three cases

1) Absence of error.
2) Presence of error.

To detect whether error occurred on that data received on decoder firstly control bits are generated by the operations on the bits received.

- C1 = data [1^3^5^7]
- C2 = data [2^3^6^7]
- C3 = data [4^5^6^7]
- S = data [1^2^3^4^5^6]

The control bits that generated and the S bit generated as per the XOR operations will determine the position of the error bit and that will be corrected as per the programming condition.

Table 4: Error position on basic decoder section

S BIT	C1 C2 C3	POSITION OF ERROR
0	000	NO ERROR
0	001	1
0	010	2
0	011	3
0	100	4
0	101	5
0	110	6
0	111	7
1	000	8

f) Enhanced Hamming Encoder

Enhanced Hamming encoder section which has 2D Hamming encoder with selective bit placement strategy. 2D Hamming encoder on which an array of data is processed. Then it follows Selective bit placement strategy. Here 4×4 bit of data which is given and it converts 4×4 bit data into 4×12 bit encoded data.

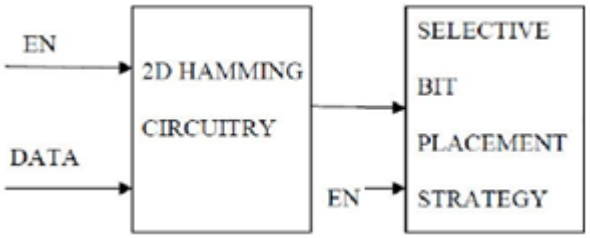


Figure 3: Enhanced hamming encoder

Now lets first have the encoder section. 2D Hamming encoder section on which data is processed in matrix format initially we have 4×4 bits of data as input which encoded as 4×12 bits through the following steps:

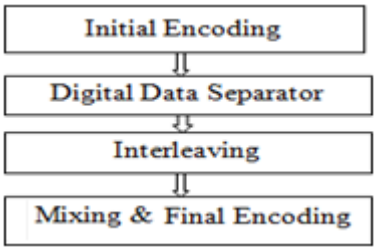
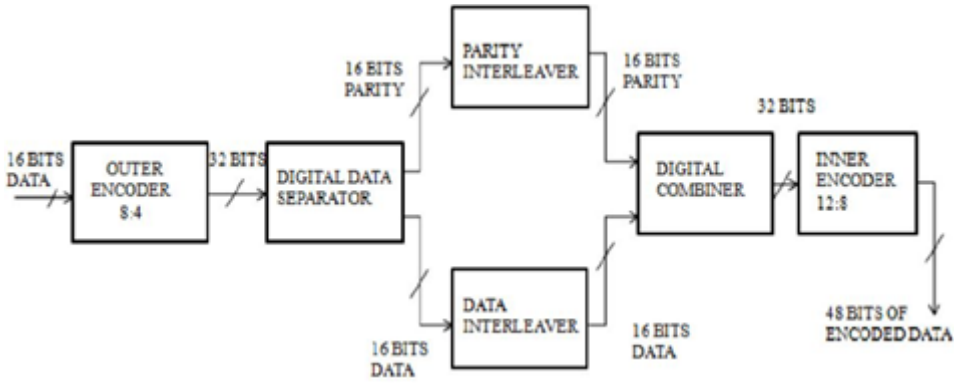


Figure 4: 2D Hamming encoder

Block Diagram



Hamming Encoders are specified as n:k where n is the number of encoded bits and k is the number of message bits. The following are the steps involved.

Step 1: A 16 bit input data is split into 4 sets of 4 bits and are sent to the outer encoder (8:4) block.

Step 2: Outer encoder encodes each 4 bit data to 8 bit (4 data bits and 4 parity bits). Thus the output of outer encoder will be a total of 32 bits (4x8=32).

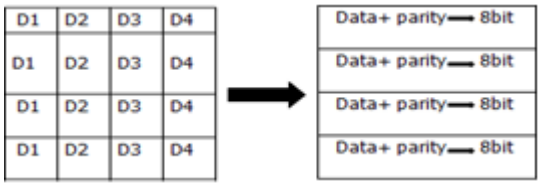


Figure 5: Initial encoding

Step 3: Data and parity in each set is separated. Thus 4 bits of data and 4 bits of parity is obtained from each set.

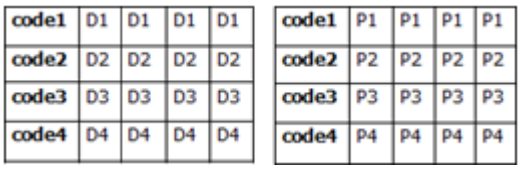


Figure 6: Digital Data Seperator

Step 4: The separated sets of data and parities are interleaved by using data interleaver and parity inter leaver using transpose of a matrix concept. Interleaving is the process for making forward error correction more robust with respect to errors. Interleaving is the method making data retrieval more efficient by rearranging the data.

Step 5: The interleaved data and parity are mixed. The first set of interleaved data and the first set of interleaved parity, the second set of interleaved data and the second set of interleaved parity, the third set of interleaved data and the third set of interleaved parity, the fourth set of interleaved data and the fourth set of interleaved parity.

Step 6: The output of the mixer will be 4 sets of 8 bits (4x8=32).

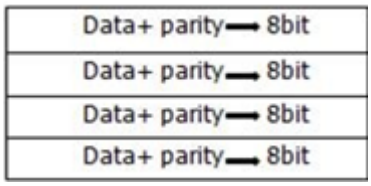


Figure 7: Mixer Output

Step 7: Each 8 bits of data from the mixer are sent to inner encoder (12:8) block;

Step 8: The Inner encoder encodes each set of 8 bits of data to 12 bits (8 bits data and 4 bits parity)

- C1= data [8^7^5^4^2]
- C2= data [8^6^5^3^2]
- C3= data [7^6^5^1]
- C4= data [4^3^2^1]

Step 9: The 4 sets of 12 bits are combined together to form a 48 bit encoder data block. Thus the output of inner encoder will be a total of 48 bits data (12x4=48) . And the final data out from the encodersection with data and parity bits are in this format:
Data out = [C1,C2,data[8],C3,data[7],data[6],data[5] C4,data[4],data[3],data[2],data[1]]

Selective Bit Placement Strategy

The encoder side includes a 12:8 encoder and bit placement block. The encoder block intakes 4 sets of 8 bit data which is in turn encoded into 4 sets of 12 bits data ie, a total of 48 bits. These 4sets of 12 bits data is given to bit placement block which arranges the bits.

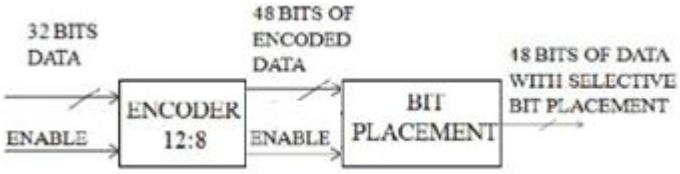


Figure 8: Encoder section

Selective bit placement strategy; it increases the probability of detecting adjacent errors. The objective of this method is to reorder the bits of the code word to maximize the adjacency. Selective bit placement strategy for adjacent error detection for coded data 4×12 is:

Table 5: Selective Bit Placement Strategy

BIT Placement											
1	2	3	4	5	6	7	8	9	10	11	12
1	12	2	3	6	8	7	9	4	10	5	11

Hamming (12, 8) code, the following adjacent error combinations are found: 1–12, 2–12, 3–12, 4–9, 4–10, 4–11, 5–8, 5–10, 5–11, 6–8, 6–9, 6–11, 7–8, 7–9 and 7–10.

The only combination including adjacent bits is 7–8 so only 1 of the 11 adjacent pairs are correctly detected when using the normal order. But by this selective bit placement strategy the detection of error combinations is high than the normal order.

2. Conclusion

Soft error protection on SRAM based FPGA for space application here we proposes efficient Enhanced Hamming codes, it includes 2D Hamming encoder section with selective bit placement strategy and decoder section with lexicographic check matrix for error detection and correction. This system performs reliably and it has advantageous features like faster reconfiguration, increased system throughput, and higher error tolerance.

Faster reconfiguration: The time required to reprogram couple of frames is very short. For instance, partial reconfiguration using I/O blocks takes only few microseconds in commercial FPGAs. Hence, the proposed in situ EC circuit can provide faster reconfiguration using internal bus only.

Higher FPGA system throughput: The read-back can be

performed as a background operation. Hence, the Enhanced Hamming product code performed without stalling the system function. Only the short reconfiguration of erroneous frames results in interruption of the system operation. Hence the system continues with its normal operation.

High error tolerance: The physical dimension of the frame buffer is very small compared to the entire FPGA, and the time required for performing Enhanced Hamming codes will be few micro seconds with internal FPGA clock frequency of 200 MHz. Hence, the probability of having soft errors in the buffer during the maintenance period is negligible.

When compared to other error detection and correction techniques this Enhanced Hamming codes provides very good performance improvement. This technique provides protection to multiple events upsets (MEU). Both column and row access of the data will take place here maximum protection to SRAM based FPGA for mission critical space applications. Simple built-in architecture that provides long survival for FPGAs in mission-critical space applications. The simulation results show that enhanced hamming codes can provide very high rate of error corrections. The time required to reprogram couple of frames is very short. Hence it provides faster configuration. This is performed without stalling the system function so by this the system throughput increases. The physical dimension of the frame buffer is very small compared to the entire FPGA, and the time required to perform can be few micro seconds with internal FPGA clock frequency. Hence, the probability of having soft errors in the buffer during the maintenance period is negligible. We can conclude that the application of enhanced hamming codes is dependable method for soft error protection to SRAM based FPGA especially for mission critical applications.

References

[1] Sang Phill Park, Student Member, IEEE, Dongsoo Lee, and Kaushik Roy,“ Soft-Error- Resilient FPGAs Using Built-In 2-D Hamming Product Code”, IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 20, no. 2, February 2012 .
[2] Sanchez- Macian, P. Reviriego and J. A. Maestro, "Enhanced Detection of Double and Triple Adjacent Errors in Hamming Codes through Selective Bit Placement," IEEE Transactions on Device and Materials Reliability, July 2012.
[3] X. She, N. Li and D.W. Jensen, "SEU Tolerant Memory Using Error Correction Code," IEEE Transactions on Nuclear Science, vol.59, no.1, pp.205-210, February 2012.
[4] R. K. Lawrence and A. T. Kelly, “Single event effect induced multiple-cell upsets in a commercial 90 nm CMOS digital technology,” IEEE Transactions on Nuclear Science, vol. 55, no. 6, pp. 3367–3374, Dec. 2008.
[5] M. Richter, K. Oberlaender, and M. Goessel, “New linear SEC-DED codes with reduced triple bit error miscorrection probability,” in Proc.14th IEEE IOLTS, July 2008, pp. 37–42.
[6] X. She, N. Li and D.W. Jensen, "SEU Tolerant Memory Using Error Correction Code," IEEE Transactions on Nuclear Science, vol.59, no.1, pp.205- 210, February 2012