Implementation of Protection Techniques against Single Event upsets for IIR Filters

R.Selvanathan
Anna University of Technology Coimbatore,
Coimbatore-641 047, India.

K.N.Vijeyakumar
Anna University of Technology Coimbatore,
Coimbatore-641 047, India.

ABSTRACT
Integrated Circuits operating under radiation may be affected by undesirable effects caused by charged particles located in the space environment. Soft errors, especially those produced in harsh environments (as for example radiation), are a major concern for digital circuits. When a particle hits the silicon, it loses its energy and transmits it to the silicon, causing a current burst. In the case of Single Event Upsets (SEUs), these can randomly change the content of storage cells. To protect storage cells of integrated circuits from this phenomenon is by designing circuits that able to detect an SEU event and act accordingly to prevent error propagation and guarantee full reliability in the system. Those based on redundancy, as Triple Modular Redundancy (TMR) and Hamming Codes, are especially popular, since they are quite straightforward to implement. We present new approach using single Hamming encoder and common decoder to protect storage cells present in the IIR filter from this type of Single Event Upsets (SEU). This design is coded using VHDL language and synthesized using Xilinx ISE. The design is compared with shared hamming encoder and decoder design. The implementation result shows that the gate occupancy of proposed design is better compared to conventional protection technique.

Keywords: Digital filters, hamming codes, reliability, single Event Upset (SEU).

1. INTRODUCTION
In some environments, for example Space, radiation sources are abundant. The effects of radiation are a well-known cause of errors in microelectronic circuits [1]. These errors range from temporary failures of the system (which most of the times produce a restart of the operations) to serious and permanent damage of the devices. One type of temporary effects is Single Event Effects (SEEs), that cause changes in the values of flips-flops (SEUs) or combinational logic (SETs) [2].

When a particle hits the silicon, it loses its energy and transmits it to the silicon, causing a current burst. In the case of SEUs, these can randomly change the content of storage cells. To protect storage cells of integrated circuits from this phenomenon, several approaches may be followed. One is by technology hardening of memory cells [3][4] and another one is by designing circuits able to detect an SEU event and act accordingly to prevent error propagation and guarantee full reliability in the system. Triple Modular Redundancy (TMR) and Error Detection and Correction Code (EDAC), like Hamming Code, are examples of such methods.

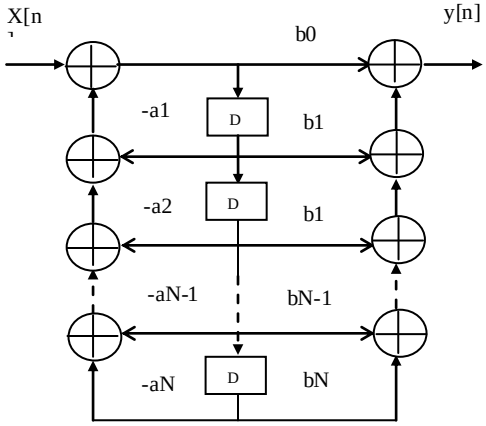


Figure 1. IIR Filter Direct form II realization (M = N)

Hamming codes are a simple class of block codes using the Hamming rule to determine the parity bits based on the number of information bits. This rule is articulated by the inequality given in

$$d + p + 1 \leq 2^p \quad (1)$$

Where d is the number of data bits and p is the number of parity bits. These codes have a minimum distance of three, which describes the number of different bits between two valid codewords, and thus they are capable of correcting all single errors within a block. This capability is defined as Single Error Correction (SEC). Syndrome decoding is especially suited for Hamming codes. In fact, the syndrome can be formed to act as a binary pointer to identify the error location.

In signal processing, the function of a filter is to remove unwanted parts of the signal, such as random noise, or to extract useful parts of the signal, such as the components lying within a certain frequency range. Filters are commonly used in digital communication systems for equalization, signal separation, noise reduction, etc. As communications are fundamental to space borne applications, such as satellites, unmanned missions, etc., digital filters play an important role in space systems [5].

There are two main types of digital filters: the recursive and the non-recursive filters. They are referred as infinite impulse response (IIR) filters and finite impulse response (FIR) filters, respectively. This paper introduces optimizations for the use of Hamming Codes to protect IIR filter, described by (2) & (3) and illustrated in Figure 1.

$$\omega[n] = -\sum_{k=1}^{N} ak.\omega[n-k] + x[n] \quad (2)$$

$$y[n] = \sum_{k=0}^{M} bk.\omega[n \quad (3)$$

Figure 1 shows the direct form II realization (N = M) of a Nth order IIR filter structure, where the input vector x[n] is stored into the first tap D of the delay line, and then shifted through all the taps. In this process, the content of the taps are multiplied with their corresponding coefficient *ak*, *bk* and the sum of these products yields the filter output y[n].

2. RELATED WORKS

As mentioned, TMR and Hamming EDAC are successful methods to protect a design from SEUs. These techniques were discussed in [6] for FIR filters. TMR is the simplest and effective way for protecting a design. The area consumption of this method is obviously up to three times higher, depending on which implementation is chosen. The optimal design of the TMR logic is discussed in [7], where Lima et al. researched TMR logic implementations of a digital FIR filter for FPGAs.

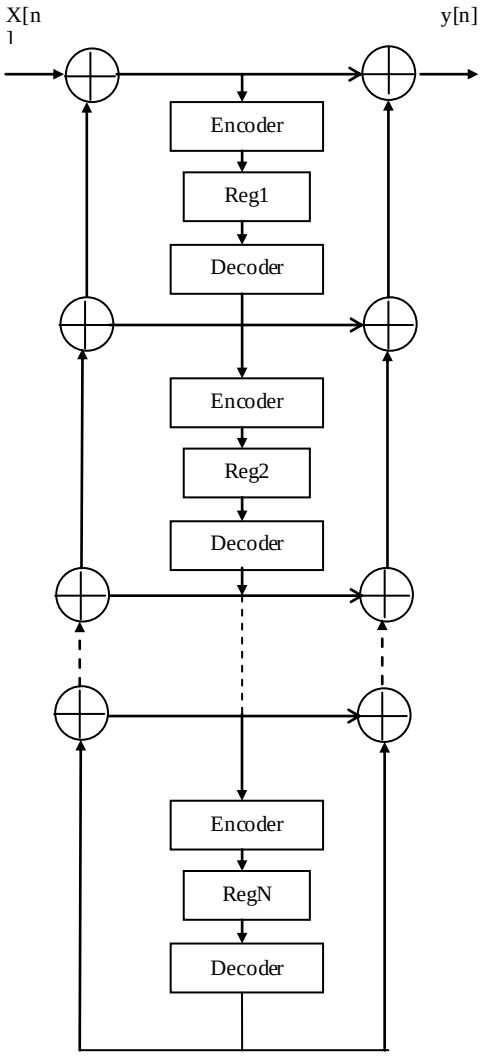


Figure 2. Existing Hamming EDAC protection approach

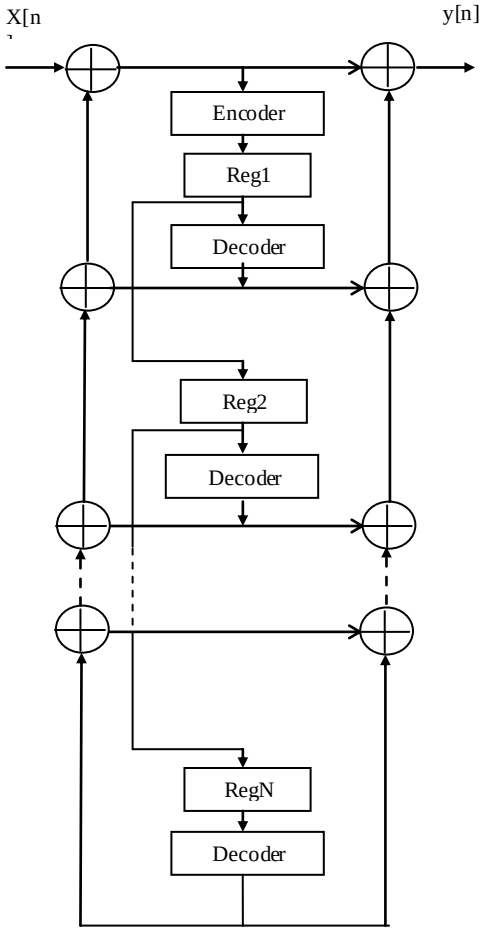


Figure 3. IIR Filter protection with a single encoder

Hamming encoders and decoders perform specific combinational operations on the data in order to generate parity bits (in the case of encoders) and to correct errors (in the case of decoders). A previous approach to protect IIR filters using Hamming codes consists in adding one encoder and one decoder before and after each register that is going to be protected [6], as illustrated in Figure 2.

This way of using EDAC codes to protect the circuit incurs a bigger delay in the critical path with respect to the use of TMR, which will be discussed further in this paper for IIR filters. The area is obviously larger than in the case of the unprotected IIR, as additional registers are needed in each tap plus the combinational logic for the encoders and decoders, but for some implementations

3. PROPOSED WORK

Please In this section, several enhancements are proposed to reduce the number of encoders. These are based on the specific system knowledge of the IIR implementation, an approach that has been previously used to protect other circuits

3.1 Hamming Single Encoder

One of these enhancements would be to remove the encoders from the delay line, as shown in Figure 3, as they are only used if there are errors in the circuit and even in that situation, if only a single error is present in the register, it can still be corrected with the decoder at each stage. In summary, these additional encoders are useful only if we assume that a tap value will be hit by more than one SEU at different time instants, as it propagates through the delay line.

3.2 Additional Hamming Data Protection

A further improvement would be to use the output of each decoder to feed the data bits of the next register while the parity bits are taken directly from the previous stage. This would allow recovering from multiple errors that occur in the data bits as long as they happen in different clock cycles. This is achieved without additional encoders.

3.3 Shared Hamming Decoder

A more sophisticated approach to reduce the complexity is shown in Figure 4, where the Hamming decoder is broken apart yielding a syndrome calculator and an error corrector. The syndrome is calculated through XOR operations of the data and parity bits as shown in Figure 4(a). This should contain only zeros when there are no SEUs. When the data bits contain an SEU, then there will be 1's in the syndrome for identifying the SEU position in the locator, using the syndrome information. The locator sends out an error vector with the exact position of the bit-flip to the corrector. This uses the OR-combined syndrome as an enable to initiate with the received error vector the correction of the faulty bit.

In this way, the locator logic is shared among all taps reducing overall complexity under the assumption that only one SEU per cycle will occur.

4. RESULTS AND DISCUSSION

In the previous sections, some Hamming EDAC protection techniques have been discussed that can be used for IIR filters. In this section, their effectiveness of our proposed technique for actual implementations will be evaluated. To that end, the proposed techniques have been implemented in VHDL and then the circuits have been synthesized for the Xilinx SPARTAN3E.

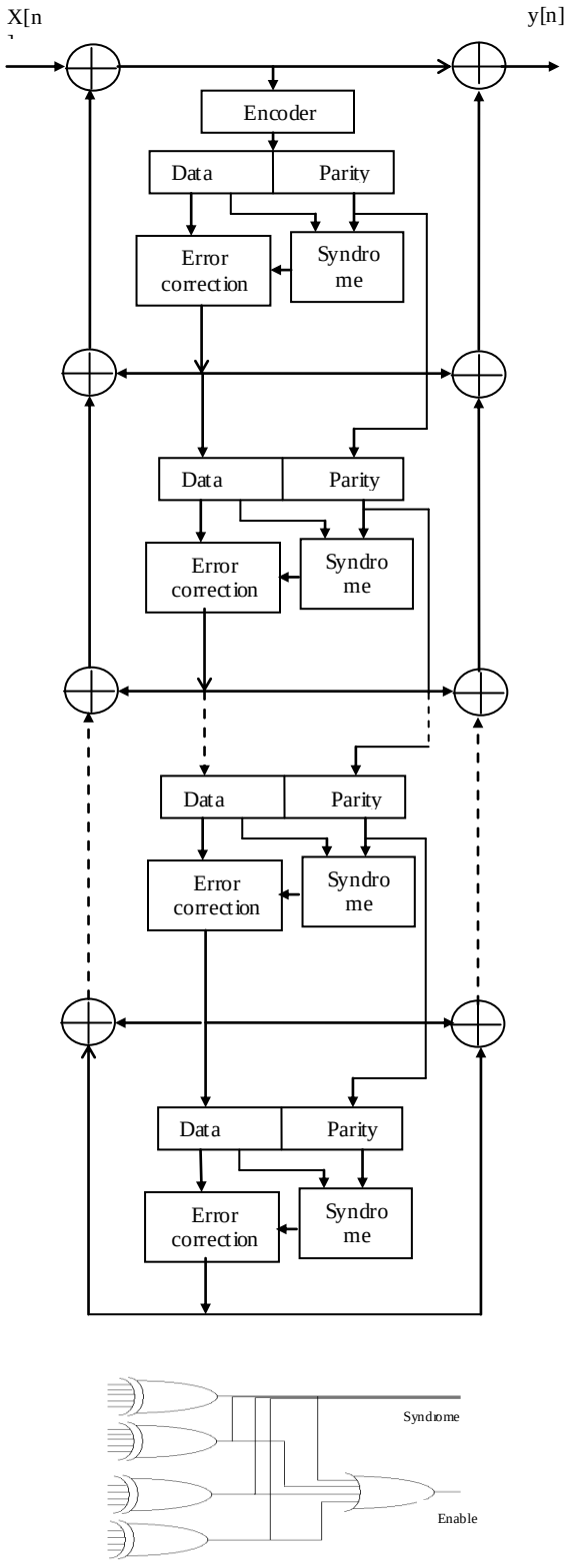
This will also allow assessing the efficiency of the proposed techniques in terms of circuit complexity and compare it with the conventional protection techniques sections.

4.1 Performance Analysis

In Table 1, 2 and 3 shows the FPGA synthesis results for 5, 10, and 15 order IIR filter with traditional and proposed protection technique, highlighting that the proposed single Hamming encoder and decoder technique gives the best performance. Comparing the synthesis results to the one of the ordinary Hamming, it achieves overall slice savings of 20% for 5 taps and a 29% for 10 taps.

4.1.1 Area Analysis

The area cost in equivalent gates and in SPARTAN3E slices of the evaluated protection techniques are provided in Table 1, Table 2 and Table 3, respectively. As it can be seen when comparing the conventional Hamming-based techniques, the proposed single hamming encoder and common decoder technique (Figure 4) offers the most competitive results, considering the combination of protection effectiveness and the total number of gates.



(a)

Figure 4. IIR with Hamming using one encoder and a common decoder; (a) Syndrome circuit

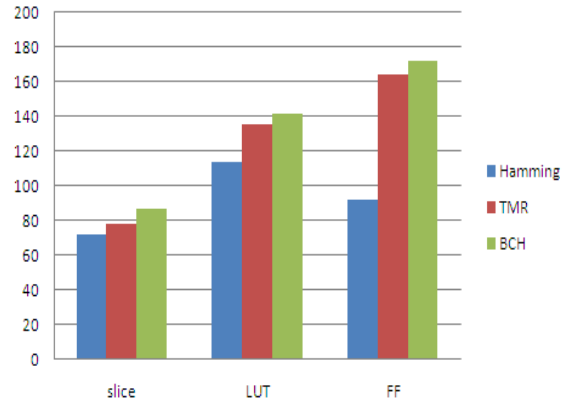


Figure 5. Comparison of Area occupancy between different protection techniques

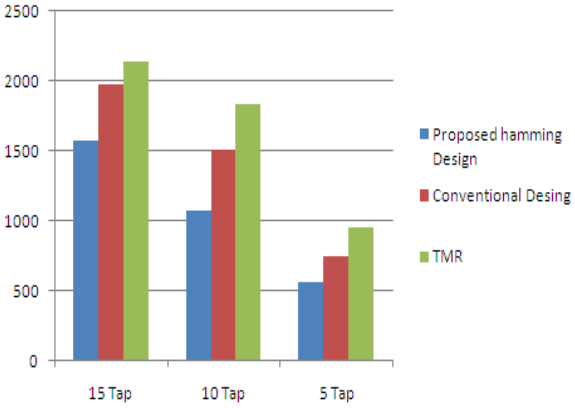


Figure 6. Slice comparison of proposed technique and conventional technique with TMR

The calculation of the area relation in percentage has been done in reference to the standard Hamming implementation. It can be seen that for 5, 10 and 15 taps, the proposed Hamming technique version uses around 20% less area in comparison to the standard Hamming version. This reduction is even larger when increasing the filter taps up to 10 with 29% in comparison to standard Hamming. Figure 5 shows that the area occupancy of hamming code compared to TMR (Triple Modular Redundancy) and BCH codes is less. So that the hamming based protection techniques for digital filter can be suitable candidate for low frequency application.

4.1.2 Frequency Analysis

In the previous subsection, the area cost in equivalent gates and in SPARTAN3E slices of the evaluated protection techniques were in comparison to the ordinary Hamming and TMR. The comparison shows that the Hamming single decoder technique is the most competitive in respect of area costs in a FPGA implementation.

Although the proposed Hamming shared decoder achieves the most competitive area cost results, it also suffers from a maximum clock frequency penalty. The impact of the penalty depends on the target architecture (FPGAs).

Table 1
FPGA Synthesis report for IIR Hamming 15 order filter

DESIGN	Slice	LUT	Freq. (MHz)	Power(mW)
Proposed Design	1580	2774	400.962	218.49
Conventional design	1973	3468	488.520	240.67
TMR	2137	3859	500.46	241.75

Table 2
FPGA Synthesis report for IIR Hamming 10 order filter

DESIGN	Slice	LUT	Freq. (MHz)	Power(mW)
Proposed Design	1075	1890	400.962	217.17
Conventional design	1509	2642	461.255	232.55
TMR	1837	2985	486.854	232.98

Table 3
FPGA Synthesis report for IIR Hamming 5 order filter

DESIGN	Slice	LUT	Freq. (MHz)	Power(mW)
Proposed Design	561	987	400.962	215.62
Conventional design	747	1308	461.255	230.73
TMR	957	1508	483.852	231.41

This penalty is produced when extra logic is added to the design, increasing the critical path and therefore, reducing the maximum clock frequency that is allowed. This is shown in Table 1, 2 and 3, comparing the results of the conventional hamming FIR filter with 461.255 MHz and the Hamming single decoder technique with 400 MHz The difference between the ordinary Hamming and the Hamming single decoder is minimal in comparison to the achieved area cost reduction.

5. CONCLUSION

A new approach for protecting the IIR filters from single event upsets (SEUs) are implemented in the design. We have focused on achieving low complexity and low gate occupancy with improved speed. The simulation results of the proposed designs are encouraging in terms of gate count and power efficiency. The implementation result shows that the proposed design is suitable candidate to protect IIR filters from single event upsets. So that, the hamming based protection techniques for digital filters can be suitable candidate for low frequency applications.

Future work includes the research of FPGA-oriented solutions for fault tolerant digital filters and the consideration of power consumption as a metric for optimization, since it is a key factor in space applications.

6. REFERENCES

[1] R. D. Schrimpf and D. M. Fleetwood, 2004, Radiation Effects and Soft Errors in Integrated Circuits and Electronic Devices. Singapore:World Scientific, 981-238-940-7.

[2] P. E. Dodd and L. L. Massengill, Jun. 2003, "Basic mechanisms and modelling of single-event upset in digital microelectronics," IEEE Trans. Nucl. Sci., vol. 50, no. 3, pp. 583–602.

[3] P. E. Dodd and L. L. Massengill, Jun. 2003, "Basic mechanisms and modeling of single-event upset in digital microelectronics," IEEE Trans. Nucl. Sci.,vol. 50, no. 3, pp. 583–602.

[4] M. P. Baze, S. P. Buchner, and D. Mcmorrow, Dec. 2000, "A digital CMOS technique for SEU hardening," IEEE Trans. Nucl. Sci., vol. 47, no. 6, pp.2603–2608.

[5] S. M. Parkes, Sept. 1998, "DSP (Demanding Space-based Processing!): The path behind and the road ahead," in Proc. 6th Int. Workshop Digital Signal Processing Techniques for Space Applications, Noordwijk, The Netherlands.

[6] R. Hentschke, F. Marques, F. Lima, L. Carro, A. Susin, and R. Reis, 2002 "Analyzing area and performance penalty of protecting different digital modules with Hamming code and triple modular redundancy," in Proc. 15th Symp. Integrated Circuits and Systems Design, pp. 95–100.

[7] F. Lima Kastensmidt, L. Sterpone, L. Carro, and M. Sonza Reorda, 2005, "On the optimal design of triple modular redundancy logic for SRAM-based FPGAs," in Proc. Conf. Design, Automation and Test in Europe, vol. 2, pp. 1290–1295.