RESEARCH　　　　　　　　　　　　　　　　　　　　　　　　　　**Open Access**

Residue code based low cost SEU-tolerant fir filter design for OBP satellite communication systems

Wenhui Yang[1], Zhen Gao[2], Xiang Chen[3*], Ming Zhao[2] and Jing Wang[2]

Abstract

With the development of satellite communications, on-board processing (OBP) obtains more and more attentions due to the increased efficiency and performance. However, the large amounts of digital circuits in the OBP transponders are sensitive to the high-energy particles in space radiation environments, which may cause various kinds of single event effect. Among these effects, single event upset (SEU) is the major potential reason for the instability of the satellite communication systems. Triple modular redundancy (TMR) is a classical and effective method for mitigating the SEU in digital circuits. However, since three identical logic modules and a voting circuit are needed in TMR, the overhead is so high that the scheme may not be applicable on the on-board digital processing platform with very limited area and power resources. Therefore, how to design a more cost-effective fault-tolerant method becomes a critical issue. Considering that FIR-like processing is frequently used on OBP platform, in this article, a dual modules (DM) plus checking module based on residue code (DM-CRC) architecture for SEU-tolerant FIR design is proposed. Although this architecture reduces the area overhead dramatically, we find that the fault missing rate is still high if single-sample checking (SSC) is used. To solve this problem, a Multi-sample checking DM-CRC (MSC-DM-CRC) is further proposed. Our analysis shows that the MSC-DM-CRC scheme can make the fault missing rate small enough without reducing the actual throughput. By simulations it is shown that, when the modulus for CRC is 7 and the number of samples for MSC is 4, the reduction of area overhead relative to TMR is over 20% and the fault missing rate is as low as 0.05%.

1 Introduction

Traditional bent pipe (BP) satellite performs only signal amplification and frequency translation. However, with the development of satellite communication applications, the demand for communication quality and capacity has increased so rapidly that BP transponders cannot afford. Instead, the on-board processing (OBP) becomes the inevitable alternative [1]. The goals of OBP are to provide single-hop connectivity to small earth station, and to enhance link performance and efficiency [2]. Compared to the two-hop connection in BP systems, single-hop connection decreases the one-way transmission delay from 540 to 270 ms, which provides more comfortable user experience. The enhanced performance

by OBP can be used to reduce the cost of small earth stations [3] or increase the system capacity.

The OBP directly related to communications can be roughly divided into two classes: intermediate/radio frequency (IF/RF) processing and switching and baseband OBP [2,3]. The IF/RF processing and switching is actually the digital channelization technique [4-6]. As an intermediate during the development from BP towards the software defined fully processed payload, digital channelization realizes the de-multiplexing, switching and multiplexing of sub-channels, without fully decoding of the information in the sub-channels [7]. Since digital channelization maintains the transparence to the physical layer transmission techniques as BP, and provides much better performance over BP, it is widely used by many successful mobile satellite communication systems, including ACeS [8], Thuraya [9], WGS [10,11] and MOUS [12,13]. In consideration of the necessity of FIR filters and FFT for digital channelization [14,15], they are considered as

* Correspondence: chenxiang98@mails.tsinghua.edu.cn
[3]Aerospace Center, School of Aerospace, Tsinghua University, Beijing, China
Full list of author information is available at the end of the article

the most basic and important modules on digital channelizing OBP.

Baseband OBP is commonly referred as fully processing satellite platform, so all the physical layer techniques, including demodulation/modulation, decoding/encoding and channel estimation/equalization, should be performed for the signal regeneration for each subchannel [3,14]. The Thuraya system is a good example of current satellite communication systems with baseband OBP platform [9]. Since complete switching is required in baseband OBP, FIR and FFT are still the basic modules. As introduced in [3,16,17], the orthogonal frequency division multiplexing (OFDM) technology is a very attractive candidate when targeting high quality and high flexibility in future mobile multimedia satellite communications systems, so FFT is still a necessary module. In addition, almost all the current mobile satellite communication systems (e.g. ACeS, Thuraya, WGS, MOUS, and so on) and future ones, no matter digital channelization or fully processed baseband OBP, would apply digital beamforming (DBF) for multiple-beam coverage [8-13], so DBF is also a necessary DSP module on OBP platforms [18,19].

SRAM-FPGA is a good option for OBP implementation because of its high density, high performance, reduced development cost and re-configurability, the last of which is quite useful for remote update and maintenance of the OBP satellite systems [20]. However, SRAM-FPGA based systems, including memories and logics, are sensitive to the radiation in space environments, so they are not reliable enough for spatial applications without any protection. single event upset (SEU) is one of the main radiation effects, and induces the majority of the function faults on OBP platform [21]. Therefore, the SEU-tolerant scheme is the key issue for the feasibility of the SRAM-FPGA based applications on OBP platforms.

As a classical fault-tolerant solution, triple modular redundancy (TMR) applies three identical modules to perform the same process, and the results are processed by a majority voter to produce a single output. So if only one of the three modules fails, the other two can help to mask the faults [22]. However, TMR introduces tripled space, weight and power, which result in impossibility to implement on the satellite platforms where resources are limited. For example, space based radar requires 100's of giga floating point operation's of OBP and 10's of Gbps data links to accomplish mission goals [23]. A TMR approach for such a program would create a system that weighs hundreds of pounds and requires kilowatts of power, which are unbearable for an OBP platform [23]. Thus, for on-board applications, the low-cost fault-tolerant designing is in demand.

The focus of this article is the low cost SEU-tolerant design for the DSPs with the structure expressed as

$$\gamma = \sum_{l=0}^{L-1} x(l) * h(l), \tag{1}$$

in which $x(l)$ and $h(l)$, $l = 0, 1, ..., L - 1$ are input data for the current operation and coefficients, respectively. Since only multiplications and additions are involved, this structure is called multiply and accumulation (MAC). The reason we choose MAC as our target is that it is the general structure of the common used DSPs in OBP, such as FIR, FFT and DBF. For FIR, $h(l)$ is the filter coefficients. For FFT, $h(l)$ is the rotation factor. And for DBF, $h(l)$ is the weighting coefficient. To facilitate the description and analysis, this article will focus on the SEU-tolerant FIR design. The analysis method and theoretical results can be easily applied to other DSPs with MAC structure.

Our key contributions in this article include three points:

(1) A simple DM plus checking module based on residue code (DM-CRC) structure is proposed to reduce the heavy cost of traditional TMR method;
(2) The fault missing problem of general residue code based checking module is revealed;
(3) Finally, we propose a multi-sample checking (MSC) solution to decrease the fault missing rate.

The rest of this article is organized as follows. In Section 2, related work and mathematical background are given. In Section 3, the single-sample checking based DM-CRC (SSC-DM-CRC) fault-tolerant scheme is proposed, and its fault missing rate analysis for the modulus with different form is given. Section 4 introduces the MSC based DM-CRC(MSC-DM-CRC) scheme to reduce the missing rate of the SSC-DM-CRC. Simulation results to validate our theoretical analysis are given in Section 5. Section 6 concludes this article.

2 Related work and mathematical background

Many fault-tolerant schemes based on residue code have been proposed to reduce the overhead of TMR for FIR design. Before introducing these schemes, the property of residue code and residue number system (RNS) is introduced firstly.

Residue code is applied by a recomputation of the remainders of the division of the operands by a given number [20]. The interesting character of the residue code is that, it maintains the arithmetical and logical properties of operand and invariant for linear operations, including additions/substractions and multiplications. It means that, for the given operands X and Y, the following equation always holds.

$$(X \text{ op } Y)_m = ((X)_m \text{ op } (Y)_m)_m, \tag{2}$$

where m is the modulus and op represents the linear operator.

Reference [24] gives the details of RNS applied to fixed-coefficient inner product computation. The RNS is defined by a set of p pairwise relatively prime integers, $\{m_1, m_2,..., m_p\}$. The dynamic range of RNS is given by: $M = m_1 * m_2 * ... * m_p$. Then an integer $X \in [0, M)$ can be uniquely expressed in RNS as

$$X \xrightarrow{\text{RNS}} \{(X)_{m_1}, (X)_{m_2}, \ldots, (X)_{m_p}\}, \tag{3}$$

where $(X)_{m_i}$ means $X \bmod m_i$. In RNS, linear operation between X and Y, including additions, subtractions and multiplications in two's complement system, can be transformed to a set of operations on residues as

$$Z = X \text{ op } Y \rightarrow \begin{cases} (Z)_{m_1} = ((X)_{m_1} \text{ op } (Y)_{m_1})_{m_1} \\ (Z)_{m_2} = ((X)_{m_2} \text{ op } (Y)_{m_2})_{m_2} \\ \vdots \\ (Z)_{m_p} = ((X)_{m_p} \text{ op } (Y)_{m_p})_{m_p}, \end{cases} \tag{4}$$

the result Z can be recovered based on $(Z)_{m_i}$ by Chinese remainder theorem (CRT) as [25]

$$Z = \text{CRT}\{(Z)_{m_1}, (Z)_{m_2}, \ldots, (Z)_{m_p}\}. \tag{5}$$

Based on RNS, a scheme named redundant RNS (RRNS) [26,27] is proposed for fault tolerant of single branch. In RRNS, redundant branches with new modulus are added to the original RNS system. With the dynamic range maintained, if one of the branches fails because of SEU, the correct result can still be recovered from other branches.

Since only redundant branches with simplified computations are added, the SEU-tolerant overhead of RRNS is much smaller than that of TMR. The problem of RRNS is that several CRT modulus are added for fault detection of parallel computation branches, but the CRT modulus themselves are not protected from SEU. If TMR is applied for CRT, the advantage of low overhead will disappear.

To avoid the heavy implementation cost of multiple CRT modules and the corresponding protections, several researches are proposed to apply residue code just for checking based on Equation (2). One of such schemes is the duplication with comparison combined with concurrent error detection (DWC-CED) proposed in [28]. As shown in Figure 1, the structure duplicates the normal logic module, and a CED module is added to each logic module for self-checking. As introduced in [28], the CED can be based on time redundancy or hardware redundancy. Considering that the time redundancy based CED introduces extra delay, which will degrade the throughput performance of the system, the hardware redundancy based CED is preferable, and the most common choice for hardware redundancy is residue code. However, fault missing issues exist in the residue code based DWC-CED. For example, the modulus is 7 and the correct result of the logic block and the CED block should 72 and 2, respectively, at some moment. If there is no fault, we have $(72)_7 = 2$, which is equal to the result of the CED block. But if a fault occurs, and changes the result of the logic block from 72 to 79, we will have $(79)_7 = 2$, which is still equal to the result of the CED block. In this situation, the fault is unidentified. The probability of the unidentified fault is called fault missing rate in this article.
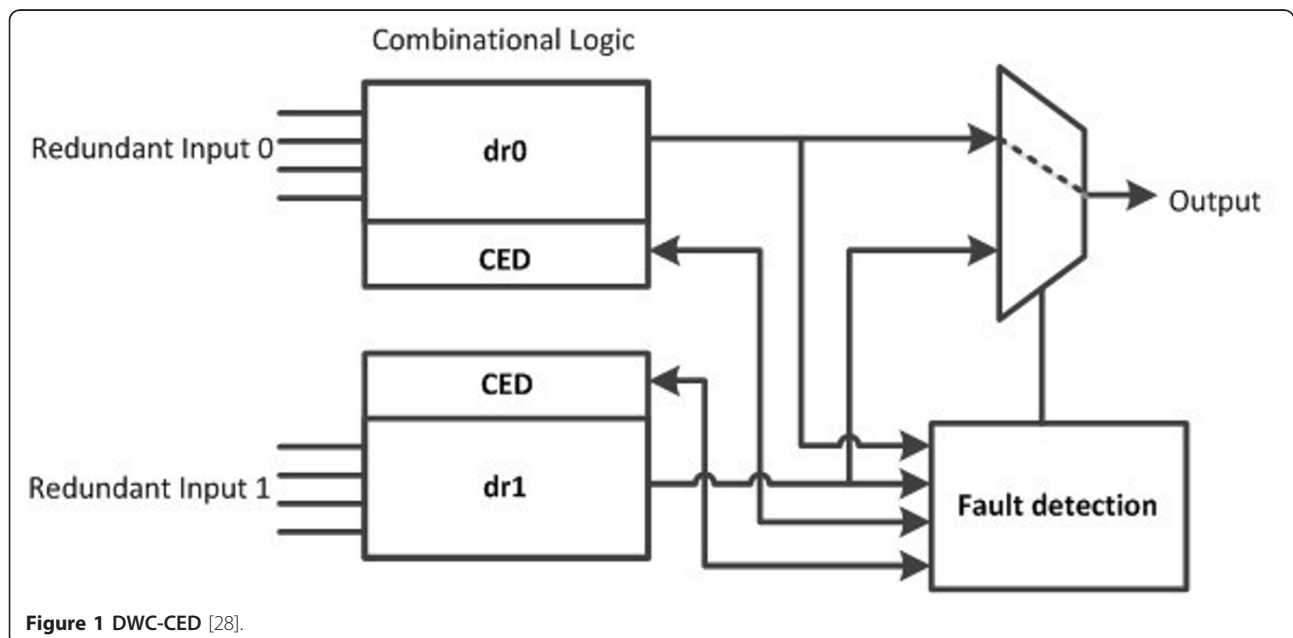

Figure 1 DWC-CED [28].

Another residue code based scheme, FIR plus two checking modules based on residue code (CRC), was proposed by [29]. As shown in Figure 2, this scheme includes one normal FIR module and two checking modules of residue code based FIR. Y_C is the correct output and Y is the faulty output of the main FIR filter, and $Y = Y_C + E$. In [29], E is the possible arithmetic errors modeled as $2^{\pm j}$, where $j = 0, 1, ..., \omega_y - 1$, ($\omega_y$ is the length of Y). V_{A_1} and V_{A_2} are the outputs of the checking modules, based on which Ψ_{A_1} and Ψ_{A_1} are defined as $\Psi_{A_1} = \left\| Y - V_{A_1} \right\|_{A_1} = \left\| E \right\|_{A_1}$ and $\Psi_{A_2} = \left\| Y - V_{A_2} \right\|_{A_2} = \left\| E \right\|_{A_2}$. Then, $\{\Psi_{A_1}, \Psi_{A_1}\} \neq 0$ represents the occurrence of an error. Based on Table 1, single bit error can be located, and corrected by a syndrome-to-error mapping logic and a conventional subtractor.

But there are three important issues neglected in [29]. Firstly, the correct logic is only effecitve for single bit error in the outputs. Actually, this kind of errors occupies only a small part of the fault models. When single error occurs in the intermediate data, multiple bits may upset in the output. For example, for binary computation 11 * 1000 = 11000, when "1000" turns into "1001" by SEU, we obtain 11 * 1001 = 11011. Obviously, two bits are upset in the result, which cannot be corrected by the correction method proposed in [29]. Secondly, the case that an error occurs in the checking module is

Table 1 One-to-one mapping of arithmetic error and syndromes for $A_1 = 7$ $A_2 = 15$ [29]

j	$\{\| 2^j \|_7\}, \{\| 2^j \|_{15}\}$	$\{\| -2^j \|_7\}, \{\| -2^j \|_{15}\}$
0	(1,1)	(6, 14)
1	(2,2)	(5, 13)
2	(4,4)	(3, 11)
3	(1,8)	(6, 7)
4	(2,1)	(5, 14)
5	(4,2)	(3, 13)
6	(1,4)	(6, 11)
7	(2,8)	(5, 7)
8	(4,1)	(3, 14)
9	(1,2)	(6, 13)
10	(2,4)	(5, 11)
11	(4,8)	(3, 7)

not taken into account in [29]. What's more, the fault missing problems are also ignored in [29].

It should be noticed that, all the fault-tolerant designs mentioned above behave in such a way that a decision is made for each output. This approach is called SSC in this article.

3 SSC based DM-CRC

Aiming at the problems of the two residue code based schemes mentioned in Section 2, a new residue code based fault-tolerant architecture is proposed in this section, and the fault missing rate is analyzed in detail.

3.1 Architecture and working procedure of SSC based DM-CRC

As shown in Figure 3, the proposed architecture includes DM of normal FIR filters, denoted as M_1 and M_2, and one error DM-CRC, denoted as CRC. The outputs of the three modules are processed by the synthesis logic, which is expected to output the correct result if one of the three branches is failed by SEU. In this section, we suppose that the synthesis logic works according to the usual SSC approach, and a more effective approach will be introduced in next section. Compared with the DWC-CED scheme, the proposed architecture combines the two CED modules into one CRC module, which simplifies the whole system structure. In addition, the working procedure introduced below will show that
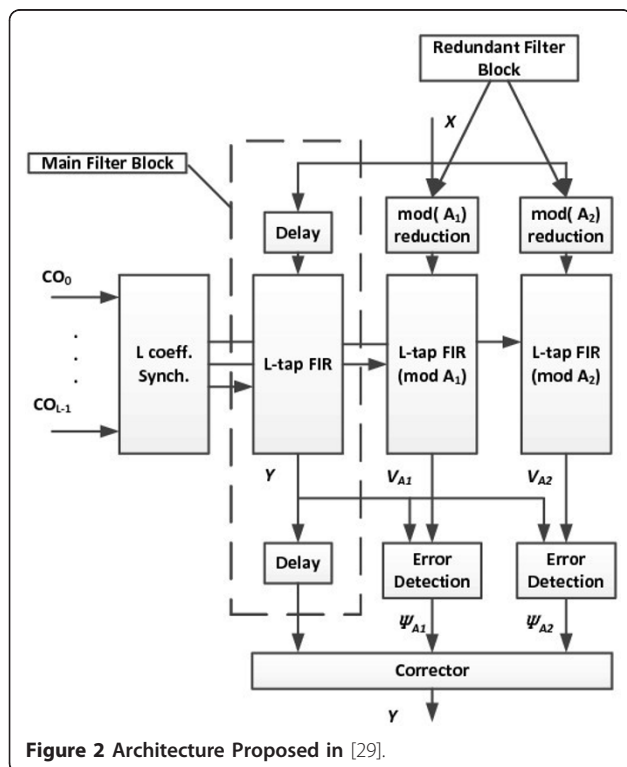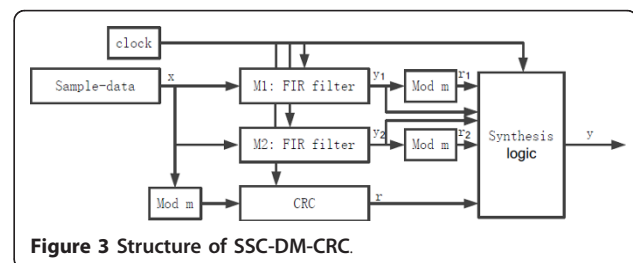

Figure 2 Architecture Proposed in [29].

Figure 3 Structure of SSC-DM-CRC.

the proposed structure also takes the SEU in CRC module into account without any simplified fault model, so the problems of the design of "FIR plus two CRC modules" is overcome.

Suppose the same input data stream is feed to the three modules, and the outputs of the two normal FIR modules and the checking module at some moment are y_1, y_2 and r, respectively. The working procedure for the SSC based synthesis logic is shown in Figure 4, and further explained as follows:

(1) If $y_1 = y_2$, y_1 is chosen as the output $y = y_1$. At the same time, $(y_1)_m$ is used to check whether CRC module runs correctly. Two cases may be met here:

(a) $(y_1)_m = r$: the checking module runs correctly and no action is taken;
(b) $(y_1)_m \neq r$: an SEU occurs in the checking module, so it should enter the recovery process.

(2) If $y_1 \neq y_2$, r is used to check $(y_1)_m$ and $(y_2)_m$ according to Equation (2). Four cases may be met here:

(a) $(y_1)_m = r$ and $(y_2)_m \neq r$: y_1 is output and M_2 enters the recovery process;

(b) $(y_1)_m \neq r$ and $(y_2)_m = r$: y_2 is output and M_1 enters the recovery process;
(c) $(y_1)_m = r$ and $(y_2)_m = r$: an error happened, but the checking module cannot identify it, so y_1 or y_2 is chosen randomly as the output. In this case y_1 and y_2 are called congruent samples.
(d) $(y_1)_m \neq r$ and $(y_2)_m \neq r$: it means more than one branches fail. This case is out of the range of this article.

The recovery processing mentioned in (1)-b, (2)-a and b is used to avoid SEU accumulation in the system, and can be accomplished by scrubbing (or reconfiguration) [30]. (2)-c is actually the fault missing event, which is the key problem for residue code based checking and has been ignored in DWC-CED and "FIR plus two CRCs". As mentioned above, an SEU in SRAM-FPGA may not only change the data stored in a memory unit, but may also change the function logic. In this article, we assume that the result of logic damage is always so serious that (2)-b will happen, so the fault missing event is only caused by SEU in a memory unit. In the following sub-sections, this fault missing rate for L-tap FIR filters is analyzed.
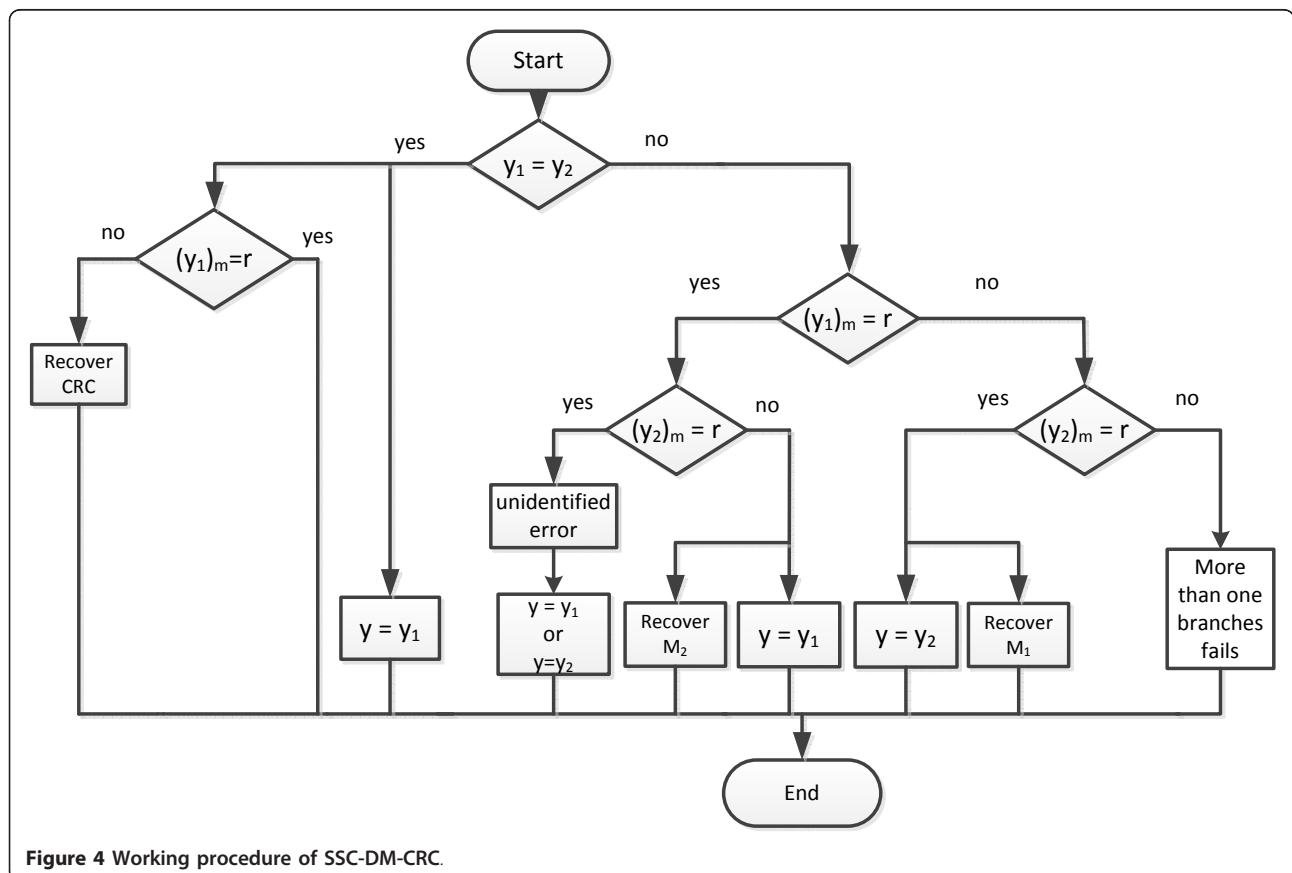

Figure 4 Working procedure of SSC-DM-CRC.

3.2 Analysis of the fault missing rate for SSC-DM-CRC
3.2.1 Fault models
In this section, only the basic FIR filters composed of MAC units are considered for the analysis of fault missing rate. Before studying on the performance of SEU-tolerant systems, the SEU fault models are established firstly in this subsection.

For L-tap FIR filters, the output at some moment can be expressed as

$$y = x_0 * h_0 + x_1 * h_1 + \cdots + x_{L-1} * h_{L-1}, \qquad (6)$$

where x_l is the lth input, h_l is the lth coefficients, $l = 0, 1, \ldots, L - 1$. Since FIR filters are composed of MAC units, the fault model can be classified into multiplier-input fault (MIF) model and adder-input fault (AIF) model.

x_l and h_l are peer-to-peer in Equation (6), hence, the effect of SEU on x_l is equivalent to that on h_l for MIF in SSC-DM-CRC system. In this section, the fault missing rate analysis for MIF only focuses on the SEU in h_l, and the analytical result is also suitable for the SEU on x_l. Assuming h_l is changed to $h'_l = h_l \pm 2^q$, where q is the position of the upset bit, then the faulty filter output for MIF can be expressed as

$$
\begin{aligned}
y_f &= x_0 * h_0 + x_1 * h_1 + x_2 * h_2 + \cdots + x_l * h'_l + \cdots + x_{L-1} * h_{L-1} \\
&= x_0 * h_0 + x_1 * h_1 + x_2 * h_2 + \cdots + x_l * (h_l \pm 2^q) + \cdots + x_{L-1} * h_{L-1} \\
&= x_0 * h_0 + x_1 * h_1 + x_2 * h_2 + \cdots + x_l * h_l + \cdots + x_{L-1} * h_{L-1} \pm x_l * 2^q.
\end{aligned} \qquad (7)
$$

The AIF is equivalent to introduce a SEU in the final output, so the faulty output for AIF can be expressed as

$$y_f = x_0 * h_0 + x_1 * h_1 + x_2 * h_2 + \cdots + x_{L-1} * h_{L-1} \pm 2^q, \quad (8)$$

Equations (7) and (8) will be used for fault missing rate analysis in following subsections.

3.2.2 Computation of fault missing rate for MIF model
Assuming that an SEU on the qth bit of h_l in M_1 branch leads to an unidentifiable error (event of (2)-c in Section 3.1.1), we have

$$
\begin{cases}
y_1 = y_f, \\
y_2 = y.
\end{cases} \qquad (9)
$$

Substituting Equations (6) and (7) into Equation (9), we get

$$(y_1 - y_2)_m = \pm(x_l * 2^q)_m = 0, \qquad (10)$$

so the fault missing rate for the SSC-DM-CRC with module m, which is the probability of the happening of Equation (10), can be expressed as

$$P_{m\_\mathrm{MIF}} = \mathrm{Prob}\big((x_l * 2^q)_m = 0\big). \qquad (11)$$

The cases leading to $(x_l * 2^q)_m = 0$ include the following three cases:

- **Case A:** $(x_l)_m = 0$;
- **Case B:** $(2^q)_m = 0$, $(x_l)_m \neq 0$;
- **Case C:** $(x_l)_m \neq 0$, $(2^q)_m \neq 0$, but $(x_l)_m * (2^q)_m = km$, where k is a natural number.

For $(x_l)_m$, we proved by simulation (not given in this article due to page limited) that, for random inputs x_l with Gaussian, Rayleigh or Rician distribution, $(x_l)_m$ is uniformly distributed between 0 and $m - 1$, or $(x_l)_m \sim U[0, m-1]$.

The analysis of Equation (11) is directly related to the value of m. According to [31], due to low complexity for implementation, 2^n and $2^n \pm 1$ are the most common choice for m, so this subsection will focus on the analysis of P_{m_MIF} when $m = 2^n$ or $2^n \pm 1$, respectively.

For $m = 2^n$, P_{m_MIF} is related to the SEU position q

(1) If $q \geq n$, $(2^q)_m = 0$ is always true, so Case B causes $P_{m\_\mathrm{MIF}} = 1$, which means the SEU on a bit that is higher than n will never be identified for $m = 2^n$.

(2) If $q < n$, $(2^q)_m \neq 0$ always holds, so Cases A and C are considered. For Case A, $(x_l)_m$ is always uniform distributed on $[0, m-1]$, so $\mathrm{Prob}((x_l)_m = 0) = \frac{1}{m} = \frac{1}{2^n}$. For Case C, as $2^q < m = 2^n$, we have $(2^q)_m = 2^q$. If $(x_l)_m * (2^q)_m = km$ ($k \in N$), we obtain $(x_l)_m = k * 2^{n-q}$. As mentioned above, $(x_l)_m \in [1, 2^n - 1]$, thus, the number of value of $(x_l)_m$ that makes $(x_l)_m * (2^q)_m = km$, or the number of $k \in N$ satisfying $(x_l)_m * (2^q)_m = km$, should be $\left\lfloor \frac{2^n - 1}{2^{n-q}} \right\rfloor = 2^q - 1$. For example, when $n = 4(m = 16)$, if $q = 1$, $(2^q)_m = 2$, so only $(x_l)_m = 8$ can make $(x_l)_m * (2^q)_m = 16$ for k = 1; if $q = 2$, $(2^q)_m = 4$, $(x_l)_m = 4, 8$ and 12 can make $(x_l)_m * (2^q)_m = 16, 32$ and 48 for $k = 1, 2$ and 3, respectively. Since the probability for each value of $(x_l)_m = k * 2^{n-q}$ is $\frac{1}{2^n}$, the fault missing rate for Case C can be calculated as $(2^q - 1)/2^n$. Actually, if we consider Case A and a special case of Case C for $k = 0$, the combined case becomes $(2^q)_m \neq 0$, $(x_l)_m * (2^q)_m = km$, $k \in Z$, $k \geq 0$, and it's easy to find out that the fault missing rate for this case is $2^q/2^n$ for $q < n$. In summary, for $m = 2^n$, the fault missing rate under MIF model can be expressed as

$$
P_{m\_\mathrm{MIF}}^{2^n} =
\begin{cases}
\dfrac{2^q}{2^n}, & q < n; \\
1, & q \geq n.
\end{cases} \qquad (12)
$$

For $m = 2^n \pm 1$, $(2^q)_m \neq 0$ always holds, so Cases A and C are considered. For Case A, since $(x_l)_m \sim U[0, m-1]$, then $\mathrm{Prob}((x_l)_m = 0) = \frac{1}{m}$. For Case C, there is no natural value k satisfying $(x_l)_m * (2^q)_m = km$. This is because:

(1) If $q \leq n$, we have $(2^q)_m = 2^q$, so the value of k for $(x_l)_m * (2^q)_m = km$ should be $(x_l)_m * 2^q / (2^n \pm 1)$, which is not an integer since the denominator cannot be the factor of the numerator;

(2) If $q > n$, we have $(2^q)_m = 2^q - p(2^n \pm 1)$, where p is a natural number, so the k for $(x_l)_m * (2^q)_m = km$ is $\frac{(x_l)_m * (2^q - p(2^n \pm 1))}{2^n \pm 1} = \frac{(x_l)_m * 2^q}{2^n \pm 1} - (x_l)_m p$, which is

neither an integer since $\frac{(x_l)_m * 2^q}{2^n \pm 1}$ is not an integer.

Finally, we have $\mathrm{Prob}((x_l)_m * (2^q)_m = km) = 0$ for non-zero $(2^q)_m$ and $(x_l)_m$. As a result, the fault missing rate for a modulus $m = 2^n \pm 1$ is

$$P_{m\_\mathrm{MIF}}^{2^n+1} = \frac{1}{m} = \frac{1}{2^n \pm 1}. \qquad (13)$$

3.2.3 Computation of fault missing rate for the AIF model

Now we consider that an SEU on the qth bit of the adder input in M_1 branch leads to an unidentifiable error (event of (2)-c in Section 3.1.1). Substituting Equations (6) and (8) into Equation (9), we have the fault missing rate for the SSC-DM-CRC with module m expressed as

$$P_{m\_\mathrm{AIF}} = \mathrm{Prob}((2^q)_m = 0). \qquad (14)$$

According to (14), for $m = 2^n$, when $q < n$, $(2^q)_m = 2^q$, so $P_{m\_\mathrm{AIF}}^{2^n} = \mathrm{Prob}((2^q)_m = 0) = 0$. Otherwise, when $q \geq n$, $(2^q)_m = 0$ always holds, so $P_{m\_\mathrm{AIF}}^{2^n} = \mathrm{Prob}((2^q)_m = 0) = 1$.

Furthermore, for $m = 2^n \pm 1$, $(2^q)_m \neq 0$, so $P_{m\_\mathrm{AIF}}^{2^n+1} = 0$.

Based on the above analysis, although the implementation of CRC with $m = 2^n$ is the simplest, the fault missing rate is much higher than that with $m = 2^n \pm 1$ for both fault models, so we conclude that $m = 2^n$ is not suitable for CRC. On the other hand, the CRC with $m = 2^n \pm 1$ achieves much smaller fault missing rate with negligible implementation overhead. So the analysis in Section 4 will focus on the CRC with $m = 2^n \pm 1$.

4 MSC based DM-CRC

This section proposes a solution to the fault missing problem of SSC-DM-CRC and analyzes the related fault missing rate.

4.1 Working procedure of MSC based DM-CRC

The analysis in Section 3 shows that, for the SSC-DM-CRC, smaller modulus brings higher fault missing rate. For example, the missing rate of MIF is 33.3% for $m = 3$. Although the missing rate can be reduced by increasing modulus, the area of CRC module becomes unacceptable [32]. In other words, the SSC-DM-CRC cannot achieve low overhead and small fault missing rate simultaneously.

To solve this problem, the MSC is proposed to replace the SSC for identification of fault-free module in the synthesis logic (see Figure 3). The general idea of MSC is that, when the unidentified SEU happens, the current outputs of M_1 and M_2 are buffered, and the next outputs are used for checking again. This process continues until the fault-free module is identified. Then, all the buffered outputs of the fault-free module are output together. The detailed working procedure of MSC-DM-CRC is shown in Figure 5. A potential problem of MSC-DM-CRC is that if the buffered samples of the fault-free module experience SEU, faulty results will be output. In this article, we assume that the buffering cycle is much smaller than the interval between two SEUs, so that only one SEU may happen during the buffering of N samples.

It should be emphasized that, since the buffered outputs of the fault-free module are output as a block, the equivalent throughput maintains unchanged during the MSC processing.

4.2 Analysis of the fault missing rate for MSC-DM-CRC

The analysis in Section 3 tells that only MIF may be unrecognizable, so this section will focus on the fault missing rate analysis for MIF model. For the MSC-DM-CRC, the effect of SEU to filter coefficients is different from that to input samples. If the SEU occurs on one filter coefficient at some instant, all the outputs henceforth are affected, so we have a lot of chance to identify the fault module by buffering enough outputs. In this case, assuming the samples are mutually independent, the fault missing rate of MSC-DM-CRC for a given buffered length N can be expressed as

$$P_{m-\mathrm{MSC}}^{2^n \pm 1} = (P_{m\_\mathrm{MIF}}^{2^n \pm 1})^N, \qquad (15)$$

which equals to $(2^n \pm 1)^{-N}$ for $m = 2^n \pm 1$. The simulations in Section 5 will show that $n = 3$, $N = 4$ can ensure the $P_{m-\mathrm{MSC}}^{2^n \pm 1}$ as low as 0.05% in practice.

On the contrary, if the SEU occurs in input samples, the fault only affects the results that related to the fault sample, so the chances for MSC to identity the fault-free module is very limited. For example, if a sample experiences an SEU one clock before it shifts out from the filter, only one output is affected by the fault, so the MSC has no more chance to identity the fault-free module than SSC. While if a sample experiences an SEU two clocks before it shifts out from the filter, two outputs will be affected by the fault, so the MSC has one more chance to identity the fault-free module compared with SSC. In general, if x_l (see Equation (6)) experiences SEU, the minimum missing probability of the un-recognizable fault should be

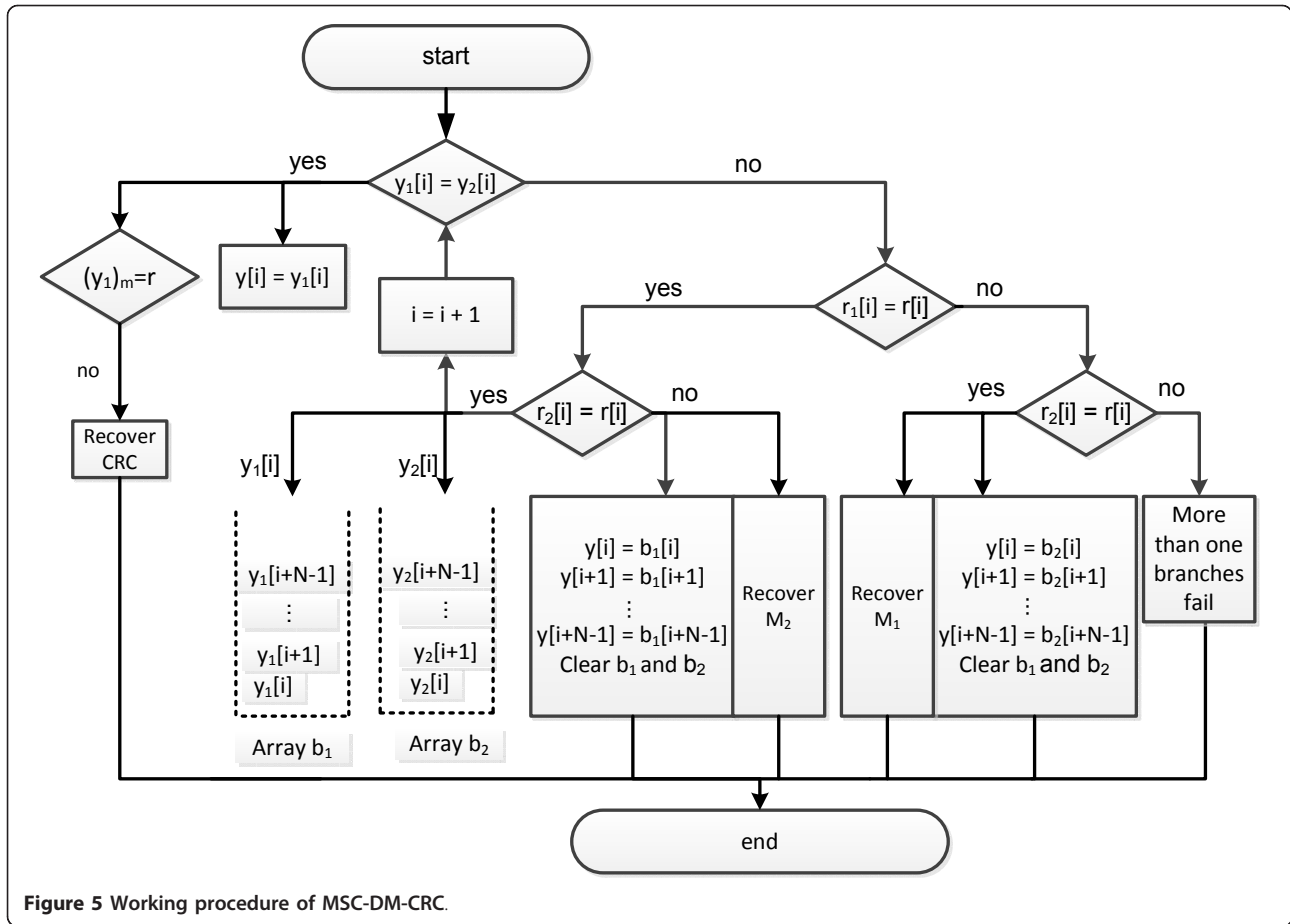$$\mathrm{Prob}_{m,l} = \left(P_{m\_\mathrm{MIF}}^{2^n \pm 1}\right)^{L-l}, \qquad (16)$$

Figure 5 Working procedure of MSC-DM-CRC.

then the average fault missing rate for SEU on samples can be expressed as:

$$\bar{P}_{m-\mathrm{MSC}}^{2^n\pm1} = \frac{1}{L}\sum_{l=0}^{L-1}\mathrm{Prob}_{m,l} = \frac{1}{L}\sum_{l=0}^{L-1}\left(P_{m\_\mathrm{MIF}}^{2^n\pm1}\right)^{L-l} = \frac{P_{m\_\mathrm{MIF}}^{2^n\pm1}(1-P_{m\_\mathrm{MIF}}^{2^n\pm1}{}^L)}{L(1-P_{m\_\mathrm{MIF}}^{2^n\pm1})}. \quad (17)$$

Considering that the un-recognizable faults for MSC are mainly from the samples at the end of the filtering procedure, e.g. $l = L - 1$ or $L - 2$, the registers storing these samples should be taken as "VIPs" and hardened by individual protection, such as TMR.

Now the hybrid scheme combining MSC and TMR can guarantee a low fault missing rate, even the SEU happens on input samples.
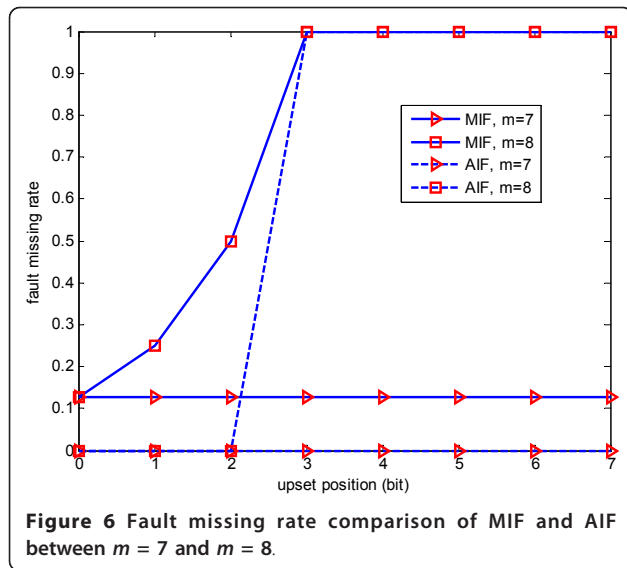
5 Simulation results
In this section, FPGA based fault injections are performed to show the overhead reduction of the DM-CRC architecture relative to the TMR, and to validate the fault missing analysis for the SSC-DM-CRC and MSC-DM-CRC based FIR designs in Sections 3 and 4. In the simulations, a 16-tap FIR filter with 8-bit inputs and 8-bit coefficients are implemented in ISE12.3 for Xilinx FPGA Virtex-4 XC4VLX100.

Firstly, the resource usages of the DM-CRC based FIR design and the TMR based FIR design are compared in Table 2. It can be seen that, smaller modulus brings larger overhead reduction. For example, about 30 and 20% resources are saved when $n = 2$ and $n = 3$, respectively, relative to that of TMR. When n is as large as 6, the overhead of slices and LUTs for the DM-CRC is even larger than that for the TMR. This is because the overhead introduced by the module operation for all the addition and multiplication is larger than the reduction brought by the FIR implementation with shortened input samples and filter coefficients.

Secondly, fault missing rate in MIF and AIF models are compared in Figure 6 for $m = 2^3 = 8$ and $m = 2^3 - 1 = 7$.

Table 2 Overhead comparison

Logic utilization	TMR protected	Proposed method protected				
		$n = 2$	$n = 3$	$n = 4$	$n = 5$	$n = 6$
Number of slices	1085	742	846	902	1028	1089
Slice flip flops	1533	1008	1142	1156	1233	1272
4-input LUTs	1537	1175	1344	1453	1535	1892

Figure 6 Fault missing rate comparison of MIF and AIF between *m* = 7 and *m* = 8.

Figure 7 Distribution of congruent samples.

Since the fault missing rate is related to the upset position when $m = 8$, each bit of the input of multipliers and adders is upset in turn. From Figure 6, we see that the fault missing rate for $m = 7$ keeps to be 14.2% and 0 in MIF and AIF model, respectively, no matter which bit is upset. This is consistent with Equation (13) and the analysis in Section 3.2.3. For $m = 8$, when $q = 0, 1$ and 2, the fault missing rates in MIF model are 12.5, 25 and 50%, respectively, and that in AIF model keeps to be 0. When $q \geq 3$, the fault missing rate keeps to be 100% in both MIF model and AIF model. This is because the fault, caused by the upset of bit higher than the third one, will not change the result of modulo-8, which means the fault will be missed definitely. These results are consistent with Equation (12) and the analysis in Section 3.2.3.

To show the necessity of the MSC-DM-CRC scheme, the probability for congruent outputs between correct FIR branch and the faulty FIR branch are examined. In the test, 136,000 samples are input to the two 16-tap FIR filters, and SEU is injected randomly into the input samples and filter coefficients for one of the filters. The two output streams then goes through the modulo-m operation, where $m = 2^n - 1$ with $n = 2, 3, 4, 5$ and 6. The results show that, the number of congruent samples for $n = 2, 3, 4, 5$ and 6 are 45184, 19752, 9352, 4416 and 2376, respectively, which are about $\dfrac{1}{2^2 - 1}, \dfrac{1}{2^3 - 1}, \dfrac{1}{2^4 - 1}, \dfrac{1}{2^5 - 1}$ and $\dfrac{1}{2^6 - 1}$ of total samples. These values tell that the congruent outputs are not events with low probabilities, so congruent problem is serious for the SSC-DM-CRC scheme, especially with small modulus. In addition, many of the congruent pairs of samples are consecutive, so the MSC scheme is very necessary. Figure 7 shows the
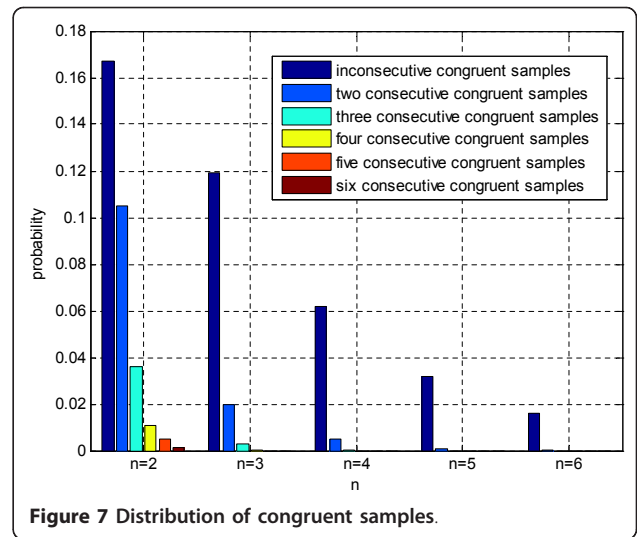
percentage of single, two, three, four, five and six consecutive congruent pairs of samples for different moduli ($n = 2, 3, 4, 5$ and 6). For a specified n, the leftmost bar represents the probability of single congruent samples, the rightmost bar represents the probability of six consecutive congruent samples, and the sum of the probability of each bar is approximately the fault missing rate of the system (seven or more consecutive congruent samples are ignored because of their low probabilities). These results will be used later to check the correctness of results for SSC/MSC-DM-CRC testing.

Figure 8 shows the fault missing rate for different modulus and different number of samples for checking in the MSC-DM-CRC based FIR design. The modulus is in the form of $2^n - 1$, and both the simulation and theoretical results are given for different moduli ($n = 2, 3, 4, 5$ and 6) and different number of samples for MSC-DM-CRC ($N = 1, 2, 3$ and 4). The MSC-DM-CRC with

Figure 8 Fault missing rate of SSC-DM-CRC and MSC-DM-CRC.
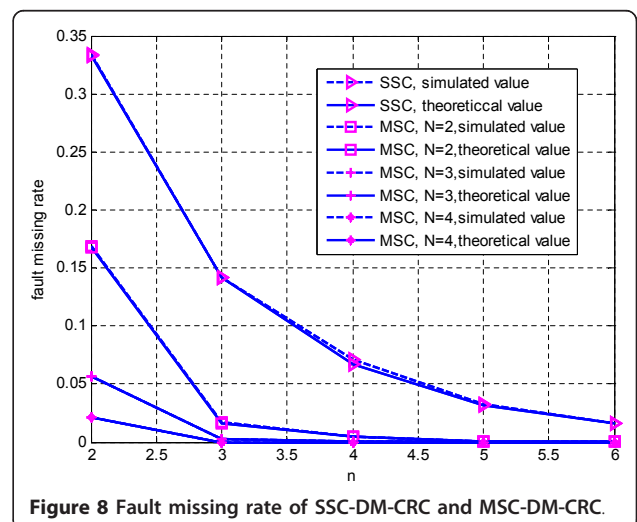
$N = 1$ is actually SSC-DM-CRC. From the figure, it's clear that: (1) larger modulus brings lower fault missing rate; (2) more samples for MSC produces lower fault missing rate; (3) the simulation results match the theoretical results very well. For a specified n, the sum of probabilities for all bars in Figure 7 correspond to the fault missing rate of SSC in Figure 8, and the sum of probabilities for 2-6 consecutive congruent samples correspond to the fault missing rate of MSC with $N = 2$. In this way, we can verify the consistency of Figures 7 and 8. For example, the fault missing rate for $n = 2$ and $N = 3$ can be read from Figure 8 as about 5.25%. Since three samples for checking can detect all the faults those producing single congruent output and two consecutive congruent outputs, the fault missing rate should be the sum of the probabilities of 3-6 consecutive congruent outputs. In Figure 7, the probability for 3, 4, 5 and 6 consecutive congruent outputs are 3.6, 1.1, 0.4 and 0.12%, respectively, and the sum of these probabilities are 5.22%, which is very closed to 5.25%. The remaining 0.03% fault missing rate should come from seven or more consecutive congruent outputs.

Since the properties of fault missing rate are similar for $m = 2^n \pm 1$, the simulations for $m = 2^n + 1$ are not given here. It should be pointed out that, the numerical results given in this section are only based on the basic implementation of a 16-tap FIR filter, and other FIR implementations may produce different results. However, our analysis method and the general conclusions given in this article should hold for all CRC based SEU-tolerant FIR designs.

6 Conclusion

To reduce the area and power overhead of TMR protected on board processing (OBP) transponders, a simple and effective SEU-tolerant design is proposed for the basic FIR implementation. The design is based on a structure with DM of normal FIR and one DM-CRC. The fault missing problem, which is common for residue code based checking schemes, is firstly revealed in this article, and the fault missing rate for SSC is analyzed in detail based on the DM-CRC structure. Then, the MSC is proposed to solve the fault missing problem. The MSC-DM-CRC based FIR design achieves smaller overhead and lower fault missing rate simultaneously. Fault injection campaigns show that, for a 16-tap FIR filter with 8-bit input samples and coefficients, if the modulus for checking branch is 7 and the buffering length is 4, the proposed MSC-DM-CRC FIR design can save more than 20% area overhead relative to the TMR design, and the fault missing rate can be reduced to 0.05%. The analysis method and theoretical results can also be applied to the SEU-tolerant design of other On-Board DSPs with MAC structure, such as FFT and digital beamforming.

Author details

[1]Department of Communication Engineering, Xiamen University, Xiamen, China [2]Research Institute of Information Technology, State Key Laboratory on Microwave and Digital Communications, Tsinghua National Laboratory for Information Science and Technology (TNList), Tsinghua University, China [3]Aerospace Center, School of Aerospace, Tsinghua University, Beijing, China

References

1. MA Jolfaei, KRA Jakobs, Concept of on-board-processing satellites, in *Proceedings of Universal Personal Communications* Dallas, TX, 1–4 (2002)
2. J Farserotu, R Prasad, A survey of future broadband multimedia satellite systems, issues and trends. IEEE Commun Mag. **38**(6), 128–133 (2000). doi:10.1109/35.846084
3. M Ibnkahla, Q Rahman, A Sulyman, H Al-Asady, J Yuan, A Safwat, High-speed satellite mobile communications: technologies and challenges. Proc IEEE. **92**(2), 312–339 (2004). doi:10.1109/JPROC.2003.821907
4. S Verma, E Wiswell, Next generation broadband satellite communication systems, in *Proceedings of the 20th AIAA International Communication Satellite Systems Conference and Exhibit 2002*, Montreal, Quebec, Canada, 1–7 (2002)
5. D Gupta, T Filippov, A Kirichenko, D Kirichenko, I Vernik, A Sahu, S Sarwana, P Shevchenko, A Talalaevskii, O Mukhanov, Digital channelizing radio frequency receiver. IEEE Trans Appl Superconduct. **17**(2), 430–437 (2007)
6. TC Butash, JR Marshall, Leveraging digital on-board processing to increase communications satellite flexibility and effective capacity, in *Proceedings of the 28th AIAA International Communications Satellite Systems Conference 2010*, Anaheim, California, 113–122 (2010)
7. P Angeletti, R De Gaudenzi, M Lisi, From "Bent pipes" to "software defined payloads": evolution and trends of satellite communications systems, in *Proceedings of the 26th AIAA International Communications Satellite Systems Conference 2008*, San Diego, CA, 1–10 (2008)
8. K Wiemann, The ACeS digital channelizer–the next generation in regional digital satellite telephone communications, in *Proceedings of the 19th Digital Avionics Systems Conference, 2000* Philadelphia, USA, 1–6 (2000)
9. D Sunderland, G Duncan, B Rasmussen, H Nichols, D Kain, L Lee, Megagate ASICs for the thuraya satellite digital signal processor, in *Proceedings of International Symposium on Quality Electronic Design, 2002*, Washington. DC, USA, 479–486 (2002)
10. R Kumar, D Taggart, R Monzingo, G Goo, Wideband gapfiller satellite (WGS) system, in *Proceedings of the IEEE Aerospace Conference, 2005* Big Sky, MT, USA, 1410–1417 (2005)
11. W Ludong, D Ferguson, WGS air-interface for AISR missions, in *Proceedings of the IEEE Milcom 2007 Conference, 2007*, Orlando, FL, USA, 1–7 (2007)
12. JS Sadowsky, DK Lee, The MUOS-WCDMA air interface, in *Proceedings of the IEEE Milcom 2007 Conference, 2007*, Orlando, FL, USA, 1–6 (2007)
13. D Wadsworth, Military communications satellite system multiplies UHF channel capacity for mobile users, in *Proceedings of IEEE Military Communications Conference, 1999*, Atlantic City, NJ, **2**, 1145–1152 (1999)
14. MO Kolawole, *Satellite Communication Engineering*, (Marcel Dekker, New York, 2002)
15. G Maral, M Bousque, *Satellite Communications Systems: Systems, Techniques and Technology*, 5th edn. (Wiley, New Jersey, 2009)

16. W Fernando, R Rajatheva, Performance of COFDM for LEO satellite channels in global mobile communications, in *Proceedings of Vehicular Technology, 1998,* Ottawa, Ont, **2**, 1503–1507 (1998)

17. A Papathanassiou, A Salkintzis, P Mathiopoulos, A comparison study of the uplink performance of W-CDMA and OFDM for mobile multimedia communications via LEO satellites. IEEE Personal Communication. **8**(3), 35–43 (2001). doi:10.1109/98.930095

18. J Litva, TK Lo, *Digital Beamforming in Wireless Comuunications* (Artech House, Inc. Norwood, MA, 1996)

19. I Chiba, R Miura, T Tanaka, Y Karasawa, Digital beam forming (DBF) antenna system for mobile communications. Aerosp Electron Syst Mag. **12**(9), 31–41 (1997). doi:10.1109/62.618017

20. FL Kastensmidt, L Carro, R Reis, *Fault-Tolerance Techniques for SRAM-Based FPGAs* (Springer, New Haven, 2006)

21. D Kaschmitter, JL ans Shaeffer, N Collea, C McKnett, P Coakley, Operation of commercial R3000 processors in the low earth orbit (LEO) space environment. Nucl Sci. **38**(6), 1415–1420 (1991). doi:10.1109/23.124126

22. C Carmichael, Application Note: Triple Module Redundancy Design Techniques for Virtex FPGAs (XAPP197 (v1.0.1)), (XILINX INC, 2006)

23. PL Murray, D VanBuren, Single event effect mitigation in reconfigurable computers for space applications, in *Proceedings of IEEE Aerospace Conference, 2005,* Big Sky, MT, USA, 1–7 (2005)

24. A Wrzyszcz, D Milford, EL Dagless, A new approlach to fixed-coefficient inner product computation over finite rings. IEEE Trans Comput. **45**(12), 1345–1355 (1996). doi:10.1109/12.545965

25. GC Cardarilli, A Nannarelli, M Re, Residue number system for low-power DSP applications, in *Proceedings of Conference Record of the Forty-First Asilomar Conference on Signals, Systems and Computers, ACSSC 2007,* Pacific Grove, CA, 1412–1416 (2007)

26. MH Etzel, WK Jenkins, Redundant residue number systems for error detection and correction in digital filters. IEEE Trans Acoust Speech Signal Process. **28**(5), 538–544 (1980). doi:10.1109/TASSP.1980.1163442

27. L Li, J Hu, Error correcting properties of redundant residue number systems. IEEE Trans Nucl Sci. **57**, 2332–2343 (2010)

28. FL Kastensmidt, G Neuberger, L Carro, R Reis, Designing fault-tolerant techniques for SRAM-based FPGAs. IEEE Circ Syst Soc. **21**(6), 522–562 (2004)

29. JJ Rodríguez-Navarro, M Gansen, TG Noll, Error-tolerant FIR filters based on low-cost residue codes, in *IEEE International Symposium on Circuits and Systems (ISCAS), 2005* **5**, 5210–5213 (2005)

30. C Bolchini, A Miele, M Santambrogio, M Politecnico di Milano, TMR and partial dynamic reconfiguration to mitigate SEU faults in FPGAs, in *Proceedings of Defect and Fault-Tolerance in VLSI Systems, 2007,* Rome, 87–95 (2007)

31. WK Jenkins, BJ Leon, The use of residue number systems in the design of finite impulse response digital filters. IEEE Trans Circ Syst. **24**, 191–200 (1977). doi:10.1109/TCS.1977.1084321

32. A Nannarelli, M Re, GC Cardarilli, Tradeoffs between residue number system and traditional FIR filters, in *Proceedings of the 2001 IEEE International Symposium on Circuits and Systems (ISCAS), 2001,* Sydney, NSW, **2**, 305–308 (2001)