

Fault-Tolerant Design of SRAM-FPGA Register Based on Dual-Mode Redundancy Structure

Zijian Huang

Shenzhen University, Shenzhen, China

Keywords: FPGA; SRAM; Register; Dual Modular Redundancy.

Abstract: As an important part of the design of SRAM-FPGA system, registers are vulnerable to soft errors caused by high-energy particles in the space environment. The three modular redundancy design method can protect the register from fault tolerance, but it brings a lot of resources and power consumption. Aiming at this problem, this paper designs a dual-mode redundancy protection structure based on parity check, which takes flip-flops as units and takes advantage of the structure characteristics of FPGA lookup table. The two flip-flops are designed as a group of fault-tolerant devices. By choosing the correct result output through parity check, the data bits of registers can be misshielded from fine-grained aspects. This method is verified in the history of Xilinx Virtex®-6FPGA. Experimental results show that the design of registers relative to three modular redundancy can reduce 16.7% of trigger resources and 50% of look-up table resources.

1. Introduction

Field Programmable Gate Array (FPGA) based on SRAM has the advantages of high performance, high flexibility, repetitive programmability and short development cycle. It is widely used in electronic equipment of aerospace system [1]. Because the space environment is full of all kinds of high-energy particles, the impact of high-energy particles on the working electronic devices will cause radiation effects (such as single event flip effect), and lead to device failure. SRAM-FPGA contains a large number of memory cells, such as configuration memory, user memory and registers. Single event flip will cause the logical state of memory cells to flip. Device errors caused by single event flip are soft errors, which can be restored to normal state by system reset, re-charging or re-writing [2].

Registers are widely used in the design of the FPGA system. As one of the sensitive resources of the system, once a fault occurs, erroneous instructions will be generated, erroneous data will be sent, resulting in abnormal system functions. At present, many fault-tolerant technologies have been widely used in the protection of SRAM-type soft errors in space environment. In reference [3], an anti-radiation enhanced register design is used to reinforce the structure of the register from the circuit to prevent soft errors, but this method is only applicable to ASIC circuit design. Literature [4] uses a scrubbing technique to erase and rewrite the configuration memory periodically. This method can eliminate the influence of soft errors by reconfiguring the logic function of the FPGA. Although it can repair the faults, it can not shield the faults. In document [5], the error correction code (ECC) technology is used to add additional check data bits to the back end of the data packet to achieve the accuracy of reading/writing data, and to a certain extent, the memory data bits can be detected and corrected. In [6], ECC is used to detect and repair registers' errors, but this method requires adding ECC coders and decoders to each register, which results in additional resource overhead. At the same time, ECC decoders introduce a lot of delay to the circuit. This method is not suitable for high-speed circuit implementation. Redundancy design is the most widely used register fault tolerant design method at present. For example, the three module redundancy design structure in [7-8] is used to design registers for fault tolerance. In the literature [9-10], the three module redundancy structure and dynamic reconfiguration technology are combined to achieve register error shielding and repair. However, the three module redundancy design usually brings more than two times of additional

logical resources and power consumption, and it is difficult to meet the design requirements of complex electronic systems with high resource occupancy rate.

In view of the shortcomings of existing schemes, a dual-mode redundancy design structure based on parity check is proposed to protect registers in SRAM-FPGA circuits. The design structure is based on flip-flop, which shields the data bits of registers from a fine-grained point of view. The design structure is verified on Xilinx Virtex®-6 FPGA.

2. Register protection design method

2.1 Three module redundancy protection structure

SRAM-FPGA is mainly composed of programmable logic block (CLB), programmable IO module and programmable internal connection. CLB is the main configurable resource for the design and implementation of FPGA circuits. It is mainly composed of LIUTs and Flip-Flops, which are used to realize the combinational logic and sequential logic of circuits. Taking Xilinx Virtex®-6 FPGA as an example, each CLB contains two Slices, and each Slice contains four 6-input lookup tables and eight triggers. Each lookup table can realize a 6-input-1-output logic function or a 5-input-2-output logic function.

The memory circuit of registers is composed of flip-flops. A flip-flop can store one bit of binary number, so N flip-flops can form N-bit registers. Figure 1 shows a register register of three modulus redundancy protection structure. The design structure duplicates three copies of the same register, and adds a majority voter to the output port to select the data output. The majority voter output adopts a three-to-two selection method. When one of the registers is wrong, the majority voter can shield the error output, so as to achieve the purpose of fault tolerance. In the implementation of the FPGA circuit, each majority voter takes up three inputs and one output of a lookup table. However, the two majority voters can not share one lookup table and the port of the lookup table can not be fully used. For a register with 2N data bits, the implementation of three mode redundancy design requires at least 2N look-up tables and 6N flip flops.

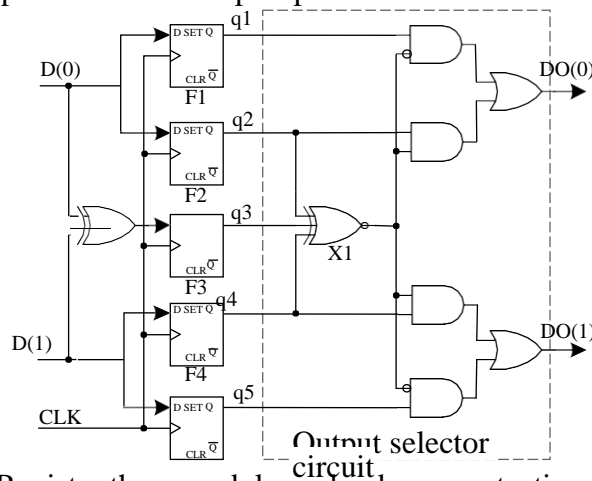


Fig. 1 Register three modular redundancy protection structure

2.2 PC-DMR Protection Structure

In view of the shortcomings of three module redundancy design structure lookup table port utilization and high resource cost, this paper designs a parity check dual modular redundancy (PC-DMR) with parity check bits to protect registers.

Figure 2 shows a PC-DMR protection structure for two flip-flops. The structure makes use of the structure characteristics of the FPGA lookup table and designs two flip-flops as a group. Each group contains two data bits and one parity bit. The parity bit is the exception or result of two input values. In the structure, each group of triggers corresponding to data bits is copied separately, and a trigger is added to store parity bits, parity bits are used to detect whether the output of the trigger is wrong, and

the output selection circuit shields the error output according to the input combination. Next, the specific single event overturn fault is analyzed with Figure 2.

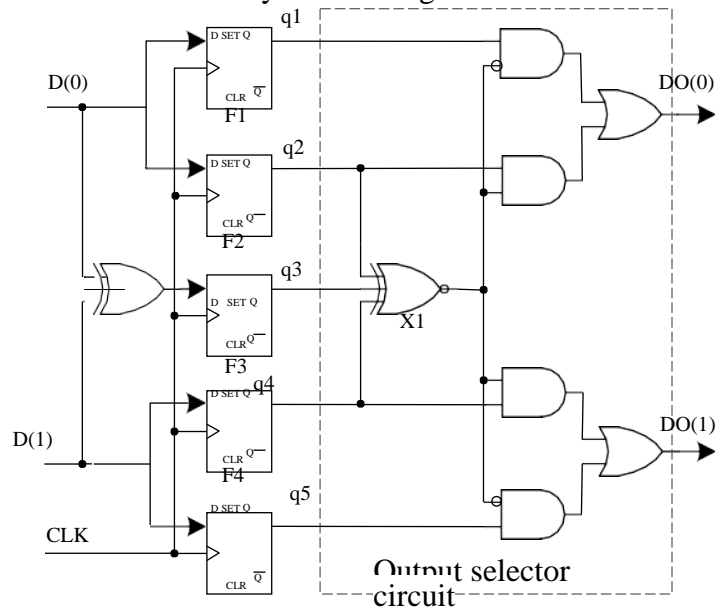


Fig. 2 PC-DMR structural unit

D0 and D1 are input of two data bits, F1 and F2 are a group of flip-flops with the same input, F4 and F5 are another group, F3 is used to store parity bits, and output selection circuit is used to select the correct output of flip-flops. When the output of F2, F3 and F4 is correct, the output of X1 with three inputs and X1 without gates is 1, the output of F1 and F5 with shielding flip-flop is 1, and the value of F2 and F4 flip-flop is output at the same time. When one of the flip-flops F2, F3 and F4 has a single event flip-over, that is, the logical state of storage has been flipped, X1 or X1 output is zero, and the error output of F2 and F4 flip-flops is shielded, and F1 and F5 flip-flops are selected as the output. The design structure can tolerate the logic state abnormality of any trigger in the structure unit and the simultaneous abnormality of the triggers F1 and F5.

For a register with $2N$ data bits, the triggers corresponding to each two adjacent data bits of the register are designed as a group of PC-DMR. Each group is designed as a PC-DMR structure unit. For $2N$ -bit registers, it can be divided into N groups of PC-DMR structure units, as shown in Figure 3. The data bits of registers commonly used in the system are even bits. For odd bits, a redundant bit can be added to the highest bit and then PC-DMR design can be carried out. For each group of data bits, the protection structure only needs three additional triggers, one lookup table and one XOR gate. PC-DMR protection structure is used to design a register with $2N$ data bits, which requires at least $5N$ flip-flops, N lookup tables and N two-input XOR gates. The two-input XOR gates in the design structure can be implemented by the two-input XOR arithmetic logic in Slice, without taking up additional look-up table resources. It can be seen that the PC-DMR protection design for a $2N$ bit register can reduce the resource cost of N look-up tables and N flip flops compared with the traditional three mode redundancy design.

3. Reliability assessment

The function of each module in the FPGA is determined by the configuration of the configuration memory, which contains several bits representing the configuration information. In the radiation environment, when the single event flip effect occurs in the circuit module of the FPGA, the function of the module will not be abnormal if any bit of the module flips. The function of the module is often determined by some key bits that can change the logic state of the module [11]. Let the failure rate of a single flip-flop be λ_m , which is determined by the flip-over rate of a single λ_{bit} in unit time and the key bit number corresponding to the flip-flop. Usually $\lambda_m = m\lambda_{bit}$. In the literature [12], the typical

ratio of Xilinx FPGA in a high radiation environment is typically $\lambda_{\text{bit}} = 2.7 \times 10^{-10}$ upsets/bit/s. The reliability of the system under continuous-time Markov model is discussed[13]. The reliability of the normal function of a single trigger in time t decreases exponentially with time, and its reliability can be expressed as follows:

$$R(t) = e^{-\lambda mt} \quad (1)$$

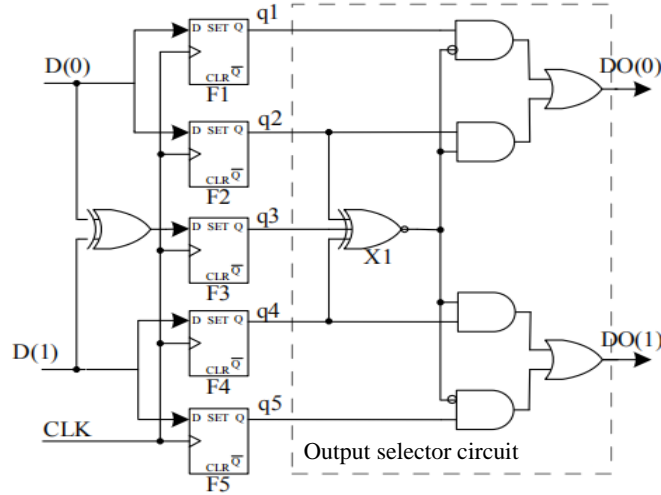


Fig. 3 Protection design of register PC-DMR with 2N data bits

For a N-bit register without any protective structure, its function is normal within time t , that is to say, to satisfy the normal logic state of all the transmitters constituting it, its reliability can be expressed as follows:

$$RN(t) = e^{-\lambda m Nt} \quad (2)$$

For a three modular redundant system, the protection function will fail only when two or more than two modules fail, assuming that the logic function of the three module redundancy system is normal. For a N bit register containing N flip-flop, three mode redundancy protection is designed, that is, the three mode redundancy design of the trigger corresponding to the N bits, and the reliability of the logic state in time t is normal.

$$RTMR(t) = (C^2 R^2(t)(1 - R(t)) + C^3 R^3(t))N = (3R^2(t) - 2R^3(t))N \quad (3)$$

In the PC-DMR structure unit, two triggers in the register are grouped together for protection. When two or more triggers fail, the protection function will fail. In Figure 2, because the output logic gives priority to the correct output of flip-flops F2 and F4, the wrong output of F1 and F5 can be shielded when F2 and F4 have no errors. At this time, the simultaneous errors of flip-flops F1 and F5 will not lead to the failure of the protective structure. The reliability of the structural unit in normal logical state in time t can be expressed as

$$R^{\text{PC-DMR}}(t) = C^5 R_5^5(t) + C^4 R^4(t)(1 - R(t)) + R^3(t)(1 - R(t))^2 = R^3(t) + 3R^4(t) - 3R^5(t) \quad (4)$$

PC-DMR protection design for N-bit registers containing N flip-flops, that is to say, to satisfy the normal flip-flops of each group of PC-DMR structural units, the reliability of the normal logic state of the registers in time t can be expressed as follows:

$$R_N^{\text{PC-DCM}}(t) = (R^{\text{PC-DMR}}(t))^{\frac{N}{2}} \quad (5)$$

The mean time to failure (MTTF) is used to measure the reliability of a module, that is, the average working time between two adjacent faults. MTTF in continuous-time Markov model can be expressed as a function of reliability.

$$MTTF = \int_0^{\infty} R_m(t) dt \quad (6)$$

$R_m(t)$ represents the reliability of the module. The reliability of the unprotected registers is compared with the reliability of the three modular redundancy protection structure and the PC-DMR protection structure respectively. Fig. 4 (a) Fig. 4 (b) are the time-varying reliability diagrams of 16-bit registers and the average fault-free time diagrams of 8-bit 16-bit 32-bit registers at the failure rate $\lambda = 1.35 \times 10^{-9}$ times/s, respectively.

From the graph, we can see that when the failure rate is λ , the reliability of registers will gradually decrease with the increase of the use time. The higher the number of registers, the more resources they occupy, the corresponding MTBF will be reduced, and the register reliability under the PC-DMR structure is slightly lower than the reliability of registers under the three modular redundancy design. However, the reliability of unprotected registers has been considerably improved.

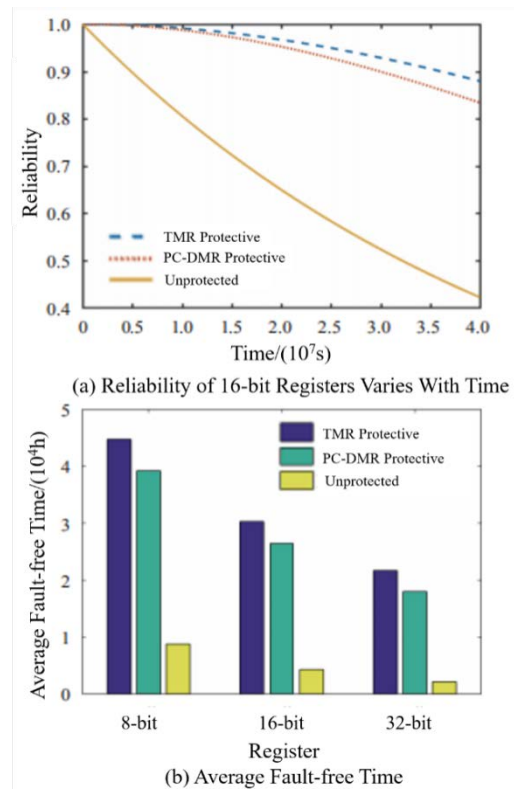


Figure. 4 Reliability comparison between unprotected registers and TMR, PC-DMR protection structures

4. Experimental results and analysis

In this paper, the PC-DMR design structure is designed and validated on Xilinx Virtex #6 FPGA by ISE14.7 design tool. Firstly, PC-DMR is designed for 32-bit registers, and the fault-tolerant function of the design is verified by simulation. RTL-level description design of registers is carried out by HDL design language. 32 bit data bits of registers are divided into two adjacent groups. Each group is designed as a PC-DMR structure unit. PC-DMR structure unit is described in Figure 2. 32 bit registers under PC-DMR protection design contain 16 groups of PC-DMR structure units.

The 32-bit register under this design structure is simulated by flipping the logic state of the trigger to simulate the register error. The simulation results are shown in Fig. 5. Before t_1 , the logical state of each group of flip-flops in the register is normal, and the output DO of the register equals the input D of the register. At t_1 , the logical state of the flip-flop Q3 in the third group of structural units

is reversed, but the output DO of the register is still the same as the input D of the register. At time t_2 , the logic state of trigger Q2 in the 16th group of structural units reverses, and at time t_3 , the logic state of trigger Q1 and trigger Q5 in the 8th group of structural units simultaneously abnormal, but the output of registers has not been wrong, and the fault-tolerant function of registers is normal.

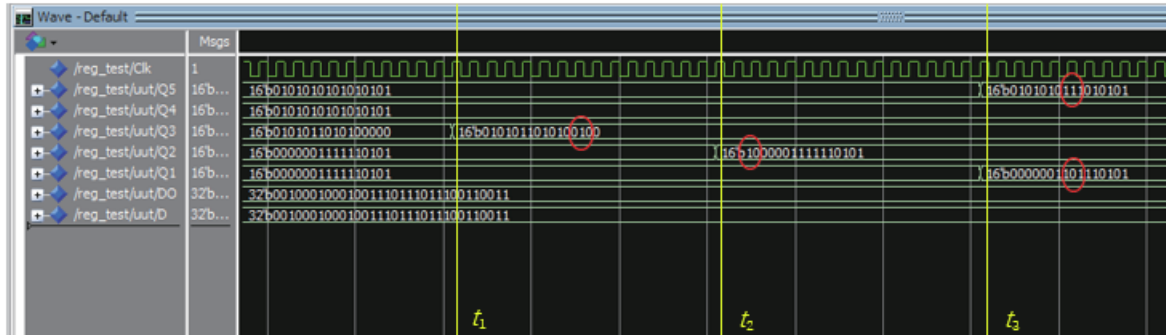


Figure. 5 Experimental simulation diagram of 32-bit register

In order to verify the resource cost of registers under the design structure, three module redundancy design and PC-DMR design for 8 bit 16 bit 32 bit independent registers are carried out respectively. The combined resource overhead is shown in Table 1. Experimental results show that the register designed by PC-DMR can effectively reduce the trigger resource and 50% look-up table resource cost compared with the traditional three module redundancy design register.

Table 1 Register resource overhead

Register	Unprotected structure		TMR		PC-DMR		Save the proportion /%	
	Trigger	Lookup Table	Trigger	Lookup Table	Trigger	Lookup Table	Trigger	Lookup Table
8-bit	8	0	24	8	20	4	16.7	50
16-bit	16	0	48	16	40	8	16.7	50
32-bit	32	0	96	32	80	16	16.7	50

PC-DMR design structure protects registers from fine-grained point of view. The design structure can also be used to protect sequential logic circuits containing a large number of triggers. The three module redundancy design and PC-DMR design of 8 bit 16 bit 32 bit counters with enable terminals are designed respectively. The comprehensive resource cost is shown in Table 2. Experimental results show that the counter designed by PC-DMR can save 16.7% trigger resources and 28% look-up table resources compared with the traditional three module redundancy design counters. Due to the influence of the original logic structure of the counter on the lookup table of the PC-DMR design structure, the number of lookup tables is increased, and the circuit delay introduced is also slightly increased. However, compared with the traditional three mode redundancy design, the considerable resource cost can still be saved.

Table 2 Counter resources and latency overhead

Counter	Unprotected structure			TMR			PC-DMR		
	Trigger	Lookup Table	Delay /ns	Trigger	Lookup Table	Delay /ns	Trigger	Lookup Table	Delay /ns
8-bit	8	8	1.46	24	25	2.01	20	19	50
16-bit	16	16	1.52	48	48	2.10	40	35	50
32-bit	32	32	1.70	96	97	2.28	80	67	50

5. Conclusion

Aiming at the fault-tolerant problem of the circuit registers of the FPGA in space environment and some shortcomings of the existing schemes, a dual-mode redundancy protection structure based on parity check is proposed in this paper. The structure takes flip-flop as unit and two as a group. It shields the data bits of registers from the fine-grained point of view. The structure can also effectively protect the circuit containing a large number of triggers, and can realize fault tolerance and eliminate the accumulation of faults by combining Scrubbing technology. The design structure is implemented on Xilinx Virtex -6 FPGA. Experimental results show that the reliability of this design structure is slightly lower than that of the traditional three module redundancy design, but it can significantly save resources and reduce costs.

References

- [1] DONTI S, HAGGARD R L. A survey of dynamically reconfigurable FPGA devices[C]//Proceedings of the 35th Southeastern Symposium on System Theory. Morgantown: IEEE, 2003: 422-426.
- [2] ASADI H, TAHOORI M B. Analytical techniques for soft error rate modeling and mitigation of FPGA-based designs [J]. IEEE Transactions on VERY LARGE SCALE INTEGRATION (VLSI) Systems, 2007, 15(12): 1320-1331.
- [3] Huang Zhengfeng, Ni Tao, Ouyang Yiming, et al. Tolerance of single mode multi node flip three mode interlocking latch [J]. [J]. Journal of University of Electronic Science and technology of China, 2016, 45 (5):
- [4] STODDARD A, GRUWEEL A, ZABRISKIE P, et al. A hybrid approach to FPGA configuration scrubbing [J]. IEEE Transactions on Nuclear Science, 2017, 64(1): 497-503.
- [5] DEMIRCI M, REVIRIEGO P, MAESTRO J A. Implementing double error correction orthogonal Latin Squares Codes in SRAM-based FPGAs [J]. Microelectronics Reliability, 2016, 56: 221-227.
- [6] REVIRIEGO P, PONTARELLI S, MAESTRO J A, et al. A method to construct low delay single error correction (SEC) codes for protecting data bits only[J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2013, 32(3): 479-483.
- [7] STERPONE L, VIOLANTE M, PANARITI A. Layout-aware multicell upsets effects analysis on TMR circuits implemented on SRAM-based FPGAs[J]. IEEE Transactions on Nuclear Science, 2011, 58(5): 2325-2332.
- [8] GHADERI Z, MIREMADI S G, ASADI H, et al. HAFTA: highly available fault-tolerant architecture to protect SRAM-based reconfigurable devices against multiple bit upsets[J]. IEEE Transactions on Device and Materials Reliability, 2013,13(1): 203-212.
- [9] SAEN M, TOBA T, KANNO Y. Soft-error-tolerant dual- modular-redundancy architecture with repair and retry scheme for memory-control circuit on FPGA[J]. IEICE Transactions on Electronics, 2017, 100(4): 382-390.
- [10] MARTINS V M G, VILLA P R C, NETO H C C, et al. A TMR strategy with enhanced dependability features based on a partial reconfiguration flow[C]//IEEE-Computer-Society Annual Symposium on VLSI (ISVLSI). Montpellier: IEEE, 2015: 161-166.
- [11] FENG Z, JING N F, CHEN G S, et al. IPF: In-place X-filling to mitigate soft errors in SRAM-based FPGAs[C]//2011 International Conference on Field Programmable Logic and Applications (FPL). IEEE, 2011: 482-485.

[12] LEE D S, WIRTHLIN M, SWIFT G, et al. Single-Event Characterization of the 28 nm Xilinx Kintex-7 Field- Programmable Gate Array under Heavy Ion Irradiation[C]// Radiation Effects Data Workshop. IEEE, 2014: 1-5.

[13] SHOOMAN M L. Reliability of computer systems and networks: fault tolerance, analysis and design [M]. New York: John Wiley & Sons, Inc., 2002:158-161.