MSW MANAGEMENT -MULTIDISCIPLINARY, SCIENTIFIC WORK AND MANAGEMENT JOURNAL
ISSN: 1053-7899
VOL. 36 ISSUE 2, 2026, PAGES 287-289

MODIFIED AREA-EFFICIENT TMR USING FAST FIR TECHNIQUES FOR IMPROVED FAULT TOLERANCE

M.Ravikumar[1], Santhosh T[2],Sanjay Prasath S[3],Sriharihara G[4],Thangamani J[5]
[1,2]Assistant professor, ECE, Mahendra Engineering College, Namakkal, Tamilnadu
[2,3,4,5,]Student , ECE,Mahendra Engineering College,Namakkal,Tamilnadu

Abstract:
Today, silicon-based processors are increasingly starting to suffer from reliability problems like heat-induced fault, Electromagnetic Interference and Transient Soft Error, which impact system and application performance and reliability, especially in mission critical applications. These problems can lead to incorrect calculations, unexpected failures and reduced life cycles, and traditional CPU designs aren't optimal for harsh and real-time systems. In order to tackle these issues, the proposed system utilizes a hardware optimized fault-tolerant CPU architecture that guarantees error-free CPU operations in the presence of the failure of a single processing module with Triple Modular Redundancy (TMR). Besides, the concepts of Fast FIR algorithm are adopted to minimize the large number of the hardware components for the power consumption and silicon area overhead. The design is modeled in HDL, simulated with fault injection, and implemented in FPGA platform to test the performance under the thermal and interference condition. This method provides reliability, efficiency and stability in embedded systems found in the aerospace, medical, automotive and industrial sectors.

Keywords- Fault Tolerance, Triple Modularity, Soft Errors,Hardware Optimization, Embedded Systems

I.INTRODUCTION

To ensure reliability in today's computer systems, safety-critical and mission-critical embedded systems like the controllers in aircrafts, medical appliances, vehicle electronics, and industrial automation units require fault tolerance as a basic requirement [1]. As the scaling down of semiconductor technology continues, Silicon based processors are being subjected to more transient faults that are likely to cause reliability problems due to reasons such as high temperature, electromagnetic interference, voltage variations and radiation induced faults [2]. These can cause soft errors, bit flipping and timing errors that reduce computation accuracy and thus can cause catastrophic failure of the system [3]. The traditional processor architectures have a major focus on speed and throughput, and largely ignore built-in resilience to these environmental and operational challenges [4]. As a result, ensuring reliable real- time performance under harsh conditions has become a major engineering challenge. To improve the reliability of processors, several fault detection and fault mitigation techniques have been proposed such as parity checking, error-correcting code, duplication with comparison, and checkpoint recovery mechanisms [5]. Of these techniques, Triple Modular Redundancy (TMR) offers tolerance to the faults by executing several copies and majority voting logic, making it widely popular [6]. The use of TMR greatly enhances the reliability, but regular approaches increase the complexity of the hardware, silicon area and power consumption [7]. This overhead makes many designs unsuitable for small embedded systems, as well as for energy optimization and resource efficiency as these two are equally important in such applications [8]. In addition, there are several existing approaches that do not have efficient architectural optimization methods to minimize hardware redundancy without decreasing fault coverage [9]. To overcome these drawbacks, in this work, a new Triple Modular Redundancy (TMR) based fault-tolerant CPU architecture optimized for these hardware is proposed with the incorporation of resources saving computational optimizations. The architecture is developed in Hardware Description Language (HDL) and tested using fault injection simulations to verify the architecture against a controlled environment and error conditions [10]. The design incorporates optimized data-processing structures to lower the number of registers, flip-flops, and buffer elements, resulting in minimized power consumption and hardware overhead, as well as strong fault masking capability. The proposed system also features redundant components along with internal monitoring signals to monitor fault events, voter decision-making, and processing time. This allows not only to analyze performance and monitor it in real time but also to be able to do so in mission-critical environments. The architecture is synthesized and placed on FPGA platforms to test the feasibility of the practical implementation and real-time behavior with different operational stresses. The proposed design not only enhances the reliability of the processor but also optimizes the usage of hardware resources in an efficient manner. It combines redundancy and architectural optimisation to provide a smaller, more energy efficient and more reliable processing solution for next-generation embedded applications in tough environments

II. LITERATURE SURVEY

A few studies aimed at enhancing reliability and fault detection in sensor/embedded systems that are subject to severe dynamic or harsh environments. In [1] a fault detection and localization technique for wind turbine sensors utilizing only output data, which does not involve extra hardware redundancy, was proposed. The energy-efficient task offloading approach to fog-enabled IoT (FET-IoT) networks was introduced in the RETO scheme from [2] to address energy efficiency and system reliability while satisfying deadline-aware processing. In [3] Xia et al. discussed the Single Event effects induced by heavy ions in SiGe Heterojunction Bipolar Transistors, and the TCAD simulation results demonstrated the sensitivity of heavy ion radiation induced soft errors in semiconductor devices. In [4] Sharma and Verma presented a fault tolerant FPGA implementation with redundancy techniques with the help of ECG denoising techniques, which proved that FPGA level redundancy in the hardware level can help to increase the reliability. In [5], Singh and Kumar suggested another voter architecture for Triple Modular Redundancy (TMR) which maintains fault masking ability and reduces area overhead. The following research works have also been conducted recently on TMR-based circuit design and architectural optimizations for improving the reliability and efficiency. In [6], Silva et al. introduced an integrated circuit for magnetic encoder sensing systems for industrial positioning applications, which focuses on reliable operation in industrial settings. In [7] Pires and his work colleagues built optimized CNN structures for TDMR systems, with emphasis on minimizing computational costs and hardware resources. To enhance the sensing accuracy and noise immunity, Dong et al. in [8] suggested a high-resolution TMR readout circuit with offset calibration techniques. In [9] Papadonikolakis and Bouganis demonstrated an efficient parallel architecture using an FPGA to accelerate the solution and in [10] Lotfi-Kamran et al. proposed a hybrid-switched Network-on-Chip (NoC) architecture to improve the communication efficiency of chip multiprocessor systems. However, a lot of these works concerned with performance or sensing accuracy did not focus on hardware optimization and comprehensive fault tolerance to be integrated within a single CPU architecture. Other work has been done in the area of energy efficiency and soft-error resistance at the architectural level. An efficient temporal data prefetcher was introduced in [11] by Bakshalipour et al., which could enhance the cache performance and thus aid in the processing efficiency. Lee [12] discussed the energy-efficient scheduling approaches for real-time applications on multicore processors, and Jeyapaul et al. [13] proposed a multicore architecture to enhance the soft-error reliability. In [14] Modarressi et al. suggested virtual point-to-point connections to improve reliability of communication in Network-on-Chip architectures, whereas Lee et al. suggested techniques to reduce peak power consumption in multicore systems under certain real-time constraints in [15]. However, most previous research is directed towards either reliability improvement or energy saving, separately. A need still exists for architecture of a CPU optimized for hardware with integrated fault masking through redundancy and with minimal hardware overhead, which is also energy efficient in real-time execution.

III. PROBLEM STATEMENT

The processor architecture of conventional silicon is very susceptible to faults caused by heat, electromagnetic interference, radiation, and transient soft errors, particularly in mission critical applications [3]. These errors may introduce bit-flips, timing issues and computation mistakes, resulting in sub-optimal performance or even complete system failure. Traditional CPU designs focus mainly on the speed and computational power of the CPU and lack adequate, built-in mechanisms for fault tolerance [4]. Error correcting codes and duplication-based redundancy have been proposed, but they offer limited fault masking capability and/or high hardware overhead and hardware power consumption [5] A popular approach to providing fault tolerance is to use Triple Modular Redundancy (TMR),

MSW MANAGEMENT -MULTIDISCIPLINARY, SCIENTIFIC WORK AND MANAGEMENT JOURNAL
ISSN: 1053-7899
VOL. 36 ISSUE 2, 2026, PAGES 287-289

however, the traditional TMR systems require more silicon area, more power consumption and more voter complexity [6]. Moreover, a lot of the current systems do not combine how to optimise hardware to remove extra components and retain real-time reliability [7]. Hence, architecture for hardware-based fault-tolerant CPU which guarantee its reliable operation, low resource usage and energy efficiency in severe environment is desired.

IV. PROPOSED SYSTEM

The proposed system uses a hardware optimized fault tolerant CPU architecture by combining the triple modular redundancy with the techniques of resource-efficient design. The system is designed so that if one of the processing modules fails, the system will continue to operate without any faults. The design incorporates architectural optimisation and redundancy, ensuring greater reliability while minimising hardware resources [5], [8]. The main architecture of the chip is three identical processors running identical instructions simultaneously. The output of each of the three modules is compared against each other in a majority voter circuit that chooses the dominant output. This mechanism will implement fault masking and continuous real time processing even when a module fails [6]. The design philosophy followed to achieve above mentioned objective is to maximize the number of registers, flip-flops, input buffers and output buffers for the computation without compromising the complexity of the hardware design, by applying Fast FIR-based optimization principles. This optimization helps save silicon area, power consumption, and improved processing efficiency [7]. The architecture is formalized with HDL and then implemented on FPGA boards for testing. The behavior of the system is evaluated by fault injection simulations for transient fault and thermal fault conditions. Internal monitoring signals are added to monitor the occurrence of a fault, voter decision, and computational delay to be analyzed for performance evaluation

.V. SYSTEM ARCHITECTURE

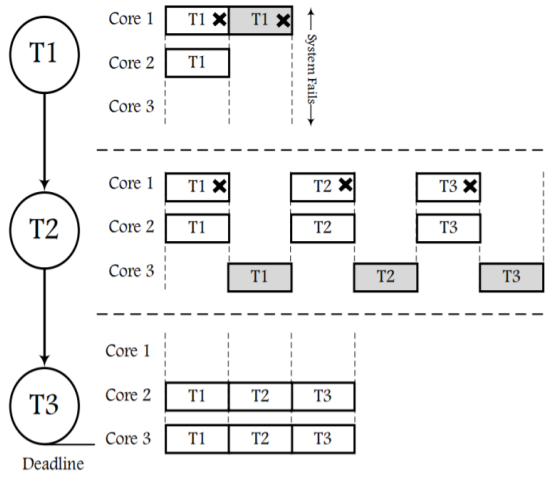

Fig 1 system Architecture

The proposed fault-tolerant CPU system architecture structures three identical processing cores, a majority voter unit, an optimization block and a monitoring unit. The three processing modules process in parallel, with the same instructions and data. The Triple Modular Redundancy (TMR) mechanism [5] is based on this parallel execution. The three modules are connected to a "Majority Voter" circuit to get their outputs. The voter performs a bitwise comparison of the votes and chooses the one with the most votes as the system's output. If one module fails due to a fault, the other two modules have a reliable output as they correct the faulty output [6]. Each processing module contains a hardware optimization block, which is based on fast FIR principles. This block removes redundant storage elements and buffering components, and hence it reduces the amount of hardware needed and power dissipation [7]. The optimized architecture allows making the changes in reliability without wasting too many resources. It includes a monitoring unit for checking the internal signals, including fault flags, timing delays, and decision from the voters. This unit provides data for checking performance and reliability. The whole architecture is synthesized on an FPGA platform, and tested for real-time operation under various fault scenarios. The key building blocks of the proposed system architecture are a clock synchronization and centralized clock distribution network, which ensures identical timing behaviour across the three processing modules, and a synchronized reset control. This synchronization is very important to provide a consistent output prior to majority voting. Optimized routing paths between modules and between the voter and the module are used to ensure minimum propagation delay and no timing mismatch between the voter and the module. Additionally, the architecture is scalable, and can include more layers of redundancy or adaptive reconfiguration if a greater level of fault coverage is needed. By designing the architecture using modules, it can be flexible to be integrated into larger System-on-Chip (SoC) or embedded controller platforms.

VI. WORKING METHODOLOGY

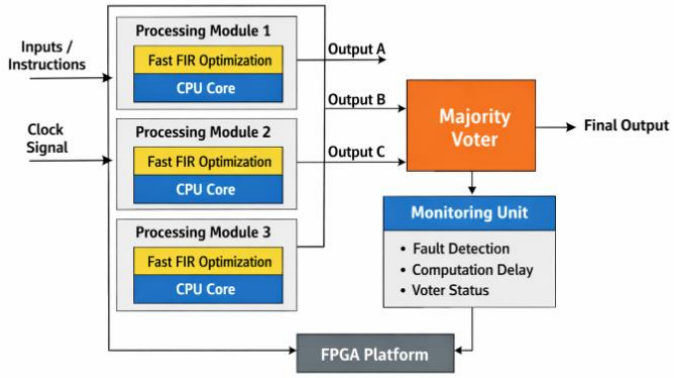

Fig 2 working process

The working approach of the proposed system starts with executing the input instructions at the same time concurrently by three identical processing modules. Each of the modules is running the same computation in parallel to provide redundancy [6]. A transient soft error or thermal disturbance to one module can cause its output to be different from the correct result during operation. The majority voter circuit is continuously comparing the results of all three modules, and setting the output, according to a majority logic. When a pair of outputs are equal, the equal value is chosen as the correct result, the on-column mask is realized and the faulty module is masked. Fast FIR-based optimization in every module minimizes unnecessary registers and buffering elements in order to make the data flow efficiently and to have less power consumption. The optimization keeps processing as fast as real-time and also reduces hardware overhead [7]. To analyze the ability of the system to withstand errors, fault injection is performed. The FPGA implementation proves the practicality and confirms that the system also functions reliably in the event of faults, thus enabling adoption for mission-critical embedded applications. At run-time the system continuously runs in parallel execution mode with identical input data being concurrently delivered to all three processing modules. Most voters process the output dynamically, so no system restart or rollback mechanism is needed to be able to mask all the faults in a single clock cycle. In case repeated faulty events are found in a specific module, the monitoring unit marks the module for diagnostic analysis and in an FPGA implementation, prevents the module from being used or reconfigures it. The fault detection and correction process runs in real-time to ensure continuous processing, computational stability, and consistent system reliability even during thermal stress, radiation, or electrical noise situations

.VII. RESULTS AND DISCUSSION

A fault-tolerant CPU architecture was modeled and synthesized in HDL codes and tested in FPGA platform using real time performance and reliability approach under fault conditions. A set of fault injection techniques was used to inject transient soft errors, thermal disturbances and signal interference at the module level of individual processing modules. A monitoring unit was used to observe the system behavior by recording the internal signals. The experimental results support the validity of the Triple Modular Redundancy (TMR) technique in hiding the effects of single module faults, while providing system operation. In normal operating, all three processing modules were able to produce same outputs as commands were executed, proving same output result with same timing and synchronized execution. In fault injection cases, the results of one of the modules were simulated as faulty, the rest of the modules' output were found to be the same, and the majority voter correctly picked the same outputs from the rest of the modules. The system output was consistent and correct with no errors, indicating the effectiveness of redundancy. The hardware utilization analysis reveals that the use of Fast FIR optimization has resulted in

MSW MANAGEMENT -MULTIDISCIPLINARY, SCIENTIFIC WORK AND MANAGEMENT JOURNAL
ISSN: 1053-7899
VOL. 36 ISSUE 2, 2026, PAGES 287-289

substantial reduction of the number of registers, flip-flops and logic elements used, compared to conventional TMR implementation. This reduction directly helped to reduce silicon area consumed and architectural compactness. Additionally, power analysis shows that the overall dynamic power consumption has been reduced by eliminating unnecessary storage elements and optimizing data-paths. Timing analysis after FPGA synthesis shows the architecture is able to keep the clock functioning reliably and have a reasonable propagation delay even under situations of simulated stress. The overall results show that the proposed system can be applied to mission-critical embedded systems, since it can provide a good trade-off between high fault tolerance, low overhead on the hardware and energy efficiency. VIII. CONCLUSION This paper introduced a combination of hardware optimized TRIPLE MODULAR REDUNDANCY and resource-efficient Fast FIR-based optimizations, which were integrated into a CPU architecture that suits for fault-tolerant applications. The proposed system can provide continuous and reliable operation even though a few transient soft errors, thermal disturbances and signal interference are present. Most of the voter logic "covers up" faults, and hardware optimizations minimize power consumption and area utilization. The system once validated and tested using FPGA showed the real-time performance to maintain stable throughout the validation without compromising the efficiency of the system performance through the fault injection test. The architecture is also well suited for applications that require reliability and stability, such as aerospace, medical, automotive, and industrial automation systems

.Table. 1 Comparative Analysis of Power, Area & Delay on Proposed FPAM

Module	Power Consumption (mw)	Area (device utilization)			delay (ns)
		IOB	LUT	FF	
Existing Method	1.801	32	140	120	7.758
Proposed method	1.065	32	96	96	7.137

IX. FUTURE SCOPE

Adaptive redundancy techniques that adjust fault tolerance in accordance with the environmental conditions and workload requirements can be introduced as an enhancement for the future. By incorporating machine learning-based fault prediction models, the system could be integrated with the hardware to detect degradation and take preventative measures before it becomes an issue. Advanced power management can be used to further optimize the use of energy, such as dynamic voltage and frequency scaling. Furthermore, the architecture can be extended to support multicore and SoC architectures to increase the system's usefulness in high-performance and safety-critical computing applications.

REFERENCE

[1] Fault Detection and Localization of Wind Turbine Sensors Using Output-Only Measurements, IEEE Sensors Journal, 2025. doi: 10.1109/JSEN.2025.3570223

[2] RETO: Energy-Efficient and Reliable Task Offloading Scheme with Deadline in Fog-Enabled IoT Networks, IEEE COMSNETS, 2025. doi: 10.1109/COMSNETS63942.2025.10885694

[3] H. Xia, Z. Wu, L. Zhang, and Y. Song, “Effects of Different Factors on Single Event Effects Introduced by Heavy Ions in SiGe Heterojunction Bipolar Transistors: A TCAD Simulation,” Electronics (MDPI), 2023. doi: 10.3390/electronics12041008

[4] S. K. Sharma and P. K. Verma, “Fault Tolerant FPGA Implementation on Redundancy Techniques and ECG Denoising,” arXiv preprint, 2023. doi: 10.48550/arXiv.2304.08165

[5] A. K. Singh and R. Kumar, “An Improvised Voter Architecture for TMR with Reduced Area Overhead,” IEEE ICICICT Conference, 2022.

[6] Silva, João, et al. "Integrated circuit for magnetic encoder sensing in tmr-based industrial positioning system." IEEE Transactions on Industrial Electronics 71.8 (2023): 9904-9913.

[7] Pires, Jorge, et al. "Architecture Optimization of a CNN Media Noise Estimator for TDMR." IEEE Transactions on Magnetics 60.5 (2023): 1-6.

[8] Dong, Tian, et al. "A 3.5 MHz-BW 128nT rms Resolution TMR Readout Using Ping-Pong Auto-Zeroing and SAR-Assisted Offset Calibration for Contactless Current Sensing." ESSCIRC 2023-IEEE 49th European Solid State Circuits Conference (ESSCIRC). IEEE, 2023.

[9] M. Papadonikolakis and C.-S. Bouganis, “Novel cascade FPGA accelerator for support vector machines classification,” IEEE Trans. Neural Netw. Learn. Syst., vol. 23, no. 7, pp. 1040–1052, Jul. 2012.

[10] P. Lotfi-Kamran, M. Modarressi, and H. Sarbazi-Azad, “An Efficient Hybrid-Switched Network-on-Chip for Chip Multiprocessors,” IEEE Transactions on Computers (TC), vol. 65, pp. 1656–1662, May 2016.

[11] M. Bakhshalipour, P. Lotfi-Kamran, and H. Sarbazi-Azad, “An Efficient Temporal Data Prefetcher for L1 Caches,” IEEE Computer Architecture Letters (CAL), vol. 16, no. 2, pp. 99–102, 2017.

[12] W.Y. Lee, “Energy-Efficient Scheduling of Periodic Real-Time Tasks on Lightly Loaded Multicore Processors,” IEEE Trans. Parall. Distr. Syst., vol. 23, no. 3, pp. 530-537, March 2012

[13] R. Jeyapaul, F. Hong, A. Rhisheekesan, A. Shrivastava, K. Lee, “UnSync-CMP: Multicore CMP Architecture for EnergyEfficient Soft-Error Reliability,” IEEE Trans. Parall. Distr. Syst., vol. 25, no. 1, pp. 254-263, Jan. 2014,

[14] M. Modarressi, A. Tavakkol, and H. Sarbazi-Azad, “Virtual Pointto-point Connections for NoCs,” IEEE Transactions on ComputerAided Design of Integrated Circuits and Systems, vol. 29, no. 6, pp. 855–868, Jun. 2010.

[15] J. Lee, B. Yun, and K. G. Shin, “Reducing Peak Power Consumption in Multi-Core Systems without Violating Real-Time Constraints,” IEEE Trans. Parall. Distr. Syst., vol. 25, no. 4, pp. 1024-1033, April 2014