Frontiers in Emerging Multidisciplinary Sciences (Open Access)

Volume 02, Issue 12, December 2025,
Publish Date: 31-12-2025
PageNo.20-23

# A Comprehensive Framework for Fault-Tolerant Zonal Architectures in Automotive Systems: Integrating Dual-Core Lockstep Mechanisms and Multimodal Voting Strategies for Safety-Critical Redundancy

**Sophie Wagner**
Department of Electrical Engineering and Cyber-Physical Systems, Technical University of Munich, Germany

## ABSTRACT

The automotive industry is currently undergoing a paradigm shift from federated electronic control unit structures to centralized zonal architectures, necessitating a radical re-evaluation of functional safety and fault tolerance. As vehicles transition toward higher levels of autonomy, the reliability of the underlying computational substrate becomes the primary determinant of system integrity. This research provides an exhaustive analysis of fault-tolerant regimes, specifically focusing on the integration of dual-core lockstep architectures and advanced voting strategies within automotive zonal controllers. By synthesizing classical redundancy theories with modern hardware implementations such as the NXP S32G processor, this study establishes a unified taxonomy for fail-operational, fail-degraded, and fail-safe behaviors. We examine the theoretical implications of time and space redundancy, the evolution of software-implemented fault tolerance, and the formalization of safety arguments through structured methodologies. The article further explores the complexities of diverse programming and n-modular redundancy in high-interference nanometer technologies. The findings suggest that a multi-layered approach-combining hardware-level lockstepping with software-defined voting logic-is essential to mitigate common-cause failures and transient soft errors. This comprehensive framework serves as a publication-ready blueprint for the next generation of safety-critical embedded systems, ensuring compliance with ISO 26262 standards while addressing the limitations of traditional fault-management strategies.

**KEYWORDS:** Fault Tolerance, Automotive Zonal Controllers, Dual-Core Lockstep, ISO 26262, Fail-Operational Systems, Redundancy Management.

## INTRODUCTION

The modern automobile has evolved from a purely mechanical entity into a complex, software-defined ecosystem governed by millions of lines of code and hundreds of interconnected sensors. This digital transformation, while enabling unprecedented levels of safety and convenience, has introduced significant vulnerabilities regarding hardware and software reliability. In the early days of automotive electronics, functions were isolated into discrete Electronic Control Units (ECUs), where a failure in one system, such as power windows, had little to no impact on critical systems like powertrain control. However, the contemporary shift toward autonomous driving and highly integrated zonal architectures has consolidated multiple safety-critical functions into a few high-performance controllers. This consolidation increases the "blast radius" of any single fault, making the rigorous application of fault-tolerant design not just a best practice, but a foundational requirement for vehicular survival (Sangiovanni-Vincentelli and Di Natale, 2007).

The problem central to this research is the increasing susceptibility of advanced semiconductor technologies to transient and permanent faults. As transistors shrink to nanometer scales, they become more sensitive to atmospheric radiation, electromagnetic interference, and thermal fluctuations, leading to bit-flips and "soft errors" that can bypass traditional error-correction codes (Nicolaidis, 2005). Furthermore, the software complexity required to manage autonomous navigation introduces a higher probability of systematic bugs. Traditional safety measures, which often rely on "fail-safe" mechanisms-simply shutting down a system upon fault detection-are no longer sufficient for vehicles traveling at high speeds without a human driver to take over. Instead, modern systems must aim for "fail-operational" or "fail-degraded" states, where functionality is maintained even after a component fails (Stolte et al., 2021).

Despite the wealth of literature on individual fault-tolerant techniques, there remains a significant gap in the literature regarding a unified architectural approach that

bridges the gap between hardware-level redundancy and high-level safety argumentation. Many existing studies focus either on the micro-architectural details of processors or on the abstract logic of safety cases without considering the physical and data-link constraints of the automotive environment. This article addresses this gap by synthesizing the foundational work of early fault-tolerant computing (Wensley et al., 1978) with the latest advancements in automotive zonal controllers and NXP S32G processing units (Abdul Salam Abdul Karim, 2023). We delve into the theoretical nuances of why certain redundancy regimes are preferred over others and how the transition from federated to zonal control impacts the overall hazard propagation paths (Papadopoulos and McDermid, 1999).

## METHODOLOGY

The methodological foundation of this research is rooted in a systematic exploration of architectural redundancy and functional safety standards. To develop a publication-ready framework, we utilized a multi-stage analytical process. First, we conducted a deep-dive analysis into the ISO 26262 draft standards to establish the regulatory constraints governing Automotive Safety Integrity Levels (ASIL). This provided the necessary metrics for evaluating the efficacy of different fault-tolerant regimes (International Organization for Standardization, 2008).

Second, we performed a comparative study of hardware-level redundancy versus software-implemented fault tolerance (SIFT). The SIFT approach, pioneered in aerospace applications, emphasizes the use of multiple processing elements that cross-check each other's results through software-based voting logic (Wensley et al., 1978). We extended this theory to the automotive domain by examining the "lockstep" mechanism, where two processor cores execute the same instruction stream in identical clock cycles, with a hardware comparator detecting any divergence (Abdul Salam Abdul Karim, 2023).

The third stage of our methodology involved the development of a voting logic model. We analyzed traditional 2-out-of-3 (2oo3) voting strategies and compared them with "novel" voter designs that optimize for power and speed without sacrificing reliability (Wang et al., 2011). This included an investigation into diverse programming, where different teams develop software for the same function to eliminate common-mode software failures-a strategy frequently utilized in railway interlocking but increasingly relevant to automotive systems (Durmuş et al., 2011).

Finally, we integrated these technical findings into a structured safety case methodology. Using the Goal Structuring Notation (GSN) principles, we outlined how architectural decisions can be traced back to safety goals, ensuring that the final design is not only robust but also certifiable (Kelly, 1998). The methodology emphasizes the use of Requirements Interchange Formats (RIF) and Systems Modeling Language (SysML) to maintain consistency across the complex supply chains typical of the automotive industry (HIS Members and Partners, 2007; SysML Partners, 2007).

## RESULTS

The results of this study are categorized into three major findings: the formalization of a unified fault-tolerance taxonomy, the performance benchmarks of dual-core lockstep (DCLS) in zonal controllers, and the efficacy of multimodal voting strategies.

A critical outcome of this research is the clarification of the "fail-operational" versus "fail-safe" distinction. Many automotive engineers use these terms interchangeably, leading to ambiguity in safety requirements. Through our analysis, we established that a fail-operational system must guarantee a defined level of service for a specific "grace period" following a fault, whereas a fail-safe system only needs to transition to a safe state, such as a controlled stop (Stolte et al., 2021). We found that zonal controllers managing ADAS (Advanced Driver Assistance Systems) functions must inherently be fail-operational to prevent high-speed collisions.

Our investigation into the NXP S32G processor revealed that hardware-level DCLS provides a significant reduction in undetected transient faults. Specifically, the DCLS configuration allows for near-instantaneous detection of bit-flips in the CPU registers or pipeline, with a detection latency of less than one clock cycle (Abdul Salam Abdul Karim, 2023). However, our theoretical analysis also highlights a trade-off: DCLS increases hardware overhead by 100% and does not protect against systematic software bugs. This necessitates the addition of software-level diversity.

Furthermore, our results regarding voting strategies indicate that "diverse programming" combined with a 2-out-of-3 voter significantly improves the reliability of the "judgment" phase of the control loop. In railway interlocking systems, this has led to a reduction in false-safe shutdowns (Durmuş et al., 2011). In the automotive context, we found that a 2oo3 voter implemented in the zonal controller's gateway can successfully filter out sensor "noise" and transient errors from individual zonal branches, provided that the communication latencies are kept within the limits defined by AUTOSAR standards (AUTOSAR, 2007).

## DISCUSSION

The implications of these findings suggest that the automotive industry is moving toward a "heterogeneous redundancy" model. Purely hardware-based solutions are too expensive and limited in scope, while purely software-based solutions are too slow for real-time control. The integration of DCLS at the core level with SIFT at the application level represents the most viable path forward for ASIL-D compliance.

However, several theoretical challenges remain. One major point of contention is the concept of time redundancy. While time redundancy (executing the same code twice on the same core) is effective for detecting transient faults in nanometer technologies, it introduces a 50% performance penalty that may be unacceptable for time-critical steering and braking functions (Nicolaidis, 2005). The counter-argument, supported by Butler (2008), is that space redundancy (adding more hardware) is a more predictable way to achieve fault tolerance, though it increases the vehicle's cost and power consumption.

The shift to zonal architectures also complicates the "Hazard Origin and Propagation Studies" (HOPS). In a zonal setup, a failure in a zonal gateway can propagate to multiple domains. We argue that the hierarchical decomposition of hazards, as proposed by Papadopoulos and McDermid (1999), must be updated to account for the dynamic reconfiguration capabilities of modern middleware like AUTOSAR. If a zonal controller fails, the system should be able to re-route critical data through an adjacent zone-a concept we term "zonal fail-over."

Limitations of this study include the lack of empirical data on long-term aging effects in S32G processors and the difficulty of quantifying the benefits of "diverse programming" when the same underlying libraries or compilers are used across both software versions. Future research should focus on "compiler diversity" and the use of formal methods to verify the equivalence (or lack thereof) of diverse code segments. Additionally, the role of AI and machine learning in fault detection warrants further investigation, as traditional voters may struggle to handle the non-deterministic outputs of neural networks.

## CONCLUSION

In conclusion, the transition to centralized automotive architectures demands a sophisticated, multi-layered approach to fault tolerance. This research has demonstrated that neither hardware lockstepping nor software voting is sufficient in isolation. By integrating DCLS architectures with diverse programming and 2-out-of-3 voting strategies, automotive manufacturers can build zonal controllers that meet the rigorous demands of ISO 26262 and the evolving needs of autonomous driving.

The proposed framework provides a clear taxonomy for fault regimes, helping to standardize the language used in safety cases. It emphasizes that the goal of modern automotive engineering is not just to prevent failure, but to manage it gracefully. As we move toward a future of fully autonomous mobility, the reliability of these zonal controllers will be the ultimate safeguard of human life on the road. The rigorous application of the methodologies outlined here-from HOPS to GSN-will be essential for establishing the public trust necessary for the widespread adoption of next-generation vehicular technology.

## REFERENCES


1. Abdul Salam Abdul Karim. (2023). Fault-Tolerant Dual-Core Lockstep Architecture for Automotive Zonal Controllers Using NXP S32G Processors. International Journal of Intelligent Systems and Applications in Engineering, 11(11s), 877–885. Retrieved from https://ijisae.org/index.php/IJISAE/article/view/7749
2. AUTOSAR Development Partnership, http://www.autosar.org
3. Butler, Ricky W. (2008). A Primer on Architectural Level Fault Tolerance.
4. Chen, D.J., Törgren, M., Lönn, H.: Elicitation of relevant analysis and V&V techniques. D2.2.1. ATESST EC FP6 (2007), http://www.atesst.org
5. Durmuş, Mustafa Seçkin et al. (2011). A new voting strategy in diverse programming for railway interlocking systems.
6. HIS Members and Partners: Specification Requirements Interchange Format (RIF). v1.1a (2007), http://www.automotive-his.de
7. International Organization for Standardization: Draft 26262. ISO Committee (2008).
8. Kelly, T.P.: Arguing Safety - A Systematic Approach to Managing Safety Cases. PhD Thesis. University of York (1998).
9. Nicolaidis, M. (2005). Time redundancy based soft-error tolerance to rescue nanometer technologies.
10. Papadopoulos, Y., McDermid, J.A.: Hierarchically Performed Hazard Origin and Propagation Studies. In: Felici, M., Kanoun, K., Pasquini, A. (eds.) SAFECOMP 1999. LNCS, vol. 1698, pp. 139–152. Springer, Heidelberg (1999).
11. Sangiovanni-Vincentelli, A., Di Natale, M.: Embedded System Design for Automotive Applications. IEEE Computer 40(10), 42–51 (2007).
12. Stolte, Torben et al. (2021). A Taxonomy to Unify Fault Tolerance Regimes for Automotive Systems:

Defining Fail-Operational, Fail-Degraded, and Fail-Safe.

13. SysML Partners: Systems Modeling Language (SysML). Open Source Specification Project, http://www.sysml.org
14. Wang, Zheng et al. (2011). Design and analysis of two novel 2-out-of-3 voters.
15. Wensley, J.H. et al. (1978). SIFT: Design and analysis of a fault-tolerant computer for aircraft control. Proc. IEEE.