AUTOMATIC CONFIGURATION AND IN-MISSION RE-CONFIGURATION IN AVIONICS SYSTEMS: WHY, WHERE, AND HOW?

M. Homolya*, J.-P. Kühn*, B. Lüttig*, B. Annighöfer*
* University of Stuttgart, Institute for Aircraft Systems, Pfaffenwaldring 27, Stuttgart, Germany

Abstract

Current avionics platforms rely on strictly static configuration data to comply with safety requirements. This ensures predictable behaviour but brings disadvantages in the platforms' efficiency. Unused payload, rarely activated reserved bandwidth, and standby redundancy have a huge potential for improving network performance, reliability, resource allocation, and power consumption just to mention a few.
The Plug-and-Fly Avionics Platform (PAFA) was introduced at the University of Stuttgart, demonstrating automatic configuration in a self-configuring octocopter testbed. This work extends the concept to a passenger aircraft cabin demonstrator. We evaluate the requirements, benefits, and limitations of automatic configuration and in-mission re-configuration.
An important cornerstone of in-mission re-configuration lies within determining the trigger events and detecting the needs and conditions for re-configuration and deriving new configuration parameters. For instance, when introducing new systems in an aircraft, re-configuration must be limited to parts that actually support this new feature and the not effected system parts still must work. While this work aims at the less safety-critical cabin domain, there are interfaces within safety-critical aircraft domain. Whereas we do not intend the re-configuration of safety-critical flight-controls, we see potential in the automatic configuration of communication interfaces from the aircraft to the cabin domain, e.g., for health monitoring or cabin dashboards.
This paper quantifies the need and potential applications for automatic configuration, how it potentially affects development cost and future technological benefits with respect to safe dynamic (re-)configuration.

Keywords

avionics; configuration; re-configuration; automation

1. INTRODUCTION

Modern avionics communication systems are increasingly complex, imposing stringent requirements for certification, reliability and safety. Aircraft networks both cabin and flight-control networks must support a wide range of functions, flight-critical systems, passenger services, and also maintenance and ground operations. With the wide variety of functions alongside traditional safety demands adaptability and run-time flexibility also became essential.
To ensure predictable behaviour, current avionics platforms rely on static, manually engineered configuration data. These configurations define communication paths, function allocation, bandwidth assignments, and redundancy strategies prior to deployment and introduce limitations. The strict safety requirements create overhead in multiple components in the form of reserved bandwidth, idle redundant components, and underutilised or empty payloads. All these components have to be configured which results in millions of configuration parameters and makes updates, retrofits, or customer-specific adaptations time-consuming, error-prone, and costly when performed manually.

Traditional configuration methods do not support dynamic adaptation at run-time, but rely on static, predefined configuration data to guarantee predictable and secure operation. While this ensures correctness, it prevents systems from responding flexibly to changing conditions. Dynamic adaptation, however, would enable platforms to react to configuration changes, integrate subsystems automatically, or adjust resource allocations in response to varying network traffic demands in different flight phases. As modern avionics platforms become increasingly complex, i.e., extending network protocols, growing number of software components, varying level of safety criticality, and real-time requirements, the design, engineering, and maintenance effort also grows. To cope with the described complexity, adaptability is needed to to mitigate the human effort induced by complexity. Automatic configuration and re-configuration provide an approach to solve this.
This paper examines the needs, benefits, and challenges of automatic configuration and in-mission re-configuration in avionics systems with the following questions: (RQ1) What reconfiguration approaches are suitable for avionics systems that are both automatically configurable and re-configurable? (RQ2)

What scenarios in the cabin require automatic configuration and re-configuration of avionics subsystems? It explores potential applications from routine operations to fault-recovery scenarios. Furthermore, it examines approaches to reduce development effort and cost, improve resource utilisation, and enable more flexible and adaptive system behaviour. It also considers future benefits of integrating dynamic configuration and re-configuration to increase efficiency and reliability to meet evolving demands.
This paper is according to the following structure. Section 2 introduces the background and related work, section 3 presents the re-configuration scheduling. Section 4 details the requirements, section 5 discusses the automatic configuration. Section 6 describes general framework and the re-configuration scenarios. Section 7 explores the s the system architecture and workflow, section 8 answers the research questions and concludes the paper.

2. RELATED WORK

There is an overall tendency for increasing automation and integrating semi-automatic configuration in other domains too [1]. These tend to aim towards to easy software integration, allow reusability, automated testing, and to increase compatibility, while still allowing a reduction in development costs and to keep up with the safety and security standards and requirements [2]. Previous studies in avionics and other safety-critical systems have explored the effect of automatic or semi-automatic configuration, to reduce manual effort and improve flexibility [3]. These approaches allow the dynamic detection and integration of new modules, subsystems or functional components, this also means lower system downtime and engineering overhead [4] [5].
Platforms like PAFA (Plug-and-Fly Avionics), integrate automatic configuration with resource management, and task allocation [6]. This approach demonstrates improvement in efficiency, fault tolerance, and flexibility. Despite these advances, there are plenty of challenges remaining for a fail-safe in-mission re-configuration that fulfils the requirements for avionics platforms. Where fail-safe means that any re-configuration must ensure the system continues to operate correctly, preventing faults or inconsistencies.

3. RE-CONFIGURATION SCHEDULE

A re-configuration has to be carefully timed in order to maintain the predictable system operation and safety while changes are being applied. Several strategies are common in distributed and safety-sensitive environments:

1) **Gradual Reconfiguration:** Only a subset of components is updated at a time, for example one partition, one line-replaceable unit, or one network segment. The remaining components continue to operate with the old configuration, ensuring redundancy. After successful verification, the roll-out proceeds to the next defined subset (see Figure 1, as also presented in [7].

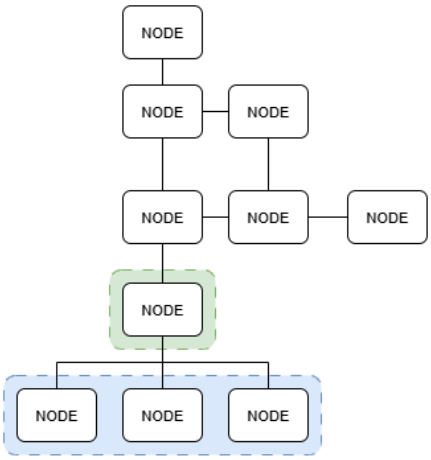

FIG 1. Gradual Reconfiguration. The nodes updated in the first subset are highlighted with blue, while the update proceeds to the next subset marked green.

2) **Bank Switching:** For this scheduling the configuration files can be created on ground, or a new configuration can be generated in-mission. Which is then deployed and validated in parallel to the old one, and stored on the device, once all needed configurations are deployed, the shadow configuration is switched over with the running configuration [8] [9].
3) **Graceful Switchover:** The re-configuration occurs in explicitly defined time windows, aligned with low system load, or ground operations. With system-wide synchronization it is ensured that all affected nodes switch configuration simultaneously, and so no intermediate configuration states are present [7] [9]. This is especially relevant when consistency across distributed notes is critical.
4) **Event-Triggered Partial Updates:** In case of an event, only those devices are re-configured that are affected by the trigger event. For example, if an end device joins the network, only the devices that are involved in the communication management of the new device need to be updated and extended. This defines a network segment for the re-configuration where it has to take place, and limiting the number of devices effected by it, similar concepts have been applied in Software Defined Networks [10]. However, this requires high-level monitoring, rollback mechanism, and well-defined trigger events to ensure that the activation of the new configuration is correct and consistent.
5) **Safe-State Re-configuration:** In this case before a re-configuration is applied, subsystems are placed into a defined "safe-state", such as a fallback mode or redundancy takeover [9]. This guarantees that no unsafe intermediate state is exposed to operations, and that all parameters not

in the new configuration are removed or reverted to default state.

6) **Predictive Scheduling:** This approach requires continuous monitoring of the active and planned processes, and transmissions, to predict the time window when the re-configuration would cause minimal disruption of the overall system, similarly to [11]. The re-configuration is scheduled proactively, to ensure that critical changes are not applied during peak load or highly dynamic transmission phases.
7) **Hybrid Approach:** Multiple studies in different domains [12], [13], [14] show that, integrating multiple re-configuration strategies can be a solution to increase reliability and adaptability with automatic configuration and re-configuration. For example, combining static and dynamic scheduling, as demonstrated in [12], shows improved reliability for real-time tasks in re-configurable devices. Partial re-configuration techniques, presented in [14], provide mechanisms to mitigate hardware faults, while maintaining operational continuity for critical applications. Overall, hybrid re-configuration strategies can ensure that critical system functions remain safe and predictable.

4. REQUIREMENTS FOR SAFETY AND SECURITY

For avionics systems safety and security requirements are central; the following sections outline them, and how they effect the configuration handling.

4.1. Safety

The design of avionics communication platforms is tightly bound to stringent safety and certification requirements, which directly shape the way configuration mechanisms must be designed and validated. Avionics systems must provide determinism and real-time guarantees. Communication between critical systems is subject to hard latency and jitter bounds, and schedules are configured in a way that ensures predictable end-to-end performance. This requires the configuration mechanism to ensure predictable performance, maintaining schedules that respect these bounds. This is consistent with standards such as DO-178C [15] and ARINC 653 [16], which emphasize determinism and partitioning in safety-critical software and systems. Comparable requirements also apply to data networking standards such as ARINC 664 Part 7 (AFDX) [17], which specifies deterministic Ethernet for avionics. Therefore, configuration frameworks must preserve deterministic timing across connected nodes.

Design Assurance Levels (DAL) classify the criticality of avionics functions, with higher levels (A and B) requiring stricter safety and reliability measures and the relevant objectives to be fulfilled. Network configurations must tolerate single-point failure, ensure seamless switchover between redundant paths, and maintain consistency across distributed systems. These requirements can be supported by feedback-based adaptive loops such as MAPE-K (Monitor, Analyze, Plan, Execute with Knowledge), which continuously monitor system states, analyze deviations, plan safe adjustments, and execute configuration changes while preserving deterministic and fail-safe operation. This creates a strong requirement for configuration frameworks, as toolsets that manage and verify system configurations, while operating fail-safe and coordinating across multiple nodes without introducing inconsistencies.

Avionics certification further requires traceability and verifiability of all configuration artifacts. Automated or dynamic configuration mechanisms therefore not only have to produce correct schedules but also demonstrate predictability and reproducibility, since traceable and verifiable artifacts allow certification authorities to confirm compliance. Relevant standards such as DO-297 [18] for Integrated Modular Avionics (IMA) and DO-326A [19] for Airworthiness Security, establish that both functional partitioning and protection against unauthorised modification are required for safe and secure operation.

The growing complexity of modern aircraft also introduces scalability and integration requirements. Modern avionics platforms have to integrate an increasing number and range of subsystems, from flight control and navigation, to passenger services, within shared networks. In such system-of-systems environments, configuration approaches must guarantee that determinism is not compromised while supporting flexible integration.

4.2. Security

Another key aspect of avionics systems is cybersecurity and integrity. Misconfigurations, system faults, or malicious actions - can negatively affect the systems security and integrity. A recent work on cybersecurity challenges in self-adaptive avionics, using the Plug & Fly Avionics (PAFA) platform as a case study, shows that conventional security solutions are often inadequate and suggests that emerging technologies, such as blockchain, Zero-Trust architectures, and multilayered security, offer promising solutions [20]. Standards such as DO-326A [19] require that configuration process incorporate the protection against unauthorised modification and ensure integrity throughout the lifecycle. Configuration and re-configuration mechanisms implement access control, verification for integrity, and validation for the deployed configurations.

Finally, avionics configurations must address security, maintainability, and lifecycle considerations. The aircraft remains in service for decades, and updates or re-configurations must be manageable without undermining certification artifacts. The long certification cycles require configuration frameworks that are stable and adaptable throughout the operational

service.

5. AUTOMATIC CONFIGURATION

Automatic configuration refers to the systematic process of generating and deploying configuration data based on formal system and component descriptions, rather than manual engineering input. It defines the relationships between modules, assigns configuration parameters, and establishes the communication structure automatically during system initialization.
Automating the generation and deployment of configurations simplifies system setup and integration, thereby reducing engineering workload, and human errors. This can improve reliability and consistency, while also enabling reusability of components.
Automatic configuration relies on formalised system and component models to allow the detection of available modules, the determination of their dependencies, and to configure the communication interfaces. The use of validated configuration modules further decreases engineering effort for review and certification.
The automatically generated initial configuration provides a verified baseline, defines network and system state, from which re-configuration can proceed. This supports efficient integration, installation, and testing, ensuring system consistency.

6. GENERAL FRAMEWORK AND RE-CONFIGURATION SCENARIOS

Based on the requirements outlined above, the general framework for automated configuration must be reliable that aligns with safety, certifiability, and lifecycle demands. Therefore has to serve as a base for distributed configuration, fail-safe operation and updates, defining how components monitor, decide, and implement configuration changes across the system.
A good example to illustrate this is cabin electronics. While cabin systems are typically less safety-critical (maximum DAL B) compared to flight-control (DAL A) or navigation systems (DAL B), they present an environment suited for automatic configuration and reconfiguration, supporting the addition and extension of cabin subsystems. Cabin subsystems like passenger service units, cabin lighting, and infotainment are frequently updated or extended during the overall aircraft lifecycle. Airlines might add new in-flight entertainment modules, or integrate new, or exchange other technologies. These changes have connection for the underlying network configuration, and additional traffic flows must be integrated without compromising determinism or interfering with safety-critical domains. The cabin environment therefore offers a representative case for exploring scalable, flexible, and fail-safe re-configuration processes.

6.1. Scenarios

Within this context, several re-configuration scenarios can be identified that demonstrate the value of an automated, distributed approach. These scenarios can be categorised as system integration, failure recovery, lifecycle updates, and dynamic adaptation.

6.2. System Integration: System Extension

Adding new modules, such as passenger service units or sensors, extends the network and needs additional communication paths, and extension of signal assignment. The configuration framework must integrate these modules seamlessly, assigning proper priority classification, bandwidth, and the applicable configuration parameters, and incorporate into the network, while minimising manual errors and reduce integration time.

6.3. Failure Recovery: Cabin Network Switch Failure

If a network switch or end system fails, traffic must be rerouted automatically to maintain continuous operation. The configuration framework should dynamically update the routing, reassign and adjust traffic to ensure that critical data streams are forwarded and still meet latency and reliability requirements. Automated failure recovery reduces downtime, prevents data loss, and minimises the need for manual intervention.

6.4. Lifecycle Updates: Maintenance Software Download

Software updates are periodically applied to networked systems during maintenance. The framework must validate and apply new configurations safely, preserving certified real-time behaviour. Once validated, the framework has to apply the new configurations automatically, preserving the behaviour and ensuring that correct operation is continuous. Automated updates reduce manual errors, hence reduces maintenance time.

6.5. Hardware Replacement: Light Fixture Change

Replacing or upgrading cabin lighting requires updates to the control network. The framework must detect the change and adjust traffic allocation to ensure lighting commands remain timely and consistent.This automatic device discovery and configuration reduces manual workload and enables seamless integration of new components into the deterministic communication domain.

These scenarios show how automated configuration meets real-world avionics demands. And it demonstrates how even in a non-critical domain such as cabin-electronics automated configurations and reconfigurations can reduce workload and engineering overhead. With automatic device detection, routing,

and traffic assignment, it lowers manual intervention, the possibility of configuration errors, and shortens integration and maintenance times.

7. SYSTEM ARCHITECTURE AND WORKFLOW

This section outlines the suggested system architecture and workflow, detailing the components and their roles for reliable and adaptive configuration management. This allows the system to automatically adjust its configuration to maintain safety, reliability, and performance under changing conditions. The general workflow is illustrated in figure 2, showing the process, and the following subsections describe the main elements.
The Configuration Manager (CM) acts as the global coordinator, maintaining the system and orchestrating the configuration activities across the nodes. The Distributed Agents (DAs) act as local executors, managing the configuration and re-configuration of the respective nodes, while supporting the monitoring and fail-safe mechanisms. The optimisation functions are to maintain determinism, minimise latency, and ensure system performance for network traffic.

7.1. Configuration Manager & Distributed Agents

The Configuration Manager serves as the central authority, orchestrates and validates all configuration activities. The CM generates the configuration artifacts and deploys them to the Distributed Agents. The Distributed Agents operate on the node level, executing the configurations locally, and checks that the configuration changes have been correctly applied. Each agent can make local decisions, report status, and apply configuration updates, allowing the system to operate in a distributed and scalable manner.

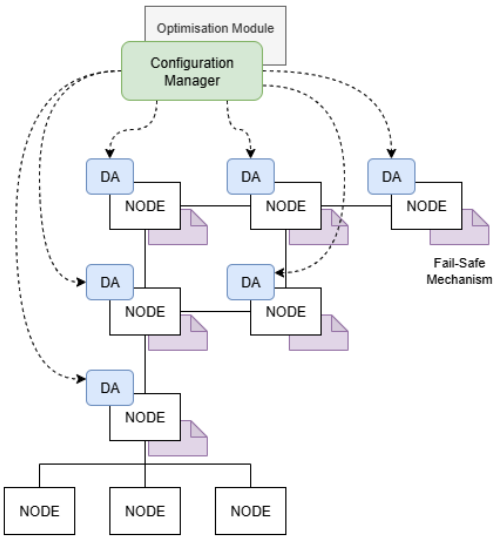

FIG 2. Overview of a configuration framework, showing the interaction between the central Configuration Manager (CM), the Distributed Agents (DAs) that act upon Nodes, the Fail-Safe mechanism (purple) assigned to the required Nodes, and the Optimisation module.

7.2. Monitoring & Triggering

This component continuously observes the network states, tracks performance metrics, and detects events that require re-configuration, such as the addition of new subsystems or changes in operational conditions. When such events are detected, the system triggers the appropriate re-configuration processes, initiating communication between the DAs and the CM.

7.3. Fail-Safe Mechanism

To maintain system reliability, the Fail-Safe Mechanism provides a controlled fallback in case of node failures, transmission errors, or misconfigurations. The main requirements are to ensure that critical functions continue to operate correctly, preserve stable system behaviour under unexpected conditions, prevent inconsistencies in intermediate configuration states, and that re-configuration does not compromise the network behaviour. These are provided through mechanisms such as predefined fallback configurations, safe rollback procedures, and verification of re-configuration steps before they are applied.

7.4. Optimisation Module

This component ensures valid allocation of available network resources, and adapts to the changes in demands. It balances priorities and optimises throughput depending on the system's requirements. It is also needed to support incremental re-configuration to minimise disruption.

8. DISCUSSION AND CONCLUSION

This section summarizes how automatic configuration and in-mission re-configuration can increase efficiency, adaptability, and resource utilisation in avionics systems, while also outlining the critical factors, benefits, and challenges associated with these approaches.
In-mission re-configuration introduces additional challenges and opportunities in safety-critical environments. To maintain safe and predictable system behaviour, the system must provide determinism, real-time guarantees, redundancy, fault tolerance, and consistency across distributed nodes. Safe-state and controlled intermediate state transitions have to be ensured to prevent compromising system integrity whenever the configuration is updated.
The potential benefits are reducing manual engineering effort and enabling faster integration of new modules with automation, improving bandwidth and resource utilisation, and minimising downtime during updates. These savings support safe, reliable, and timely mission operation. These enhance operational efficiency and mission reliability. Implementing these mechanisms is complex and introduces certification and verification challenges. Clearly defined events and scheduling mechanisms are therefore required to

ensure that re-configurations are applied safely and predictably.
The scenarios presented in Section 6 and the framework concept in Section 7 demonstrate how these benefits are achieved in practice. The discussed scenarios, system integration, lifecycle updates, hardware replacement, and failure recovery show re-configuration approaches for avionics systems that are both automatically configurable and re-configurable (RQ1), as well as cabin use cases that require automatic configuration and re-configuration of avionics subsystems (RQ2). The Configuration Manager of the framework coordinating Distributed Agents provides scalable and consistent management across complex avionics networks, enabling safe and efficient in-mission re-configuration.

8.1. Conclusion

This paper highlights the role of automatic configuration and in-mission re-configuration within avionics systems. When applied with the correct approach, they can reduce manual effort during both engineering and maintenance works. Automated frameworks support compliance with requirements and resource management already during design and integration stages. Automated configuration also reduces maintenance downtime by minimising the need for manual intervention, which saves time and reduces the risk of human error. Continuous monitoring is required to enable in-mission re-configuration, and it also contributes to system health management and enhances system reliability.

The combined insights from scenarios and the framework concept demonstrate that automatic configuration and in-mission re-configuration can provide suitable re-configuration approaches for avionics systems (RQ1) and address cabin scenarios that are relevant to automatic configuration and re-configuration (RQ2).

Contact address:

mariann.homolya@ils.uni-stuttgart.de

References

[1] Raphael Trindade, Lukas Bulwahn, and Christoph Ainhauser. Automatically generated safety mechanisms from semi-formal software safety requirements. volume 8666, pages 278–293, 09 2014. ISBN: 978-3-319-10505-5. DOI: 10.1007/978-3-319-10506-2$_19$.

[2] Jiri Barnat, Jan Beran, Luboš Brim, Tomas Kratochvíla, and Petr Ročkai. Tool chain to support automated formal verification of avionics simulink designs. 08 2012. ISBN: 978-3-642-32468-0. DOI: 10.1007/978-3-642-32469-7$_6$.

[3] Hannes Stoll, Daniel Grimm, Marc Schindewolf, Michel Brodatzki, and Eric Sax. Dynamic reconfiguration of automotive architectures using a novel plug-and-play approach. In *2021 IEEE Intelligent Vehicles Symposium Workshops (IV Workshops)*, pages 70–75, 2021. DOI: 10.1109/IVWorkshops54471.2021.9669222.

[4] Chuanwen Lin, Gang Chen, and Zhenhua Liu. Development and trend of space plug-and-play avionics. *Journal of Physics: Conference Series*, 1544(1):012171, may 2020. DOI: 10.1088/1742-6596/1544/1/012171.

[5] Bastian Luettig, Jona Dallmann, and Bjoern Annighoefer. Adima: Automatic configuration by peripheral detection and adaptive distributed task execution for integrated modular avionics platforms. In *2022 IEEE/AIAA 41st Digital Avionics Systems Conference (DASC)*, pages 1–10, 2022. DOI: 10.1109/DASC55683.2022.9925885.

[6] Björn Annighöfer, Johannes Reinhart, Matthias Brunner, and Bernd Schulz. The concept of an autonomic avionics platform and the resulting software engineering challenges. In *2021 International Symposium on Software Engineering for Adaptive and Self-Managing Systems (SEAMS)*, pages 179–185. IEEE, May 2021. DOI: 10.48550/arXiv.2103.12065.

[7] Victor Braberman, Nicolas D'Ippolito, Jeff Kramer, Daniel Sykes, and Sebastian Uchitel. Morph: a reference architecture for configuration and behaviour self-adaptation. In *Proceedings of the 1st International Workshop on Control Theory for Software Engineering*, CTSE 2015, page 9–16, 2015. ISBN: 9781450338141. DOI: 10.1145/2804337.2804339.

[8] Damian Wanta, Waldemar T. Smolik, Jacek Kryszyn, Przemysław Wróblewski, and Mateusz Midura. A run-time reconfiguration method for an fpga-based electrical capacitance tomography system. *Electronics*, 11(4), 2022. ISSN: 2079-9292. DOI: 10.3390/electronics11040545.

[9] Arne Schwabe, Elisa Rojas, and Holger Karl. Minimizing downtimes: Using dynamic reconfiguration and state management in sdn. pages 1–5, 07 2017. DOI: 10.1109/NETSOFT.2017.8004209.

[10] Ll. Gifre, F. Boitier, C. Delezoide, M. Ruiz, M. Buffa, A. Morea, R. Casellas, L. Velasco, and P. Layec. Monitoring and data analytics-triggered reconfiguration in partially disaggregated optical networks. In *2020 22nd International Conference on Transparent Optical Networks (ICTON)*, pages 1–5, 2020. DOI: 10.1109/ICTON51198.2020.9203059.

[11] Zain A. H. Hammadeh, Monowar Hasan, and Mohammad Hamad. Securing real-time systems

using schedule reconfiguration. In *2024 IEEE 27th International Symposium on Real-Time Distributed Computing (ISORC)*, pages 1–10, 2024. DOI: 10.1109/ISORC61049.2024.10551328.

[12] Abolfazl Ghavidel, Yasser Sedaghat, and Mahmoud Naghibzadeh. Hybrid scheduling to enhance reliability of real-time tasks running on reconfigurable devices. *The Journal of Supercomputing*, 76(6):4701–4730, 2020. DOI: 10.1007/s11227-019-02976-6.

[13] Bei Tian, Gang Xiao, and Yu Shen. A predictive-reactive strategy for flight test task scheduling with aircraft grounding. *Complex Intelligent Systems*, 10:4329–4349, 2024. DOI: 10.1007/s40747-024-01365-8.

[14] Charlotte Frenkel, Jean-Didier Legat, and David Bol. A partial reconfiguration-based scheme to mitigate multiple-bit upsets for fpgas in low-cost space applications. In *2015 10th International Symposium on Reconfigurable Communication-centric Systems-on-Chip (ReCoSoC)*, pages 1–7, 2015. DOI: 10.1109/ReCoSoC.2015.7238095.

[15] RTCA. Software considerations in airborne systems and equipment certification. Technical Report DO-178C, 2011.

[16] Aeronautical Radio. Avionics application software standard interface. Technical Report ARINC Specification 653, 1997.

[17] Aeronautical Radio. Avionics full duplex switched ethernet (afdx) network. Technical Report ARINC Specification 664 Part 7, 2002.

[18] RTCA. Integrated modular avionics (ima) development guidance and certification considerations. Technical Report DO-297, 2005.

[19] RTCA. Airworthiness security process specification. Technical Report DO-326A, 2010.

[20] Aisha Zahid Junejo, Mario Werthwein, and Bjoern Annighoefer. A comprehensive analysis of cybersecurity challenges in self-adaptive avionics: A plugfly avionics platform case study. In *2025 IEEE/ACM 20th Symposium on Software Engineering for Adaptive and Self-Managing Systems (SEAMS)*, pages 133–139, 2025. DOI: 10.1109/SEAMS66627.2025.00022.