State-of-the-Art Technologies for Integrated Modular Avionics and the Way Ahead

Bojan Lukić
Institute of Flight Systems
German Aerospace Center (DLR)
Braunschweig, Germany
Email: bojan.lukic@dlr.de

Alexander Ahlbrecht
Institute of Flight Systems
German Aerospace Center (DLR)
Braunschweig, Germany
Email: alexander.ahlbrecht@dlr.de

Sven Friedrich
Institute of Flight Systems
German Aerospace Center (DLR)
Braunschweig, Germany
Email: sven.friedrich@dlr.de

Umut Durak
Institute of Flight Systems
German Aerospace Center (DLR)
Braunschweig, Germany
Email: umut.durak@dlr.de

Abstract—Towards the end of the 20th century, the aviation industry started to adopt the Integrated Modular Avionics (IMA) architecture. It describes an airborne system with a unified design and standardized components. This allows application software to be used on various hardware modules that share common features. The previously established federated avionics architecture, which describes the self-containment of avionics functions as Line-Replaceable-Units, would therefore gradually become obsolete. A major advancement of IMA is its computing paradigm. With the allocation of multiple functions to single processing units, IMA systems show a higher efficiency, modularity, and maintainability compared to the federated systems. This paper motivates the journey to the next generation of IMA systems. For this new generation of avionics systems, highly adaptive and integrated structures, i.e., computing platforms, are being developed and implemented. The novel design enables the reconfiguration and reassignment of safety-critical applications to counter total system failures. The deployment of advanced airborne applications is enabled due to to more powerful processing units and wireless technologies. The study presents the state-of-the-art in designing ARINC 653-compliant IMA systems as well as development efforts for future IMA architectures. Three major points are discussed in the paper. First, IMA as a technology is presented. Second, the status quo and development efforts for IMA systems are discussed. This refers mainly to the research performed for state-of-the-art avionics systems. Third, the requirements for next generation IMA with some potential implementations are discussed. The research shows that IMA is invaluable for aviation systems and there will be a major shift to more advanced software and hardware technologies with future IMA systems. Nevertheless, there are many emerging requirements yet to be met with this next generation of IMA.

Index Terms—Integrated Modular Avionics, space and time partitioning, avionics full-duplex switched ethernet, ARINC 653 dynamic reconfiguration, avionics wireless network

I. INTRODUCTION

Integrated Modular Avionics (IMA) is a technology that has revolutionized the aerospace industry. It is an advanced avionics architecture that has been replacing the traditional federated architecture used in aircraft since the 1970s [1]. In IMA, multiple avionics applications are consolidated onto a common computing platform, which allows for more efficient system integration and management. The concept of IMA was first proposed in the 1990s by the Aircraft Electronics Association (AEA) and the Radio Technical Commission for Aeronautics (RTCA). It was developed as a response to the increasing complexity of avionics systems in aircraft and the need for a more flexible and scalable architecture. The goal was to create a modularized architecture that could be easily upgraded, maintained, and customized without requiring extensive reconfiguration or redesign. Since the proposal, several guidances and standards for the development of IMA have been introduced. Some relevant examples are [2, 3, 4] issued and published by the Airlines Electronic Engineering Committee (AEEC), RTCA, Federal Aviation Administration (FAA), and European Union Aviation Safety Agency (EASA). Today, IMA is used in various aircraft, including commercial airliners, military aircraft, and business jets. It has become the industry standard for avionics architecture and has led to significant improvements in system integration, reliability, and maintenance. Using IMA, manufacturers can reduce the time and cost associated with aircraft design and development, while also improving safety and performance [5, 6].

IMA architectures provide multiple benefits over traditional federated avionics architectures. One major advantage is the ability to consolidate multiple avionics applications onto a common computing platform. Applications are processed in partitions, with multiple partitions sharing the same hardware resources. This is an improvement to the federated avionics architecture, which describes the less efficient self-containment of avionics applications as Line-Replaceable-Units (LRUs). While increasing efficiency concerning hardware utilization and avionics modularity, the modular avionics architecture provides a spatial and temporal separation during computation. Thus, as specified in the ARINC standard 653[1] [2], a sufficient

---

[1]Software specification for space and time partitioning in safety-critical avionics real-time operating systems (RTOS).

degree of safety for use in aircraft is also ensured. The partitioning concept is illustrated in Fig. 1.
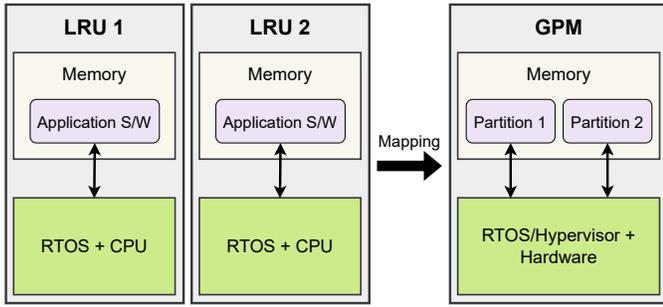

Fig. 1: Application allocation in a federated compared to a modular avionics architecture (right)

The modular design of IMA also facilitates easy upgrades, customization, and reconfiguration by replacing or adding modules without requiring extensive rewiring or redesign. One of the most significant features of IMA is its capability to support integrated, advanced system features, such as flight management systems, cockpit display alerts, and navigation databases.

Over the years, IMA developed further into reconfigurable IMA. Reconfigurable IMA is capable of changing the allocation of system applications to partitions, leveraging an even higher degree of system flexibility and efficiency [7]. The next generation of IMA systems will be deployed in an even more integrated and complex system environment with advanced flight applications. Besides the integration of multi-core processors [8], versatile application interfaces [9] and functionalities such as Graphics Processing Units (GPUs) for advanced flight assistance will be implemented on a homogeneous platform. Integration of these additional functionalities supports the development of more elaborate and complex functions such as automated collision avoidance or autonomous flight [10, 11]. However, these developments also come with increased system complexity, and challenges with reliability, certifiability, and safety. One of the key challenges is ensuring that the system components can be integrated and reconfigured dynamically to meet the changing needs of the system and meet safety requirements [10, 12, 13, 14]. On the other hand, the inclusion of wireless network technologies requires more vigilant cybersecurity measures. These challenges need to be addressed with development efforts from research institutions and the industry [6].

This paper makes a contribution to the development efforts for the next generation of IMA. Specifically the adoption of new technologies for safe IMA architectures with increased computational and dynamic reconfiguration capabilities are discussed. The remaining paper is structured as follows: In chapter II, the background on IMA as a technology is presented. The status quo in IMA development is discussed in chapter III. The main part of the paper is chapter IV, discussing the requirements for the next generation of IMA. The paper is concluded with chapter V.

II. BACKGROUND ON IMA

IMA architectures consist of several components, most of which are standardized. The essential components are Core Processing Input/Output Modules (CPIOMs), or more generally General Processing Modules (GPMs), Remote Data Concentrators (RDCs), and the network interface connecting all components for communication. The general parts of an IMA systems are discussed in more detail in the following subsections.

A. General Processing Module

GPMs are computing modules found in IMA systems. They run application software and have the ability to be replaced without any necessary changes to the software or hardware interfaces to the other modules in the IMA network. This flexibility makes the design and maintenance of IMA systems easier, as GPMs can be easily swapped out to upgrade or repair specific functions within the avionics suite. GPMs have high safety standards in their construction and adhere to strict design processes such as DO-178C[2] [15], DO-254[3] [16], and DO-297[4] [3]. In this context the two guidelines for development and safety assessment on civil airborne systems, ARP4754A [17] and ARP4761 [18], are noteworthy. An illustration of such a GPM is shown in Fig. 2.
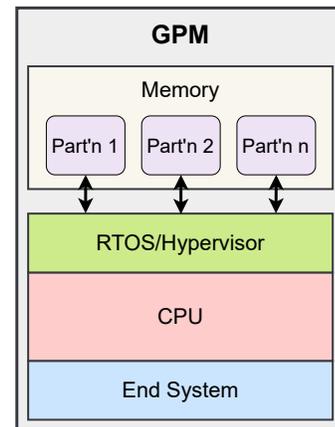

Fig. 2: High-level illustration of a General Processing Module (GPM) in an Integrated Modular Avionics (IMA) system

In its most basic form, a GPM in an IMA system contains a CPU, a hypervisor, an end system, and a real time operating system (RTOS). The end system is an interface serving as a source or receptor of data on a network. A hypervisor is software that creates and runs virtual machines (VMs) and allows a single host to have multiple VMs running on it, each with its own operating system and applications. The hypervisor

[2]Software development standard for safety-critical avionics software with a focus on safety, reliability, and performance considerations.
[3]Standard for the design assurance of avionics hardware applying to the entire hardware development life cycle, from initial planning to final certification.
[4]Standard for the certification and development of IMA systems regarding system architecture, software development, and verification.

sits between the physical hardware and the operating system of each VM, effectively decoupling the hardware from the software [19]. Depending on the type of hypervisor, an OS layer may be implemented between hypervisor and the hardware, representing the type-2 hypervisor. For simplicity reasons, the GPMs discussed in this paper assume the use of type-1, bare-metal hypervisors. Through virtualization, partitions can be created, which run applications. Depending on the system complexity, several tasks belonging to one application can be executed within a partition. Fig. 3 shows a basic partitioning architecture with two partitions in one GPM. Partition 1 runs task 1.1 and partition 2 runs tasks 2.1 and 2.2.
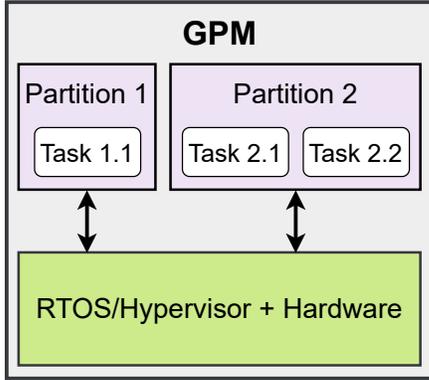

Fig. 3: Basic partitioning scheme for a GPM in an IMA system

Each partition receives a designated time on the CPU for computing. The computing of all partitions in one GPM happens in the time of one major frame. The major frame in a GPM can be divided into partition time windows of arbitrary length for each partition, but at most the length of one major frame. The partition schedule is enforced by the RTOS which dictates the partition access for executing tasks on the processor. Once the partition schedule is finished, it starts from the beginning. This process is repeated indefinitely until some stopping criterion takes effect [20]. Due to the spatial and temporal separation of partitions being guaranteed according to ARINC 653 [2], applications of mixed critically can be run on one GPM. Spatial separation refers to tasks or processes not interfering with one another by running them in different memory locations or addressing spaces. An exemplary illustration for the partition time windows of two partitions in one major frame is shown in Fig. 4.
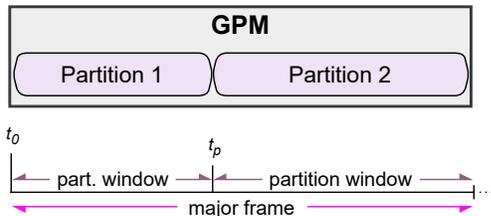

Fig. 4: Computing times for two partitions in a GPM

The tasks inside a partition are running within the time

frame of the partition time window. If no task is executed, the CPU is idle. An illustration of the designated computing times for tasks inside a partition is illustrated in Fig. 5.
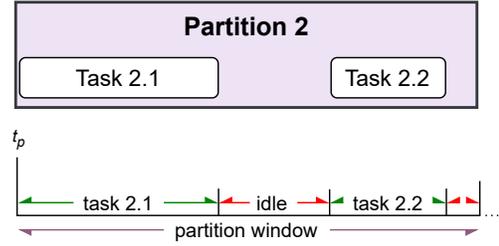

Fig. 5: Computing times of two tasks in a partition

One more component of GPMs are the end systems, which communicate between Input/Output Modules (IOMs), such as GPMs, and the switches in the IMA system. The network along with its relation to the IOMs is discussed in the next subsection.

B. Network

In IMA systems, the network interface serves as a vital component in facilitating communication between avionics modules. The ARINC 664 standard [21] describes the network of IMA alongside the general network architecture of the whole aircraft. The network is also known as the Avionics Full-Duplex Switched Ethernet (AFDX) and provides high-speed data transfer between avionics subsystems. The protocol offers determinism, prioritization, and redundancy features that are necessary for safety-critical systems. Avionics architectures need to be capable of hosting legacy systems, such as LRUs. The use of standard software interfaces simplifies the communication between interface modules with legacy avionics systems. Overall, the network interface plays a critical role in ensuring communication between computing modules and aircraft subsystems is safe, efficient and reliable. An illustration of an AFDX network is shown in Fig. 6.

The main components of an AFDX network are switches and the end systems. End systems serve as a source or receptor of data on the network and increase interoperability by means of standardized input and output interfaces [21, 22]. End systems are deployed in any IOM, such as the GPM or RDC, which is connected with the AFDX network. The information sent and received between IOMs is routed through switches. Designated input and output paths ensure a redundant information flow and eliminate two-way collisions, which can occur in other network technologies such as standard Ethernet [21]. Besides AFDX, there are the ARINC 429 and 629 communication standards which are commonly used in aircraft. There are two major advantages of the ARINC 664 network compared to 429 and 629. First, dual active redundancy is guaranteed, because data is transmitted on two buses while the receiver decides which of the two incoming packets is used. Second, AFDX offers higher throughput, with speeds of up to 100 Mb/s compared to 100 kb/s in ARINC 429 and 2 Mb/s in ARINC 629 [23].
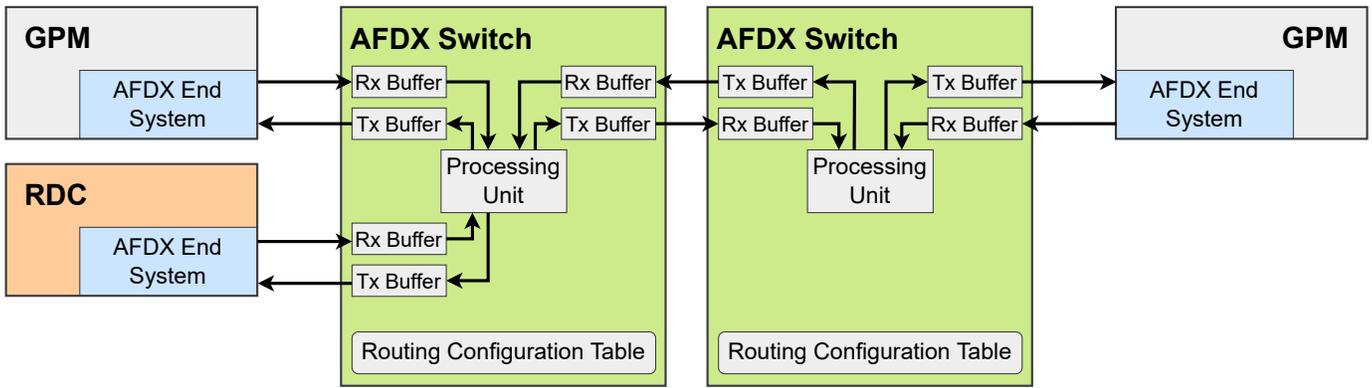
Fig. 6: High-level illustration of AFDX in an IMA System

C. Supplementary Components

IMA incorporates Remote Data concentrators (RDCs) to connect multiple other units to the IMA architecture and relay data to various modules. Some legacy hardware, such as LRUs, might be required for IMA system functionality and can be incorporated as "plug and play" units through RDCs. Sensors are critical components that provide real-time data to assist with flight control and other vital functions. Lastly, effectors and actuators enable the mechanical execution of actions, for instance on the wings. Interoperability of these components is increased with RDCs acting as interfaces between them and other modules in the system. RDCs, sensors, and effectors are essential hardware components in IMA systems that ensure safe and dependable operation aboard modern aircraft [22]. An illustration of a high-level IMA architecture is shown in Fig. 7. For reasons of simplicity, the end systems in the RDCs are ommited in this illustration.
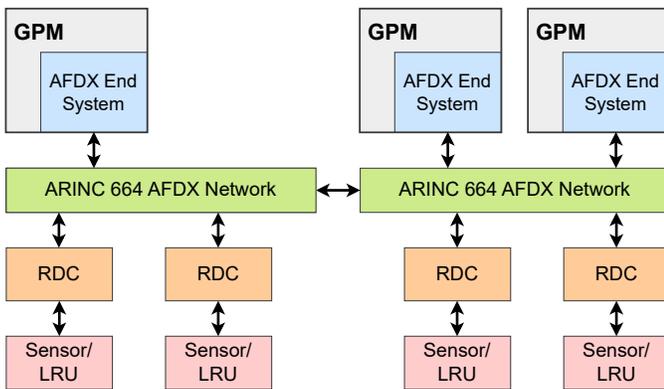

Fig. 7: High-level illustration of an IMA architecture

As described in the previous subsections, the IOMs are connected through a network with end systems and switches. Additional information, such as sensor data which may be used in the computation of certain applications, is processed in RDCs and routed to the GPMs.

III. RELATED WORK

The research presented in this section poses a selection of important publications throughout the last decades posing the state-of-the-art in developing IMA. In an effort to tackle the high complexity of IMA, a lot of research has been done on model-based engineering for IMA, which constitutes a significant part of the related work presented. Besides general model-based design for IMA, the content can further be categorized into requirements and general development efforts for IMA systems, and specifically ARINC 653 compliant implementations of processing modules with robust partitioning capabilities. Networks are treated as a part of the state-of-the-art IMA systems without setting particular focus on them.

Some of the earlier papers introducing the concept of IMA come from Prisaznuk. In the paper [24] the author introduces the concept of IMA and its benefits for the aviation industry. A highly-integrated digital avionics architecture that allows for substantial cost savings compared to earlier avionics implementations is demonstrated. In a later paper [25], Prisaznuk refers to the ARINC specification 653 and its role in the development of IMA. The author identifies the interface boundary between avionics software applications and the core executive software. ARINC 653 represents a paradigm shift for avionics development as it recognizes the RTOS as a key component of an IMA system. In his publications [26, 27] Watkins provides guidance for developing the methodology and tools to efficiently manage the set of shared intersystem resources. The author proposes that all platform properties can be organized into three categories: Interface/service definitions, Interface/service operating characteristics, and Resource allocations. Modular Avionics and the AFDX network can improve software design, fault isolation, and configuration management. However, component certification is needed for its use. Some other relevant research on IMA as a general technology comes from Fraboul [28] and Lopez [12].

The publication [29] by Jafer and Durak shows the challenges of designing, developing, and testing avionics software. The authors propose model-based approaches and agile development practices as major advancements that can help overcome these and also some of the aforementioned challenges.

The book chapter presents an overview of avionic systems and software, advances in avionics software development with a special emphasis on agile development, and model-based approaches as an enabler.

Significant research on the development of next generation IMA comes from Halle and Thielecke. In multiple publications [30, 31, 32], the authors discuss the challenges of avionics platform development, integration, and testing. As a result they propose a custom tool chain, AvioNet, as a solution. The papers highlight that model-based systems engineering (MBSE) methods are essential to cope with the increasing complexity of avionics systems. They also suggest that domain-specific models help to structure and express artifacts of different aspects. Modeling tools applicable to avionic systems are available but there are shortcomings for avionics specific parameters, constraints, and automation. The papers also highlight the importance of standardization of system tests including avionics, and how to formalize data formats and data management.

Björn Annighöfer et al. published several papers [33, 34, 35, 36] on model driven development and simulation of IMA architectures along with a framework for configuration and development of IMA systems. The main findings are that model-driven development (MDD) can provide a structured approach to modeling and simulation, which can help to reduce the complexity of system development. The study also showed that MDD can improve the quality of system design, reduce development time, and increase productivity. Moreover, the authors emphasize the importance of finding an architecture that is compliant with requirements and optimal in terms of quality measures like cost and weight. Concluding, MDD can be successfully implemented for developing and simulating a computer-integrated manufacturing and management system.

An important contribution to the next generation of IMA comes from Gaska [6]. The paper highlights the challenges that IMA systems will face going forward and the need to develop an IMA architecture that can incrementally improve both data fusion and autonomy. The paper also discusses emerging challenges in performing effective change containment in the IMA environment through architectural features and structure of the certification artifacts. Additionally, the need for refinement in criteria for IMA cybersecurity in connected aircraft and the emergence of ARINC 653-independent hybrid architectures is addressed. The contribution of Gaska for the next generation of IMA is the basis for some of the technological advancements discussed in the next section, with some concrete examples for their implementation.

IV. NEXT GENERATION IMA

Next generation IMA systems are characterized by better maintainability, higher modularity, and better system integration while increasing functionality. The system domain capability requirements are met by an IMA network consisting of computing modules that communicate with each other in real-time. Some of the upcoming technological advances in IMA systems include multi-core processing, dynamic reconfiguration capabilities, and wireless networks.

There are many requirements which need to be met for this new generation of IMA system to be certifiable and usable in commercial aircraft. Some of the requirements are listed in [3] and [4]. The technologies with their challenges are further discussed in the next subsections.

A. Computational Capabilities

In the foreseeable future, avionics will deploy new functionalities which require high computing power and may not be achievable with current processors. Rather, special GPMs for computationally intensive applications will be developed and integrated in IMA architectures. While model-based engineering can facilitate the development of designated GPMs for computationally demanding applications, the main requirements for their implementation are software- and hardware-based.

Due to strict certification and regulatory requirements in aviation, processing modules in aircraft have generally been developed with single-core CPUs. The rigorous testing and verification of avionics systems required by certification authorities can be more complex and time-consuming with multi-core processors. Additionally, software running on multi-core processors must be carefully designed and tested to prevent issues such as race conditions and priority inversions, which can cause safety-critical problems [37, 38]. However, the switch to multi-core processing will be inevitable in the future when considering the surge in complexity of avionics applications and decreasing availability of single-core processors [39].

Multiple nonbinding guidances discussing the use of multi-core processors in airborne systems were published, such as [37, 38, 40]. With supplements 4 and 5 of ARINC standard 653 part 1, official guidelines supporting the use of multi-core processors in aviation were provided. There are currently two multi-core scheduling models [20, 40]. Symmetric multi-processing (SMP) refers to multiple cores simultaneously running different tasks from the same partition. The SMP scheme is shown in Fig. 8. In this example, some tasks in partition 1 have dependencies. Tasks 1.4 and 1.5 can only be executed after task 1.3, which in turn can only be executed after tasks 1.1 and 1.2. The tasks in partition 2 don't have dependencies.
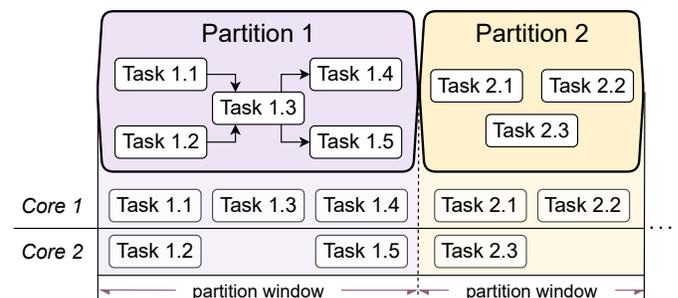

Fig. 8: Symmetric multi-processing scheme with two cores

As illustrated, SMP allows for parallel computation of different tasks from one partition on different cores, such as tasks 1.1 and 1.2 from partition 1. An advantage of SMP is that it obeys temporal and spatial separation according to ARINC 653 eliminating the need for changing existing guidelines. However, while this scheduling scheme is highly applicable for some airborne applications like the flight management system, computation of legacy applications might be prone to inefficiency [40]. The second scheduling model, asymmetric multi-processing (AMP), describes multiple cores simultaneously running different partitions with each partition only utilizing one core at a time. The AMP scheme is illustrated in Fig. 9. Here, two additional partitions, partition 3 and partition 4, are added for completeness.
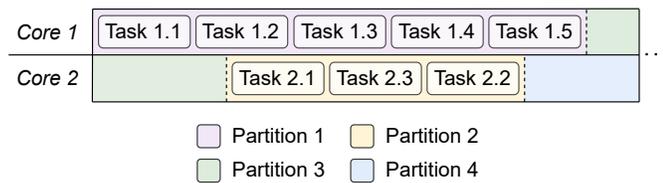

Fig. 9: Asymmetric multi-processing scheme with two cores

In AMP, single partitions are still executed inside a single-core, thus legacy application compatibility is close to the classical single-core GPM configuration. Further, AMP scales with the number of cores deployed. The biggest drawback of this scheduling model is that robust partitioning might not be guaranteed at the highest level of criticality due to potential inter-core interference [40].

Besides multi-core processors, the research on GPU in safety-critical applications in aviation is becoming more prevalent. With the advance of technology, processing units with superior performance and processing power are required. One example is the use of machine learning applications for object detection in aircraft, as described in [11]. GPUs with their strong parallel computation capabilities greatly outperform CPUs concerning tasks which involve a large number of simple calculations, such as graphics processing [41, 42]. Graphical processors are in some form already in use for less stringent safety-critical applications in aircraft today [43], for instance for display and surveillance systems. A subset of the Open Graphics Library (OpenGL), the OpenGL Safety Critical (SC) standards 1.0 and 2.0, is provided for graphics-based general-purpose computing techniques [44]. Two non-binding guidances [45, 43] were published discussing the use of graphical processors in airborne systems. The use of GPUs for safety critical applications, such as flight advisories based on object detection, however, is not yet intended. Concerns have been expressed regarding the use of graphics processors in airborne applications [46], with one example being the incompatibility of GPU programming languages with the safety standards imposed by DO-178C, especially in the highest criticality level DAL A. Most of these programming languages, like CUDA and OpenCL, rely on dynamic features such as dynamic memory allocation, pointers, and on-the-fly code compilation

[44, 47]. In fact, even in the less stringent automotive domain GPU programming languages have proven not to be certifiable yet [47]. A viable alternative to GPUs for graphics processing is the Field Programmable Gate Array (FPGA). An FPGA is an integrated circuit that can be programmed using a hardware description language or schematic design. Determinism is given by ensuring a consistent execution workflow and producing identical outputs for identical inputs, which makes certification for these processors simpler. Additionally, FPGAs have shown higher performance than GPUs in the past [48].

Despite the difficulties concerning certification, graphics processors will, just like multi-core CPUs, be of high importance and inevitable for next generation aircraft. The use of dedicated processing modules with nonstandard processing units in a hybrid architecture [6] should be considered. These processing modules could be integrated in an advanced IMA architecture as isolated units with additional safety and redundancy measures. It should be noted that currently such configurations are speculative and their implementation in modern avionics architectures requires further research.

B. ARINC 653 Compliant Dynamic Reconfiguration

Dynamically reconfigurable IMA describes the dynamic reconfiguration of computing units in IMA systems. The reconfiguration happens during runtime and prevents catastrophic system failures. In the context of ARINC 653, this refers to the partitioning configurations in GPMs being reconfigured either with predefined configurations or just in time. In case applications are faulty, certain redundancy mechanisms guarantee that required functionalities can be provided. This safety concept ensures that necessary applications, such as flight control, can continue to be computed.

There are currently three reconfiguration principles, which are discussed shortly in [49] and presented exemplary in [10]. The first option is to have a spare partition per GPM alongside active partitions. In case a partition is faulty, the spare partition can be activated and take over required functionality. An illustration of such a reconfiguration can be seen in Fig. 10.

Another option for reconfiguration is for each partition to contain all required applications. In case an application in a partition is faulty, a redundant application with the same functionality can be used instead of the faulty one, as illustrated in Fig. 11.

A third option is to run all partitions at all times and have functionality redundant partitions act as hot spares. This ensures an even transition from one partition to another in case of reconfiguration and that all partitions can be monitored while not having to be active. This third approach, however, requires a high amount of resource allocation to redundant partitions.

Dynamic reconfiguration has certain implications. Configuration tables are required for the system to look up valid configurations of the system. This implies one of two things: Either, all possible (valid) configurations of the IMA system need to be tested and certified according to DO-178C during design time. Or, each reconfiguration needs to be validated
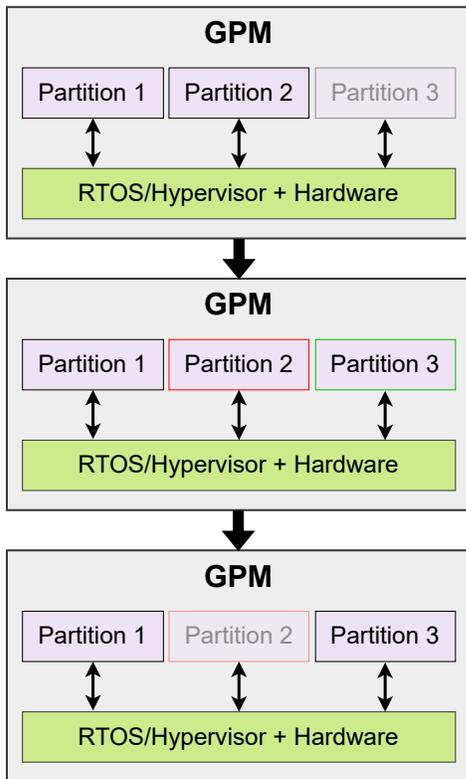
Fig. 10: Reconfiguration of one GPM due to partition failure

Fig. 11: Reconfiguration of one application due to application failure

during runtime and certified just in time, as mentioned in [50]. Both approaches are complex and come with significant development work. Besides the reconfiguration for computation of applications, the changing communication paths after reconfiguration need to be considered. This and other additional challenges are discussed in more detail in [10].

C. Networks and Security

The aviation industry is putting significant effort into developing new network technologies [51, 52], with one major technology being Avionics Wireless Networks (AWN). In general, wireless networks pose an important technology in the transportation industry. Reducing the amount of wiring and thus the weight of vehicles increases efficiency and can contribute to significant operating cost savings [53]. Specifically in the complex aviation systems, technologies such as Wireless Avionics Intra-Communication (WAIC) reduce the wire installation complexity and contribute to safety critical concepts such as dissimilar component redundancy and improved system reconfigurability [54].

WAIC is an aircraft network connecting its different components using wireless technology. It does not consider air-to-satellite, air-to-ground, or in-flight entertainment services. WAIC is currently developed by the Aerospace Vehicle Systems Institute [55] and documented in recommendations [56, 57] from the International Telecommunication Union, Radiocommunication Sector (ITU-R) and other partners. In the final agreement [58], the 4.2–4.4 GHz frequency band
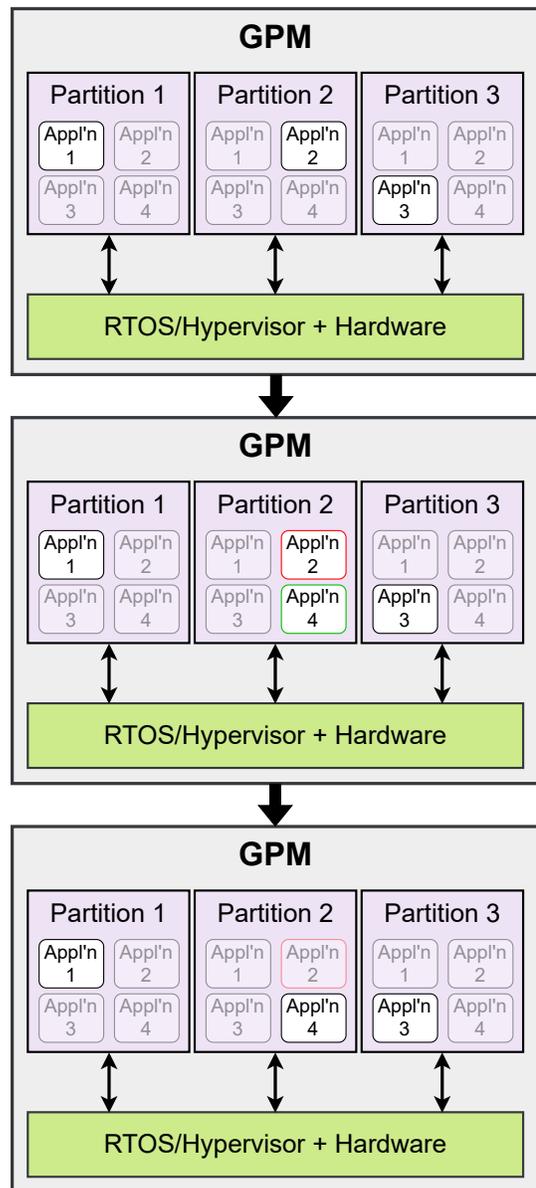
has been allotted as the primary spectrum band for WAIC. EUROCAE published an additional design guidance [59] for the coexistence of WAIC in aircraft and is working tightly with ITU-R on its compliance standards and certification. The wireless network allows for a transfer range of up to 100 m and transfer rates of up to 200 Mb/s, with the latter significantly exceeding the established AFDX network. This arguably makes WAIC suitable for IMA architectures and for its integration between remote peripherals and the core IMA system, with AFDX acting as the network backbone [60, 22]. Route segregation and redundant radio links provide dissimilar redundancy and mitigate risks of single points of failure [54]. An applicable use case is spatially distributed controllers, actuators, and sensors that use WAIC instead of

current fieldbus technologies. The analogy are advanced IMA architectures with WAIC transponders and receivers deployed in remote components such as RDCs and Sensors. A visionary implementation is shown in Fig. 12.
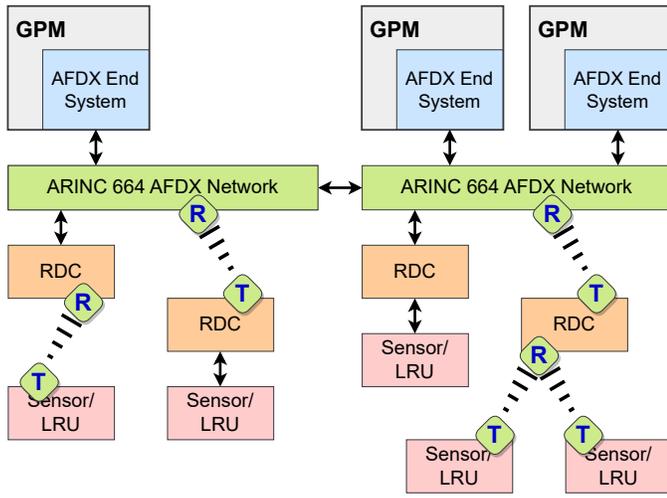

Fig. 12: High-level illustration of an advanced IMA Architecture with WAIC for communication between remote components and the core system using transmitters (T) and receivers (R)

Albeit WAIC comes with many advantages, there are major security and safety concerns. Safety concerns stem from wireless networks not being able to deliver safety guarantees according to current certification standards. Security concers stem from attacks on the wireless nodes of the network. Some examples are spoofing, replay, man in the middle, and denial of service attacks, as explained in [61, 62, 63]. Secure channel protocols, addressing some potential attacks were presented in the past [64, 65] with further research on mitigation of major security breaches in AWNs still ongoing.

V. CONCLUSION

The leap to the next generation of IMA is a highly topical matter. Many new technologies, such as multi-core processors, wireless networks, and highly adaptable processing modules are enablers for advanced airborne applications. The most promising technologies for next generation IMA, which were identified in this paper, are multi-core and graphical processors, dynamically reconfigurable processing modules, and wireless networks.

As pointed out in the main section, these technologies have major benefits by improving the efficiency of aircraft and enabling advanced applications like automatic takeoff and landing. While the implementation of these technologies is advantageous and desired, they adhere to strict certification standards like DO-178C, DO-254, and DO-297. Their development with formal methods and thorough testing for verification, validation, and certification will be inevitable. The duration of those development efforts and the time of adoption and transition to the next generation of IMA is not certain.

This is particularly evident when considering that many of the novel technologies are not even certifiable in the less stringent automotive industry yet.

It is up to the research institutions and aviation industry as a whole to address these challenges and guarantee safety and security with the transition to the next generation of IMA. Some of the research presented in this paper will be picked up in future research on ARINC 653-compliant avionics architectures. A focus will be set on model-based engineering as a formal method for developing safe and robust next generation avionics. The establishment of formal safety guarantees and simulation of prospective IMA architectures will be essential.

REFERENCES

[1] Ian Moir, Allan Seabridge, and Malcom Jukes. *Civil Avionics Systems*. Aerospace Series (PEP). John Wiley & Sons Ltd, 2013. ISBN: 978-1-118-34180-3. DOI: 10.1002/9781118536704.

[2] Airlines Electronic Engineering Committee. *ARINC 653: Avionics Application Software Standard Interface*. 1997.

[3] Radio Technical Commission for Aeronautics. *DO-297 – Integrated Modular Avionics (IMA) Development Guidance and Certification Considerations*. 2005.

[4] European Union Aviation Safety Agency. *AMC 20-170, 'Integrated modular avionics (IMA)'*. 2018.

[5] Cary R. Spitzer, Uma Ferrell, and T. K. Ferrell. "Digital Avionics Handbook". In: Taylor & Francis, 2015. ISBN: 9781315216980. DOI: 10.1201/b17545.

[6] Thomas Gaska, Chris Watkin, and Yu Chen. "Integrated Modular Avionics - Past, present, and future". In: *IEEE Aerospace and Electronic Systems Magazine* 30.9 (2015), pp. 12–23. DOI: 10.1109/MAES.2015.150014.

[7] Mirko Jakovljevic and Astrit Ademaj. "2nd generation IMA: Extended virtualization capabilities for optimized architectures". In: *2013 IEEE/AIAA 32nd Digital Avionics Systems Conference (DASC)*. 2013, pp. 7D5-1-7D5–11. DOI: 10.1109/DASC.2013.6712646.

[8] Anika Christmann et al. "Integrating Multi-/Many-Cores in Avionics: Open Issues and Future Concepts". In: *2021 IEEE/AIAA 40th Digital Avionics Systems Conference (DASC)*. 2021, pp. 1–8. DOI: 10.1109/DASC52595.2021.9594458.

[9] Sven Friedrich et al. "Assuring APEX with a versatile Rust API". In: *embedded world 2023 Conference Proceedings*. WEKA FACHMEDIEN GmbH, Mar. 2023, pp. 298–305. ISBN: 978-3-645-50197-2.

[10] Wanja Zaeske et al. "Towards Enabling Level 3A AI in Avionic Platforms". In: Mar. 2023. DOI: 10.18420/se2023-ws-18.

[11] Jasper Sprockhoff et al. "Model-Based Systems Engineering for AI-Based Systems". In: *AIAA SCITECH 2023 Forum*. Jan. 2023. DOI: 10.2514/6.2023-2587.

[12] Víctor López-Jaquero et al. "Supporting ARINC 653-based Dynamic Reconfiguration". In: *2012 Joint Working IEEE/IFIP Conference on Software Architecture and European Conference on Software Architecture*. 2012, pp. 11–20. DOI: 10.1109/WICSA-ECSA.212.9.

[13] Pierre Bieber et al. "Preliminary design of future reconfigurable IMA platforms". In: *SIGBED Rev.* 6 (2009). DOI: 10.1145/1851340.1851349.

[14] G. Jolliffe. "Producing a safety case for IMA blueprints". In: *24th Digital Avionics Systems Conference*. Vol. 2. 2005, 14 pp. DOI: 10.1109/DASC.2005.1563468.

[15] RTCA and EUROCAE. *DO-178 – Software Considerations in Airborne Systems end Equipment Certification*. 2011.

[16] RTCA and EUROCAE. *DO-254 – Design Assurance Guidance for Airborne Electronic Hardware*. 2000.

[17] Society of Automotive Engineers. *Guidelines for Development of Civil Aircraft and Systems ARP4754A*. 2010.

[18] Society of Automotive Engineers. *GUIDELINES AND METHODS FOR CONDUCTING THE SAFETY ASSESSMENT PROCESS ON CIVIL AIRBORNE SYSTEMS AND EQUIPMENT ARP4761*. 1996.

[19] Harry Katzan. "Operating Systems Architecture". In: *Proceedings of the May 5-7, 1970, Spring Joint Computer Conference*. AFIPS '70 (Spring). Atlantic City, New Jersey: Association for Computing Machinery, 1970, pp. 109–118. ISBN: 9781450379038. DOI: 10.1145/1476936.1476960. URL: https://doi.org/10.1145/1476936.1476960.

[20] Patrick Huyck. "ARINC 653 and multi-core microprocessors — Considerations and potential impacts". In: *2012 IEEE/AIAA 31st Digital Avionics Systems Conference (DASC)*. 2012, 6B4-1-6B4–7. DOI: 10.1109/DASC.2012.6382387.

[21] Airlines Electronic Engineering Committee. *ARINC 664: Aircraft Data Network, Part 7, Avionics Full-Duplex Switched Ethernet Network*. 2009.

[22] Cary Spitzer. *Avionics: Development and Implementation*. Dec. 2006. ISBN: 9781315222233. DOI: 10.1201/9781315222233.

[23] Paul M. Simon and Scott R. Graham. "Comparison of Real-Time, Mobile Data Bus Architectures". In: *2016 IEEE 17th International Conference on Information Reuse and Integration (IRI)*. 2016, pp. 484–489. DOI: 10.1109/IRI.2016.71.

[24] Paul J. Prisaznuk. "Integrated modular avionics". In: *Proceedings of the IEEE 1992 National Aerospace and Electronics Conference NAECON 1992*. 1992, 39–45 vol.1. DOI: 10.1109/NAECON.1992.220669.

[25] Paul J. Prisaznuk. "ARINC 653 role in Integrated Modular Avionics (IMA)". In: *2008 IEEE/AIAA 27th Digital Avionics Systems Conference*. 2008, 1.E.5-1-1.E.5–10. DOI: 10.1109/DASC.2008.4702770.

[26] Christopher B. Watkins. "Integrated Modular Avionics: Managing the Allocation of Shared Intersystem Resources". In: *2006 ieee/aiaa 25TH Digital Avionics Systems Conference*. 2006, pp. 1–12. DOI: 10.1109/DASC.2006.313743.

[27] Christopher B. Watkins and Randy Walter. "Transitioning from federated avionics architectures to Integrated Modular Avionics". In: *2007 IEEE/AIAA 26th Digital Avionics Systems Conference*. 2007, 2.A.1-1-2.A.1–10. DOI: 10.1109/DASC.2007.4391842.

[28] Xinying Li and Huagang Xiong. "Modelling and simulation of integrated modular avionics systems". In: *2009 IEEE/AIAA 28th Digital Avionics Systems Conference*. 2009, 7.B.3-1-7.B.3–8. DOI: 10.1109/DASC.2009.5347426.

[29] Shafagh Jafer et al. "Advances in Software Engineering and Aeronautics". In: *Advances in Aeronautical Informatics: Technologies Towards Flight 4.0*. Ed. by Umut Durak et al. Springer International Publishing, 2018, pp. 87–102. URL: https://elib.dlr.de/121246/.

[30] Martin Halle and Frank Thielecke. "Next generation IMA configuration engineering - from architecture to application". In: *2015 IEEE/AIAA 34th Digital Avionics Systems Conference (DASC)*. 2015, 6B2-1-6B2–13. DOI: 10.1109/DASC.2015.7311445.

[31] Martin Halle and Frank Thielecke. "Tool chain for avionics design, development, integration and test". In: *2019 Workshops of the Software Engineering Conference*. CEUR workshop proceedings. 2019, pp. 79–82. DOI: 10.15480/882.2598.

[32] Martin Halle and Frank Thielecke. "Avionics Next-Gen Engineering Tools (AvioNET): Experiences With Highly Automised and Digital Processes for Avionics Platform Development". In: *2021 IEEE/AIAA 40th Digital Avionics Systems Conference (DASC)*. 2021, pp. 1–8. DOI: 10.1109/DASC52595.2021.9594509.

[33] Björn Annighöfer, Ernst Kleemann, and Frank Thielecke. "Model-based Development of Integrated Modular Avionics Architectures on Aircraft-level". In: Sept. 2011.

[34] Björn Annighöfer, Erik Stallkamp, and Frank Thielecke. "Eclipse Framework for an Integrated IMA Tool Chain". In: Oct. 2011. DOI: 10.4271/2011-01-2635.

[35] Björn Annighöfer. "Model-driven Development and Simulation of Integrated Modular Avionics (IMA) Architectures". In: *SNE Simulation Notes Europe* 28 (June 2018), pp. 61–66. DOI: 10.11128/sne.28.tn.10414.

[36] Bjoern Annighoefer et al. "Holistic IMA Platform Configuration using Web-technologies and a Domain-specific Model Query Language". In: *2020 AIAA/IEEE 39th Digital Avionics Systems Conference (DASC)*. 2020, pp. 1–10. DOI: 10.1109/DASC50938.2020.9256726.

[37] European Union Aviation Safety Agency. *AMC 20-193 Use of multi-core processors*. 2022.

[38] Certification Authorities Software Team. *CAST-32A Multi-core Processors*. 2016.

[39] Paul Parkinson. *The challenges of developing embedded real-time aerospace applications on next generation multi-core processors*. Apr. 2016.

[40] European Union Aviation Safety Agency. *MULCORS - Use of MULticore proCessORs in airborne Systems*. 2011.

[41] Jyothish Soman, Kothapalli Kishore, and P J Narayanan. "A fast GPU algorithm for graph connectivity". In: *2010 IEEE International Symposium on Parallel & Distributed Processing, Workshops and Phd Forum (IPDPSW)*. 2010, pp. 1–8. DOI: 10.1109/IPDPSW.2010.5470817.

[42] John Owens et al. "A Survey of General-Purpose Computation on Graphics Hardware". In: *Computer Graphics Forum* 26 (Mar. 2007), pp. 80–113. DOI: 10.1111/j.1467-8659.2007.01012.x.

[43] Certification Authorities Software Team. *CAST-29 Use of COTS Graphical Processors (CGP) in Airborne Display Systems*. 2007.

[44] Marc Benito et al. "Comparison of GPU Computing Methodologies for Safety-Critical Systems: An Avionics Case Study". In: *2021 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. 2021, pp. 717–718. DOI: 10.23919/DATE51398.2021.9474060.

[45] European Union Aviation Safety Agency. *EASA CM-SWCEH-001 Development Assurance of Airborne Electronic Hardware*. 2018.

[46] CoreAVI. *Eight Concerns with Graphics Processors in Safety Critical Applications*. 2018.

[47] Matina Maria Trompouki and Leonidas Kosmidis. "Brook Auto: High-Level Certification-Friendly Programming for GPU-powered Automotive Systems". In: *2018 55th ACM/ESDA/IEEE Design Automation Conference (DAC)*. 2018, pp. 1–6. DOI: 10.1109/DAC.2018.8465869.

[48] Andrew Boutros et al. "Beyond Peak Performance: Comparing the Real Performance of AI-Optimized FPGAs and GPUs". In: *2020 International Conference on Field-Programmable Technology (ICFPT)*. 2020, pp. 10–19. DOI: 10.1109/ICFPT51103.2020.00011.

[49] Claire Pagetti et al. "Reconfigurable IMA platform: from safety assessment to test scenarios on the SCARLETT demonstrator". In: *Embedded Real-time Software and Systems, ERTS 2012*. Toulouse, France, Feb. 2012. URL: https://hal.science/hal-02170919.

[50] Björn Annighöfer. *Model-based Architecting and Optimization of Distributed Integrated Modular Avionics*. Vol. 2015,1. Schriftenreihe Flugzeug-Systemtechnik. Aachen: Shaker Verlag, 2015. ISBN: 9783844034202.

[51] European Organization for Civil Aviation Equipment. *ED-246 - Process Specification for Wireless On-board Avionics Networks*. 2017.

[52] Radio Technical Commission for Aeronautics. *SC-236, Standards for Wireless Avionics Intra-Communication System (WAIC) within 4200-4400 MHz*. 2023.

[53] Oroitz Elgezabal Gómez. "Fly-by-wireless: Benefits, risks and technical challenges". In: *CANEUS Fly by Wireless Workshop 2010*. 2010, pp. 14–15. DOI: 10.1109/FBW.2010.5613788.

[54] Uwe Schwark. *Wireless Avionics Intra-Communication (WAIC)*. https://www.sae.org/binaries/content/assets/cm/content/attend/2017/aerospace-standards-summit/waic.pdf. SAE Aerospace Standards Summit Spring 2017. Apr. 2017.

[55] Aerospace Vehicle Systems Institute. *Written Ex Parte – AVSI Project Report 76S2-REP-03, GN Docket No. 18-122*. 2021.

[56] International Telecommunication Union, Radiocommunication Sector. *Operational and technical characteristics and protection criteria of radio altimeters utilizing the band 4 200-4 400 MHz*. 2014.

[57] International Telecommunication Union, Radiocommunication Sector. *Compatibility analysis between wireless avionics intra-communication systems and systems in the existing services in the frequency band 4 200-4 400 MHz*. 2014.

[58] International Telecommunication Union, Radiocommunication Sector. *FINAL ACTS of the World Radiocommunication Conference (WRC-15)*. 2015.

[59] European Organization for Civil Aviation Equipment. *ED-260A - MASPS for Coexistence of Wireless Avionics Intra-Communication Systems within 4 200-4 400 MHz*. 2022.

[60] Pangun Park et al. "Wireless Avionics Intracommunications: A Survey of Benefits, Challenges, and Solutions". In: *IEEE Internet of Things Journal* 8.10 (2021), pp. 7745–7767. DOI: 10.1109/JIOT.2020.3038848.

[61] Peter Reji, Karuppannan Natarajan, and K.R. Shobha. "Challenges in implementing a wireless avionics network". In: *Aircraft Engineering and Aerospace Technology* ahead-of-print (Feb. 2020). DOI: 10.1108/AEAT-07-2019-0144.

[62] Peter Reji, K Natarajan, and K.R. Shobha. "Secured Wireless Network for Aircraft Intra Communication". In: *2020 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT)*. 2020, pp. 1–6. DOI: 10.1109/CONECCT50063.2020.9198688.

[63] Sathish A.P. Kumar and Brian Xu. "Vulnerability Assessment for Security in Aviation Cyber-Physical Systems". In: *2017 IEEE 4th International Conference on Cyber Security and Cloud Computing (CSCloud)*. 2017, pp. 145–150. DOI: 10.1109/CSCloud.2017.17.

[64] Raja Naeem Akram et al. "Security and performance comparison of different secure channel protocols for Avionics Wireless Networks". In: *2016 IEEE/AIAA 35th Digital Avionics Systems Conference (DASC)*. 2016, pp. 1–8. DOI: 10.1109/DASC.2016.7777966.

[65] Raja Naeem Akram et al. *An Efficient, Secure and Trusted Channel Protocol for Avionics Wireless Networks*. 2016. arXiv: 1608.04116 [cs.CR].