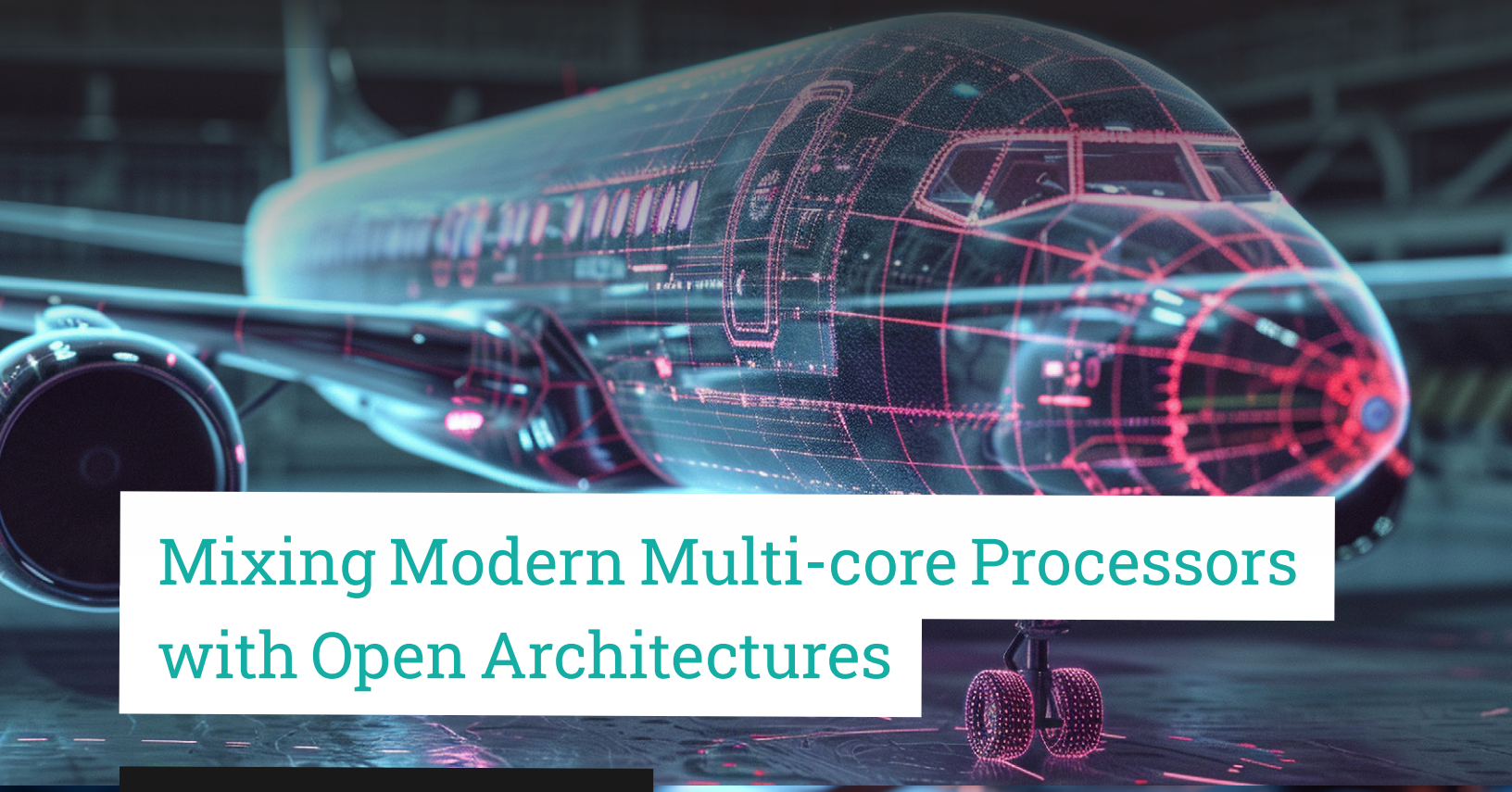
WNDRVR

CoreAVI
INNOVATING A SAFER TOMORROW

ScioTeq



Mixing Modern Multi-core Processors with Open Architectures

Experiences of Developing a State-of-the-Art Smart Avionics Display

Paul Parkinson
Field Engineering Director, Wind River

Dominiek Saelens
Senior Product Manager Avionics, ScioTeq

Damian Fozard
Chief Executive Officer, CoreAVI

EXECUTIVE SUMMARY

In recent years, the aerospace industry has sought to provide increased situational awareness for pilots and to improve operational efficiency of aircraft. This has resulted in new requirements for avionics display systems in the handling and display of more complex information.

At the same time, there has been a disruptive technology change with the adoption of multi-core processors, which can provide significant benefits in terms of size, weight, and power (SWaP) but can also present challenges for RTCA DO-178C/EUROCAE ED-12C avionics software and RTCA DO-254/EUROCAE ED-80 avionics hardware safety certification.

To make the challenge complete, the relationship between original equipment manufacturers (OEMs), system integrators (Tier 1), and component providers (Tier 2) has changed drastically. OEMs are demanding more involvement in and control of the system design and more flexibility and configurability of the solution — in short, a more modular and more open system design approach, as reflected in the U.S. Department of Defense (DoD) Modular Open Systems Approach (MOSA) procurement directives.

This paper will discuss the development of a next-generation smart avionics platform, including:

- The use of an ARMv8 multi-core processor
- The use of hardware virtualization and a hypervisor for isolation and management of applications
- Runtime environments for safety-critical real-time applications
- The use of advanced Vulkan® SC-based GPU acceleration capabilities on modern graphics processing units (GPUs)
- The portability of software applications through use of ScioTeq's MOSArt® and the ARINC 653 software architecture

This paper will also cover the technical challenges of adopting the state-of-the-art technologies listed above, along with the results of the project, with a comparison to previous-generation platforms. Finally, we will present the lessons learned from the project.

Contents

EXECUTIVE SUMMARY 2

THE EVOLUTION
OF SMART AVIONICS DISPLAYS 4

THE ROLE OF SOFTWARE IN AVIONICS DISPLAYS AND THE DEVELOPMENT OF MOSART 5

REQUIREMENTS FOR A NEXT-GENERATION SMART AVIONICS DISPLAY PLATFORM 6

Data Processing Requirements (CPUs) 7

Graphics Processing Requirements (GPUs) 8

DEVELOPMENT OF SCIOTEQ SMART DISPLAY PLATFORMS AND CHALLENGES ENCOUNTERED 9

Arm Cortex-A72 Core L1 Cache Protection 10

Arm A53 Core L2 Cache Partitioning 10

Cross-Core Clock Synchronization 10

GPU Scheduling in Mixed-Criticality Systems 10

Porting Helix Platform to NXP i.MX8 Quad Max 12

RESULTS AND LESSONS LEARNED 13

CONCLUSION 14

ACKNOWLEDGMENTS 14

REFERENCES 14

THE EVOLUTION OF SMART AVIONICS DISPLAYS

Avionics cockpit dials and gauges originally performed a single dedicated function and were deployed in many civil and military aircraft as part of a federated avionics architecture utilizing line-replaceable units (LRUs). Pioneering display research undertaken by NASA's Langley 737 flying laboratory[1] enabled the presentation of raw aircraft system data and flight data as an integrated picture on electronics digital displays, which became known as a glass cockpit. This concept has been widely adopted in both civil and military aircraft as the technology has matured, enabling the replacement of multiple analog dials and gauges (Figure 1) with digital counterparts (Figure 2).

Figure 1. C-130H analog cockpit displays (Source: Jeff Melvin, USAF, public domain)

Figure 2. C-130H digital cockpit displays (Source: Samuel King, Jr., USAF, public domain)

Modern digital displays and display computers are designed to perform the equivalent functions of multiple analog displays and are often implemented as multifunction displays (MFDs) comprising a glass display surrounded by fixed-function mechanical buttons that enable the appropriate content to be selected and displayed on the screen (Figure 3).

Figure 3. Multifunction display showing different applications (Source: ScioTeq)

Control displays and control devices in general have followed the integration evolutionary trend, providing control and interaction for the pilot with an increased number of subsystems (Figure 4). The advent of Touchscreen Control Units has resulted in further integration of multiple cockpit functions onto a single device (Figure 5).

Figure 4. Control display and management systems (Source: ScioTeq)

Figure 5. Touchscreen Control Unit (Source: ScioTeq)

There is also an ongoing market trend toward larger display size and higher resolution. Fifteen-inch displays are now common in civil air transport, such as the Boeing 787 Dreamliner and the Airbus A380 and A350 aircraft. In the defense sector, following the adoption of a panoramic cockpit display[2] by the Lockheed Martin F-35 Lightning II (aka Joint Strike Fighter), this trend toward large-area displays is progressing in other military fast jet aircraft and trainer aircraft and in the rotary-wing market. This is further driving higher levels of integration and providing less cluttered instrument panels, creating new challenges in relation to redundancy, safety, and performance.

Consolidating historically separate functional units is not the only change at the application level. Capabilities such as sensor fusion, synthetic display, vision augmentation (particularly in helicopter reduced-visibility solutions), among others, often require a software-defined level of integration. The demand for increased capabilities, consolidation of functionality, and more powerful processor hardware (both CPU and GPU) have led inevitably to the rise of mixed-criticality systems on common hardware.

THE ROLE OF SOFTWARE IN AVIONICS DISPLAYS AND THE DEVELOPMENT OF MOSART

Many digital avionics systems are being deployed and are subject to continuous evolution due to changes in operational requirements and diminishing manufacturing sources and material shortages (DMSMS)[3] during their in-service lifetime, resulting in the need for recurring design iterations that include migration to newer processor architectures. Many of these legacy systems are still based on monolithic architectures and proprietary interfaces, and end users can face serious challenges in funding the sustainment of such systems.

The challenge for the industry in both defense and civil avionics is to develop application frameworks that are hardware-agnostic and offer extensible functionality to meet the ever-increasing demand for new capabilities as hardware performance increases. Customers are not only looking for multi-supplier options to combat supply risks, such as obsolescence, but they also want to protect application investment, which typically runs to tens or even hundreds of millions of dollars over an application's lifetime.

These challenges for increasing the return on investment (ROI) can be mitigated by using an open standards–based software architecture, a modular design approach, and a hardware abstraction layer and associated APIs, following the trends of many industries of making the hardware elements more software defined. Another industry trend driving the need for improved portability is the desire of airframe manufacturers to increase independence from the primes (i.e., system integrators) and to gain flexibility in sourcing applications, with a typical segregation between certified safety-critical and generally noncertified mission-critical.

Recognizing these challenges, in 2000 ScioTeq started development of the Modular Open System Architecture platform for avionics-certifiable real-time processing (MOSArt®),[4] to provide a flexible framework based on ARINC 653[5] software architecture and using an ARINC 653–conformant commercial real-time operating system (RTOS) (Figure 6).

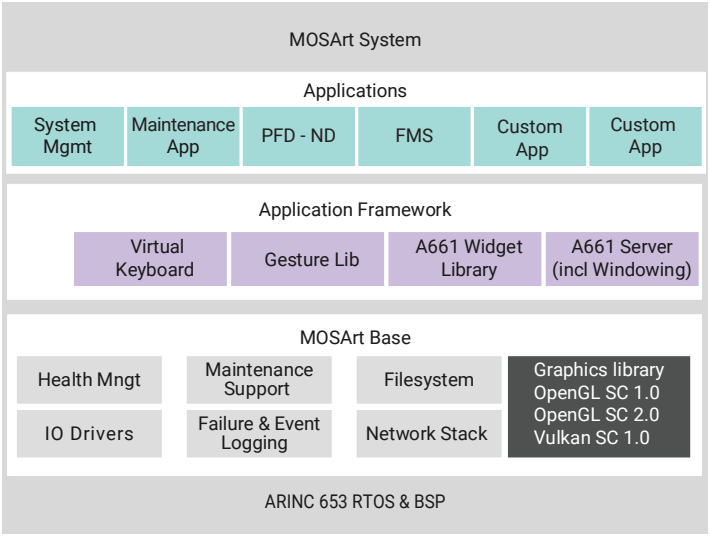


Figure 6. MOSArt V high-level architecture

MOSArt was designed to enable the integration of multiple applications at different levels of safety criticality while providing abstraction from the underlying platform, enabling portability to future evolutions of the platform and preserving investment and reuse of intellectual property. MOSArt has been successfully deployed on multiple civil and military programs combined with VxWorks® 653. An interesting use case is an AgustaWestland (now Leonardo Helicopters) program previously discussed in a case study[6] and conference paper.[7]

REQUIREMENTS FOR A NEXT-GENERATION SMART AVIONICS DISPLAY PLATFORM

Many smart display designs have undergone RTCA DO-254[8]/EUROCAE ED-80[9] avionics hardware safety-certification and DO-178C[10]/ED-12C[11] avionics software safety-certification processes and have entered into service during the last 20 years.

OEMs and avionics system integrators who want to maximize reuse and ROI for their software intellectual property have been looking for open standards–based software architectures and abstraction layers to minimize dependencies on specific hardware architectures and to maximize software portability. Adding incrementally new capabilities while minimizing the cost of recertification is also part of the wish list.

For more than 20 years, the commercial aerospace sector has supported the development of portable avionics software applications through the evolution of the ARINC 653[12] avionics software standard as well as commercial-off-the-shelf (COTS) RTOSes from a range of commercial suppliers that have achieved ARINC 653 conformance, enabling the development of portable ARINC 653 applications.

In the military aerospace sector, the Future Airborne Capability Environment® (FACE®)[13] initiative created by the U.S. DoD and the PYRAMID[14] initiative of the U.K. Ministry of Defence are defining reference architectures and standards that individually incorporate several of the following commercial open standards: ARINC 653, ARINC 661,[15] POSIX®,[16] OpenGL®SC1[17] OpenGL SC2, EGL, and Vulkan SC.[18] These open standards target ease of portability across platforms.

As a precursor to the FACE Technical Standard, ScioTeq has, since 2004, offered MOSArt to meet the need for increased independence, flexibility, and long-term investment protection through portability on ScioTeq computing platforms. The FACE Technical Standard and PYRAMID further extend those objectives by targeting portability across hardware vendors by defining APIs to be adopted as industry standards, where MOSArt, in its legacy form, provides a proprietary set of APIs. ScioTeq is committed to alignment of MOSArt with the FACE Technical Standard, and MOSArt will support integration with other components that are also aligned with the FACE Technical Standard or have been certified as FACE Conformant (Figures 7 and 8).

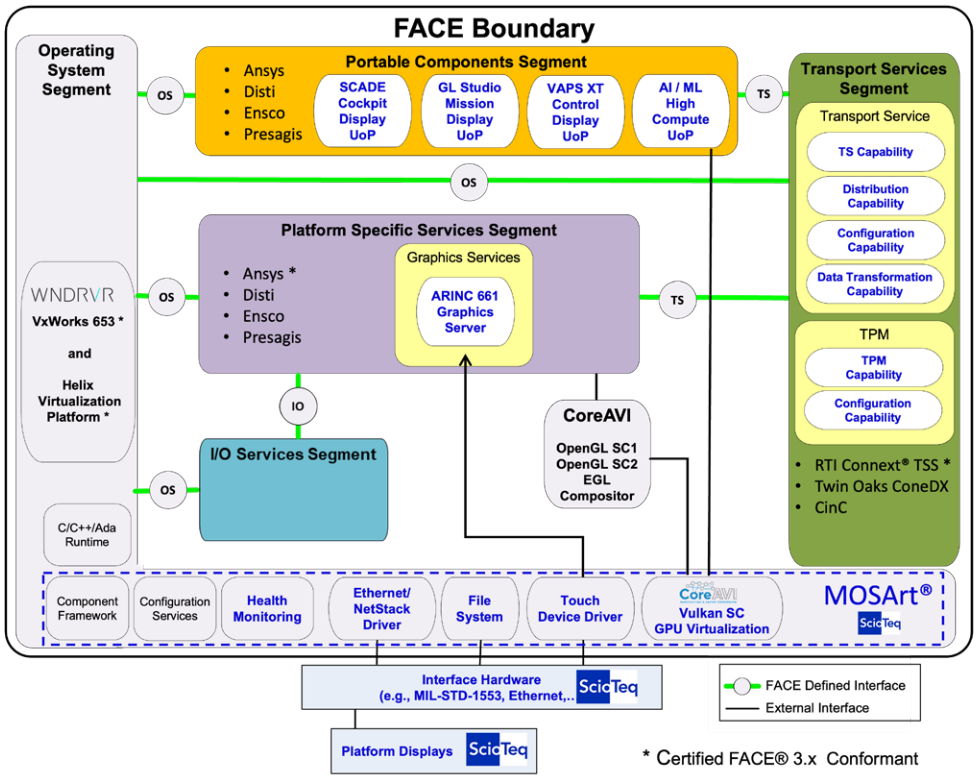


Figure 7. MOSArt alignment with FACE Technical Standard reference architecture
(Credit: FACE Reference Architecture, The Open Group)

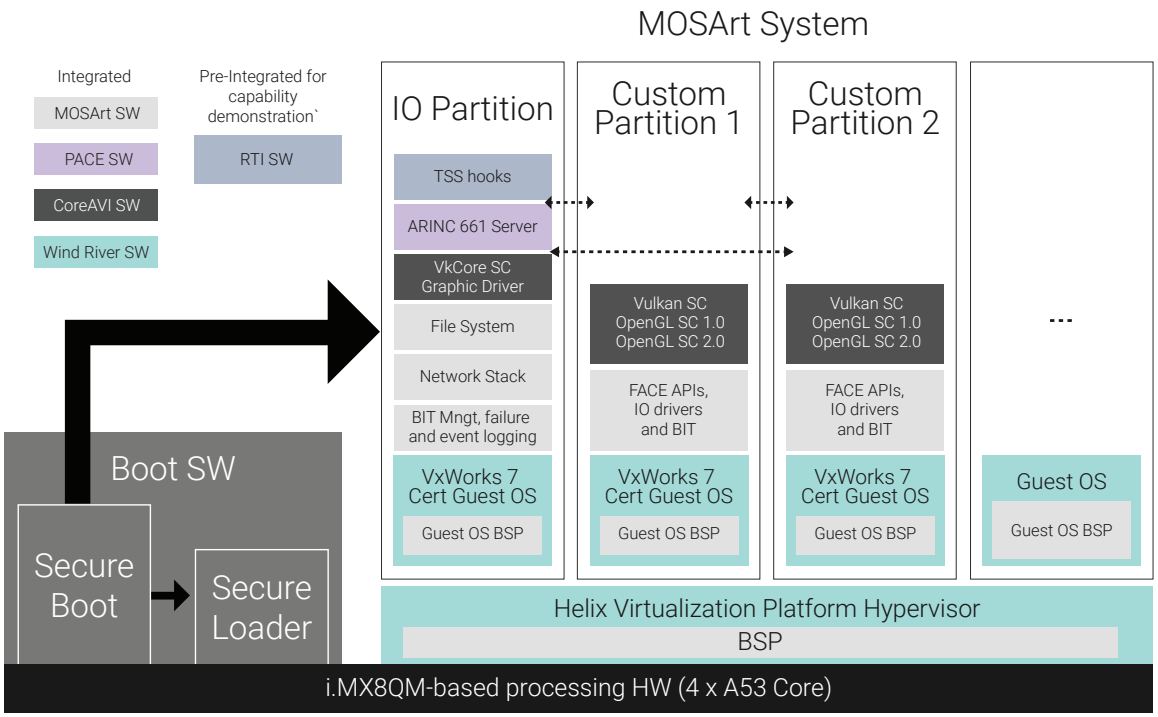


Figure 8. ScioTeq MOSArt V architecture and integration with components aligned with FACE Technical Standard or certified as FACE Conformant

Data Processing Requirements (CPUs)

Many avionics displays designs are based on Power Architecture®, which has a proven track record in avionics due to long silicon availability lifetimes and support for certification activities from NXP through the Multi-core for Avionics (MCFA) industry working group. Similarly, discrete GPUs have been used with different types of CPUs in many different avionics display applications. However, significant changes in technology trends, when combined, have resulted in a disruptive change to the avionics sector.

Adoption of Multi-core Processors

The global semiconductor market has seen a dramatic shift from single core processors to multi-core processors over the last decade, driven by the need to provide increased performance and support for more complex applications. The diminishing availability of single core processors has resulted in multi-core processors becoming the de facto choice for next-generation avionics platforms. This use of multi-core processors in avionics introduces new challenges for determinism and safety certification, which is being addressed through regulatory guidance provided by EASA AMC 20-193[19] and FAA AC 20-193,[20] and on some programs through a gradual transition to multi-core capabilities by initially utilizing only a single core within the multi-core processor.

Power Architecture Planned Obsolescence

Many aerospace programs have benefited from the long silicon production lifetimes of many Power Architecture processors, in particular the widespread use of the P3041 and T2080 in avionics systems. However, the lack of new Power Architecture designs means that silicon availability will eventually decline. NXP's Product Longevity program[21] currently indicates that the QorIQ T-series will end production in 2035, an impressive 23 years after the processor architecture was launched. Such timescales mean that Power Architecture is viable for fulfilling the production requirements of many current aerospace programs, but new programs with in-service dates planned well beyond this out-of-production date will most certainly consider alternative processor architectures.

Market Trend Toward Integrated GPUs

A number of discrete GPU designs were previously available, but the dominant market trend is now toward system-on-chip (SoC) designs with integrated GPUs. SoCs often share system memory (RAM) between many components, including the CPU and GPU. With respect to discrete GPUs, these designs often provide lower power at the cost of lower performance. The increased complexity of SoCs also creates new challenges for the industry, which requires collaboration

between organizations. One challenge is the decreasing availability of discrete GPUs to enable avionics display designs that incorporate different permutations of discrete CPUs and GPUs to satisfy the market need for a range of performance requirements. Additionally, the advent of integrated GPUs that enable systems to be designed with lower SWaP but that provide fewer options for system design from the perspective of performance and capability.

Graphics Processing Requirements (GPUs)

Modern GPUs provide both 2D/3D graphics processing and parallel processing for data-intense applications, such as video encoding/decoding and, increasingly, AI and sensor fusion. In high-reliability applications, hardware abstraction of graphics functions is supported by OpenGL SC1 and OpenGL SC2 open standards and referenced in the FACE Technical Standard as normative standards that are widely used by avionics suppliers and supported by hardware manufacturers with development tools and existing applications. Vulkan SC is a new GPU abstraction layer for safety-critical applications that provides more flexibility and control of the GPU functionality at the application level, including enabling General-Purpose GPU (GPGPU) functions for compute applications alongside graphics processing.

ScioTeq's requirements for the latest generation MOSArt V5 included the ability to support OpenGL SC1/SC2 and Vulkan SC to enable support for existing MOSArt-based applications, as well as providing a path for future capabilities with Vulkan SC. The integrated graphics driver allows for coexistence of applications using either standard API language.

Graphics Software Evolution from OpenGL to Vulkan SC

OpenGL SC was released in 2003 and has been the standard for the implementation of safety-critical graphics in the A&D market. Many existing applications rely on the OpenGL framework. However, OpenGL SC1 and SC2 are monolithic drivers that impose significant overhead on the CPU and are unable to take advantage of more modern innovations, such as multi-core CPUs and the low-level acceleration capabilities of next-generation GPUs.

Vulkan was developed to address the limitations of OpenGL and to provide application developers with more low-level control and more efficient use of GPUs. It removed the limited support for multi-core CPU architectures that had created a bottleneck for OpenGL applications on more modern systems. Following the initial Vulkan announcement by the Khronos Group in 2015, support for Vulkan has been implemented on a range of semiconductor manufacturers' GPUs.

A benchmark comparison between Vulkan and OpenGL by Basemark[22] (Figure 9) revealed that Vulkan performance exceeded OpenGL's by 2.9x on average. It also noted that in GPU-intensive applications, the OpenGL driver would quickly become the limiting performance factor.

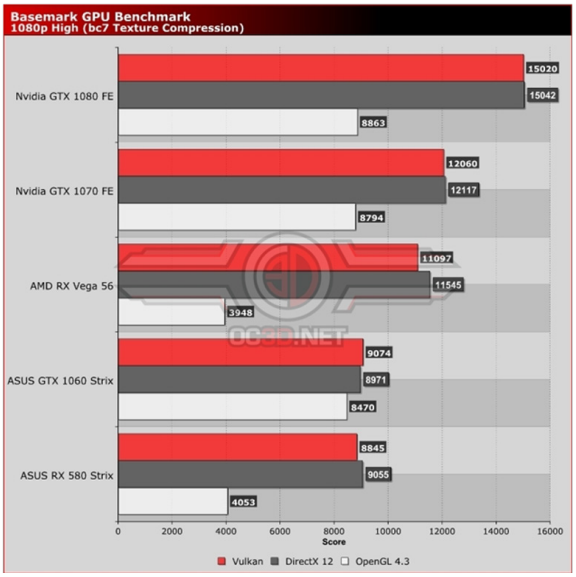


Figure 9. Basemark GPU benchmark
(Source: Mark Campbell, Basemark, published November 29, 2018)

Vulkan has become widely supported by GPU hardware manufacturers, and there are now Vulkan SC–supported GPUs available from AMD, NXP, Nvidia, Intel®, and Arm® licensees. This means that the avionics market will benefit from a wider choice of GPU and SoC products across a range of power and performance options.

However, some legacy applications that use OpenGL SC 1.0 will encounter performance challenges, since the latest GPU designs no longer provide direct hardware acceleration support for OpenGL SC 1.0. All new GPUs are based on a fully programmable pipeline aligned with either OpenGL SC 2.0 or Vulkan SC. While the number of Vulkan programmers continues to grow, they are still a scarce resource, so some programs continue to use OpenGL for now.

CoreAVI recognized that the transition to Vulkan SC, though it offers many performance benefits and better accessibility to computational capabilities of the GPU, could nonetheless require costly redevelopment of existing applications. This

limitation was addressed by the development of OpenGL SC1 and Open GL SC2 libraries that run on top of Vulkan SC. This architecture simplifies the requirements for future GPU hardware adoption to support of the Vulkan SC standard only. The ability of the OpenGL SC libraries to coexist with Vulkan SC means it is also possible to execute OpenGL and Vulkan applications concurrently, supporting legacy application code while opening possibilities to higher performance and novel functionality through Vulkan SC (Figure 10).

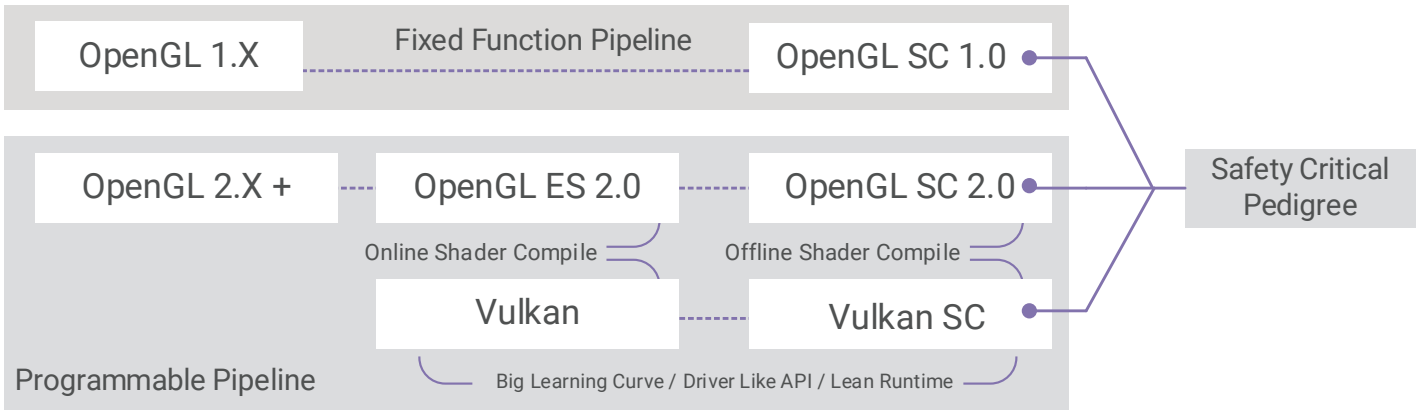


Figure 10. Programmable pipeline (Source: CoreAVI)

DEVELOPMENT OF SCIOTEQ SMART DISPLAY PLATFORMS AND CHALLENGES ENCOUNTERED

In 2020, driven by the trends described above and by DMSMS, ScioTeq initiated the design of a next-generation, modular, open, high-integrity avionics platform. This would become the fifth-generation MOSArt platform (Table 1).

Table 1: ScioTeq Avionics Platform Generations

Development Period		2002-2006	2005-2012	2006-2016	2006-2018	2013-2017	*2012-2022*	*2020-*
Product Lines		CDMS-2	PU-2	MFD-2	CDMS-3	MFD-3054	*4th Gen*	*5th Gen*
Hardware	Application processor	MPC8270	PPC755	PPC755	MPC8270	QorIQ P3041	QorIQ P3041	*i.MX8 QM*
	Application Graphics processing	3DLabs P9	3DLabs P9	3DLabs P9	Fujitsu CoralP	SW rendering	*Matrox GPU*	*i.MX8 QM*
Kernel/RTOS	Wind River VxWorks 5.5	■						
	Wind River VxWorks 653		■	■	■	■		
	GreenHills Integrity-178 tuMP						■	
	Wind River Helix Platform + VxWorks							■
Platform SW	MOSArt V2 - SIMphony V1	■	■	■	■			
	MOSArt V3					■		
	Open Platform V4						■	
	MOSArt V5 - SIMphony V5							■
Graphics library	OpenGL SC 1.0 - ScioTeq	■	■	■	font lib			
	OpenGL SC 1.0 - Subset - Ensco					■		
	OpenGL SC 1.0 - ScioTeq						■	
	OpenGL SC x.0 & Vulkan SC 1.0 - CoreAVI							■
Certification DAL level		NA	NA	DAL A	DAL B	DAL B	DAL A	DAL A

ScioTeq selected the NXP i.MX 8QuadMax (i.MX8QM) processor[23] for its fifth-generation PU-5000 Avionics Display Computer[24] platform, MDF-5000 Multi-Function Display[25] platform, and any custom smart product. The i.MX8QM is a complex SoC device comprising clusters of Arm processor cores; dual integrated GPUs; and many integrated peripherals for computation, connectivity, and video processing. It was assessed as providing optimal characteristics for use in avionics display computing due to its combination of sufficient performance, from both the CPU and the GPU perspective, and its very low power requirements.

Given the complexity of the design challenge, a number of development risks were identified:

- Ability to meet application performance requirements
- Complexity of the Arm core-based SoC (including integrated CPU and GPU function in one device, access to Arm proprietary design and test documentation, support from NXP and Arm in relation to use of the processor in a safety-critical avionics environment, and subsequent late discovery of architectural issues with the SoC)
- Lack of prior references for avionics certification of Arm-based multi-core architecture
- Ability to meet CAST-32A[26]/AMC 20-193/AMC 20-193 objectives (including mitigation of multi-core interference channels between processor cores and GPU)
- Complex integration of the COTS software stack (Helix Platform and CoreAVI graphics driver and libraries)

ScioTeq selected Helix Platform[27] to provide the RTOS environment running on the display platform's i.MX8QM processor. Helix Platform is an evolution of VxWorks 653, which has successfully been deployed in many safety-critical avionics applications running on single core and multi-core Power Architecture processors, including the Airbus A3R program, which has completed DO-178C certification with Design Assurance Level (DAL) A applications running simultaneously on multiple cores.[28] Helix Platform also provides an operating system segment (OSS) that has been certified to conform to the FACE Technical Standard 3.0 Safety Base Profile and Security Profile. This enables existing ARINC 653 and FACE applications to be ported to Helix Platform.

The technical implementation of Helix Platform on the Arm multi-core processor involved some changes compared to VxWorks 653 on Power Architecture multi-core, due to architectural differences. This included changes to software implementation of virtual timers on Arm, due to challenges when being used with ARINC 653 time partitioning. The result was a modification to the implementation of the Helix Platform Type 1 hypervisor that decoupled the hypervisor and guest OS scheduling while enforcing synchronization. This resulted in a significant improvement in the impact of guest OS execution time on the hypervisor.

During the development program, further analysis of the processor architecture revealed some insights and potential issues for the use of i.MX8QM processor architecture in a safety-critical avionics application.

Arm Cortex-A72 Core L1 Cache Protection

The Arm Cortex-A72 dual cores each have an L1 instruction cache with parity-bit protection and an L1 data cache without protection. This is adequate for some types of applications but is inadequate for DO-178C/ED-12C DAL A safety-critical avionics applications, where detection and correction of single-bit errors is necessary and can be achieved using error correction code (ECC) hardware. To mitigate this issue, the decision was taken to use only the i.MX8QM's Arm A53 quad cores, which each have an L1 instruction cache with parity-bit detection and an L1 data cache with single error correction/double error detection (SECDED). The unused cores are put into idle mode and monitored to ensure that they do not cause multi-core interference.

Arm A53 Core L2 Cache Partitioning

The Arm shared L2 cache implementation does not provide cache partitioning capabilities equivalent to those available in the NXP QorIQ T2080. However, this could potentially be addressed via cache coloring through use of non-contiguous memory per partition. The potential impact on performance would need to be assessed.

Cross-Core Clock Synchronization

There are multiple potential clock sources in complex SoCs that can be used by individual processor cores. However, in an ARINC 653 periodic processing system, a common time reference is needed to ensure alignment of ARINC 653 minor frames on the individual processor cores and minimization of jitter.

GPU Scheduling in Mixed-Criticality Systems

Multi-core considerations remain a significant challenge for avionics systems developers, as applications try to take advantage of improved hardware performance. Many of the advances in both CPU and GPU performance come from preemptive and scheduling techniques that are difficult to predict or control and therefore create challenges in ensuring deterministic system execution. In addition, there are the development challenges of managing shared resources

across independent processing units. While developers can use many approaches to address these issues — semaphores and interrupts, for example — these solutions can prove detrimental to system determinism. The industry has proceeded with caution in implementing CPU multi-core solutions, yet GPU coordination represents a similar challenge. The GPU is a highly parallel processing unit, and, unlike CPUs, it offers no ability to control the scheduling activities of the processor.

The method of addressing this in graphics applications has been to essentially decouple the operation of the GPU from the CPU and main application thread, using ring buffers. In simple form, when the application is ready to display an image, it takes an image from the last completed ring buffer and passes it to the display controller, even though the GPU might be working on a new image at that moment. As long as the GPU can sustain a rate of image processing that matches or exceeds the application requirements, the decoupling of the processors is invisible to the end user. The application needs to be able to determine that the worst-case execution time (WCET) of any image is less than the maximum allowed rendering time for the application performance. This challenge now needs to be addressed in the context of using hypervisors to support mixed-criticality applications sharing GPU resources. Application developers can maximize the performance of the CPU hardware and reduce system footprint by taking advantage of multi-core SOCs such as the NXP i.MX8QM. However, this creates new elements of deterministic behavior and resource management that must be addressed within the GPU framework. The method of ensuring safe execution of the GPU in mixed-criticality environments can vary and is determined by a combination of hardware support and application requirements. For the ScioTeq Display solution, a method of IPC was the most appropriate means of communication between lower- and higher-criticality elements of the system. Given that the GPU cores of the i.MX8QM cannot support independent operation, for example, memory space is shared between cores. If a driver was used in direct mode, lower-criticality applications would reside in the same memory space as the GPM, which directly communicates with the GPU, resulting in the potential for the DAL C application to impede the operation of the DAL A application. Using the IPC protocol for lower-criticality application partitions enforces the required isolation for a mixed-DAL deployment. High-criticality application partitions have the freedom to use either IPC or direct call communication methods (Figure 11).

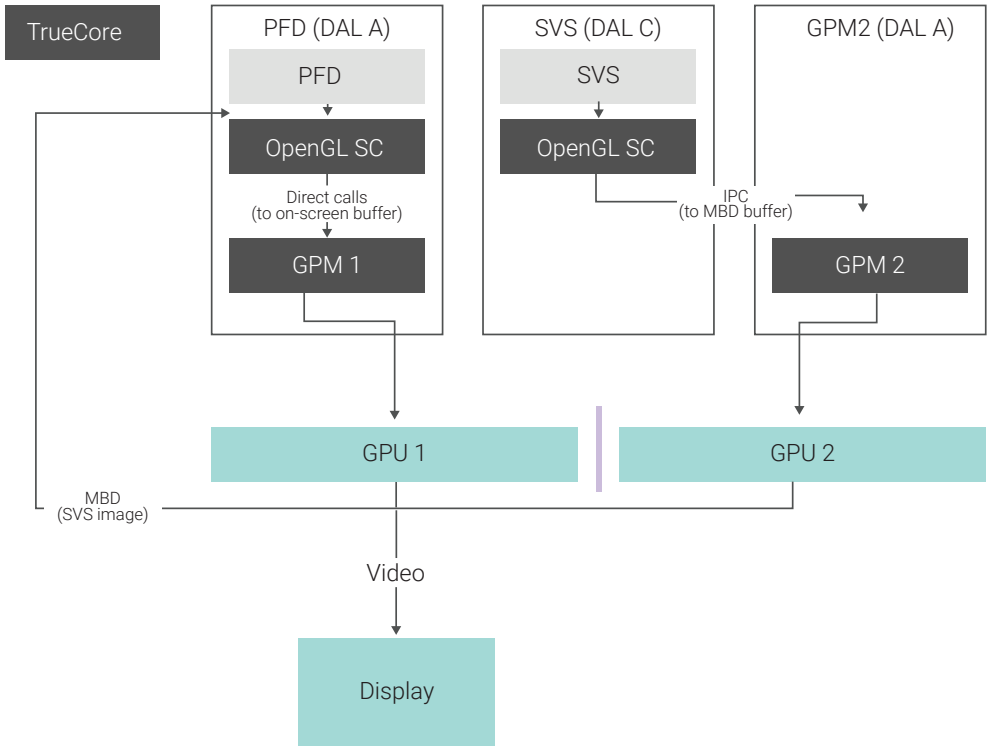


Figure 11. Dual GPU architecture

CoreAVI provides two technologies that are essential to enable multi-core and mixed-criticality implementations: Vulkan SC[29] and TrueCore™[30]. Modern CPUs with increased numbers of cores achieved increased performance by spreading workload over those cores. However, this increase is achieved by reducing the performance of individual cores in favor of a higher count of cores. The multi-core structure means that monolithic code (anything that cannot take advantage of multiple cores) is likely to experience performance constraints. OpenGL SC and other older drivers were designed for single CPU implementations and do not take advantage of additional cores. Vulkan is designed for multi-core and puts a lower performance overhead on individual cores, while being able to benefit from spreading the workload over multiple cores.

TrueCore provides software-based monitoring to detect failure of the GPU processor. By implementing TrueCore in the DAL A partition in a mixed-criticality environment, the correct functioning of the GPU is assured for all partitions, regardless of criticality. TrueCore is used to replace hardware-based monitoring (often implemented in an FPGA), which can limit the flexibility of design while adding to hardware cost and complexity. TrueCore enables a more software-defined–centric architecture and thereby gives application designers more flexibility.

Porting Helix Platform to NXP i.MX8 Quad Max

The implementation of Helix Platform involved hardware-dependent and hardware-independent code. The hardware-dependent code is responsible for low-level architecture-specific and device-specific initialization and control within the hypervisor and guest operating systems, and much of this is implemented in the respective board support packages (BSPs). The hardware-independent core provides an abstraction from the underlying hardware and presents a portable abstract interface to applications through ARINC 653 APEX APIs and VxWorks Guest OS (GOS) APIs. The implementation of these interfaces enabled porting of existing applications and middleware from a VxWorks 653 environment to Helix Platform while minimizing changes.

Independent build, link, and load (IBLL),[31] which was pioneered in VxWorks 653 to support RTCA DO-297 processes and enable incremental updates and incremental safety certification, was ported to Helix Platform. This has enabled continued use of existing DO-297 processes and provides the ability to minimize the recertification costs of future incremental updates to the system.

Continuous built-in test (CBIT) was implemented on unused cores to verify that they remained deactivated and did not cause multi-core interference. Techniques that can be used to determine and mitigate multi-core interference at the system level are discussed in a separate paper.[32]

The implementation of the ARINC 653 Health Management Framework in Helix Platform on Arm architecture was able to provide fine-grain control to achieve greater resilience of avionics systems, similar to what had been implemented in VxWorks 653 Multi-Core Edition previously. This was achieved due to the Arm processor providing an independent watchdog per processor core, such as the NXP PowerPC e6500 architecture. (However, this is not the case for all multi-core processors available on the market, since some semiconductor manufacturers' designs have a single watchdog for the processor.) The ARINC 653 Health Management Framework in Helix Platform is configured via XML tables, and the behavior for a watchdog time-out can be configured for the table entry HM_GOS_DEADLINE_MISSED to perform a default recovery action, such as cold restart of the partition or call a user-defined handler. This enables reset of an individual partition on watchdog expiry without impacting other partitions or cores. The resulting ScioTeq system architecture is shown in Figure 12.

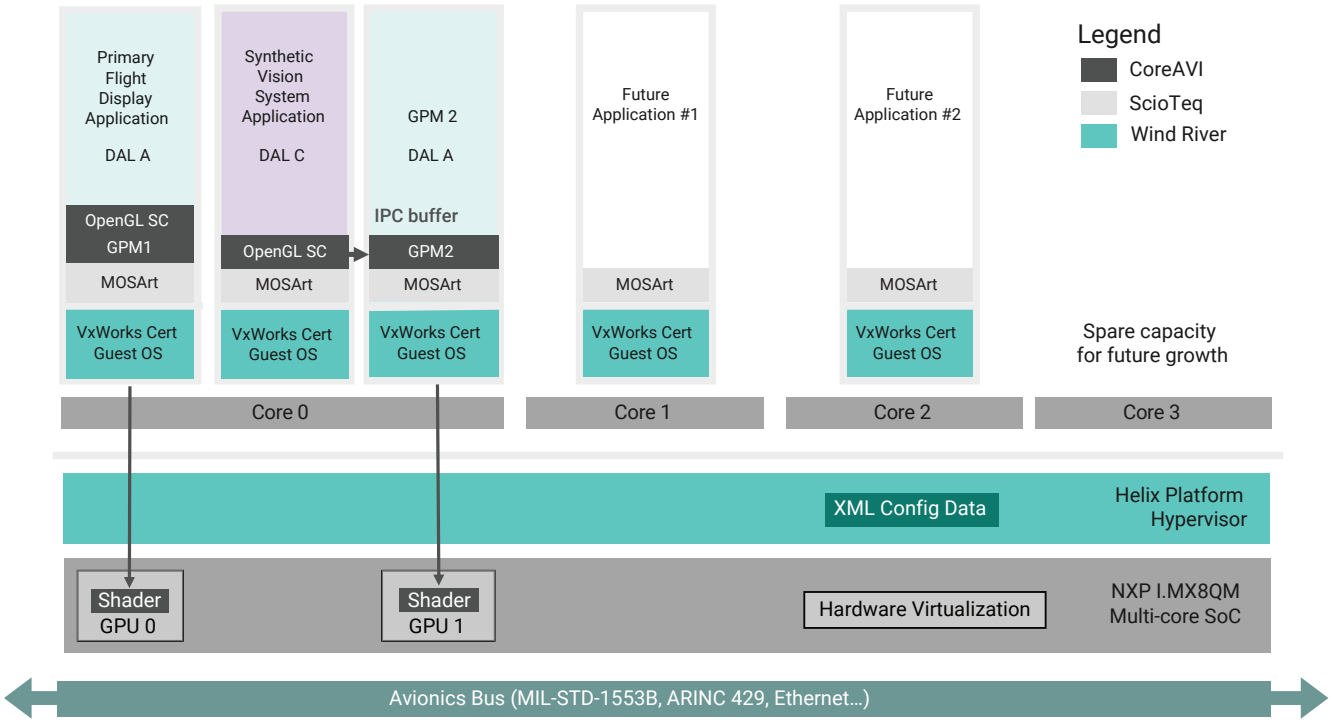


Figure 12. ScioTeq display platform architecture with example configuration

RESULTS AND LESSONS LEARNED

ScioTeq has made significant progress on the development of the next-generation smart display platform, demonstrating prototype computers and displays (see Figure 13) since late 2023, including at the Aerospace Tech Week 2024 conference and exhibition in Munich. The company plans to enter formal qualification testing in the summer of 2024, and customer contracts for the new display platform have already been publicly announced.[33,34] ScioTeq has already undertaken many DO-178C/ED-12C avionics software safety and DO-254/ED-80 avionics hardware safety certification activities for the next-generation platform and plans to complete SOI4 in late 2024/early 2025.

Figure 13. MFD-5068 multi-function display (Source: ScioTeq)

The development of a next-generation platform presented a number of challenges due to the transition from Power Architecture to complex Arm multi-core, transition to a different GPU, and the software runtime environment. Challenges included technical unknowns and program risk but also the need to develop strategic alignment with new stakeholders in the supply chain and close relationships with additional subject matter experts.

Technical risks related to performance and mixed-DAL rendering have been mitigated such that risk sheets are now closed. The lesson learned in this instance is that the project should always have alternative options (a Plan B) in case the selected approach (Plan A) does not work out. Engaging in codevelopment and integration with selected partner suppliers on complex new technology also requires trust and a flexible, cooperative mindset with sufficient and recurring interaction to secure alignment and common understanding.

For the first custom fifth-generation smart display, ScioTeq decided to rely on a previously DAL A–certified safety architecture and to secure performance on one Cortex-A53 core. The use of a SWaP-optimized SoC has resulted in significant weight and power reduction. Application migration has been de-risked through the use of OpenGL SC, and V1.0 has provided the required performance.

In terms of GPU performance, while the GPU cores do not have OpenGL SC 1.0 acceleration, the graphics performance when using the GPU cores in companion mode is equivalent to the Permedia 9 GPU. The GPU cores outperform the Permedia 9 when using Vertex Buffer Objects (VBO) programming, a feature available in the library extensions. Transitioning to OpenGL SC 2.0 and/or Vulkan SC 1.0 will result in significant performance improvements compared to those of that latter GPU.

The ScioTeq tradition of application software abstraction and open standards support (ARINC 653 and OpenGL) through MOSArt and the RTOS made a significant positive impact on portability and reuse of existing IP developed and deployed on other platforms. This made a very important contribution to the reduction of technical risk on the program and helped minimize the impact of hardware architecture changes.

The instrumentation capabilities of Helix Platform were used to measure the boot-time performance of a system. Analysis of the timing data revealed that the access to the configuration database was slow and adversely contributed to the boot time. This indicated that there was potential for optimization of device tree access, which have been incorporated into the platform.

CONCLUSION

The selection of a fundamentally different processor architecture and the introduction of a multi-core processor for a next-generation smart avionics display platform presents multiple challenges and unknowns. However, experience gained on ScioTeq's program demonstrates that close collaboration among industry partners can result in a successful outcome.

The use of software abstraction layers and open standards provides significant benefits in minimizing dependencies on underlying hardware architectures and application portability, preserving investment in previously developed applications and other intellectual property.

ACKNOWLEDGMENTS

The authors would like to thank the following colleagues for their input to this paper: Dennis Rice, software architect, Wind River; Daniel Herring, principal software engineer, and Geoff Rivers, senior software developer, CoreAVI.

REFERENCES

1. L. Wallace, "Airborne Trailblazer: Two Decades with NASA Langley's 737 Flying Laboratory," NASA, 1994, ASIN: B000VBQLH4 ntrs.nasa.gov/citations/19940028287
2. "Panoramic Display Avionics in F-35 Joint Strike Fighter Ignites Industry Debate on the Size of Cockpit Screens," Military & Aerospace Electronics, July 21, 2010 www.militaryaerospace.com/home/article/16724144 panoramic-display-avionics-in-f-35-joint-strike-fighter-ignites-industry-debate-on-the-size-of-cockpit-screens
3. "Diminishing Manufacturing Sources and Material Shortages (DMSMS)," U.S. DoD www.dsp.dla.mil/Programs/DMSMS/
4. ScioTeq MOSArt® Platform www.scioteq.com/en/avionics/mosart
5. "ARINC 653 P0-3: Avionics Application Software Standard Interface – Overview", November 15, 2021 www.sae.org/standards/content/arinc653p0-3/
6. "Wind River Helps Develop Safety-Critical Helicopter Touch Screen Unit," Wind River case study www.windriver.com/success-stories/agusta-westland
7. "Development and Certification of a Safety-Critical Avionics Touch Screen Display Using Open Standards," Addressing Systems Safety Challenges, Proceedings of the 22nd Safety-Critical Systems Symposium, Brighton, U.K., February 5–7, 2014 https://www.amazon.co.uk/Addressing-Systems-Safety-Challenges-Safety-critical/dp/1491263644/
8. RTCA, "DO-254, Design Assurance Guidance for Airborne Electronic Hardware," April 19, 2000 my.rtca.org/productdetails?id=a1B36000001IcjTEAS
9. EUROCAE, "ED-80: Design Assurance Guidance for Airborne Electronic Hardware," 2000 eshop.eurocae.net/eurocae-documents-and-reports/ed-80/#non-member

REFERENCES (CONT'D)

10. RTCA, 2011, "DO-178C: Software Considerations in Airborne Systems and Equipment Certification," 2011 my.rtca.org/productdetails?id=a1B36000001IcmqEAC
11. EUROCAE, "ED-12C: Software Considerations in Airborne Systems and Equipment Certification," 2012 eshop.eurocae.net/eurocae-documents-and-reports/ed-12c-with-corrigendum-1/
12. ARINC653P0-3, "Avionics Application Software Standard Interface Part 0 Overview of ARINC 653," 2015 www.sae.org/standards/content/arinc653p0-3/
13. Wind River, "What Is FACE™?" www.windriver.com/solutions/learning/face
14. U.K. Government, "PYRAMID" www.gov.uk/government/publications/pyramid
15. ARINC 661, "Cockpit Display System Interfaces to User Systems, Part 1" www.sae.org/standards/content/arinc661p1-8/
16. IEEE and the Open Group, POSIX™ Certification posix.opengroup.org/
17. OpenGL www.opengl.org
18. Khronos Group, "Vulkan® 1.3.280: A Specification (with All Registered Extensions)" registry.khronos.org/vulkan/specs/1.3-extensions/pdf/vkspec.pdf
19. EASA, "AMC 20-193: Use of Multi-core Processors," January 2022 www.easa.europa.eu/sites/default/files/dfu/annex_i_to_ed_decision_2022-001-r_amc_20-193_use_of_multi-core_processors_mcps.pdf
20. FAA, "Advisory Circular 20-193: Use of Multi-core Processors," January 2024 www.faa.gov/documentLibrary/media/Advisory_Circular/AC_20-193.pdf
21. NXP, "Product Longevity" www.nxp.com/products/nxp-product-information/nxp-product-programs product-longevity:PRDCT_LONGEVITY_HM
22. GPU Score, "Basemark GPU" www.gpuscore.com/benchmarks/basemark-gpu/
23. NXP, "NXP i.MX8QM Applications Processor Reference Manual" www.nxp.com/security/login?TARGET=https%3A%2F%2Fwww.nxp.com%2Fwebapp%2Fsecure%2Flogin.SAMLSecuredController.sp%3Faction%3DforwardToDestination
24. ScioTeq PU-5200 www.scioteq.com/en/product/certified-avionics-display-computers/pu-5200
25. ScioTeq MFD-5068 www.scioteq.com/en/product/smart-displays/mfd-5068-new
26. Certification Authorities Software Team, FAA, "Multi-core Processors, CAST-32A Position Paper," November 2016 www.faa.gov/aircraft/air_cert/design_approvals/air_software/cast/cast-32a.pdf
27. Wind River, "Enabling the Migration to Future Aerospace & Defense Systems" www.windriver.com/resource/wp-enabling-the-migration-to-future-aerospace-and-defense-systems
28. Paul Parkinson, "Airbus A3R Completes Certification with Support from Wind River Safety-Critical Multi-core Platform," Wind River, January 23, 2023 www.windriver.com/blog/airbus-a3r-completes-certification-with-support-from-wind-river-safety-critical-multi-core-platform
29. CoreAVI, "Vulkan SC Safety-Critical Graphics and Compute Library" coreavi.com/wp-content/uploads/CoreAVI-White-Paper-Vulkan-SC-Safety-Critical-Graphics-and-Compute-Library.pdf
30. CoreAVI , "TrueCore™ GPU Safety Monitor" coreavi.com/product_category/gpu-safety-monitor/
31. Paul Parkinson, "Safety-Critical Software Development for Integrated Modular Avionics," Wind River www.windriver.com/resource/safety-critical-software-development-for-integrated-modular-avionics
32. Rapita Systems and Wind River, "Mitigation of Interference in Multi-core Processors" www.rapitasystems.com/downloads/mitigation-interference-multicore-processors-webinar
33. ScioTeq, "ScioTeq Advanced Avionics Displays Selected for Airbus A330 MRTT Multi-Role Tanker Transport Aircraft," September 29, 2023 www.scioteq.com/en/news/scioteq-advanced-avionics-displays-selected-airbus-a330-mrtt-multi-role-tanker-transport
34. General Atomics, "GA-ASI Selects ScioTeq to Support Detect and Avoid Program," June 20, 2023 www.ga.com/ga-asi-selects-scioteq-to-support-detect-and-avoid-program

WNDRVR