dap TECHNOLOGY

Getting Started with AS5643

Prepared by:

Dr. Michael Vonbank, CEO

DAP Technology B.V.
Zutphenstraat 67
7575EJ Oldenzaal
The Netherlands

Phone: +31 541 532941
Email: michael@daptechnology.com

1   Objective

The objective of this proposal is to present a starting point for the development effort for an AS5643 avionics network system and/or subcomponents of such a system.

It is the dedicated goal to identify necessary steps to build a knowledge base, relevant accompanying literature and documentation as well as suitable tools.

This document is not intended to itemize individual development steps. Especially during the System Architecture and System Engineering phases the necessary tasks depend too much on program specific details and therefore cannot be summarized in such a document.

2   Scope

This document outlines tasks needed to jumpstart and conceptualize any 1394B/AS5643 development project.

3 Project Outline

3.1 Phase A: Understand Technology

IEEE-1394-2008 is the base specification on which AS5643 builds upon. A good understanding is essential to later on efficiently architecture a sound avionics network. While portions of IEEE1394 are only optional for the implementation of AS5643 it is highly recommended to also study these (i.e., isochronous data transmission, Bus Management, …) as they are vital to the big picture understanding of the technology.
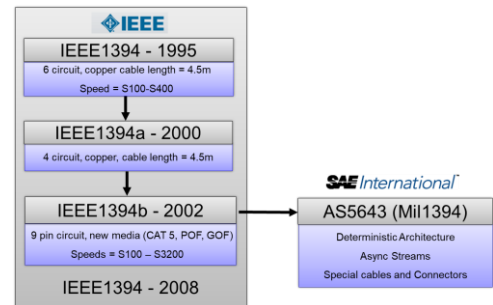
The following standards are essential to the understanding of the technologies involved:

- IEEE-1394-2008 Specification   Source: IEEE
- AS5643[1] Interface Requirements   Source: SAE International

Helpful tools at this stage for understanding the bus arbitration, tree formation, the messaging layers, the various protocols, etc. are:

- 1394 Bus analyzer[2] with AS5643 module   Source: DapTechnology
- 1394/AS5643 nodes (repeaters/host adapters/…..)   Source: DapTechnology

Furthermore, comprehensive training seminars are likely a good source of information material and can be delivered custom tailored to customers' needs.

- 1394 & AS5643 Technology Training   Source: DapTechnology

3.2 Phase B: Breadboard Sample Implementations

This is an important and often overlooked stage. In reality AS5643 is a requirements framework rather than a hardcore specification. The way it is written leaves room for several implementation options and variants. Identifying these options and determining their pros and cons are essential steps in the process of developing an AS5643 avionics system from the scratch. The goal is to test/evaluate/qualify/validate alternative system architecture concepts.

Examples:

a) The STOF frame rate of 12.5ms is a rough guideline. Many programs have adopted this rate as it gives a good compromise between sensor/actuator latencies,

---

[1] 07/2018: currently in Rev B
[2] At this stage a single-node analyzer is sufficient. However, in the light of further reuse a multi-node analyzer is probably a better investment.

responsiveness, dataflow and implementation complexity[3]. However, if the processor on the device is not powerful enough then it could be beneficial to vary the STOF rate in order to mitigate such device/network constraints. Getting an understanding of the behavior of STOF timing, TX/RX offsets and the associated data exchange path and their overall interrelationship is critical.

b) AS5643 and its underlying IEEE1394 data exchange requires particular attention to signaling quality, proper cabling, right cable selections, cable distance, and EMI immunity. By using typical components mentioned in the specification the majority of criticalities can be overcome but the specification cannot address application and implementation specific challenges arising from EMI bursts, etc.….

The following standards/documents are essential for this stage:

- Everything from previous phases
- AS5643-1 S400 Copper Media Slash sheet   Source: SAE International
- AIR5654 Applications Handbook   Source: SAE International

The following components are needed in this phase:

- Multi-node 1394 Bus analyzer with AS5643 module   Source: DapTechnology

  Bus analyzer is now used to analyze the bus traffic and stimulate traffic. For example the FireSpy Generator function can be used to generate STOF frames for various frame rates. Furthermore ASM packets can be generated to stimulate responses from other network devices. Specifics of the encapsulated data payload can be tested, Like, for example, what happens when a VPC calculation fails? Or, what actions need to be taken if the heartbeat is a received stream of packets is not updating? The FireSpy line of analyzers are perfectly suited to perform all these tests and checks.

- Passive devices (Repeater, Media Converters, …)   Source: DapTechnology

  Passive devices (PHY Layer only) will be handy to use as "dumb" network devices. Their function will be to represent network devices and simulate a network. They will be needed to study the behavior of loop-breaking and provide enough notes for simulating an avionics redundancy (on a physical layer level).

- Active devices (Interface cards, …)   Source: DapTechnology

  Active devices (PHY and Link Layer) can be used for the same reason as the passives but have the additional benefit of performing early analysis on the ASM RX/TX. This will allow for tests on the ASM packet payload structure and the respective RX/TX timing.

---

[3] It is advisable to keep the STOF frequency aligned with the base 1394 PHY clock of 24.576 Hz and keep one frequency domain.

A requirement for host adapter cards is the presence of suitable Software stack. Standard OHCI interface cards are supported by some operating systems but are not well supported. Instead, using a dedicated SW stack like FireStack is beneficial since it is actively maintained and supported by DapTechnology. While using OHCI adapter cards provides a suitable entry level their effectiveness needs to be questioned. The reason for this is that standard OHCI Link Layer controllers (LLCs) are fabricated from standard 1394 silicon that is not "AS5643 aware". Meaning there is no intrinsic support for the AS5643 requirements (protocol encapsulation, AS5643 compliant filtering, message integrity verification, message timing). All these functions need to be manually addressed in the SW application layer which can create a sizeable abstraction layer & entry barrier[4]. Dap's FireTrac is a solution dramatically lowering this entry barrier. While a bit more expensive, the extra price of FireTrac is easily justifiable by a streamlined path to a working AS5643 device. And FireTrac is fully supported by Dap's FireStack, which is available for VxWorks, Windows and Linux environments and can later on ported over to other real-time operating systems.

3.3   Phase C: Develop a System Architecture[5]

This phase defines the fundamental organization of the avionics system, embodied in its components, their interrelationships and to the environment, and the principles governing its design and evolution. It shall create a representation of a system, including a mapping of functionality onto hardware and software components, a mapping of the software architecture onto the hardware architecture, and interaction with these components. The objective is to develop an Interface Control Document ICD) for the aircraft's network system.

The main tasks involve – but are not limited to - architecting the overall bus system and topology, the needed level of fault tolerance (single, dual or triple redundancy), security questions (encryption or not), the messaging system (ASM data format, data content encapsulation), TX/RX/DataPump timing, Cross Channel Data Link, data transaction management (what content is exchanged, how and when do CCs engage into this type of communication, …). What sort of device I/O shall be used (off-the-shelf silicon or FPGA)? What are my longest cables runs between individual nodes? What is my data communication speed and what wiring gauge do I need for that? How much weight does the wiring add to my overall aircraft weight? Are all of my choices sustainable over

---

[4] At a later stage other concerns become relevant. Host resource utilization can be addressed by offloading these tasks to AS5643-enables LLC devices. These do not exist as off-the-shelf silicon but are provided via solutions with LLC-IP layers.

[5] Only needed when a complete avionics system from ground up in designed by the prime contractor/integrator. Subcomponent vendors/manufacturers will typically be given a detailed component specification and very likely will not be confronted with the phase.

---

the life span of the end product? How can the entire solution be manufactured? This list is by no means complete and details depend on corporate expertise, the need to reuse existing technologies, investments and processes as well as high level strategies as many of these question affect not only corporate but also national security aspects/concerns.

Things are not getting easier that in fact only a handful of "freelance"[6] technology experts exist worldwide that can assist and consult regarding these matters. Their involvement can be helpful but will require addressing the adequate security concerns etc. While this might not always be possible it is essential to have the right tools at hand. Decisions like the ones mentioned above will require simulation, verification and validation processes before a stable and matured system architecture can be derived.

At this point DapTechnology suggests paying particular attention to the choice of the interface I/O. In normal project planning the particulars of the network I/O typically would be addressed in the next phase (i.e. System Engineering). However, the current AS5643 standard was developed based on the availability of off-the-shelf silicon from only one vendor (i.e. Texas Instruments). This has in many regards limited the overall I/O architecture to a degree where the resulting consequences need to be addressed on the system level. Certain very much desirable features/functions are simply not possible with the TI silicon. DapTechnology therefore strongly suggests to inspect the overall system design from a very different angle and consider the usage if IP cores for the device design.

The following standards/documents are essential for this stage:

- Everything from previous phases

The following components are needed in this phase:

- Everything from previous phases
- FPGA IP Cores (FireCore, FireLink, FireGate)   Source: DapTechnology

3.4   Phase D: System Engineering

With an ICD now in place the aircraft's system engineering can now commence. It is the objective to address issues such as requirements engineering, reliability, logistics, coordination of different teams, testing and evaluation, maintainability and many other disciplines necessary for successful system development, design, implementation, and ultimate decommissioning of the avionics system.

As the avionics network now has to address specifics of a particular avionics program (aircraft) the necessary tasks become rather difficult to specify. They also differ

---

[6] The term "freelance" refers to individuals that are not affiliated/employed/… by specific avionic corporations and/or government agencies. They might be part of corporate entities that specifically deal with AS5643 and 1394 technology but have no particular affiliation with specific avionics programs.

significantly depending to on level of involvement (e.g. top tier system integrator vs subcomponents manufacturer).

3.5   Phase E: Develop a Test Procedure

For IEEE-1394 and AS5643 enabled no Compliance Certification Authority exists.

Therefore a comprehensive test methodology has to be developed in order to ensure compliance with the design specification and performance criteria by the main system integrator and/or the subcomponent vendors.

The SAE organization has developed test plans (AS5657) for verification of the device compliance and Dap encourages all involved parties to engage in rigorous testing at a very early stage.

The following standards/documents are essential for this stage:

- AS5657 Test Plan for AS5643 Interface Req.   Source: SAE members only
- AS5703 Test Plan for AS5643-1 S400 Copper Media   Source: SAE members only

The following components are strongly recommended in this phase:

- AS5657 Automated Test Environment   Source: DapTechnology

   The AS5647-ATE is designed to automate device compliance testing according to AS5657. The purpose of this tool is to automate and simplify the task of AS5643 compliance verification and testing, which of course is essential for device compatibility within avionics and aerospace programs. The tool utilizes and benefits from proven test and interface components already developed by DapTechnology and adds test executive functionality together with predefined test scripts and a customization and expansion environment.[7]

---
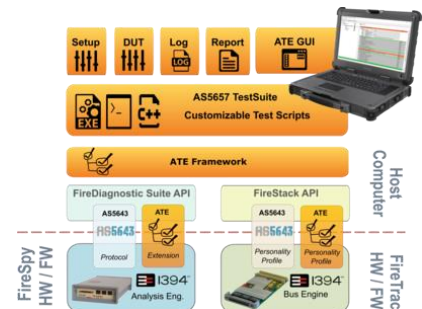
[7] The AS5657-ATE is useful throughout the entire Life Cycle of a product starting at an early stage of engineering, extending through verification and production and even expands into sustainment of a typical avionics program. Thanks to its expansion capability – the ATE Framework and the test framework can be modified and extended – customization and adaptation to reflect specific project requirements can be implemented easily and with straight forward methods. Usage areas include but are not limited to:

- Requirements Definition
- Software Systems Design
- Software Quality Assurance
- Verification and Validation of hardware modules
- Product Acceptance
- Systems/Network Integration
- Systems Simulation (includes load testing, HIL, PIL, etc.)
- Module/LRU Qualification Testing (e.g. VMC, Flight Control Actuator, etc.)
- …

- [FireMatrix](#)   Source: [DapTechnology](#)

  FireMatrix is an example of an integrated 1394 analysis and test platform. In its simplest case it can function as an out-of-the-box 1394 Bus analyzer solution without the need for external PCs, laptops etc. However, it can also be utilized as a platform to build large and integrated test systems that are relying on FireSpy features and functionality. Higher-level applications can easily interface to the analyzer's API and thus enable a variety of test applications.

  Besides its well-proven standalone and card-based FireSpy bus analyzer solutions, DapTechnology can offer such out-of-the-box FireMatrix solutions. The FireMatrix in general is comprised of a portable PC platform hosting a flexible number of 1394 analyzer cards. Limited only by the number of available PCI slots, the FireMatrix is a truly scalable system with regards to the number of 1394 bus analyzer engines. As an added feature the FireMatrix also offers support for IRIG-B time stamping.

- [Cable Tester](#)   Source: [DapTechnology](#)

  The Cable Test System is a perfect example of a special solution built on existing DapTechnology building blocks. The system, which was developed with the cooperation of a key A&D cable harness supplier, replaces a cable test system with proven FireSpy Technology. The new system makes use of the FireSpy engine in combination with custom hardware and software to improve test times and reliability. The system is a true functional test system that verifies if AS5643 signaling can pass a cable under test with a series of cable extensions.