Health Management Architecture and Logic for a Distributed Complex Integrated Modular Avionics System

Yang Li *

Shanghai Aircraft Design & Research Institute of COMAC, Shanghai, China

* Corresponding author e-mail: liyang5@comac.cc

Abstract. More and more advanced aircraft systems are equipped with Integrated Modular Avionics (IMA) system, which presents a series of challenges related to fault detection and reporting. This paper addresses some of those challenges and proposes a new Health Management (HM) architecture for a typical distributed complex IMA system. A significant objective of this HM architecture is to ensure that the IMA system integrity and availability meet the system safety requirements. Due to the highly complexity and the integration of IMA system, it is very difficult for both designers and verifiers to develop a practical HM application in charge of detecting, isolating and reporting failures to the related hosted applications and other non-IMA systems. For this purpose, this paper proposes a new IMA health management logic for fault isolation and reporting.

Keywords: IMA; Health Management; BIT; Fault Isolation.

1. Introduction

Integrated Modular Avionics (IMA) "is a shared set of flexible, reusable, and interoperable hardware and software resources that, when integrated, form a platform that provides services, designed and verified to a defined set of safety and performance requirements, to host applications performing aircraft functions" [1]. Nowadays, more and more on-board avionics systems in both military and civil areas equip with IMA systems [2]. Compared with federal avionics architecture, IMA uses a set of new technologies and features to provide enhanced functionality, reduced development and maintainability costs, and a new architecture that is configurable and easily accommodates hardware updates due to parts obsolescence [3]. Within an IMA architecture, the IMA system is comprised of an IMA platform and several hosted applications. The IMA platform provides shared resources including common processing modules (CPM), communications network modules (SWITCH), and Remote I/O Units or Remote I/O Modules (RIUs/RIMs), while the hosted applications performing aircraft functions are allocated to these shared resources. Several allocations increment that tailor the many reusable components are performed to meet the overall requirements of timing, bandwidth, and performance [4].

For high integrity and high availability, IMA system introduces the functions of health monitoring and fault management to ensure expected behaviors in the presence of system faults, which are to identify and report the failures of hardware, operation systems and applications, and locate and handle the specific fault to conduct trouble-shooting work after it receives the report from specific health management module [5]. Usually, the IMA platform health management consists of detecting faults, failures and errors, correctly identifying them when they occur and performing the appropriate response, which are independent of the hosted applications.

Although the IMA architecture has gone a long way to improve the significant challenges of weight and reusability, a series of system problems come up along with the increased integration complexity due to the very flexible and configurable nature. In the form of state correlation, the problem impacts between hosted applications' states is inherent on the IMA platform. When one shared resource fails and loses its function, the hosted applications using this resource may fall into fault states and the related aircraft systems will lose the expected functions. Therefore, analyses on state correlation are essential to find out potential hazards considering components' states of IMA platform and its modules due to the integration of multiple applications and resource sharing. Additionally, IMA platform is capable of prognosing the functional operation condition and provides the information to

Onboard Maintenance System (OMS) and Flight Crew Alerting System (FCAS). Moreover, the designer of IMA platform shall have a thorough understanding of health management methods for optimal selection of monitoring strategies, tools, and algorithms needed to detect and isolate[6~9].

This paper proposes an IMA health management architecture with new faults isolation and reporting logic. The rest of the paper is organized as follows. An overview of IMA system architecture and Health Management function and architecture are presented in section 2. Then, a faults isolation and reporting logic for the IMA platform is proposed in section 3. Finally, section 4 makes a conclusion of our work and some further work.

2. IMA Health Management Architecture

2.1 IMA System Architecture

Figure 1 illustrates an example that identifies boundaries of a distributed complex IMA system and presents considerations for fault management and reporting [1]. This IMA system consists of common processing modules (CPM), communications network modules (SWITCH), Remote I/O Units (RIUs) and I/O Modules (IOMs). The RIU and the IOM are similar devices, one is located remotely and the other at the resource center. Aircraft systems can use the distributed complex IMA system resources by several ways, which are dependent on the specific aircraft system functions and requirements. Usually, all of the IMA system resources are defined by specific configuration data to allocate to a number of hosted applications that work with IMA system and other related federated Line Replaceable Units (LRUs), sensors and effectors together to perform the expected aircraft functions.

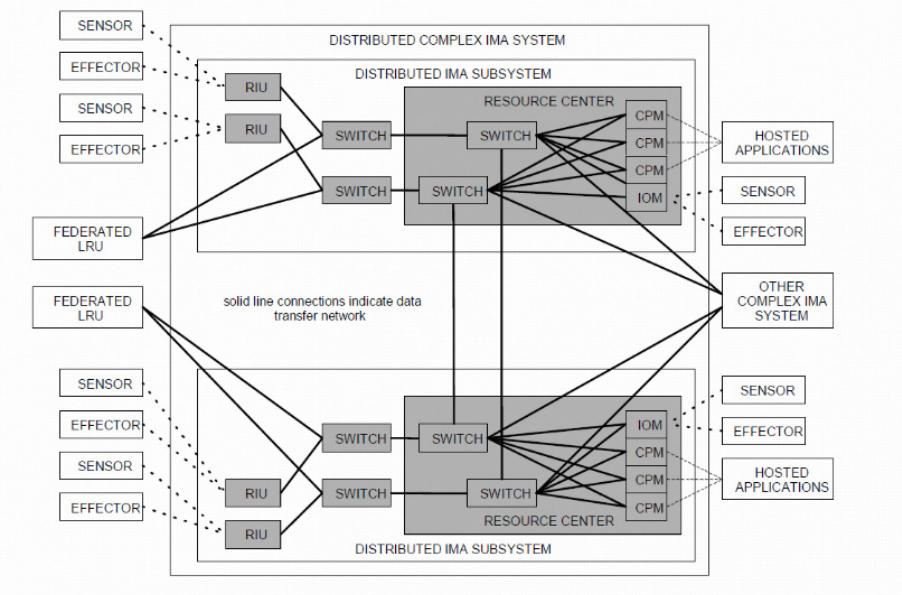


Figure 1. Example of the Distributed Complex IMA System [1]

2.2 IMA Health Management Function

The IMA platform provides Health Management (HM) function to detect and report failures in the IMA system resources including all the hardware and software. In addition, HM function provides an interface to Onboard Maintenance System (OMS) and Flight Crew Alerting System (FCAS), in order to communicate the IMA system health for the purpose of dispatch and maintenance, as well as appropriate flight operation actions. The detailed descriptions of IMA HM function are listed in Table I.

Table 1. IMA System Health Management functions

No.	Functions Description
FUN-HM-01	Detect faults within the IMA platform, as well as faults with regard to the physical interfaces between the platform and other aircraft systems.
FUN-HM-02	Filter detected faults to mitigate nuisance fault reporting.
FUN-HM-03	Report IMA fault status to Flight Crew Alerting System (FCAS) for indication to the flight crew.
FUN-HM-04	Report IMA fault status to OMS, for indication to the maintenance personnel.
FUN-HM-05	Retrieve fault history upon request from OMS.
FUN-HM-06	Manage IBIT requests from OMS and report IBIT status to OMS.
FUN-HM-07	Report IMA platform status to the Hosted Applications, for them to react to abnormal conditions.
FUN-HM-08	Log detected hard faults to Non-Volatile Memory (NVM) for failure analysis.

2.3 IMA Health Management Objectives and Consideration

The Health Management primary objective of IMA platform is to ensure the system integrity and availability meeting the system safety requirements, as well as to maximize aircrew confidence. In addition, it should maximize fault detection, properly handling 100% faults that have safety implications, with minimum impact on IMA system operation. In addition, each fault in the redundant function shall be considered to avoid reducing the operation of the system, and must be reported in order to take appropriate operation and maintenance measures. Last but not least, the BIT reporting shall provide information for fault isolation to a unit/module level whenever possible.

2.4 Built-In Test and Health Management Application

Built-In Test (BIT) is a test approach using built-In test equipment or self-test hardware and software that is internally designed into the supported end item to test all or part of that item. Herein, as a practical means of health management, Built-In Tests (BITs) are used to detect faults within the IMA platform, as well as faults regarding interfaces between the IMA system and other systems. There are three modules of BIT applied in IMA health management:

- Power-Up BIT (PBIT) that executes during power-up to detect faults regarding the platform configuration, hardware, as well as other faults that cannot be monitored on a continuous basis.
- Continuous BIT (CBIT) that executes continuously during normal platform operation to monitor the status of the platform, after PBIT is complete, as background, non-interfering tasks. All safety critical functions are monitored by CBIT.
- Initiated BIT (IBIT) that is performed upon receipt of a command and may include operator interaction.

2.5 IMA Health Management Architecture

IMA Health Management function distributes BITs on each IMA module and unit (CPMs, SWITCHs, RIUs, and IOMs) that provide the detection of hardware faults and other aspects of the platform status. BIT is a distributed function, meaning each unit or module is responsible for performing its own internal tests and for reporting status. Therefore, a specific application hosted in the IMA system, named as Health Management Software Application, or HMSA, is designed to act as the central system status manager for executing the IMA health management function.

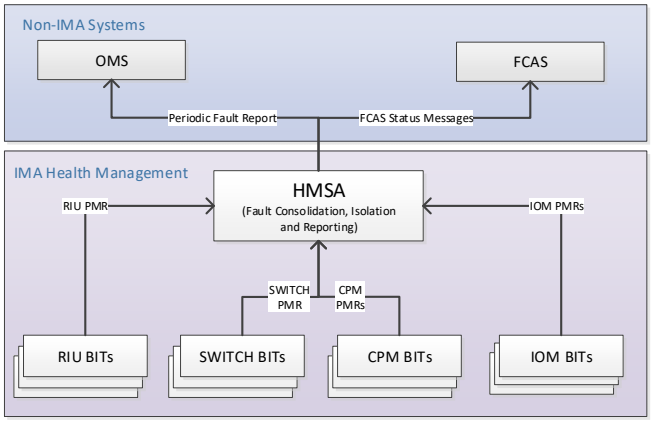


Figure 2. IMA Health Management Architecture

Figure 2 shows the basic flow of the IMA platform level fault consolidation, isolation, and reporting. IMA module/unit level BITs and the IMA HMSA work together to effectively report problems in the IMA system. The CPM BITs, SWITCH BITs, RIU BITs, and IOM BITs perform fault detection and containment, and isolate failures to the functional level within a unit or module. And the BITs are also responsible for consolidating test failures into Periodical Monitor Reports (PMR) that are periodically sent to the IMA HMSA. The IMA HMSA uses the fault information to isolate failures and report them to the OMS and FCAS. The fault logs in each LRU/LRM are used at shop-level for debugging units/modules that are returned for repair.

In addition to the above, the IMA HMSA performs ADN (Aircraft Data Network) connectivity by gathering Periodical Monitor Reports (PMR) data from the switches and the ADN Health Monitor Periodic Reports (AHMPR). Based on the status of the resource center, the status reported by associated units and modules, and ADN connectivity results, the IMA HMSA determines overall system status and provide the appropriate advisory messages to FCAS and/or OMS.

Last, the IMA platform provides resources to multiple hosted applications performing aircraft functions, so the specific effects of IMA platform failure conditions are dependent on how hosted applications use the IMA platform resources. The safety objectives for the IMA platform are generic and agnostic to the physical implementation of functions using the IMA Platform.

3. IMA Health Management Logic

Most of the fault isolation within the IMA system is performed by default and requires minimum fault isolation logic at the IMA platform level. Any failure reported as an internal failure is automatically isolated to that unit or module. This should pertain to most of the failures within the IMA platform. Failures which result in complete loss of communication with a unit or module will be considered isolated to the unit/module.

Due to the redundant communication architecture of the IMA platform the only situations that should result in total loss of communication with a unit/module would be one of the following:

- Power is not being supplied to the LRU/LRM.
- The unit/module is not present within the IMA platform.
- The unit/module has a critical internal failure resulting in a “dead box”.

For failures that result in loss of all communication with a unit/module, the unit/module will be reported as failed. For failures that result in loss of partial communication or other ambiguous failure effects, a fault isolation and reporting logic is defined in Figure 3 to determine the location of the failure.

3.1 Unit/Module Fails to Report BIT Data

First, the IMA HMSA collects Periodical Monitor Reports (PMR) from all IMA platform units and modules, including CPMs, SWITCHs, IOMs and RIUs, as well as ADN connectivity data. A unit/module will be considered failed if it does not respond to any communication. If a unit or module fails to report BIT data, the HMSA could not receive the data and then HMSA will report the failed unit/module in the Periodic Fault Report to OMS. Otherwise, the HMSA will receive the BIT data from each unit and module.

3.2 Unit/Module Internal Failure

If any unit/module reports an internal failure detected by PBIT or CBIT, this failed unit/module will be reported to OMS by HMSA via a Periodic Fault Report. If no failed unit/module are reported or internal failures are detected, the HMSA will check whether there is any ADN connectivity problem.

3.3 ADN Connectivity Problem

The ADN connectivity problem is analyzed by HMSA and reported in AHMPR. Because of ADN connectivity, a unit/module fails to deliver a periodic monitor report to the IMA HMSA, which will cause the unit/module to be considered as a failure. If there is no ADN connectivity problem, the HMSA will report no IMA fault to the OMS. Conversely if yes, then the HMSA will analyze whether there is any internal failure from the SWITCH.

3.4 Switch Internal Failure

Each SWITCH will conduct a CRC loopback test to verify whether the frames with incorrect or no Frame Check Sequence (FCS) sent to each port have not been received, and these frames will be collected in the management information base. The HMSA will check whether there is an internal SWITCH fault indication in the base. If yes, the HMSA will send the failed SWITCH to the OMS via a Periodic Fault Report.

3.5 Isolation the Failure to a Single Unit or Module

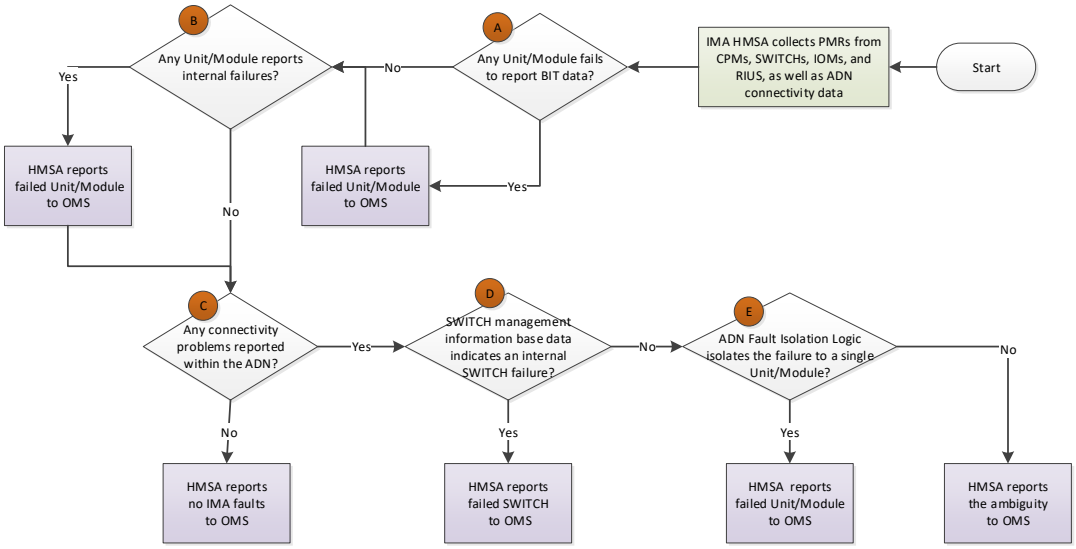


Figure 3. IMA HM Isolation and Reporting Logic

If there is an ADN connection failure other than an internal failure within any single unit/module or SWITCH, the ADN fault isolation logic of HMSA can isolate the fault within the single unit/module and SWITCH pairs, and then report it to the OMS via a Periodic Fault Report. If the location of the failure is still ambiguous, the IMA HMSA will report the existence of ambiguous failure conditions and will list the possible failed units/modules. Afterwards, the IMA HMSA will

report a different fault message to the OMS for each different ambiguous fault detected in the IMA platform. Each message can then be traced back to a specific maintenance action to repair the fault condition.

4. Conclusion

The most of motivation for research on health management techniques is to reduce maintenance costs and improve the availability of aircraft. From the safety and economic perspective, it is necessary to detect, isolate and report the failed unit/module of the IMA system. This paper presents the general introduction and functions of a typical Distributed Complex IMA System. Furthermore, this paper introduces a new Health Management architecture that distributes three types of BITs on each IMA modules and units and sets up a central system health status manager, as Health Management Software Application, or HMSA. At last, this paper proposes a fault isolation and reporting logic for the IMA platform to isolate and report the failed unit/module or ambiguous failures to the OMS. Further work may be conducted on the fault isolation logic considering the different network topology.

References

[1] DO-297, Integrated Modular Avionics (IMA) Development Guidance and Certification Considerations, RTCA Inc., Nov. 2005.

[2] C. B. Watkins and R. Walter, ‘‘Transitioning from federated avionics architectures to Integrated Modular Avionics,’’ in Proc. IEEE/AIAA 26th Digit. Avionics Syst. Conf. (DASC), Dallas, TX, USA, Oct. 2007, pp. 2.A.1-1–2.A.1-10.

[3] L. Sagaspe, G. Bel, P. Bieber, F. Boniol, and C. Castel, ‘‘Safe allocation of avionics shared resources,’’ in Proc. IEEE Int. Symp. High-Assurance Syst. Eng., Oct. 2005, pp. 25–33.

[4] P. Parkinson and L. Kinnan, ‘‘Safety-critical software development for integrated modular avionics,’’ Embedded Syst. Eng., vol. 11, no. 7, pp. 40–41, 2003.

[5] R. Hilbrich and K. R. van Kampenhout, ‘‘Dynamic reconfiguration in NoC-based MPSoCs in the avionics domain,’’ Proc. 3rd Int. Workshop Multicore Software. Eng., 2010, pp. 56–57.

[6] G. Niu, Data-Driven Technology for Engineering Systems Health Management, (2017). doi:10.1007/978-981-10-2032-2.

[7] P. Parkinson and L. Kinnan, ‘‘Safety-critical software development for integrated modular avionics,’’ Embedded Syst. Eng., vol. 11, no. 7, pp. 40–41, 2003.

[8] M. Ben-Daya, Failure Mode and Effect Analysis. London, U.K.: Springer, 2009, pp. 75–90.

[9] ASAAC Standards, Fault Management, NASA, Tech. Rep. DEFSTAN 00- 78/1-2005, Jan. 2005, vol. 2.