Distributed Integrated Modular Avionics

Miguel T. Barros, José A. S. Neves
GMV
PORTUGAL
mtavares@gmv.com jose.neves@gmv.com
José R. Parizi Negrão, Sérgio D. Penna
Embraer S/A
BRASIL
parizi@embraer.com.br sdpenna@embraer.com.br
Marco A. A. Ortiz
Embraer S/A - Defense & Security
BRASIL
mortiz@embraer.com.br

ABSTRACT

Multi-mission certifiable unmanned vehicles require an optimized avionics approach.

This paper describes the reasoning behind using Distributed IMA platforms for the development of unmanned vehicles. These platforms facilitate the achievement of SWaP requirements while adding other IMA benefits such as scalability, reusability, modularity, integration and incremental certification.

This paper also presents a concept for a Distributed IMA platform with reconfiguration capabilities, where the reconfiguration is triggered by changes in the payloads attached to the system. This concept was explored and implemented in the DIMA test bench and its successes are studied.

1.0 INTRODUCTION

Currently, unmanned vehicles (UVs) are based on federated architectures, mostly using general-purpose operating systems and only a few hardware suppliers. In such systems, when it is necessary to integrate a new sub-system from a different supplier, the solution is often to add new hardware and new software components and reintegrate the whole system. An alternative is to integrate everything under the same computational module, leading to a more complex, and an even less flexible system configuration. For an unmanned system, where volume, weigh and energy consumption are frequently main design drivers, both alternatives are non-optimal.

The aeronautical Integrated Modular Avionics (IMA) concept enables sub-systems to be integrated easily without interfering with other sub-systems or the actual base platform. When matured, this will open the possibility of having "UV platforms as a service" that can be effortlessly integrated with different payloads from different suppliers to meet the requirements set by the stakeholders, while reducing integration and testing effort. Indeed, by maintaining an underlying platform based on IMA open standards, functionalities similar to "Plug-and-Play" for payloads can be implemented in UVs. Additionally, an architecture that is common among different types of UV platforms can improve the portability of sub-systems that can be reused. An example of this application are payloads that can be potentially reused across different platforms without modification.

A key enabler of this vision is the introduction of reconfiguration mechanisms into IMA platforms.

2.0 THE DIMA PROJECT

A typical Distributed Integrated Modular Avionics (DIMA) platform is composed by several avionic components as Core Processing Modules (CPMs) and Remote Data Concentrators (RDCs), connected through an avionics data-bus. In the DIMA platform, RDCs are introduced to distribute I/O across the aircraft. RDCs also have local processing capabilities, handling the system's payloads and allowing their cryptographic verification.

In a distributed platform such as this, the reconfiguration needs to occur at several levels. Changing the payload attached to the avionics system results in the need to load/unload software executing in the avionics platform. Hence, one or more CPMs will require reconfiguration to adapt to the new payload configuration by loading/unloading the payload specific software. In addition to this module reconfiguration, the avionics network has to be considered as well; the network routes need to be modified to reflect the new distribution of applications throughout the distributed platform. Finally, changing a payload can impact aircraft or flight parameters. Thus, new parameters may need to be propagated to aircraft's system to reflect the change in payload.

The DIMA project has addressed this problem through the definition and development of an automatically reconfigurable distributed avionics platform, where the trigger for reconfiguration is a modification in the payloads attached to the avionics platform. The DIMA platform is responsible for detecting the payload attached to the vehicle, identifying the configuration to be loaded and executing the reconfiguration procedures throughout the aircraft. At run time, the platform is able to automatically load the correct configuration for the payload detected; the reconfiguration mechanism of the DIMA platform was designed to require no human interaction, besides the change in payload and the reconfiguration consent and acknowledgement.

3.0 ARCHITECTURAL SOLUTION

For the distributed system reconfiguration to take place safely and automatically, all involved actors need to be aware of the current state of the system, when the state transition will occur, and what the target state is. Once the reconfiguration has been triggered, the system must consolidate the new state to ensure that all involved actors have successfully transitioned to the new configuration; otherwise the system may be left in a hybrid state, potentially leading to catastrophic failure.

In order to drive the reconfiguration, the payloads firstly need to be detected. RDCs are the architectural elements that interact directly with the payloads and, therefore, they are responsible for their detection and validation. Security protocols are part of this service to ensure that only previously identified payloads may trigger a reconfiguration of the aircraft. Once the payload is detected and recognized, the system reconfiguration starts and the state consolidation described before is initiated.

This reconfiguration framework includes mechanisms to handle the reconfiguration at module, network and application level, containing also features to manage redundancy by detecting failures occurring at module and application level and activating redundant applications.

The reconfiguration used in this project is referenced as "multi-static"; all possible configurations are statically defined and the reconfiguration services are only responsible for changing between these pre-defined configurations when a given event requires it. A DIMA avionics configuration is composed by the configurations of each one of the elements forming the avionics platform (CPMs, RDCs, network), hence, every additional configuration adds a non-trivial amount of effort to develop and validate. In the long term, this effort can be alleviated with the usage of improved tools to generate the required low-level configurations.

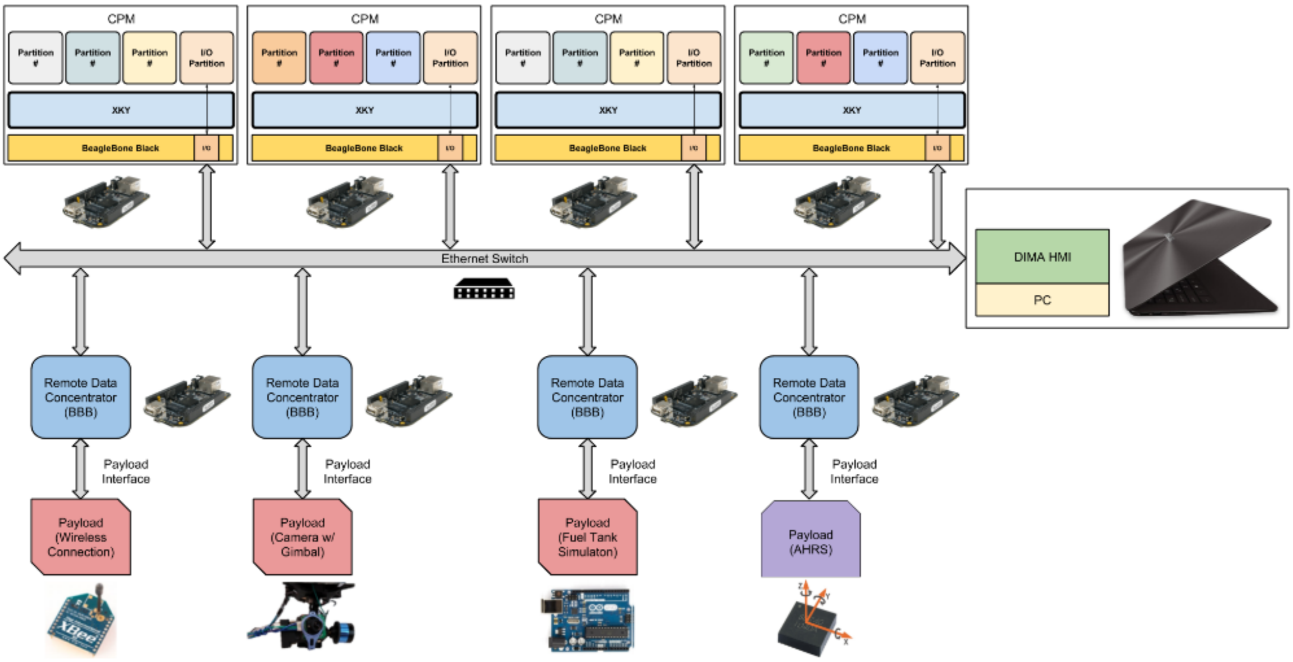

Figure 1-1: DIMA Platform.

Another key element in DIMA architecture was the usage of the XKY real-time operating system (RTOS) with a separation kernel based hypervisor. XKY is an ARINC 653 compatible partitioning RTOS that allows the execution of distinct Design Assurance Level (DAL) system applications on the same module while assuring robust time and space partitioning between them. The layered architecture provides a low level virtualization kernel on top of which several services can be implemented. XKY has carried out a central role in the scope of DIMA as the partitioning RTOS unit of the core processing modules, ensuring strictly separated partition applications that have implemented DIMA's functionalities.

4.0 CONCLUSIONS AND NEXT STEPS

The goals of the DIMA project were completely achieved: a Distributed Integrated Modular Avionics (DIMA) platform with reconfiguration capabilities, where the reconfiguration is triggered by changes in the payloads attached to the platform, was successfully developed and demonstrated using a cost-effective test bench.

To fulfil the goal of enabling payload driven reconfiguration, two main developments were undertaken. First, a unique interface, allowing the connection of payloads, was specified extending the military standard MIL-STD1760. This standard was complemented with a cryptographic handshake to permit unequivocal identification of payloads. Second, a reconfiguration framework was developed, which includes a distributed consensus protocol, providing safe avionics reconfiguration at several levels.

This reconfiguration solution was tested and demonstrated in the DIMA test bench. However, some constraints of the demonstration were also considered in the results' analysis. First, the application load in each Core Processing Module was essentially composed by the payload specific applications. These applications were built in the project to interface with the simple emulated payloads. This fact also simplified the process of generating module schedules, as the applications did not have substantial real-time requirements. Hence, the task of analyzing the behavior of the system under representative loads, with hard real-time requirements, was not part of the project's scope of work.

Despite the reconfiguration of time-predictable networks being a topic that was considered within the scope of the project, the implementation of the DIMA test bench only prototyped a non-deterministic Ethernet network. This simplification in terms of network choice has resulted in several issues being avoided and in a simplification of the configuration as no network configuration was required. Nonetheless, the impacts of the network on the reconfiguration need to be further explored as the distributed system and, therefore, the reconfiguration heavily depend on the network capabilities. The potential loss of communication during the network reconfiguration must be considered and encompassed in the reconfiguration protocol. In addition, for some network types it may not be possible to establish a bound for the network reconfiguration time. Finally, the issues resulting from data loading over deterministic networks must be reflected on the solution if the reconfiguration requires software to be loaded from non-local storage. As such, the prototyping of the DIMA reconfiguration framework over a deterministic network, addressing the aforementioned points, was left as future work.

Finally, it is considered that the process of integrating software over a DIMA platform is still burdensome and could benefit from a higher level of abstraction than that provided in typical ARINC 653 compliant systems. New methods and tools should be researched that ease this process, providing, namely, avionics functions with standardized and well specified interfaces to access common avionic on-board services and data.

As a closing remark, the success of the DIMA project fosters the introduction of IMA technologies in unmanned vehicles. By using an innovative reconfiguration process to optimally allocate the avionic system's resources, DIMA promotes scalability, reusability and modularity while reducing SWaP restrictions.