UNCLASSIFIED

Defense Technical Information Center Compilation Part Notice

ADP010970

TITLE: Transitioning to Integrated Modular Avionics with a Mission Management System

DISTRIBUTION: Approved for public release, distribution unlimited

This paper is part of the following report:

TITLE: Strategies to Mitigate Obsolescence in Defense Systems Using Commercial Components [Strategies visant a attenuer l'obsolescence des systemes par l'emploi de composants du commerce]

To order the complete compilation report, use: ADA394911

The component part is provided here to allow users access to individually authored sections of proceedings, annals, symposia, etc. However, the component should be considered within the context of the overall compilation report and not as a stand-alone technical report.

The following component part numbers comprise the compilation report:
ADP010960 thru ADP010986

UNCLASSIFIED

Transitioning to Integrated Modular Avionics with a Mission Management System

M. Gangkofer, Dr. H. Kader, Dr. W. Klöckner, C.G. White
ESG Elektroniksystem- und Logistik GmbH, Fixed Wing Aircraft - Advanced Avionics,
PO Box 80 05 69, 81605 Munich, Germany; e-mail: asaac@esg-gmbh.de

1 SUMMARY

This paper presents an incremental approach towards the adoption of an Integrated Modular Avionics (IMA) architecture, via the implementation of a Mission Management System using present-day Commercial Off-The Shelf (COTS) technology.

While standardised IMA modules are planned to be developed in the medium term, the approach presented enables the maximum benefits to be obtained from those aspects of the IMA concepts which are the most advanced, while exploiting the availability of today's COTS hardware. This approach is embodied in the Mission Management System, which is under development at ESG.

The Mission Management System is a computer-based system which is intended to host advanced mission management applications focussed on crew assistance functions, including mission planning and terrain-based display. The hardware of the Mission Management System comprises a single unit, the Mission Management Computer. The system, and in particular its software structure and system management functions, are based on the IMA concepts developed in the ASAAC (Allied Standardised Avionics Architecture Council) programme, which are applied here to the extent that they are compatible with the available COTS components. The Mission Management System implements those elements of the ASAAC IMA concepts that are most suitable for near-term adoption, and in particular those related to the software structure, which is based on the use of a COTS Real-Time Operating System.

In implementing the IMA concepts from ASAAC, the Mission Management System is designed around an open system architecture, using COTS hardware and software components. This approach provides technology transparency, and supports the substitution of system components, such as the replacement of system hardware or the Real-Time Operating System implementation, to mitigate the effects of hardware and software component obsolescence.

The paper first presents the approach adopted in transitioning towards the IMA architecture via the use of current-day COTS components. The Mission Management System is then described from the system architecture, software architecture and hardware architecture points of view, noting the implementation constraints of current COTS components. The system characteristics which are achieved through the adoption of the relevant IMA principles together with open systems and COTS practices are presented.

The mission management functions to be implemented on the system are defined, and an example is then presented of a complete avionics system built using the transitional technology of the Mission Management System in a number of Integrated Computers to provide a complete computing core.

Some certification issues are discussed, and the adoption of an incremental certification approach is recommended. A path forward towards the development of a true IMA system implementation is proposed, including further development of the Mission Management System, and the migration to a modular implementation.

2 INTRODUCTION

Integrated Modular Avionic concepts for military aircraft have been developed under a number of national [Ref. 1] and international projects, the latter including in particular the completed European EUCLID RTP 4.1 programme [Ref. 2], and the key ASAAC programme [Ref. 3, Ref. 4, Ref. 5].

The IMA concepts developed, and particularly those of the ASAAC programme, are based on the principles of modular systems, open systems and COTS. In an IMA architecture, the computing capacity is concentrated into a 'Core', which consists of interchangeable processing modules of a limited number of standardised types, particularly for data, signal and graphics processing. IMA systems provide a high level of technology transparency by being based on a set of open standardised interfaces, so facilitating the replacement of hardware components without affecting the application software. In addition, the use of open standardised interfaces directly supports the use of COTS components, which is of great benefit in combating the effects of component obsolescence. IMA systems also implement fault tolerance, so that when a module becomes defective, the system reconfigures and a spare module takes over the functionality of the failed module. The IMA concepts developed in the ASAAC programme have been adopted as the basis for the Mission Management System reported on in this paper.

Paper presented at the RTO SCI Symposium on "Strategies to Mitigate Obsolescence in Defense Systems Using Commercial Components", held in Budapest, Hungary, 23-25 October 2000, and published in RTO MP-072.

The development of the required set of ASAAC IMA hardware modules will be a substantial task, and its completion lies some way off in the future. Even given the availability of hardware modules, a very considerable certification effort will be required to qualify an ASAAC IMA-based system, particularly due to the system-wide configurability under software control it exhibits.

It is, however, possible to exploit the elements of the ASAAC IMA system, software and hardware concepts which will have been developed before such a stage is reached, while using the COTS-based board-level hardware components available today. Such advanced IMA concepts may be implemented in systems without expending significant additional effort in comparison with a more conventionally based solution. In this way a significant number of the benefits promised by the IMA architecture may be achieved today: a key example is the use of technology transparency to combat obsolescence.

A transitional step towards the implementation of the true IMA architecture is therefore now being taken with the development of a Mission Management System. In defining the Mission Management System, the aim has been to define a demonstrator which will allow the implementation of a representative set of application functions, using an architecture based on the IMA architecture of ASAAC. This will permit progress to be made towards the implementation and certification of an IMA-based avionics system, by gaining experience in the implementation of an ASAAC IMA-based software structure, and evaluating the consequences of hosting applications on such a structure.

The Mission Management System is designed to exploit:

- *The ASAAC IMA concepts:*
  Currently available elements of the ASAAC IMA concepts have been realised.
- *Open system architectures:*
  In accordance with the principle of offering an open architecture, open standards have been adopted. The use of an open architecture supports especially technology transparency, application portability and system scalability.
- *Available COTS hardware and software technology:*
  Extensive use is made in the Mission Management System of COTS components, both hardware and software. The use of COTS supports particularly the lowering of system costs and reduction of development time.

By the adoption of these concepts in the Mission Management System, it is aimed to achieve the following characteristics typical of IMA systems:

- *Application Portability:*
  The portability of application functions between hardware platforms is supported in turn by technology transparency.
- *Technology Transparency:*
  Technology transparency embraces hardware independence, which supports hardware replacement for future system upgrading and to counter component obsolescence, network independence, which enables the network technology to be upgraded, and also extends to the software technology.
- *Scalability:*
  Scalability supports the application to avionic systems of different sizes and roles, as well as future avionic system growth. The scalability characteristic is in turn supported by technology transparency, in particular by hardware and network independence.
- *System Reconfigurability:*
  By reconfiguring the system depending on the system mode, the total resource requirements of the system may be reduced. Reconfigurability may be used on the occurrence of faults to support fault tolerance.
- *Fault Tolerance:*
  Fault tolerance may be used to improve system reliability, and is supported in turn by system reconfigurability.

3 SYSTEM ARCHITECTURAL CONCEPT

3.1 Key Concepts

The architecture of the Mission Management System (MMS) is a transitional architecture between that of the current generation of federated architectures, and future IMA architectures. It is designed to be compatible with the avionics system architectures of both new-build aircraft designs and retrofit applications.

The architecture defined for the MMS utilises the elements of the ASAAC concepts which are the most advanced, but which are also compatible with conventional federated avionics systems. The aspects of the ASAAC concepts which have been adopted lie mainly in the Software Architecture and System Management areas, although some aspects of the hardware concepts have also been employed. The following key concepts have been applied:

- Use of a standardised interface between Application Software and the Operating System Layer.
- Hardware abstraction by software.
- Use of a system management structure and 'Blueprints' which together support system configuration and reconfiguration.

- Use of a standardised software interface for all data communication.

Due to the requirement to be able to integrate the MMS within a conventional federated avionics system, the aspects of the ASAAC IMA concepts which extend throughout the entire avionics system have necessarily found only partial application. These include for instance the system health and configuration management concepts.

3.2 Open Architecture

Open architectures are characterised by the use of widely accepted and supported standards set by recognised standards organisation or the commercial market place. As the standards are available to all, a system based on an open architecture is open to the incorporation of components from potentially any source. The IMA architecture defined in ASAAC is such an open architecture, as it both defines its own open standards, and supports the use of available open standards in its implementation.

An ASAAC IMA system may be regarded as comprising a set of application functions hosted on an open architecture platform. The applications are not dependent on the underlying technology and hardware, as the interface between the applications and the system functions is established as an open standard, so allowing different manufacturers of software and hardware components to contribute to the system.

A common example of an open architecture in the commercial world is the POSIX system [Ref. 6], which allows Unix systems and applications to be developed independently of the underlying hardware, regardless of the hardware manufacturer. While POSIX is unsuitable for an IMA System, which requires fully predictable real-time behaviour of its components, applications in ASAAC IMA systems are hosted on an open platform with an open interface, the Application to Operating System layer (APOS) interface (see Sec. 4.3).

3.3 Modes and Configurations

In the MMS, the various mission management functions are each only required to operate during the relevant phases of the aircraft's mission. For example, different mission planning functions are likely to be applicable to different mission phases, and display functions for high-altitude combat air patrol would differ from those for low-level target ingress.

In order to optimise the utilisation of the hardware resources in the MMS, and so reduce the total system resource requirements, the same hardware in the MMS will be used to host different application functions at different times, according to the requirements of the mission phase.

Because of the strict separation of application function design from the system hardware design, it is possible for the MMS application functions to be distributed over the hardware in a number of different ways, and hence for an application to be easily transferred from one processor to another.

At the transition from one mission phase to another the MMS will be reconfigured by halting and removing the relevant software components from the processors, and loading and starting new components from the Mass Memory Unit.

A set of MMS modes is defined to cover the various phases of the mission, where a specific set of functions runs in each mode. Within each mode, a number of different configurations of the functions on the various hardware resources may also be possible.

3.4 Fault Tolerance

ASAAC IMA systems offer fault tolerance by reconfiguring on the occurrence of faults. A new module may be substituted for a faulty module, and the application functions reconfigured accordingly. The fault tolerance concept of the MMS is derived from that of ASAAC IMA systems.

It is not possible to implement the same degree of redundancy in the MMS as in the ASAAC IMA concept, as the latter relies on full hardware redundancy between modules. The components of the MMS, on the other hand, are integrated with one another at a lower level: powering on and off individual hardware components is not supported within the MMS, for instance.

The MMS implements a limited degree of fault tolerance. Some redundancy is likely to be available between the multiple instances of the various hardware components, such as the Single Board Computers (SBCs) which are used within the MMS. Where such redundant capacity is available, when a component becomes defective, the system is reconfigured so that a spare component takes over the functionality of the failed one. Alternatively, where no extra resources are available, non-critical functions may be dropped, to free resources for higher priority functions, or reversionary implementations of functions, with lower resource requirements, may be used. In this way, a significantly greater fault tolerance capability is achieved than with conventional systems.

3.5 System Management

ASAAC IMA systems implement a standardised system management structure that is responsible for performing the following major functions:

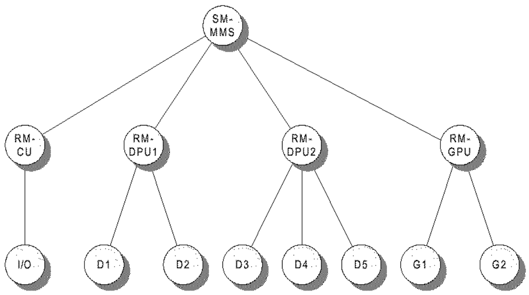
- Initialisation and shutdown management.
- Configuration management, including reconfiguration on mission mode transitions and on faults.
- Fault management, including health management such as the processing of Built-In Test (BIT) data.

For the Mission Management System, elements of the ASAAC IMA system management have been adopted as appropriate. Whereas the ASAAC system management concepts, such as for instance health management and configuration management, generally encompass the entire avionics system, it has only been possible to implement these within the scope of the MMS.

One of the prime responsibilities of system management is managing the control of the application functions, which includes their instantiation, start, suspension and removal from the system. System management is also responsible for configuring communications between applications, which is performed in a similar manner to the process scheduling, in order to guarantee predictable communications.

The system management functions manage the system in accordance with the blueprints. The blueprints provide the definition of the system resources, and define the possible configurations of the system. The configurations are deterministic, and are defined at design time: such strict system control will be necessary in order to certify the system. As well as the configurations themselves, the blueprints also define the reconfiguration processes which are carried out on the occurrence of faults.

System management in the ASAAC IMA concept is performed throughout the avionics system on a hierarchical basis. In the MMS, system management is performed at two levels. The top-level manager is the System Manager, and this controls the Resource Manager, which manages a particular single board computer hosting a number of application functions. Figure 1 shows the MMS system management hierarchy.



SM	System Manager	RM	Resource Manager
CU	Communications Unit	DPU	Data Processing Unit
GPU	Graphics Processing Unit	I/O	Input / Output
D1-Dn	Data Processing Processes	G1-Gn	Graphics Processing Processes

Figure 1: System Management Hierarchy

The MMS system management differs from that of the ASAAC concepts primarily as follows:

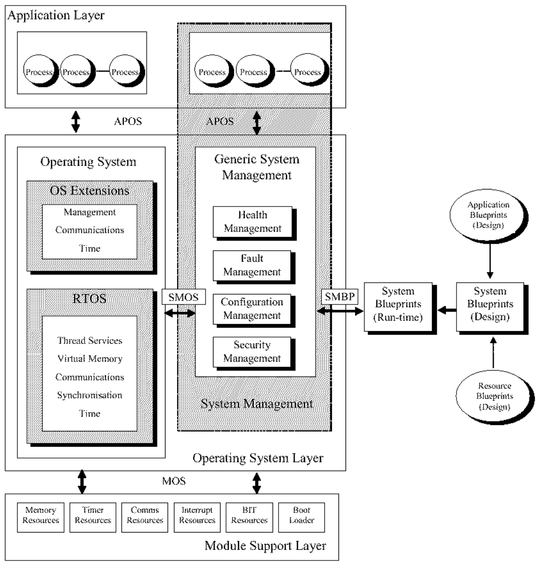
- The system manager hierarchy is limited to two levels.
- Fault detection is dependent on the capabilities of the COTS components.
- Only limited reconfiguration is supported, particularly for fault tolerance.
- The ASAAC System Management Blueprint (SMBP) interface between the GSM and the blueprint data is not implemented.

The system management functions are implemented by the software of the Generic System Management (GSM: see Sec. 4) together with the blueprint data, which are provided as part of the software load.

4 SOFTWARE ARCHITECTURE

4.1 The ASAAC Software Architecture

The software architecture of the Mission Management System is based on the ASAAC software architecture, modified to suit the COTS-based hardware architecture of the MMS.



APOS	Application to Operating System Layer Interface
BIT	Built-In Test
MOS	Module Support Layer to Operating System Interface
OS	Operating System
RTOS	Real-Time Operating System
SMBP	System Management Blueprint Interface
SMOS	System Management to Operating System Interface

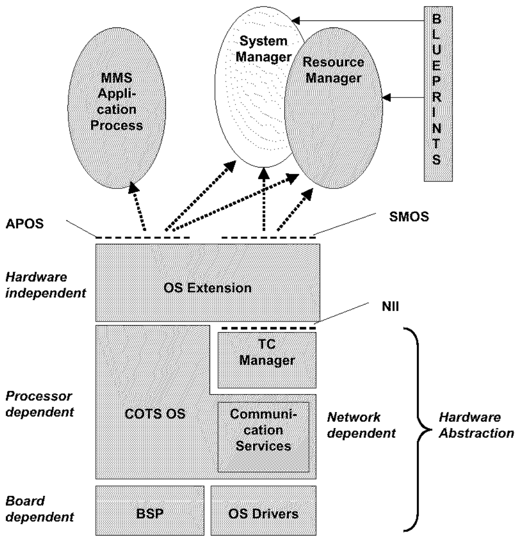
Figure 2: The ASAAC Software Architecture

The main components of the ASAAC software architecture are the Application Functions, the Operating System (OS), the Generic System Management (GSM) and the Runtime Blueprint representation (RTBP), as depicted in Figure 2. The GSM defines and manages the processing and communication resources required by the application, and the operating system provides the application with access to these resources. The blueprints provide the definition of the resources and of the possible system configurations. The GSM and the runtime blueprints are associated with both the system hardware and the application-independent operating system layer.

This architecture is based on the principle of a three-layer software stack, where the layers have the following properties:

- *Application Layer*
  Application Dependent, Hardware Independent
- *Operating System Layer*
  Application Independent, Hardware Independent
- *Module Support Layer*
  Application Independent, Hardware Dependent.

This architecture as implemented in the MMS is illustrated by Figure 3.



APOS Application to Operating System Interface
BSP Board Support Package
NII Network Independent Interface
OS Operating System
SMOS System Management to Operating System Interface
TC Transfer Connection

Figure 3: Design of the MMS Software Stack

In this design, the ASAAC software architecture has been adapted to the requirements and constraints of a computer architecture based on COTS VME hardware and a COTS real-time operating system:

- There is no Module Support Layer: hardware abstraction is achieved by the architecture of the real-time operating system.
- The GSM function is simplified into two kinds of management functions, a top level System Manager, which controls the overall computer, and a Resource Manager for every single board computer, which controls the board local resources.

4.2 Hardware Abstraction

A prime property of the ASAAC software architecture is the independence of the application software from the hardware, as the highest costs that arise in the replacement of obsolete hardware are incurred in the consequential adaptation of the application software.

In the ASAAC software architecture, a hardware abstraction layer is provided in the form of the Module Support Layer (MSL). This hardware abstraction provides the system with hardware independence: the higher software layers are independent from the hardware details, so that hardware changes do not affect either the operating system layer or the application layer software, so countering the effects of component obsolescence. In the Mission Management Computer, this architecture has been adapted to the architecture of a COTS real-time operating system. This adaptation provides the same hardware abstraction characteristics as specified for the Module Support Layer.

This hardware independence is provided at the level of source code compatibility, as binary compatibility would require a much higher degree of standardisation. Standardisation down to the level of binary compatibility has to be very detailed, and is therefore very restrictive, so limiting the openness of the architecture. Source code compatibility appears for this reason to be the appropriate choice.

In addition to the issue of source code compatibility, hardware modifications are likely to result in changes in the resource characteristics of the hardware, such as enhancements to the processing power or communication capacity, which would affect the system operation. This issue is addressed by separating the configuration data from the system management functions, and encapsulating it in the form of blueprints. A change of hardware or the porting of an application to another system would therefore only require the adaptation of the blueprints and the recompilation of the relevant software, including the operating system layer and application code.

4.3 The Operating System Layer

The Operating System Layer includes the Operating System itself, together with the system management components, ie. the system managers and the run-time representation of the system's blueprints.

One of the chief benefits from the use of the ASAAC software architecture for the Mission Management System is application portability, which supports the reuse of the application software. Application portability is provided primarily by the use of a standardised Application to Operating System (APOS) layer interface. As it is the only interface of the ASAAC software architecture visible to the application, the application is dependent only on this interface for the satisfaction of its processing and communication resource requirements. As the Mission Management System offers the same APOS as any ASAAC system, the mission management applications are therefore portable to other systems built using the ASAAC standards.

A COTS real-time operating system is used as the core of the operating system layer. In comparison with the alternative approach, which is the development of a bespoke operating system, the COTS approach exhibits greater flexibility and cost efficiency due to the commercial support provided for the adaptation to new hardware components. The use of a COTS operating system supports the efficient implementation of the operating system layer on the underlying COTS hardware, both for the MMS hardware, and for ASAAC IMA systems. A COTS Real-Time Operating System is therefore well suited to support the transition from the federated system architecture of the MMS to an IMA system, providing a stable Application Programming Interface together with off-the-shelf compatibility with COTS hardware.

The COTS operating system is complemented in the operating system layer by additional software which provides the adaptation to the ASAAC operating system interfaces, and in particular to the APOS.

4.4 Application Function Software Structure

The application process is the basic software element of an application function, and represents the configuration unit of a system. Each application process depends on processing and communication resources, namely on threads and virtual channels, and can only be executed when the system management function provides its required processing and communication resources.

In the Mission Management System, three different kinds of application processes have to be accommodated: data processing processes, graphics processes and mass memory -related processes. The last of these includes such processes as database applications, and is implemented in the form of file access functions. While data processing and mass memory applications may be freely located on any processor, graphics processing is restricted to the specific hardware capable of providing the OpenGL interface and functionality.

The mission management application is built from a number of small but simple processes, most of which contain only a single thread, and which are configured in accordance with the currently active mission mode. Each of these processes depends on both transient data and persistent information: the transient data is provided via the MMS interface, and the persistent information from a central database, which is represented by an application process that provides information on request.

4.5 Properties of the Software Architecture

The software architecture of the MMS offers a number of beneficial properties in comparison with conventional systems.

Application portability and technology transparency are both provided by the use of the APOS as a standardised interface between the application software and the operating system layer. This permits the reuse of the application software on other systems offering the ASAAC APOS interface, and also the replacement of the MMS hardware without modification of the application software source code. Performing the application design independently of the underlying hardware is also supported.

In addition, hardware abstraction below the level of the APOS provides protection against component obsolescence, by supporting hardware independence.

Technology transparency in the MMS also extends to software technologies: as the APOS is independent of the underlying COTS OS used, the COTS OS may also be replaced without affecting the application functions.

The GSM and blueprints support the ability to reconfigure the system in accordance with the mission requirements, so providing for efficient hardware use, and also supporting fault tolerance, which contributes to improved system reliability.

The open standards used include the open commercial standard OpenGL, and the ASAAC standards. COTS components used include the COTS operating system, together with its board support packages and drivers.

Due to the intention to maximise the use of COTS components in the development of ASAAC modules, the approach adopted for the Mission Management Computer of employing the ASAAC software stack with a COTS operating system is likely to remain effective throughout the continuing transition from today's federated system architecture to tomorrow's IMA architectures.

5 HARDWARE ARCHITECTURE

5.1 The Mission Management Computer

While current work is focussed on the software structure, using laboratory development hardware, the Mission Management Computer (MMC), which would form the hardware of the Mission Management System in a real aircraft application, and on which the application functions would run, has also been defined.

In an avionics system application, the Mission Management System would be integrated with and exchange data with other avionics system computers and peripheral devices, including sensors, effectors and displays and controls: a possible system implementation is shown in Sec. 7.

In order for the MMS to be able to perform the required application functions, the MMC must support the relevant low-level functions, which include data processing, graphic display processing, provision of mass memory, and communication. The MMC has been defined for hosting the MMS functions discussed in Sec. 6.

5.2 Structure and Packaging

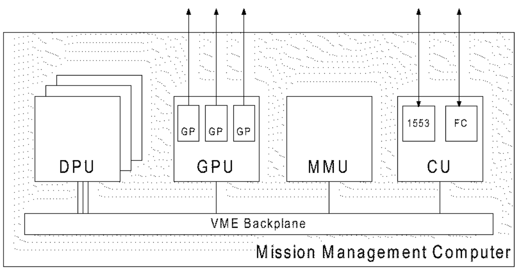
The packaging concept of the MMC differs considerably from that of IMA systems. Whereas in an IMA system, the line-replaceable items are the individual modules, the MMC is, in line with contemporary avionic systems, itself the line-replaceable item.

The basic principle behind the construction of the MMC is the exploitation of available COTS components, and in particular the use of VME boards. This enables the demonstrator to be built using standard commercial grade racks and boards, and a ruggedised version suitable for service use to be produced using components qualified to the appropriate environmental standards. In an aircraft application, the MMC would take the form of a standard ATR format unit, which would be mounted on an ATR rack in the avionics bay.

While the application of the IMA hardware concepts to the MMS is limited by the use of pure COTS hardware for the MMC, the hardware structure of the MMC has been influenced by the ASAAC concepts where possible, in order to support the ASAAC software and system architecture concepts employed, and to support the future development path towards an IMA system.

The structure of the MMC therefore conforms with the ASAAC concepts in the segregation of the data processing, graphics processing and mass-memory capabilities. Areas in which the structure of the MMC diverges from the ASAAC structure include the separation of the external communications interface from the processing capacity, and the lack of a Module Support Unit local to each of the processors. Power-Up and Continual Built-In Test are implemented to the degree that these are supported by the COTS hardware.

The structure of the MMC is shown in Figure 4 below, and discussed in the following text. The MMC is constructed using 64-bit VME boards: where required, for instance for communications, PMC modules are mounted on the VME cards.



DPU	Data Processing Units 1-3	GPU	Graphics Processing Unit
GP	Graphics Processor	MMU	Mass Memory Unit
CU	Communications Unit	1553	Mil Std 1553B
FC	Fibre Channel		

Figure 4: The Mission Management Computer

5.3 Hardware Components

The MMC consists of four main types of VME-bus component:

- Data Processing Unit
- Graphics Processing Unit
- Mass Memory Unit
- Communications Unit.

The data processing hardware comprises three Data Processing Units, each taking the form of a PowerPC Single Board Computer (SBC).

The graphic display processing implements three graphics processing channels, and so provides the capability of driving three displays with separate formats. The graphics processing hardware comprises a Graphics Processing Unit, consisting of three 3D graphics PMCs supporting OpenGL, mounted on a PowerPC SBC. Each graphics processor PMC provides the following key features:

- OpenGL implementation
- 1024 x 1024 resolution
- 1 M three-dimensional-polygons / sec.

The Mass Memory Unit is implemented as a solid state memory or hard disk interfaced with a PowerPC SBC.

The Communications Unit consists of the relevant network interfaces implemented as PMC modules mounted on a PowerPC SBC: in addition, analogue and discrete input / output is provided for as required.

5.4 Data Communications

All data communication within the MMS between processor boards and between the MMS and external equipment takes place via a standardised software network interface, termed the Network Independent Interface (NII), which has been derived as part of the ASAAC concepts. Data transfer takes place through the NII via a Transfer Connection (TC), which acts as a virtual channel, and which is established explicitly prior to the data transfer.

The use of the Network Independent Interface provides the ability to change the network technology used, for instance when reconfiguring the system and re-routing a particular transfer from an MMS-internal transfer to an external transfer. The network technology used for external transfers might also be upgraded as part of a future system improvement.

Communications within the MMC takes place using the buses available with the COTS boards, such as the VME and PCI buses.

The network technology used for external communication depends on the particular avionics system implementation into which the MMS is integrated: options range from the conventional Mil-Std 1553 command / response bus or ARINC 429 data distribution bus, to a high-capacity optical serial COTS Fibre Channel network.

5.5 Properties of the Hardware Architecture

The hardware architecture of the MMS offers a number of beneficial properties in comparison with conventional systems.

Technology transparency is supported by the Network Independent Interface, which provides for the replacement of the network technologies used while maintaining the software interface to the network, so allowing for data transfer growth.

Support is provided for scalability and system growth, particularly by the use of an open COTS-based architecture. The MMC is capable of being extended to cater for the addition of further MMS functionality, should this be required for a particular system application. Further data processing SBCs may be added to provide the required capacity, and the number of graphics processing PMCs may be chosen to feed the number of displays required. Hardware components of the computer may be replaced, providing that the availability of the relevant COTS operating system, board support package and drivers is assured.

Within the MMC, a limited degree of interchangeability between components could be offered by the use of a number of identical VME SBCs and PMC modules.

Open standards used in the MMC include open commercial standards as ATR, VME64, PCI, PMC and Fibre Channel, open military standards such as Mil-Std-1553B, and the ASAAC standards.

Use is made of Commercial (COTS) and Military Off-The-Shelf (MOTS) components as follows:

- SBCs, PMCs: COTS boards, using commercial chip-level components.
- Network: COTS, MOTS.

6 MMS FUNCTIONALITY

6.1 Generic Mission Management Functions

The possible system applications of a Mission Management System extend across a range of aircraft roles, from combat aircraft, including rotary wing types, to heavier patrol and transport aircraft. The specific functions implemented on the Mission Management System in a particular avionics system implementation would depend on the role of the aircraft and the allocation of functionality within the complete avionics system.

Typical mission management functions include the following:

- Situation Assessment
- Conflict Management
- Mission Planning
- Man-Machine Interface
- Navigation
- Flight Guidance.

The functional structure of a Mission Management System is shown in Figure 5 below.

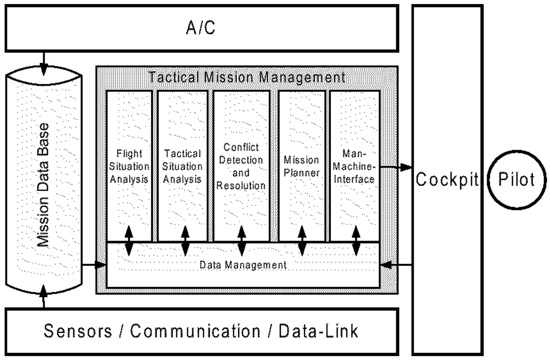


Figure 5: Functional Components of a Mission Management System

6.2 Functions Selected

For the Mission Management System under development, a representative set of functions which is broadly aimed at a multi-role fighter application has been chosen.

The primary functions selected are mission planning, representing a high-performance data processing application, and the production and presentation of perspective flight guidance information, a three-dimensional graphic application.

These functions are based on those under development in parallel activities being performed at ESG, as previously reported [Ref. 7], and are now to be ported to the Mission Management Computer from their original implementations on a laboratory workstation-based environment.

To provide the required functionality, the Mission Management System will comprise the following functional components:

- *Databases:*
  - Terrain Database
  - Navigation Database
- *Functional components:*
  - Threat Analysis
  - Mission Planner, including:
    - Transit Planner
    - Low-Level Flight Planner
    - Attack Planner
- *Graphics processing:*
  - Head-Up Display with Terrain Graphics
  - Head-Down Display with Synthetic Vision
  - Tactical Navigation Display.

External inputs received by the MMS from other avionics system components include:

- Aircraft Data (eg. Position, Velocity, etc.)
- Sensor Data (eg. Threat warnings, Datalink).

Details of the functions have been given previously in [Ref. 7].

The functional structure of the system has been designed to accommodate the addition of further functional components as and when required, depending on the requirements of the anticipated application avionics system.

7 AVIONICS SYSTEM ARCHITECTURE

7.1 Architectural Concepts

This section addresses the system architecture of an avionics system into which the Mission Management System might be integrated. The system example described here is a near-term new system design, again based on a multi-role fighter application.

The proposed system architecture is based on the use of Integrated Computers, of which the Mission Management Computer is an example, each based on the same system, software and hardware architectures as described for the MMS above.

The avionics system structure remains essentially a federated architecture. As in conventional federated systems, the overall avionics system is divided into a number of dedicated systems / sub-systems, eg. Mission Management System, Stores Management System. The overall system consists of a number of Integrated Computers, together with essentially the same dedicated equipment found in conventional federated systems, eg. Radar, Radio, Displays and Controls.

The computing capacity of the avionics system is distributed between the Integrated Computers, which are interconnected via the communication network. In comparison with other federated architectures, the Integrated Computer-based architecture is characterised by the centralisation of the processing capacity in a smaller number of higher-capacity processors.

Most of the processing of the sensor data, including the signal processing, takes place in the sensor equipment itself, whereas the subsequent system-level data processing of the sensor data is performed in the Integrated Computers. The degree to which the sensor data processing is implemented in the Integrated Computers depends on the capability provided by the particular sensors themselves.

System management, embracing moding, configuration management and fault tolerance, would initially remain implemented largely at the level of the Integrated Computers, rather than being integrated at an aircraft level, as with an ASAAC IMA system. However, for critical functions additional reversionary implementations with reduced capabilities could be hosted on alternative computers.

7.2 System Structure

Four Integrated Computers form the core of the avionics system, connected with the peripheral equipment, including sensors and effectors, and displays and controls, and also the safety-critical Stores Management System.

The Integrated Computers perform the following roles:

- Interface and Monitoring Processor
- Mission Management System
- Communications Processor
- Defence, Attack and Armament Computer.

The structure of the avionics system is shown in Figure 6 below. The equipment shown in grey is external to the avionics system.

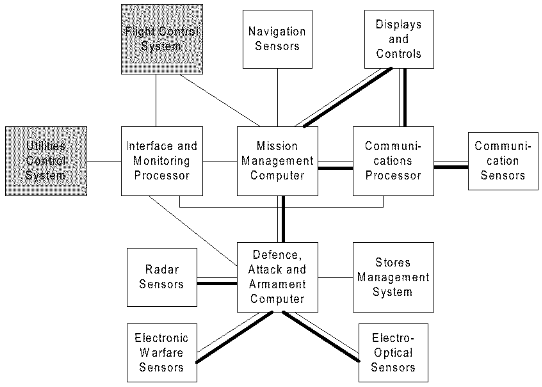


Figure 6: Example Avionics System Structure

Due to the high level of integration within the Integrated Computers, the data transfer loading between the individual equipment remains relatively moderate. Data transfer between equipment takes place either as in a conventional system via a Command / Response or Data Distribution Bus (eg. Mil-Std 1553 or ARINC 429), or by the use of a supplementary Fibre Channel overlay network, which is indicated by the broad lines in Figure 6. The latter provides for higher data rates between particular equipment, where these are required, for instance between the integrated computers, and for video data.

Interchangeability between the various Integrated Computers could be provided by the use of multiple instances of the same computer design. This should in principle be possible, due to the similarity of the requirements for the various system computers.

8 THE PATH FORWARD TOWARDS IMA

8.1 Further Development of the MMS

The Mission Management System defined in this paper represents a significant step towards implementing an ASAAC IMA architecture, a goal which is to be achieved with the introduction of the IMA hardware modules. There would, however, be a number of potential advantages in further developing the MMS concept by adding additional IMA features before progressing to a definitive IMA system:

- Multiple instances of Integrated Computers similar to the Mission Management System could be integrated in an avionics system, as shown in Sec. 7. The implementation of the ASAAC system management concept could be extended incrementally to encompass system-level management, so that such functions as initialisation and reconfiguration would be performed on a system-wide basis.
- A form of design-time blueprints could be implemented, containing resource requirements and specifications, to provide system design support.
- When available, the ASAAC network technology could be introduced, to provide the required high capacity, real-time deterministic behaviour, and network redundancy.
- Signal processing capability might also be added to the MMC, following the IMA strategy of centralising processing in the core.

8.2 Migration to an IMA Implementation

The eventual implementation of a system with ASAAC IMA modules will result in fundamental efficiency advantages for system development and operation, due to the use of a standardised, interchangeable hardware set.

The example system implementation presented in Sec. 7, based on the use of four Integrated Computers, represents an architecture which could be migrated to a full ASAAC IMA architecture, by the replacement of each of the Integrated Computers by an IMA Integration Area.

The use of individual modules will improve the hardware efficiency of reliable system designs, by permitting system reconfigurability at the module level. This represents a significant improvement, when compared with non-modular equipment such as the MMC, which is integrated as a complete unit, and which is therefore potentially susceptible to single point failures.

8.3 IMA Certification Considerations

Systems such as the Mission Management Computer that are based on IMA principles introduce a number of new factors, including in particular their reconfigurability, which are likely to affect their certification.

With traditional systems, certification has been achieved for the complete system, including both its hardware and its software. Some systems have implemented a degree of reconfiguration, but with a relatively restricted number of configuration cases, all of which have usually been designed into the system together with the hardware and application software, with the system being certified as a whole.

In contrast, in the ASAAC IMA system, and in the MMS, the hardware, operating system layer software and application functions are to be developed separately, with well-defined interfaces, and configurations determined for the integrated system.

The system configurations of the MMS are deterministic, in that all possible configurations are determined at design time, and stored in the blueprints. The total number of system configurations may however be very high, as the overall system configuration is made up of all the individual processor configurations, and as the processors are configurable at the level of individual processes.

While it will be necessary to certify each system configuration, it will not be practical to verify in detail the complete correct operation of an integrated ASAAC IMA system in all possible modes. This leads to the proposal to adopt a new approach and perform certification incrementally using a component-based method, as discussed below.

8.4 Incremental Certification

The basic principle of the incremental certification approach is to be able to modify a certified system, and to achieve certification for the modified system by certifying only the changes, without having to repeat the full certification process anew on the modified system in its entirety.

In order to be able to perform incremental certification, it is necessary to constrain, and to be able to identify, the effects of the modifications on the original certified system. In this way, the certification process for the modified system may be concentrated on the changes introduced and their resulting consequences.

Incremental certification can be applied to the development of completely new systems, by performing certification of the system at various stages in its development, as well as to the modification of in-service systems: the incremental certification approach is particularly applicable to the addition of application functions to an existing system.

The principle of incremental certification may be further developed to include component-based certification. Here, the basic principle is that each component is certified in its own right, so that when a system is assembled out of a number of components, it is then just the integration of the components which needs to be certified.

8.5 MMS Incremental Certification

It is proposed to adopt a component-based incremental certification approach for the Mission Management System. The main characteristics of the Mission Management System which support incremental certification are application portability together with the related technology transparency. These characteristics are derived primarily from the use of the APOS, the Application to Operating System layer interface. Due to the definition and standardisation of this interface, the Application

Functions are decoupled from the underlying operating system, hardware and drivers, which in turn enables the certification of the components either side of the APOS to be decoupled.

The adoption of incremental certification will require the development of an appropriate certification process by the certification authorities, and it is proposed that the MMS be used as a vehicle for development work on such a process.

When applying the principles of component-based incremental certification to the development of the MMS, there are a number of steps which may be taken to ease its certification.

Firstly, the application functions, in particular for the first MMS implementation, should be of a low criticality. In view of this, those that have been selected for the MMS are non- safety-critical, and do not require fail-safe implementation due to reliability considerations.

Further, due to the concerns regarding the certification of reconfigurable systems discussed above, it might be advisable to limit the scope of the reconfiguration mechanisms implemented in the MMS, in order to ease the first certification. One potential measure would be to limit the configurations to a small number, so that each reconfiguration step could be examined in detail. A further measure would be to exclude all fault-triggered reconfiguration, leaving only reconfiguration on mission mode changes. Once initial certification was obtained, the scope of the reconfiguration could be successively extended.

9 CONCLUSIONS

The Mission Management System described in this paper and being prototyped at ESG has been seen to feature an effective IMA-derived architecture, and to offer a representative set of mission management functions.

Through the adoption of the IMA principles from the ASAAC programme, and their implementation using COTS components on the basis of an open system architecture, the Mission Management System is able to offer the following key characteristics:

- Application portability is achieved by the application functions' use of the APOS interface.
- Hardware independence is provided by hardware abstraction, and provides protection against component obsolescence.
- Reduced development time and lower costs are supported by the use of COTS components and methods.

A number of further valuable properties are also realised:

- System reconfigurability is provided by the system management function and blueprint data, and supports the optimisation of the use of the hardware resources.
- Fault tolerance is achieved by means of system reconfigurability, and improves system reliability.
- System growth and the ability to apply the system to aircraft for a wide range of roles are supported by the use of an open system architecture.
- Network independence is provided by the use of the NII, and permits the upgrading of the network technology.
- Software technology transparency is supported by the standardisation of the APOS interface to the operating system at a level above the underlying COTS operating system.

The development of the Mission Management System as a transitional architecture implementing mission management functions should achieve a major step towards the implementation of IMA systems, and provide valuable experience for their consequent implementation and certification. In accordance with the proposed incremental certification approach, the Mission Management System is open to progressive development to incorporate further aspects of the ASAAC IMA concepts, so supporting the eventual migration to a true IMA system.

In conclusion, it is hoped that development of systems based on transitional architectures, and particularly the Mission Management System presented in this paper, will significantly ease the introduction of Integrated Modular Avionic systems.

10 REFERENCES

Ref. 1 Integrated Modular Avionics Architecture Concepts Demonstration, BDir J. Potthaus, Dr. W. Klöckner, I. Sprang, C.G. White, NATO AGARD Mission Systems Panel Symposium, Istanbul, Oct. 1996.

Ref. 2 Modular Avionic System Architecture Definition in the EUCLID R&T Programme 4.1, A. Marchetto, NATO AGARD Mission Systems Panel Symposium, Istanbul, Oct. 1996.

Ref. 3 ASAAC Phase II Programme Progress on the Definition of Standards for the Core Processing Architecture of Future Military Aircraft, G. Multedo, D. Jibb, Dr. G. Angel, ERA Avionics Conference, London, Nov. 1998.

Ref. 4 Modular Avionics: The future of Great Weapon Systems, W. Brühmann, D. Shealey, NATO RTA System Concepts and Integration Panel Symposium, Florence, Sep. 1999.

Ref. 5 Avionics Architecture Standards as an Approach to Obsolescence Management, D.J. Jibb, J.B. Walker, NATO RTA System Concepts and Integration Panel Symposium, Budapest, Oct. 2000.

Ref. 6 POSIX Standard, IEEE Std. 1003.

Ref. 7 Cognitive Concepts for Tactical Mission Management, Dr. A. Schulte, Dr. P. Stütz, Dr. W. Klöckner, NATO RTA System Concepts and Integration Panel Symposium, Florence, Sep. 1999.

11 ABBREVIATIONS

A/C	Aircraft
APOS	Application to Operating System Interface Layer
ARINC	Aeronautical Radio Inc.
ASAAC	Allied Standard Avionics Architecture Council
ATR	Air Transport Racking
BIT	Built-In Test
BSP	Board Support Package
COTS	Commercial Off-The Shelf
CU	Communications Unit
DPU	Data Processing Unit
EUCLID	European Co-operation for the Long Term in Defence
FC	Fibre Channel
GP	Graphics Processor
GPU	Graphics Processing Unit
GSM	Generic System Management
I/O	Input / Output
IMA	Integrated Modular Avionics
Mil Std	Military Standard (US)
MMC	Mission Management Computer
MMS	Mission Management System
MMU	Mass Memory Unit
MOS	MSL to Operating System Interface Layer
MOTS	Military Off-The- Shelf
MSL	Module Support Layer
NII	Network Independent Interface
OpenGL	Open Graphics Language
OS	Operating System
OSL	Operating System Layer
PCI	Peripheral Component Interconnect
PMC	PCI Mezzanine Card
POSIX	Portable Operating System Interface
RM	Resource Manager
RTBP	Run-Time Blueprints
RTOS	Real-Time Operating System
RTP	Research and Technology Programme
SBC	Single Board Computer
SM	System Manager
SMBP	System Management Blueprint Interface
SMOS	System Management to Operating System Interface Layer
TC	Transfer Connection
VME	Versa Module Europe