ABSTRACT

This paper focuses on the concepts and technologies required to develop, integrate, and test the Pave Pillar architecture in an avionics system. An in-house research team was formed at the Avionics Directorate to demonstrate and evaluate architectural and Ada software concepts that relate to modular avionics. The architectural concepts related to line replaceable modules are as defined by the Architectural Specification for Pave Pillar Avionics dated 16 January 1987. The hardware and software issues relating to multi-processing, multi-tasking, and real-time reconfiguration are discussed. The integration issues of developing and integrating the VHSIC Avionic Modular Processors (VAMPS), high speed data bus networks, and Ada software are examined. An avionic hot bench simulation was integrated to provide a closed-loop real-time test set-up called the Integrated Test Bed (ITB) Facility. The configuration and test setup for the common avionics modules were selected to provide a realistic environment and to be as close to the defined Pave Pillar architecture as possible. There are four clusters with each cluster consisting of two High Speed Data Bus Modules, two to four Mil-Standard 1750A CPU Modules, and one Mil-Standard 1553B Bus Module. The results from Demonstration III proves the concepts of common modules, reconfiguration, and modular avionics while quantifying the integration issues.

INTRODUCTION

The Pave Pillar Program defined and established a baseline avionics architecture for use in tactical and strategic aircraft using revolutionary new technologies and documented the architectural concepts relating to line replaceable and common avionic modules in the Architecture Specification for Pave Pillar Avionics dated 16 January 1987. On 16 March 1988, a Pave Pillar In-house Project Team was formed at the Avionics Directorate to evaluate the Pave Pillar architecture and concepts. The goal of the Project was to evaluate the Pave Pillar architecture in the purest configuration possible with no deviations from the original specification. Specifically, the Project's objectives were to evaluate and demonstrate the architectural concepts related to modular avionics, and to explore the hardware and software issues relating to distributed multi-processing, multi-tasking, and real-time reconfiguration. This paper discusses the development, integration, testing, and demonstration of the modular avionic components, and the integrated hot bench required for testing and demonstration. The Pave Pillar In-house Team defined three major milestone demonstrations which would evolve the avionic system configuration under test in a step-by-step integration process with the final full-up architecture being contained in Demonstration III. In addition, this paper also discusses the results and conclusions derived from the three years of research and testing on the Pave Pillar components and Ada software.

PAVE PILLAR IN-HOUSE DEMONSTRATIONS

The Pave Pillar In-house Team elected to use the Integrated Test Bed (ITB) Facility located in Building 620 at Wright Laboratory for its demonstrations because of its flexibility and ability to support real-time man-in-the-loop testing. The proposed plan was to use the existing configuration of controls and displays, Harris host simulation computer complex, and MicroVAX and VAX 11/785 support computers as a baseline system; and integrate the new PAVE PILLAR related technologies, algorithms, and avionic equipment in an incremental scheme to accomplish the three planned demonstrations. The Project Team went through a design, development, integration, and test sequence for each major milestone to assure that formal design review requirements would be met with the appropriate documentation produced

U.S. Government work not protected by U.S. copyright.
for each milestone configuration. Three ITB configurations were established to meet the requirements defined for each milestone demonstration. The following paragraphs describe the top level configuration for each major milestone demonstration.

Demonstration I milestone was basically a one year effort to set up the laboratory for a low altitude mission scenario, stand alone test the VAMP clusters, and integrate and test the Integrated Terrain Access/Retrieval System (ITARS) in the ITB. The low altitude mission scenario was selected since all the prime contractors during the Pave Pillar Definition Studies concluded that this mission scenario places the greatest stress on the avionic system especially during the low altitude terrain following, terrain avoidance, and threat avoidance modes of operation. The ITARS unit is a flyable prototype of a digital map system with a high speed capability to access and retrieve large blocks of DMA terrain, DMA cultural, threat, and other types of data used by advanced avionic systems. The Demonstration I milestone was accomplished on 31 July 1989.

Demonstration II was the first attempt at inserting the Pave Pillar architecture into the ITB configuration. This milestone focused on the insertion of technologies such as the VHSIC Avionic Modular Processors (VAMPs), High Speed Data Buses (HSDBs), Ada Avionics Real-time System (AARTS), and Ada application software modules developed around algorithms necessary for ITARS and low altitude penetration mission operation. The Demonstration II milestone was accomplished on 10 December 1990.

Demonstration III was the final milestone and included the complete integration of the selected Pave Pillar technologies including the insertion of the two high speed data buses, the four VAMP clusters, the System Mass Memory, and the Ada operational fight programs. The demonstration focused on the experimental conditions which would allow real-time reconfiguration of the processor modules, and examination of the protocol of the high speed data buses during information transfer. The low altitude penetration mission scenario was extended to include conditions that stress information transfer, showcase the use of ITARS, and research new algorithms coded in Ada application software and executing in the VAMP clusters. Demonstration III was accomplished on 18 October 1991 and marks the completion of the engineering effort except for the final analysis and finishing the final research report on the project.
contains identically or similarly configured Mission Data Processors all connected to the Mission Avionics Bus and loadable from the System Mass Memory via the Block Transfer Bus. All mission processing and data exchange operations are performed using the Mission Avionics Bus. The Mission Management System collects the health and status of all core elements and sensor/subsystem components for maintenance history, and logs the maintenance data via the Test and Maintenance Bus (TM-Bus) for later retrieval by maintenance or test personnel.

COMMON MODULES

The Pave Pillar architecture is physically configured in a building-block approach with common modules and some specially designed line replaceable modules. The Avionics Directorate undertook the design and development of a VHSIC Avionic Modular Processor (VAMP) capable of performing the Mission Data Processor function and meets the MIL-STD-1553B multiplex bus, TM-Bus, and high speed data bus interface requirements. The VAMP consists of MIL-STD-1750A Processor Modules arranged within clusters with dual PI-Bus and single TM-Bus backplane interconnects between the modules within a cluster. Each VAMP cluster usually contains two High Speed Data Bus (HSDB) Interface Modules, one MIL-STD-1553B Interface Module, two VHSIC MIL-STD-1750A Processor Modules, one Non-volatile Memory Module, one Clock Terminator Monitor Module, and Four Power Supply Modules. The VAMP has eleven useable slots with eight slots wired for VHSIC MIL-STD-1750A Processor Module usage if desired. This allows the user to reconfigure the module types into clusters to meet specific computational throughput and external interface requirements. The VAMP modules along with a brief description of the PI-Bus and TM-Bus are described in the following paragraphs.

The VHSIC MIL-STD-1750A Processor Module is a central processing unit (CPU) capable of executing two million instructions per second. The CPU can address via extended memory addressing 256K words of volatile local memory. There is 8K of EEPROM to function as non-volatile memory for start-up instructions and bootstrap. An input/output bus controller provides intelligent interface control to the PI-Bus backplane for inter module communication.

The HSDB Interface Modules link the PI-Bus to a pair of dual redundant high speed fiber optic buses. The HSDB Interface Modules consists of three major components: the fiber optic transceivers and protocol logic, a MIL-STD-1750A CPU with 256K words of memory, and dual PI-Bus interface. The modules have the capability to interface a fiber optic bus organized as two redundant channels. Each channel consists of a transmit fiber and a receive fiber. The HSDB Interface Module is capable of performing the Block Transfer Bus or Mission Avionics Bus functions in the avionic system.

The MIL-STD-1553B Bus Interface Module (BIM) links the PI-Bus to a dual redundant MIL-STD-1553B bus. The MIL-STD-1553B BIM is divided into three functional components: the MIL-STD-1553B multiplex terminal units, a MIL-STD-1750A CPU with 128K words of memory, and a dual redundant PI-Bus interface. The module interfaces with MIL-STD-1553B buses organized as redundant pairs. The module converts 1Mb/s serial data on the MIL-STD-1553B bus, performs all Manchester encoding or decoding, and provides all MIL-STD-1553B sync codes, parity, time out and other special decode functions. The transmission and reception of messages are determined by bus lists. The bus lists are statically built and loaded into the 1553B BIM control software. The 1553 Interface Module transmits messages synchronously and forwards received messages onto the PI-Bus.

The Non-Volatile Memory Module (NVMM) has a capacity of 8 million 16-bit words of read/write non-volatile memory used primarily for program store for fast load and reload of processors inside a cluster. Memory control is under a VHSIC 1750A controller. The memory access time is 250 nanoseconds with write time of 10 milliseconds per 128 word page.

The PI-Bus is a backplane bus that supports communications between all of the modules in a cluster. The communications control access protocol is based upon a message by message contention method. When a module has control of the PI-Bus, it may send or receive a single message. The messages may be directed to a specific physical module, broadcast to all modules, or transmitted to a set of all modules identified by a multicast address. The specific message, being transmitted or received, is identified by a label which is issued by all modules to uniquely identify the specific message.

The TM-Bus is a serial backplane interconnection that is used primarily for test and maintenance. Using this bus, one of the modules in the cluster can interrogate, restart, or halt any of the other modules within the same cluster. In addition, a special Time Stress Measurement Module can be added to the VAMP cluster to collect temperature, humidity, vibration and corrosion data and log it via the TM-Bus.
The High Speed Data Bus (HSDB) is the most important element of the Pave Pillar architecture. The capability of transferring large amounts of data throughout the system is critical to achieving the objectives of fault-tolerance, reconfiguration and resource sharing. The HSDB is a 50 Megabit per second (Mb/s) fiber optic network employing a token-passing access protocol to provide a highly-efficient, and robust communication medium with a distributed control philosophy. Unlike its predecessor, MIL-STD-1553, the HSDB uses a source-oriented data distribution method rather than the central controller technique of 1553. This means that each terminal or station is responsible for gaining access to the bus, transmitting its data and then passing the "token" to the next terminal in a logical ring. In the Pave Pillar avionics architecture, there are at least two distinct application areas for the HSDB.

The Mission Avionics Bus is primary medium for communication among all avionics functional elements. It connects the Mission Data Processors, Vehicle Management Data Processors, Signal Processors and other mission elements as the primary data and control highway for the system. In the Pave Pillar concept, it is the control path of the very high speed sensor, Video and Data Exchange networks as well. This application is specified as dual-redundant. In the Pave Pillar In-House Demonstrations, the mission HSDB connect the VAMP clusters to the Integrated Terrain Access/Retrieval System (ITARS), the Display Generation System (DGS), the cockpit, and the Harris real-time simulation system.

The Block Transfer Bus is an application area that makes use of the low-latency and long message block possible with the HSDB to quickly load/reload the data processors with application tasks from the System Mass Memory. In addition, large application data files can be transferred between processors or to/from System Mass Memory over the Block Transfer Bus. This application is specified as dual-redundant. In the Pave Pillar In-House Demonstrations, the mission HSDB connect the VAMP clusters to the System Mass Memory hosted on an IRONICS 68020 host computer and interfaced via a HSDB Avionic Bus Interface (ABI) card plugged into the 68020 VME backplane.

The Fiber Optic Active Star Coupler (FOASC) is a device that replaces a passive star coupler in a high speed data bus system. The passive star coupler receives optical signals on any of several input "ports" and sends the optical signal to all of the output ports. Because the HSDB protocol dictates that only one station transmit at a time, the signal goes into the star and is then broadcast to all stations (including the transmitter). A passive transmissive star coupler is made by taking "n" fibers (for "n" ports), twisting them together, and heating until a fused glass region is created. Light entering this region on one fiber is subdivided and goes out on all fibers. Since the amount of optical power that can be launched into a fiber by practical devices is rather low, the optical power "budget" of a cabling system is critical. The passive coupler will result in a division by "n" of the power, plus an excess loss. Optical inline connectors also contribute a significant loss, so a HSDB system with a large number of stations and/or a lot of connectors may have little or no power margin at the optical receiver. A FOASC boosts the optical signal to ensure that sufficient power is available at the receiver for a low bit-error rate. The FOASC was developed to prove the feasibility of using an active device to improve power margins in HSDB systems that need it.

ADA SOFTWARE AND RECONFIGURATION

All Mission Software for the Pave Pillar In-House was written in Ada and executed on the VAMPs. The VAMP Mission Software consists of an Ada Operating System and application modules operating in a real-time multi-tasking and multiple processor cluster configuration. The Mission Software performs the following functions: start-up control, application scheduling and execution, reconfiguration management, restart control, communications control, data and resource control, application mission functions, and shutdown. The Ada Operating System consists of a three part executive: the System Executive, the Kernel Executive, and the Distributed Executive. Figure 2 illustrates the Mission Software configuration and interfaces.

Figure 2: Mission Software
The System Executive resides in the Mission Data Processors. There are two copies, one of which is designated the primary (System Supervisor), and the second is designated the standby. The standby system operates in a "hot backup" mode, available to take over whenever a fault occurs in the primary. The System Executive is responsible for monitoring the state of the system hardware and software. The System Executive is responsible for keeping a fault log for all system components and for controlling reconfiguration of the system resources. The System Executive handles two types of system resource reallocation. First, as functional requirements change throughout the mission, the system components are relocated to meet these requirements. Second, when any system component currently in use is determined to have failed, reallocation of the remaining components may be necessary in order to retain any functionality lost by the failure. The reallocation of the components is called reconfiguration. With the Pave Pillar architecture, it is now possible to reallocate at the Loadable Program Unit (LPU) level of software to reload software at the module, processor, cluster, or across cluster level of reconfiguration of the avionic system. In the purest implementation of the architecture, the within-cluster reconfiguration can be handled by loading from the non-volatile memory module while the across-cluster case is loaded from the System Mass Memory. The reconfiguration strategy is based on a functional priority scheme and the actual reconfiguration is accomplished by a tree search of defined LPU's based on priority of the function and size of the software unit.

The Kernel Executive is responsible for controlling the application software assigned to a processor, providing control and communication between application tasks, controlling peripheral devices, and participating in processor level fault tolerant operations. Control of the application software is provided by a task sequencer which handles the invocation and suspension of tasks within the processor. Intertask control and communication services is provided to support the Ada (MIL-STD-1815A) tasking model which includes the concept of a rendezvous between tasks, and allowing intertask control and communications. The peripheral device control capability of the Kernel Executive permits an application task to request an operation on a peripheral device at a logical level. Processor level fault tolerant operations include the receipt of processor error indicators, analysis of the conditions, and determination of the appropriate actions. The appropriate action may be to: pass the condition to an application task for further action by an exception handler, notify the system executive, disable the processor, or any combination of these actions.

The Distributed Executive provides bus control interface capability, inter-processor data transfer operations, and general control over processor participation in the overall system operation. The Distributed Executive communication interface provides for the exchange of information between processors. Data transfer requirements will result from transactional requirements in support of application tasks and from system status monitoring and reconfiguration operations. System software and hardware configuration information is maintained by every processor in the system. The Distributed Executive is responsible for the maintenance of the system state information requirements and system operation. A Distributed Executive in one processor can interface with the Distributed Executive in other processors via the Mission Avionics Bus, and with the Kernel Executive resident within its own processor.

The Mission Application Software is built around a low altitude penetration mission and performs the functions typically performed by onboard computers. These functions include navigation, position update, guidance and steering, weapons delivery, stores management, sensor interfaces, controls and displays management, terrain following, terrain avoidance, threat avoidance, enroute replanning, and other miscellaneous functions. The Mission Software functions were integrated in such a way as to evolve the Pave Pillar experiments and demonstrations from a single VAMP cluster configuration to the full-up four VAMP cluster configuration.

DEMONSTRATED AND EVALUATION SETUP

The Integrated Test Bed Facility, located in Building 620 at the Avionics Directorate, was used to support the development, test and evaluation of the Pave Pillar avionic systems and subsystems. After installing the avionic components, the ITB provided a real-time simulation of a military aircraft performing an operational mission in the low altitude regime. While running the simulation, it generated the interface signals between the aircraft sensor suite and the avionic system such that the avionic components or equipment are subject to a data signal environment as close to the actual flight environment as possible. A simulated generic cockpit, approximately the size of an F-15 cockpit, was included as part of the simulation for
realistic evaluation of the avionic system. The aircraft model selected to provide the best aerodynamics for the low altitude penetration mission was the F-15E, and all developed avionics algorithms were fine tuned for this selection. The ITB also utilized a complex of commercial digital computers, including three Harris 800s and a number of MicroVAX series computers. Simulation software resident in the Harris 800 computers was developed to provide a real-time simulation of the F-15E aircraft in an operational environment, including aircraft dynamics, aircraft flight control, aircraft sensors, weapons, targets, and threats. The cockpit is generic in nature and equipped with enough controls and displays so that various modes of a low altitude penetration mission can be flown with closed-looped stimulus provided by an out-the-window background scene. Figure 3 illustrates the final ITB Demonstration setup.

Figure 3: Integrated Test Bed

The final configuration for the Integrated Test Bed (ITB) reflects a new baseline for the laboratory and represents a phased integration approach during the project. The ITARS equipment was the first major equipment change and was integrated with updated F-15E and terrain relational models to form the configuration for Demonstrations I. VAMP clusters, System Mass Memory, and high speed data bus equipment were then added to configuration for Demonstrations II and III. The Ada Avionics Real-time Systems (AARTS) provided the operating system and executive control software for further integration of the application software. Finally, Ada application software modules consisting of algorithms for Navigation, Terrain Following, Terrain Avoidance, Terrain Aided Navigation, Route Planner, and System Executive were integrated. System loading and reconfiguration of the VAMP modules was made possible via the addition of the Processor Control Panel, Auxiliary Control Panel, Power Distribution Panel, Startup ROM, and System Executive software control. The System Mass Memory and ITARS equipment were connected to the VAMP clusters via the Block Transfer Bus and Mission Avionic Bus respectively. All the system integration effort as defined by the Pave Pillar In-House Project was complete as of 18 October 1991 and the following top level observations or conclusions were made:

a. The use of DMA databases is an important capability and the applications will expand quickly. As the Pave Pillar Team accomplished more experiments with DMA terrain, DMA cultural, threat, and other types of databases, it became very clear that a common digital cartographic database is mandatory for future avionic systems, and for the simulation systems that test avionic systems.

b. Any digital map of the ITARS class must support perspective head-down and head-up displays. The displays should be roll stabilized to provide the proper orientation to the pilot.

c. When the head-up display is presented to the pilot, the best ridge line drawing scheme to employ is to flood the display with as many ridge lines as possible within a set range. Twenty nautical miles seems to be the ideal range.

d. The ITARS three-channel display output method is very effective, and allows for excellent control of the video and associated display formats.

e. The ITARS Plan View Clearance Plane Bands are very helpful in the terrain following, terrain avoidance, and threat avoidance modes of operation. The colored bands of red for danger, yellow for caution, and green for good terrain clearance allow the pilot to see at a glance if the coupled flight control system and algorithms are functioning properly.

f. Testing of Common Modules and highly integrated Mission Data Processors like the VAMP requires careful planning and costly test setups. For example, the simulation interface units for the HSDB technology required highly sophisticated design to support the high rates of speed. In addition, monitoring a bus of this speed is extremely difficult since no commercial products are yet available and data must be buffered directly to a memory location for later data reduction.

g. The Common Module building-block concept is valid, but does cause initial development problems. A lot of the module problems early in the integration testing of the Ada Operating System were caused by hardware failures due to interaction of more than one module, PI-Bus traffic
errors, or interactions of certain processor instructions with PI-Bus traffic. Many of these failures could only be corrected through hardware fixes. When working with high density circuits of the VHSIC class of chips, hardware fixes become very difficult or even impossible without fabricating a new batch of chips. This means delays as well as very expensive engineering changes to the system. For an experimental laboratory program with limited budgets, this is very difficult to manage and requires constant trade-offs of experimental results versus resources expended.

h. The FOASC has been shown to be a feasible, practical and reliable solution to the problem of optical power loss in HSDB systems. It is a useful addition to the ITB, and shows its potential for use in a Pave Pillar avionics architecture.

i. Distributed processor modules that form a cluster configuration and have a volatile memory, cause special loading, start-up, restart, reload, and shutdown problems. In the ITB cluster configuration for example, there are a total of 20 possible MIL-STD-1750A CPUs to load out of the System Mass Memory via the Block Transfer Bus to PI-Bus interface. The Start-up ROM (SUROM) must be able to run a Built-in-Test and then request a load. The strategy for accomplishing this task was difficult to derive and implement. When considering strategies for loading and reloading a module, the possibilities are almost endless and must be carefully structured by the system developer early in the project.

j. Ada is a good real-time embedded language for avionics development. The project team had a steep learning curve to overcome with Ada; but once the Ada concepts and language structures were mastered, production output seemed to pickup.

k. Reconfiguration using the Pave Pillar architecture is both possible and practical. The time to fully load a VAMP CPU module from the System Mass Memory across the Block Transfer Bus was two seconds or less. The integrated avionic system could detect the failure, diagnose the fault, and have the avionic reconfigured outside a cluster from the System Mass Memory in enough time to keep the navigation systems from having major disruptions. System reconfiguration within a cluster is within real-time.

l. The area of hardware BIT and 98 percent BIT coverage on modules seems to be difficult to implement and is open to interpretation. This are merits further research and definition.