①

AGARD LECTURE SERIES No. 164

Systems Engineering

S DTIC D
ELECTE
JUN 22 1989
H

89 6 20 180

NORTH ATLANTIC TREATY ORGANIZATION

ADVISORY GROUP FOR AEROSPACE RESEARCH AND DEVELOPMENT

(ORGANISATION DU TRAITE DE L'ATLANTIQUE NORD)

AGARD Lecture Series 164

SYSTEMS ENGINEERING

)

THE MISSION OF AGARD

According to its Charter, the mission of AGARD is to bring together the leading personalities of the NATO nations in the fields of science and technology relating to aerospace for the following purposes:

— Recommending effective ways for the member nations to use their research and development capabilities for the common benefit of the NATO community;

— Providing scientific and technical advice and assistance to the Military Committee in the field of aerospace research and development (with particular regard to its military application);

— Continuously stimulating advances in the aerospace sciences relevant to strengthening the common defence posture;

— Improving the co-operation among member nations in aerospace research and development;

— Exchange of scientific and technical information;

— Providing assistance to member nations for the purpose of increasing their scientific and technical potential;

— Rendering scientific and technical assistance, as requested, to other NATO bodies and to member nations in connection with research and development problems in the aerospace field.

The highest authority within AGARD is the National Delegates Board consisting of officially appointed senior representatives from each member nation. The mission of AGARD is carried out through the Panels which are composed of experts appointed by the National Delegates, the Consultant and Exchange Programme and the Aerospace Applications Studies Programme. The results of AGARD work are reported to the member nations and the NATO Authorities through the AGARD series of publications of which this is one.

Participation in AGARD activities is by invitation only and is normally limited to citizens of the NATO nations.

ii

THEME

Recent AGARD activities have indicated a strong need for more efficient avionics system engineering. There is a growing need for reducing development time, effecting savings in costs of ownership, and in extending the life-time of avionics systems. This must be accomplished along with meeting needs of the user faced with a growing threat. With the growing complexity of avionics systems (as well as other systems), it is important to develop and maintain expertise in system planning, architecture, and management.

This Lecture Series addresses the important systems engineering aspects of Requirements, Systems Integration, Prototyping, and Design. In addition, the impact of technology on system architecture will be discussed. Methodologies are described and actual case histories will serve as practical examples of modern system engineering.

This Lecture Series, sponsored by the Avionics Panel of AGARD, has been implemented by the Consultant and Exchange Programme.

* * *

Le besoin de rendre plus performant l'ingéniérie des systèmes avioniques ressort très nettement des activités récentes de l'AGARD. Il devient de plus en plus nécessaire d'écourter les délais de développement, de réaliser des économies dans les coûts globaux et de prolonger la durée de vie des systèmes avioniques, tout en répondant aux exigences de l'utilisateur face à la menace grandissante.

Vu la nature de plus en plus complexe des systèmes avioniques (et d'autres systèmes) il importe de développer et de maintenir l'expertise existante dans le domaine de la planification, l'architecture et le management des systèmes.

Ce Cycle de Conférences examine les aspects importants de l'ingénierie de systèmes en matière de besoins, d'intégration des systèmes, de réalisation de prototypes, et de conception. En outre, l'impact de la technologie sur l'architecture des systèmes, est aussi examiné. Des descriptions de différentes méthodologies sont présentées et des cas réels sont utilisés à titre d'exemples pratiques d'ingénierie de systèmes modernes.

Ce Cycle de Conférences est présenté dans le cadre du programme des Consultants et des échanges, sous l'égide du Panel AGARD d'Avionique.

iii

LIST OF SPEAKERS

Lecture Series Director: Dr Fred I. Diamond
                            Chief Scientist
                            Rome Air Development Center
                            Griffiss AFB NY 13441-5700
                            USA

AUTHORS/SPEAKERS

Dr L. Berardi
AERITALIA
Gruppo Sistemi e Teleguidati
10072 Caselle (Torino)
ITALY

Dr H.M. Paskin
Development & Operations Division
Defense & Electronics Systems Center
Westinghouse Electric Corp
Box 1693
Baltimore, MD 21203
USA

Mr M.J. Breza
Performance Group
Dirctorate of Design Analysis ASD
Wright-Patterson AFB OH 45433
USA

Mr J.D. Rowley
British Aerospace
Warton Aerodrome
Preston, Lancashire PR4 1AX
United Kingdom

Mr J. Ostgaard
AFWAL/AAAS
Wright-Patterson AFB OH 45433
USA

Mr P. Schirle
Avions Marcel Dassault
Breguet Aviation (DSA)
78 Quai Marcel Dassault
92214 Saint-Cloud
France

Mr M.Tooze
GEC Avionics
Airport Works
Rochester, Kent ME1 2XX
UNITED KINGDOM

CONTENTS

)

AVIONICS SYSTEM ENGINEERING--AN INTRODUCTION

Dr Fred I. Diamond
Rome Air Development Center
Griffiss AFB NY 13441-5700 USA

Summary

System engineering is the process used in the evolution of systems from
identification of a need through construction and/or production and deployment
in an operational environment. It is a process that involves the application
of appropriate scientific and technical knowledge (1) to transform an
operational need into a system configuration with defined parameters, through
an iterative process of analysis, design, test, and evaluation; (2) to
integrate all performance requirements, including reliability, maintainability,
supportability, etc. into the total engineering effort; and (3) to integrate
related components to insure interoperability and optimum system performance.
It is a process that also considers economic factors such development and life
cycle costs.

The life cycle process involves several key steps, many iterative, but in an
orderly and controlled manner. They include Requirements, Architecture
Specification, Design, Development/Construction, Test & Evaluation, and
Operational Use.

With the growing complexity of avionics systems, effective systems engineering
is critical. Therefore, we must p't greater emphasis on architectures,
subsystem design and interfaces and system integration. Only through a total
systems engineering approach from the very initial phases of the system life
cycle may we achieve a well-engineered system. The payoff will be reduced cost
of ownership and greater mission effectiveness.

Introduction

A growing and more sophisticated threat plus shrinking budgets are placing
even more challenges to the R&D community. Faced with complaints about
lengthy and costly developments, rapid obsolescence, and excessive costs
of ownership, we have all heard the following concerns:

    How to develop timely solutions which are responsive to new and
        changing requirements.

    How to better integrate elements into new or existing systems.

    How to integrate hardware and software functions.

    How to achieve interoperability.

    How to reduce life cycle costs and make systems more usable and
        supportable.

    How to predict system cost and performance.

    How to accomplish timely and non-disruptive technology insertion.

    How to plan for proper test and evaluation.

    How to manage large programs.

These reflect the need for more effective system engineering, and such issues
have been addressed in many activities of AGARD's Avionics Panel.[1-7] With
the growing complexity of avionics systems, system engineering is critical.
The increased use of automation has broadened the scope of avionics system
engineering. A recent AGARD study of avionics identified system engineering
as one of the more pressing problems but also, an opportunity for major
improvements.[8] Incidently, system engineering is a major problem in all
development and acquisition activities, not only in other areas of Defense
such as Command and Control, but in many other endeavors of our society.
Apparently, as the need arose, we developed many kinds of engineering
specialists in all of the conventional engineering areas such as electrical
and electronic engineering, but we may have neglected the important role
of the system engineer, the individual who has the skills, breadth, and
experience to ensure that all interacting components fit into an efficient,
cost-effective entity.

In the broadest sense, systems may not be the same, but the general methods
used in arriving at a successful system are similar. This is the function
of system engineering, which should be considered as a professional activity
and an academic discipline. Many universities are now offering programs
in System Engineering, and many books address this subject.[9-11] System
Engineering may be defined as the process used in the evolution of a system
beginning with the identification of a need and ending with the construction
and/or production and deployment in an operational environment. It is a
process that involves the application of appropriate scientific and technical
knowledge (1) to transform an operational need into a system configuration
with defined parameters, through an iterative process of analysis, design,
test, and evaluation; (2) to integrate all performance requirements, including
reliability, maintainability, supportability, etc. into the total engineering
effort; and (3) to integrate related components to insure interoperability and
optimum system performance. It is a process that also considers economic factors
such development and life cycle costs.

The life cycle process involves several key steps, many iterative, but in
an orderly and controlled manner. One of several variations includes
Requirements, Architecture, Specification, Design, Development/Construction,
Test & Evaluation, and Operational Use. The requirements phase is the
beginning step in the transformation of operational requirements to technical
requirements. It includes examination of "customer" needs and concepts of
operations, along with considerations of performance, cost, scheduling, and
reliability and maintainability. It extends to analysis of these requirements
for the formulation of system functional requirements. The Architecture
phase involves system concepts and synthesis. Alternate architectures are
defined and prototypes are described. Hardware and software aspects are
examined along with subsystem interfaces. Modelling and mathematical
representation may play an important role in order that alternative concepts
and architectures may be examined and compared. The definition of the system
architecture and a detailed functional analysis leads to Specifications.
Given, for example, such avionics elements such as sensors, navigation,
displays, communications, etc., one may describe detailed technical speci-
fications and interfaces at the subsystem and component level. Relevant
standards are also identified, along with appropriate figures of merit and
measures of performance. The system design not only involves subsystems and
components, but also takes into account the interaction and interoperability and
integration of its component subsystems, the identification of critical elements.
It requires decisions regarding the use of off-the-shelf elements rather than
development. It must, at the onset, consider hardware and software as an
integral design. It should include such factors as life cycle costs, logistics
support and measures of effectiveness.

A good example is Reliability and Maintainability. The system design must
consider not only individual components with respect to specified performance
under carefully defined environmental conditions, but it also must take
into account overall system performance and design. Decisions regarding
redundancy, self-repair, fault-tolerance, graceful degradation, and
reconfigurability must be made. For example, redundancy can be practiced
at the chip level or at the subsystem level. We may wish to assign a function
to other resources. For example, if a computer supporting an important
function fails, one might wish to have the capability to replace it with
a similar computer performing a non-critical or low-priority function at
that particular phase of a mission. And new technologies such as active
phased arrays merit consideration of cost-effectiveness because of the
graceful degradation they provide. Thus, testability and maintainability must
be part of the design process, as well as the extent of, the usage, and
the nature of modules, for easy maintenance, reduced logistics support,
reconfigurability, and follow-on modification or upgrading. Proper attention
to factors such as these early in the life cycle will avoid problems later;
ignoring them may be very costly.

As the system progresses through its life cycle, engineering decisions are
made that may impact on earlier design decisions. The system engineer must
be sensitive to this; otherwise, adverse effects, directly or indirectly,
sometimes in subtle ways, may occur.

Development/Construction involves not only satisfying specifications and
standards. It must consider process control, quality assurance, testability
and cost. Test & Evaluation includes preparation of an appropriate test
plan, specification of test methodology and equipment, calibration, data
reduction, and analysis.

Each of the steps involves analysis and simulation, and may require feedback
and iteration. Such analysis may involve system trade-offs and risk assess-
ment, analysis of alternative concepts and designs, analysis of life cycle
costs vs. performance, and prediction of performance under different scenarios.

System engineering of aircraft systems is particularly challenging. Aircraft systems are "super" systems whose very elements (e.g. avionics) are themselves complex systems. They utilize a diverse collection of sophisticated equipment for sensing, communications, and information processing. They involve the interaction of equipment and the people who operate and maintain them. A major design challenge is the integration of information to be used by the pilot who must perform navigation, flight control, fire control, etc., interacting with sensor and threat data through displays, observations, and audio inputs. Furthermore, new technology is constantly emerging—adaptive signal processing, distributed processing and data bases, artificial intelligence/expert systems, parallel processing—and these must be considered in the context of system performance including interoperability, survivability, connectivity, rapid deployment, and reliability and maintainability. The nature of this very rapid progress in electronics, electromagnetics, and information processing is creating new system questions. With the advent of VLSI and VHSIC, the distinction between devices and circuits is vanishing; progress in monolithic microwave integrated circuits raises similar system and sub-system issues. Microprocessor developments raise new questions regarding the trade-offs between hardware and software. The evolution of photonics offers new opportunities and challenges for system design and integration. These technical trends may imply the need for and utilization of more specialists, but future avionics developments will also require systems-oriented engineers.

By definition, a system is any collection of elements that interact or interrelate. With more sophisticated and more interdisciplinary technology, System concepts and system engineering apply even at the microchip level. System engineering should be of concern at all levels of engineering, and involve both hardware and software. The need for a system-oriented engineer pervades every aspect of development and acquisition.

For example, consider the antenna engineer. In the future, the antenna engineer, must be such a systems-oriented technologist. The use of radiating elements on the same substrate with monolithic microwave integrated circuits and their coupling to processing electronics are system and sub-system issues. With further advances in high speed digital processing and analog-to-digital conversion, the prospect of all-digital beam forming, scanning, and processing is upon us. A clear distinction between transmitter, receiver, antenna, and processor may no longer exist, and interrelations among many specialties will become greater. Electromagnetic radiation is still important, but digital beam forming and adaptive nulling also require a knowledge of modern control theory and information theory. The digital manipulation of data, the processing of signals, and the generation of control signals require skills in the information sciences. This digital antenna is indeed a system problem that requires interdisciplinary skills. The transition from a relatively simple dish to the digital antenna is happening in other areas as well.

From the above, we may conclude that a good system engineer needs some special attributes. Among them are:

A mastery at formulating and solving system-type problems.

The talent to translate operational needs into technical requirements.

The ability to direct and coordinate multiple, simultaneous technical activities.

A capability and willingness to make firm engineering and management decisions.

An understanding of procurement procedures and financial management.

A sensitivity to the factors affecting the acceptability of his system concept.

The technical integrity to honestly surface and face issues.

Although system engineers may evolve out of one of the engineering disciplines which are involved in the system life cycle, the new breed of system engineer must have a good working knowledge of many engineering and scientific disciplines involved, in order to recognize interdisciplinary problems and lead or participate in their solutions. And yet, as any good system engineer, he must also be the prime mover in the conceptual and analytic studies required; he must deal creatively and decisively with technology, people, and environments; for let us not forget that achievement of a successful system also requires good management of the system engineering functions.[12,13] To carry out any program, an engineering organization consisting not only of engineers and scientists is needed, but also technicians, draftsmen, programmers, model builders and

non-technical personnel for financial management, contracting, etc. The efficient use of these skills must be planned and scheduled. Goals and objective schedules must be established. Technical direction must be provided, and managerial controls must be initiated.

Modern system engineers must be equipped and trained to use modern engineering tools, especially computer-aided capabilities.[14] Modern system engineers should use work stations which are equipped with software packages for mathematical and statistical analysis; with engineering tools for design and analysis and for modelling and simulation. They should include aids for project management-- cost, schedule, and performance tracking; cost analysis and trends; report generation. And all of these require suitable graphics.

The prevailing opinion is that system engineers are born, not made. Most large engineering firms and so-called system houses assign system engineering responsibilities to their brighter, accomplished engineers who seem to "think big." Many of the skills of system engineering and management are acquired through years of experience, and talents such as creativity and the ability to deal with complexity may be innate qualities. However, formal training and education are necessary to develop and provide:

(1) Interdisciplinary skills - broad knowledge of many engineering disciplines as well as operations research, and even manufacturing processes. Subjects such as control engineering, communications theory, signal processing, electromagnetics, and information processing are vital.

(2) Mathematical skills - sufficient knowledge of advanced mathematics to understand and direct the design analyses that may be needed.

(3) Computation skills - use of computers for analysis, instrumentation, and modelling and simulation.

(4) Management and Interpersonnel skills - to better deal with people and resources.

This new breed of system engineer may learn some valuable lessons from the emergence of software engineering as a discipline. The software engineer is well aware of the steps in the life-cycle of a project. An integrated software engineering environment is emerging; he is developing tools for automated prototyping and for the evaluation of progress in the development of software. Artificial Intelligence/Expert Systems technology is being applied in the development of software tools and methodology to support the various phases in the software life cycle, and to provide corporate memory such as the rationale behind design decisions made throughout the life-cycle. These techniques for software engineering are also being applied to system engineering of both hardware as well as software.

Future avionics systems, employing new technologies, present even greater system engineering challenges. One may envision multiple sensors, communications, and electronic warfare systems sharing a common adaptive phased array; we may anticipate a local area network of processors sharing the functions of control, computation, message processing, etc.; and we can anticipate intelligent displays that will provide information in a variety of forms--graphics, imagery, text, etc. that may be accessed by voice, by keyboard, or by touch. This implies that we must put greater emphasis on architectures, subsystem design and interfaces and system integration. Only through a total systems engineering approach that considers these factors from the very initial phases of the system life cycle, as well as modularity, internal communications, standards, reconfigurability, and supportability, may we achieve well engineered systems. The payoff will be reduced cost of ownership and greater mission effectiveness.

This lecture series will introduce to the system engineer and the potential system engineer some of the basic concepts of system engineering. Considering the broad scope of activities involved, these lectures concentrate on the initial phase of the system life cycle--requirements, architecture, design. Although all aspects of the life-cycle are important, the more attention given "up front", the less likely will be the need for the occurrence of errors and costly rework in the later stages of the life-cycle.

In the following paper, Dr Paskin addresses the avionics requirements definition process, at the conceptual level, in light of changing threats, technology, and business environments. He provides a perspective of total integrated system performance, looking at broad requirements issues rather than specific subsystem specifications. His fundamental premise is that avionics requirements are driven by four factors; Information and Data

Sources, Control Opportunities and Information Needs, Concepts, and Algorithmic Techniques, and Realization Technologies. These four factors are set in a system structure which shows their interrelationships and provides the framework for conceptualizing avionics system solutions to meet particular mission needs. An examination is made of the issues of fielding and affording the solution with specific emphasis on architecture, fusion, production, and support.

Mr. Rowle will describe a top-down methodology for establishment of design requirements. He will describe the structured design of the avionics system for the UK Experimental Aircraft Program (EAP) using a design tool called CORE (Controlled Requirements Expression).

Mr. Breza will then discuss the integration of avionics into the overall conceptual design phase. He will cover methods and concepts for the introduction of avionics systems in the initial design of aeronautical systems, describing the balanced design process which relates avionics, propulsion and armament systems. This involves the extension of current aircraft design synthesis and analysis procedures to incorporate avionics in order to achieve an integrated system design for optimum aeronautical system and avionics system parameters. Quantifiable measures of merit and the optimization of performance-to-cost ratio for new aeronautical systems are described.

Some recent examples of the evolution of architectures for modern-day information systems are presented by Mr. Ostgaard, DAIS in the 1970's and Pave Pillar in the 1980's. The Digital Avionics Information System (DAIS) was a system architecture for an avionic system utilizing digital technology to reduce life cycle costs by defining and developing hardware and software core elements and standardized interfaces which could be configured and applied to many aircraft. The Pave Pillar program established an avionics architecture with a design philosophy which permits resources to be shared across subsystems. This requires a highly coupled system-wide management and control program (operating system) supported by a wide-band data distribution network, high speed processors, and extensive mass memory.

The next series of lectures will be devoted to avionics system design. Mr. Tooze will first describe an approach to avionics system design and its application to modular avionics architectures. By taking various architectures and corresponding modular sets and applying functional descriptions of the requirement, each architectural candidat may be investigated for its capability to meet the expected requirements.

Dr Berardi will talk about rapid prototyping, but with emphasis on the growing use of artificial intelligence/expert systems for rapid prototyping. In particular, he will describe a design tool called ECATE (Expert Consultant for Avionics System Transformation Exploitation), developed by Avionics and Equipment Group of Aeritalia. This is an expert system that prototypes the information handling architecture of an avionics system. He further expands upon the use of artificial intelligence and computer tools with a description of a complete, integrated environment for the rapid development of software prototypes of avionics systems.

M. Schirle will describe the use of rapid prototyping leading to detailed specifications and corresponding test plans. He will describe examples carried out in the Avions Marcel Dassault-Breguet Aviation company through the analysis and the synthesis of the rapid prototyping used within the framework of Rafale A and Mirage 2000 NC systems developments. Although emphasis will be upon the practical use of prototyping and the results gained, he will also show other uses and benefits of prototyping.

And finally, to illustrate the impact of new technology and its introduction into future avionics systems, Mr Ostgaard will describe the Pave Pace program. This program is a systematic and disciplined approach to the technology maturation process and will provide the system definition for future development and demonstration of highly capable and affordable avionics technology for new/retrofitted aircraft in the 21st century. It addresses compatibility with existing avionics architectures which provide new capabilities in machine intelligence to provide better aircrew situation awareness and decision aiding in complex scenarios, ultra-reliable electronics for surge/austere support conditions, and system level computer aided development and support tools to reduce embedded computer software costs.

References

1. "Methodology for Control of Life Cycle Costs for Avionics Systems", AGARD Lecture Series No 100, May, 1979.

2. "Modelling and Simulation of Avionics Systems and Command and Control and Communications Systems", AGARD Conference Proceedings No. 268, Oct, 1979.

3. "Advanced Avionics and the Military Aircraft Man/Machine Interface", AGARD Conference Proceedings No. 329, April, 1982.

4. "Software for Avionics", AGARD Conference Proceedings No. 330, Sept. 1982.

5. "Advanced Concepts for Avionics/Weapons System Design, Development and Integration", AGARD conference Proceedings No. 343, April, 1983.

6. "Design for Tactical Avionics Maintainability", AGARD Conference Proceedings No. 361, May, 1984.

7. "The Design, Development and Testing Complex Avionics Systems," AGARD Conference Proceedings No. 417, May, 1987.

8. "The Potential Impact of Developments in Electronic Technology on the Future Conduct of Air Warfare", AGARD Advisory Report No. 232, Oct. 1985.

9. Blanchard, B. & Fabrycky W., "Systems Engineering and Analysis", Prentice-Hall, Inc., Englewood Cliffs, N.J.

10. Chestnut, H., Systems Engineering Methods", Wiley & Sons, Inc., New York, N.Y.

11. Wilson, I. & Wilson, M., "Information, Computers, and System Design", Wiley & Sons, New York, N.Y.

12. Chase, W., "Management of System Engineering", Wiley & Sons, Inc., New York, N.Y.

13. Skinners, S., "A Guide to Systems Engineering and Management", D.C. Heath & Co., Lexington, Ma.

14. Eisner, H., "Computer-Aided Systems Engineering", Prentice-Hall, Inc., Englewood Cliffs, N.J.

AVIONIC SYSTEM REQUIREMENTS

Dr. Harvey M. Paskin
Westinghouse Electric Corporation
Post Office Box 1693, MS 1112
Baltimore, Maryland 21203 U.S.A.

SUMMARY

This paper addresses avionics system requirements at the conceptual level in light of changing threats, acquisition strategies, technology, and business environments. The objective is to provide a perspective of total integrated avionics system performance which illuminates broad requirements issues rather than specific subsystem specifications.

The fundamental tenet of this paper is that although one can intuitively relate parametric and functional avionic system requirements to mission related activities, a more global view is necessary to ensure that system requirements aptly address the gamut of factors which relentlessly bear on the ultimate system design, development, production, and support. The premise is that avionics requirements are driven by four factors; Information and Data Sources, Control Opportunities and Information Needs, Concepts and Algorithmic Techniques, and Realization Technologies. These four factors are set in a generic systems structure which shows their interrelationships and provides the framework for conceptualizing avionic system solutions to meet particular mission needs. The structure focuses on t   role of avionics in providing situation assessment, response selection, response implementation, and communications. With this structure in place, avionic system requirements are then examined within the context of architecture, techniques, technology, producibility, and supportability. Each of these factors play a major role in determining the ultimate system specifications and are significantly altering the design parameters of systems as we know them today.

The importance of developing the right avionic system requirements in tomorrow's complex weapon systems cannot be overstated. It is the critical first step in the commitment of vast resources to multidecade design, development, production, and support programs. The ultimate measure of success is a system which *is affordable in peace* time and capable in wartime. The concepts advanced in this paper are intended to stimulate the move in this direction.

The classical approach to avionic system requirements focuses on functional entities such as surveillance, targeting, communications, electronic warfare, etc. This approach very quickly drives down to specific parameters such as sensitivity, dynamic range, power, aperture size, and field of view, associated with various sensor subsystems. The approach is quite straight forward and inherently pleasing to the engineering community because it allows rapid immersion into design, development, and testing. Unfortunately, the results often leave the operational planner and user with a system which not only doesn't meet mission requirements, but imposes a support and training burden which reduces operational availability and effectiveness. The fundamental tenet of this paper is that although one can intuitively relate parametric and functional avionic system requirements to mission related activities, a more global view is necessary to ensure that system requirements aptly address the gamut of factors which relentlessly bear on the ultimate system design, development, production, training, and support.

If we step back from the microcosim of system functional requirements and take a macroscopic view of factors affecting the lifetime of avionic systems, we can readily see that avionic system requirements are driven as much by circumstance as by function. (Figure 1)

Figure 1   AVIONIC SYSTEM REQUIREMENTS ARE DRIVEN BY CIRCUMSTANCES
• Threat
• Supportability
• Policy
• Acquisition Strategy
• Funding
• Schedule

Threat is usually the primary circumstance used to justify the requirement for advanced avionic systems, and normally drives the performance specifications. Supportability is probably the second most important driver   new avionics developments. When system failure rates and repair costs exceed availability requirem ... and fiscal constraints, replacement becomes a primary consideration. Specifications in this instance are more heavily influenced by supportability considerations than threat driven perfo...   characteristics. With the need for the system justified, the evolution of the ultimate config   is heavily influenced by the remaining circumstances shown in Figure 1.

Recent policy trends are driving industry towards cost-sharing, firm fixed price development contracts and performance warranties that have a definite effect on the ultimate system. The potential financial impacts associated with these policies are inherently mitigated with lower risk, lower technology approaches to the solution. Creativity is stifled by the fiscal reality of bottom-line constraints at the corporate level.

Acquisition strategies are also playing a major role in the determination of avionic system requirements. The use of prototyping and demonstration/validation programs allows a more creative approach to the problem as well as the time and resources to advance the state-of-the-art. Non-Developmental Item (NDI) policy on the other hand, forces a trade between what is desired and what is available rather than what is desired and what can be achieved.

Funding and schedule are two additional circumstances which heav'ly influence avionic system requirements. The amount, timeliness, and continuity of funding, as well as the schedule, play dominant roles in determining what technology can be brought to bear on the problem, and hence the content of eventual solution.

A recently imposed acquisition strategy in the U.S. involves the use of modular avionic specifications derived by a Joint Integrated Avionics Working Group (JIAWG). The intent is to standardize form-fit-function specifications at the line replacement module (LRU) level for specific subsystem components such as power supplies, computers, and signal processors. This activity could have a significant impact on eventual system performance parameters as well as cost and supportability.

With an awareness of and sensitivity to the circumstances cited above, we will now address avionics requirements by examining the role avionics plays in contributing to weapon system effectiveness. In every system in which avionics are used, the role of the avionics involves one or more of the functions illustrated in Figure 2.

By examining mission objectives for the system under consideration, one can arrive at a very specific set of data and information necessary to assess the situation. Given the situation, a number of alternative responses are normally available, and upon selection, they can be implemented to change the situation in a manner which achieves the desired system mission objectives.

Situation assessment normally requires knowledge of position, movement, and intent of both friendly and enemy forces. Parameters of interest in determining avionics requirements for situation assessment include field of regard, field of view, resolution, update rates, false alarms, false dismissals, and time to formulate.

Response selection is the process of deciding which of the available options to use and when to implement. Time to select and effectiveness are the key parameters which drive the avionics requirements.

Response implementation is the process of modifying the situation. Important parameters include availability, timeliness, and effectiveness.

Communications is the vital link which ties the system together. It includes man-machine and man-to-man interfaces. Effectiveness of communication is measured by its clarity, security, timeliness, and interoperability.

The Role of Avionics

FIGURE 2

Within the context of this top-down view of the functions of avionics systems, one can now consider the factors which drive the avionics designs. The premise of this paper is that avionics specifications and designs are driven by four factors: Information and data sources (what is available); Control opportunities and information needs (what is controllable); Concepts and algorithmic techniques (Ingenuity and expertise); and Realization technologies (Mechanization-Hardware/Software). This concept is illustrated in Figure 3.

FIGURE 3

At this time, one of the two key points of this paper is made: THE FIRST CASUALTY OF WAR IS INTRICACY -- A ROBUST FORCE STRUCTURE IS REQUIRED. As shown in Figure 4, it is absolutely essential to be able to define a minimum level of effectiveness below which the engagement is lost. Increased system complexity may increase effectiveness and result in quicker success with fewer losses, but the minimum essential capability must not be sacrificed.

The four factors of avionic system specification and design can now be examined in the more familiar contexts of the development and operational communities: Architecture, Techniques, Technology, Producibility, and Supportability.

**The First Casualty of War Is Intricacy
A Robust Force Structure Is Required**
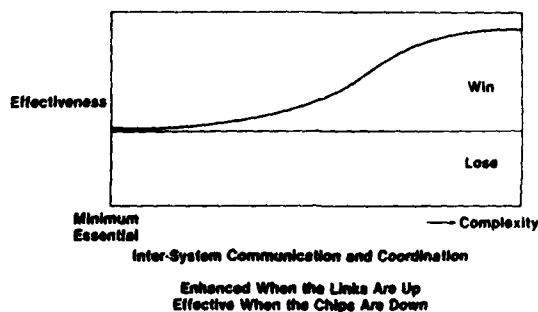
FIGURE 4

Avionics architectures are driven by three fundamental requirements; performance, availability, and cost. Performance is enhanced by architectures with high levels of interconnectivity where information and data at high bandwidths are fused and processed to provide high confidence situation estimates and response selections. Availability is addressed by architectures with low level parallelism that ensure no single point failure modes exist for achieving the desired minimum essential capability. Cost objectives are approached by designing for regular repetitive structures which use common functional hardware modules in as many cases as possible.

Revolutionary techniques will be a necessity to meet information requirements in the complex electronic, meteorological, and environmental conditions of the modern battlefield. Present avionic systems formulate a mostly one-to-one match between sensor and function. Association and fusion of multisensor data is rudimentary and limited to combinations of information which has already passed declaration thresholds. Final results are built on layers of decisions and require high confidence data near the front end. The most telling weakness of these fusion techniques is that if no one sensor declares an object, the object will not be declared. Furthermore, the problem of spatial and temporal association drives resolution, boresight, and false alarm requirements which result in design complexity and high costs. A top down process is required, one on which the final result is built on a single high level decision. The process should operate on variable confidence data from individual sensor reports and declare an object even if no one sensor so declares. The value of sensor contributions should be independent and intricate temporal and spatial association should not be required. These criteria might be used to measure the effectiveness of "fusion" efforts as avionic systems evolve.

Evolutionary technology will be the key to future avionics system implementation. Four technologies (gallium arsenide, mercury cadmium telluride, Very High Speed Integrated Circuits, and super conductivity) hold the promise of achieving levels of miniaturization, sensitivity, and processing throughput which will enable implementation of techniques not possible today.

Production goals will continue to be quantity, quality, and low cost. Concurrent engineering and application of proven techniques such as statistical process control and the Teguchi Method will make it possible to achieve the desired levels of quality at substantially reduced cost and schedule.

Supportability is one of the most critical design requirements in avionics systems. For weapons systems to be effective, they must perform adequately when they work, be easily maintained and fixed when they break, and not break, more often than is economically and operationally acceptable. It's important to recognize that most avionic systems are developed and procured in peace time, when cost is an issue. The design requirement for supportability should be a system which is affordable in peace time and sustainable in wartime. This means more than just high reliability and more redundancy. Figure 5 illustrates three approaches to avionic system architecture, each of which has substantially different impacts on availability. In the single thread architecture, every element of the system is contributing towards system effectiveness. When a failure occurs, the system effectiveness drops to zero until the failed element is repaired or replaced. The system down time is a function of the time-to-repair or time-to-replace and overall availability is further impacted by the reliability of the individual components. To compensate for inadequate reliability, back-up redundancy is often introduced into the system. In this type of redundant architecture, redundant sub-elements await "failure of the first team players" and then step in to fill the function. The system continues to meet "spec levels" until the redundant elements degrade to single thread status and a single point failure occurs. Although simple in concept, this type of architecture is the least cost effective approach to availability. The system is overburdened with "spare parts" which cost dollars and weight and only "pay for themselves" when the primary component fails.

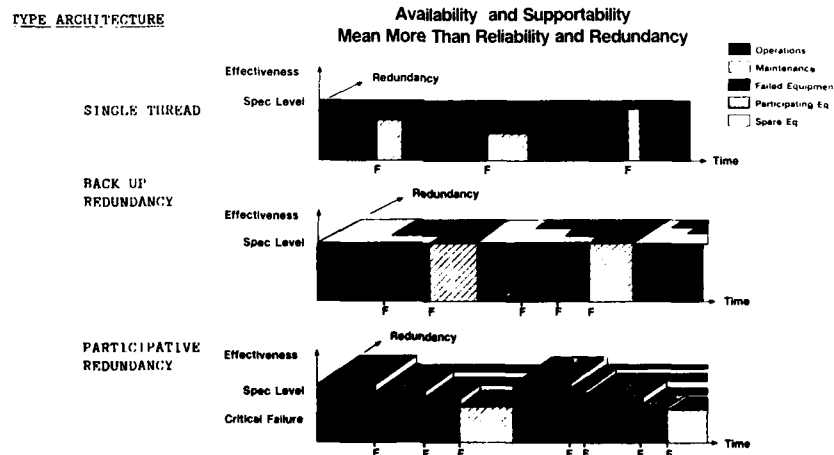AVIONICS ARCHITECTURES IMPACT AVAILABILITY

FIGURE 5

The second key point of this paper is now stated. SYSTEMS SHOULD HAVE AN ARCHITECTURE WHICH *GRACEFULLY DEGRADES TO THE MINIMUM ESSENTIAL EFFECTIVENESS BEFORE MAINTENANCE IS MANDATORY.* This concept is illustrated by the participative redundancy example in Figure 5. Redundancy is used to **provide** **capability** rather than **back** **up** **capability**. As sub-elements of the system fail, the effectiveness degrades, but the system continues to operate. Thus we introduce the notion of graceful degradation, rather than the strict "fault tolerance" in the previous architecture. The concept is analagous to designing system sub-functions in an "analog" manner so they degrade like a flashlight battery rather than a light bulb. Light bulbs fail catastrophically, where as batteries continue to provide capability even in a deteriorated condition. An air-to-air radar system would still provide effective warfighting capability even if its range dropped to 3/4, 2/3, and even 1/2 the spec value. The challenge is to develop system architectures and components to support this notion. It can be done. Electronically scanned antennas are an example (Figure 6). Antennas of this type are comprised of multiple phase control modules (PCM's) which form the antenna beam pattern. As PCM's fail, the antenna continues to operate although the main beam pattern broadens and the gain drops. Even after 20% of the PCM's fail, the sidelobes still permit effective detect and track operation (Figure 7) and as predicted by the radar range equation, the range is at +54% of full capability (Figure 8). Pilots in combat would still have sufficient capability to engage the enemy, even with a system in this degraded mode. In the final analysis, that's what avionics system requirements are all about.
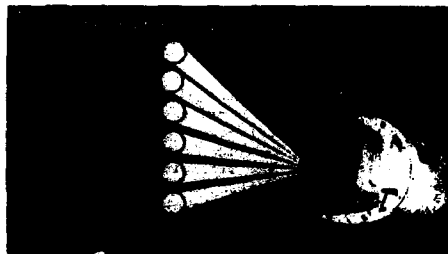

FIGURE 6

TYPICAL ELECTRONIC SCAN
ANTENNA PATTERN VERSUS FAILED ELEMENTS
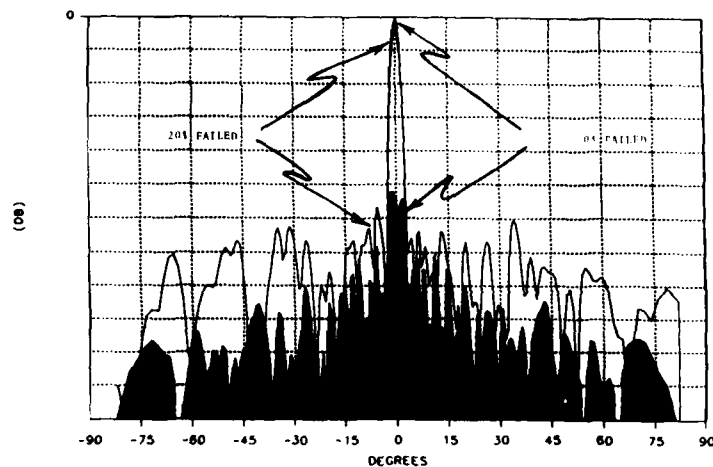
FIGURE 7

Range Performance vs Failed Elements
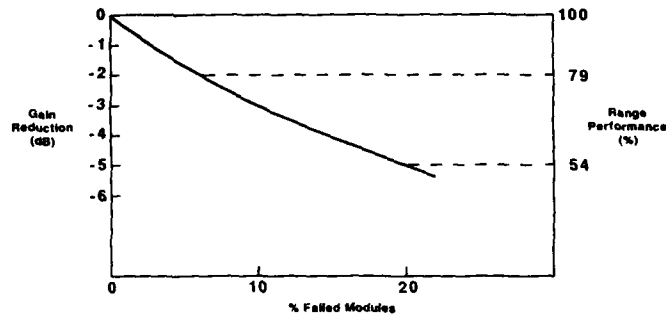
FIGURE 8

The importance of developing the right avionics system requirements in tomorrow's complex weapon systems cannot be overstated. Requirements are driven by a number of complex and interacting circumstances. Policy, acquisition strategy and funding often have as profound an effect as the threat and supportability. A global view of requirements from the perspective of situation assessment, response selection, response implementation, and communications leads one to consider the mission related effectiveness parameters rather than specific sensor specifications such as range, sensitivity, sidelobes, etc. The concept of a minimum essential level of effectiveness sets the threshold below which system performance is intolerable. This threshold is then the basis upon which architecture, techniques, and technology can be exercised to develop a cost effective, supportable, total quality system which will provide an affordable capability during peace time and a decisive combat advance during wartime.

A STRUCTURED APPROACH TO WEAPON SYSTEM DESIGN

J. D. ROWLEY
British Aerospace (Military Aircraft) Ltd
Warton Aerodrome, Preston, Lancashire, PR4 1AX,
United Kingdom.

1. INTRODUCTION

The prime purpose of this paper is to describe a structured approach to the design of a weapon system which British Aerospace (BAe) were able to develop and prove during the design of the avionics system for the Experimental Aircraft Programme (EAP) demonstrator aircraft. This aircraft first flew in the United Kingdom in August, 1986. Brief descriptions are given of the EAP avionics system, the main system design tools used, the activities carried out during the systems design process and the management and control procedures adopted. In addition a series of observations highlighting some of the findings of the project and providing pointers to the design of future weapon systems are given.

In designing the Avionics system of EAP, BAe developed the structural approach to system design called CORE - COntrolled Requirements Expression. This method of functional definition and partitioning ensure that a methodical, structured approach to the step by step process of progressive development is maintained. It also means that a clearly partitioned, well documented, unambiguous and easily traceable functional design is produced which can readily be changed without loss of integrity or design continuity.

During the functional design process outline implementation architectures are produced which enable optimum partitioning of functions to be achieved recognising practical constraints such as "off the shelf" equipment in addition to allocating functions so as to avoid duplication and reduce systems weight. Prior to EAP, equipments and sub-systems have been produced largely on a stand alone basis which has led to some duplication, excess weight and a lack of optimisation of system performance. Future aircraft will demand that weight is kept to an absolute minimum since it is critical to both the overall performance of the vehicle and its cost. It is necessary therefore that the Avionics Systems design is tackled in an integrated and disciplined manner in order to obtain maximum performance for minimum weight. A fully integrated weapons system design removes possible duplication and enables some equipments to under take several functions. It is the complexity and interdependence of the various functions involved which force the weapon system designer to look for improved design techniques which must include to following features:-

- a step by step approach which progressively develops the design rationale and which can be applied across the whole of the design. This in turn must provide a capability for planning the execution of the design and for monitoring its progress.

- a precise, consistent and unambiguous way of expressing system requirements at all levels.

- a means of applying checks at different stages of design life cycle to detect errors of specification or design in order to assure the design quality.

- an ability to demonstrate that the requirements have been met in order to assure the design quality.

- an ability to demonstrate that the requirements have been met in order to provide traceability of the requirements.

- a capability to provide configuration control of the design.

These are the principal features embedded in the structured approach which was adopted by BAe for the design of the avionic systems for the EAP and which are discussed in more detail in this paper.

2. EAP

While EAP has been created relatively quickly, its origins go back at least ten years. During this period engineers at British Aerospace, Warton, worked on various studies for a new fighter aircraft incorporating twin engines, delta wings, canards and both single and twin fins. Some of these studies were undertaken in collaboration with other Companies. In 1979 a proposal for a European Combat Fighter (ECF) was put to the British and German governments jointly by BAe and Messerschmitt-Bolkow-Blohm, while in 1980 a slightly modified design for a European Combat Aircraft (ECA) was prepared by BAe, Dassault-Breguet and MBB and put to their respective governments. Unfortunately the governments were unable to reach agreement on a common set of requirements.

During 1981 BAe continued its studies and defined the P110 project which involved the UK Avionics Industry in agreeing a weapon system architecture and producing equipment specifications. At the same time MBB were working up their TKF90 project which was very similar to the P110. Therefore in April 1982, the three companies (AIT, BAe and MBB) who

had previously co-operated to design and build the Tornado aircraft, agreed to investigate the possibilities of producing a joint specification to meet their individual national requirements. The resulting Agile Combat Aircraft (ACA) was unveiled in mock up form at the 1982 Farnborough Air Show and the UK government announced that they would provide support for a demonstrator aircraft which would be flown at the 1986 Farnborough Air Show. The demonstrator aircraft was to be called the EAP.

It was anticipated that two demonstrator aircraft based on the ACA design would be built, one in Great Britain and one in Germany. During 1983 a limited systems fit was agreed for the aircraft and due to the tight timescales of the project, equipment specifications were produced, put out for tender and Suppliers selected, very much in advance of the carrying out of a detailed system design. Unfortunately as a result of the German and Italian Governments decisions to withdraw at the end of 1983, work on the German aircraft did not proceed. However the chosen equipment suppliers from the three countries accepted BAe's invitation to continue with the design, build and supply of the numerous equipments required, from their own funds, for the single demonstrator aircraft.

In obtaining the agreement of the UK government to provide support for the demonstrator aircraft, it was necessary to agree the objectives which would be demonstrated. The areas chosen covered the fields of aerodynamics, structures, materials and systems and involved the development and demonstration of procedures necessary for the design, manufacture and test in these areas which were considered relevant to a future fighter aircraft.

This paper concentrates on the work carried out in the design and development of the avionics system involving both the MIL STD 1553B data bus and the modern electronic cockpit.

3. EAP SYSTEMS

The EAP has three major electronic systems: Flight Control System, Utilities Services Management Systems and Avionics System, the latter comprising communications, navigation and display and control subsystems. A simplified system architecture is shown in Fig 1.

3.1 Flight Control System

The EAP has a full authority, full time, digital fly by wire system to provide artificial stability and the necessary complex control functions. This system is based on the Jaguar Active Control technology aircraft which was the first aircraft to use fly by wire for flight control without mechanical backup. The system controls up to 13 surfaces simultaneously. The four identical flight control computers host the flight software which enables the pilot to fly the unstable aircraft and provides carefree manoeuvrability and increased agility. The computers also house software for failure management, reversion logic and built-in-test. The computers receive inputs from the four aircraft motion sensors, four attitude detectors, two air data computers and pilot inceptors, and provide the outputs to the control surfaces. In addition they provide air data information to the USMS and avionics systems via the two MIL STD 1553B data buses and air data attitude and heading to the reversionary instruments.

3.2 Utility Services Management System

While not originally claimed as a technological demonstration feature, the EAP has adopted an integrated computing system for the control and management of the aircraft utility system. The system comprises four system management processors connected to the dual redundant MIL STD 1553B data bus. The bus control function is embedded in two of the processors. The main utility systems which are controlled by the processors are as follows:-

- fuel management and fuel gauging

- hydraulic system control and indication

- undercarriage indication and monitoring, wheel brakes

- enviromental control system including cabin temperature control

- engine control and indication

- secondary power system

- LOX contents, electrical generation and battery monitoring, probe heating

The main benefits of this type of system for a high performance aircraft are a significant reduction in installed weight and operating costs, and a large improvement in availability. It also provides a simple interface to the avionics system in particular for the cockpit electronic displays and controls, this being one of the original drivers in evolving the system.

3.3 Avionics System

It was accepted that for the EAP, the avionics system would be a sub-set of the weapon system proposed for the ACA. This was called the "Core System" and provides the essential

features to fly a high performance aircraft namely navigation, communications, and displays and control functions. Transmission of data between the subsystems is via a dual redundant MIL STD 1553B data bus which greatly reduces the amount of wire required in the aircraft and simplifies the development and on-aircraft testing.

The navigation subsystem comprises an inertial platform with its own self-contained navigation processor and a TACAN and radar altimeter which share the same remote terminal.

The communications subsystem comprises a standard V/UHF radio and an emergency UHF radio. The control of this equipment is through an integrated control and management unit which also provides a voice warning facility. The latter supplements the normal aircraft warning system.

The displays and controls subsystem demonstrates several new technologies. Two identical waveform generators form the heart of the subsystem and are each capable of driving the three multi function colour displays and the wide angle holographic headup display. They also provide the bus control and executive control functions for the avionics system. Mission data such as waypoints, TACAN beacons, communication channels etc., are inserted by the pilot via the manual data entry facility mounted on the left hand flareshield. This information together with raw control data from the controls mounted on the consoles, throttles, control stick and displays is processed in two identical cockpit interface units prior to being transmitted on the avionics data bus. A cockpit lighting controller undertakes the task of monitoring the light sensors distributed around the cockpit and continuously regulates the power supplied to all the displays and controls to provide optimum illumination and display contrast at all times.

4.   DESIGN TOOLS

In parallel with the work being carried out on fighter aircraft studies during the 70's, BAe put considerable effort into examining ways of improving the techniques used for designing systems and also into the newer system technologies. Two specific areas which showed promise and were pursued with the support of the UK government, were:

-   means of improving the production of airborne software in terms of productivity and quality

-   investigations into the implementation of a MIL STD 1553B databus together with an all "electronic" cockpit including the multi-moding of displays and controls.

The former led to the development of an approach called Semi-Automated Functional Require-ment Analysis (SAFRA) while the latter led to the production of the Active Cockpit. Both of these tools were used to support the design of the avionic system for EAP and are described in the following paragraphs.

4.1   Semi-Automated Functional Requirements Analysis (SAFRA)

In examining ways of improving the productivity and quality of airborne software it was shown that the biggest improvements would be attained by the ability to find and eliminate errors at as early a stage as possible in the software life cycle. This led to the development of the approach called SAFRA which in particular addressed the lack of method, lack of visibility, lack of consistency and resolution of ambiguities in producing software requirements. Just as this method was applied to software requirements it was shown that it could be applied to the establishing of system requirements and was there-fore also adopted for this latter purpose.

The SAFRA approach encompasses a number of methods and tools which support the various stages of the system/software life-cycle. At the heart of SAFRA is a method called COntrolled Requirements Expression (CORE) which is used to produce system and software requirements that are unambiguous, consistent and complete. The method is based on the progressive decomposition of high level requirements in a logical and consistent manner until a level is reached where the requirements are expressed in sufficiently precise detail to allow hardware and software design to commence.

Each level of decomposition consists of a number of logical steps, eleven in all, which when applied to a higher level requirement produces the lower level components of the requirement. These steps can be collectively summarised as information gathering, estab-lishment of relationships, and the verification of relationships. The information derived at each level of decomposition is presented in diagrammatic form known as CORE diagrams. These diagrams use a precise unambiguous notation which can be checked for consistency and completeness across the whole of the systems requirement.

To assist in the production of CORE diagrams, a work station was developed which enables diagrams to be entered at a high-resolution graphics terminal and edited as required. It also provides a multi-user database in which diagrams are stored and a hard copy facility using a printer-plotter. Some automatic on-line checking of the diagrams for consistency is undertaken as they are being entered.

By producing the requirements in an unambiguous form in a computer batabase it is possible to check the data for consistency and completeness. This was done using PSL/PSA (Problem Statement Language/Problem Statement Analyser), which is a product of the ISDOS project of the University of Michigan, although other similar products such as EPOS are now available. The CORE notation is automatically described in PSL in a consistent manner

and stored in a new database. PSA is then used to provide checking and analysis of the database in numerous ways. When all the checks have been satisfactorily completed at each level of decomposition, the CORE database is made read-only by the configuration controller to allow the next stage of the design to proceed.

4.2 Active Cockpit

As the result of its continuing development studies, BAe gained considerable experience in the operation of active cockpit facilities and demonstrated their great importance in providing information on the man-machine interface to the system design process. In effect this facility provides a means by which rapid prototyping of ideas can be tested and developed with full operator interaction.

The Active Cockpit is housed in a wooden shell representing the actual cockpit. The displays and controls are positioned according to the best information available, use being made of commercial items wherever possible. Initially static displays are assessed for the development of the moding and formats. These displays are then driven dynamically in a representative manner together with other simulated functions such as engine, hydraulics, fuel etc., to fully exercise the cockpit displays and controls. To allow assessment under realistic flight conditions an outside world simulation system is provided. In addition a comprehensive fault injection system is used to allow assessment of the pilot/cockpit interface when single or multiple system failures occur.

5. SYSTEM DESIGN ACTIVITIES

The life-cycle stages in the design and development of a typical system are illustrated in figure 2. These can be grouped under three main headings namely system design, system implementation and system test. As defined, system design covers the stages of activity from the establishment of the initial high level system requirement through several levels of decomposition which produce the detailed requirements including hardware/software partitioning to the production of hardware and software specifications. System implementation covers the stages from the availability of specifications through to their realisation in either hardware or software. System test covers the stages of testing starting with individual equipments and software modules and building up individual elements to subsystems and finally integrating these to form the total system.

System design consists of 4 activities which can be summarised as:

-   Design Planning - This is the establishment of a design plan or route map showing the various levels of functional decomposition, interdependence between functional areas and thus design team tasks and documentation requirements.

    Data Gathering - This embraces the customer requirement and relevant pre-defined constraints such as off-the-shelf equipments and research findings, together with any principles and philosophies which should guide the design.

-   Functional Analysis - This is the actual functional design using the CORE tool for documenting, analysing and validating the design in a diagrammatic manner. To provide sufficient detail the Avionic system design was completed in three stages as shown in Fig 3.

-   Partitioning - This is the activity of allocating the functions to equipment, or in the case of multiprocessor units, to specific processors. This procedure is carried out at each level of functional definition and is refined as increased definition is achieved.

A good example of optimum partitioning arose during the EAP design when having determined the major functions of the avionics system, it was necessary to decide where these functions would be carried out. In examining locations for the bus control, executive control, display management and symbol generation functions, there were several possible choices. The most obvious way was to combine the bus control and executive control functions in a single equipment and the display management and symbol generators functions in a separate equipment. However from analysis of databus traffic it was shown that the databus traffic could be significantly reduced by combining all four functions into a single equipment. It was also shown that by using a single equipment significant savings in weight, volume, cabling, power and cooling would result and finally it was established with potential suppliers that this solution was viable. Thus a specification for a waveform generator which undertook all four functions, was put out to tender.

In this specific case the main operational software which included the bus control transaction tables, executive control, warnings and display Management functions were produced by BAe while other software functions for bus control algorithm, symbol generation, built-in test and basic system operating modules were produced by the equipment supplier. This combination of software within individual processors was satisfactorily achieved through the use of common software standards and tools.

6. SYSTEM MANAGEMENT AND CONTROL

The structured design process with its pre-defined life cycle stages provided the basis for the management and control strategy. Each life cycle stage has a specified start point, purpose and resultant output. Thus as the life cycle unfolded the successful

completion of these outputs provided management with the necessary measure of progress and achievement.

The management control procedures, to which all life cycle stage products were subjected can be summarised as follows:-

- Review - The aim of this is to ensure both the satisfactory completion of one stage of the life cycle before commencing the next and the correct planning of subsequent stages to achieve successful implementation. The review was both technical and managerial. Technical; to check both automatically and through independent quality control, that the design was compliant, accurate, traceable and conformed to standards, and managerial to ensure all corrective actions were resourced and progressed adequately.

- Configuration Control - The formal review having been completed a formal baseline was established for each life cycle product allowing work to proceed against that formally controlled baseline. As more detailed work continued errors were exposed the correction of which were rigorously controlled through the change control procedure.

- Change Control - Through this procedure the effect of a change on all potentially affected areas had to be assessed prior to approval being given for the change. This is particularly important in an integrated system since changes in one processor can cause changes in other processors. If the change was sufficiently large or important it would be subjected to the formal review process.

- Configuration Status Accounting - In a project with a large number of configuration controlled items and an even larger number of changes in progress it is essential that a status account of these items is maintained and circulated to all project participants. In addition to the change status it is beneficial to maintain an analysis of the change data.

7. OBSERVATIONS AND CONCLUDING REMARKS

The use of the structured design approach together with its associated tools undoubtedl: contributed to the success of the EAP in allowing the design of the avionics system to achieve a high standard in an extremely short timescale. The following specific points are considered to be worthy of noting summarising what was learnt from this exercise and in providing pointers to the design of future weapon systems.

- The structured approach together with the application of the rigorous management and control procedures enabled realistic programme plans to be produced and provided a high level of visibility in terms of progress of the design activities. The change statistics proved to be a valuable indicator of how the project expectations and requirements were being fulfilled.

- The undertaking of a freeze of the systems requirements prior to starting the functional analysis process and then the application of a strict configuration control procedure which virtually eliminated the introduction of changes to these requirements once the freeze had taken place were considered to be major factors in enabling the programme timescale to be met.

- The use of the structured approach brings about a significant increase in the amount of design documentation produced but this is greatly assisted by the use of computer aided tools which reduce the labour intensive nature of this task. This increase in design documentation is a significant step forward in overcoming past deficiencies of having insufficient information readily available. It also significantly reduces such tasks as the production of test specifications, customer manuals etc.

- The key to the production of a high quality system design was undoubtedly the insistence on the adherence to the very rigid control and management of the design process. CORE proved to be a very powerful tool not only for design but also for fault finding due partly to the extensive documentation. This was found to be very versatile in assisting the engineers to rapidly locate the problem area and correct it. It also enabled changes in the requirements to be introduced easily and rapidly.

- The structured approach to the total system design places more responsibility on the weapon system contractor who carries out the partitioning process, and therefore determines where and how the various functions will be carried out. How much of the resulting activity is undertaken by the weapon system contractor or the avionic equipment suppliers is a matter of debate. It is considered that avionics suppliers will continue to implement the specialist functions such as sensors, displays and processor hardware but the definition of the on-board software will become more the responsibility of the weapon system contractor.

- As well as the tools associated with SAFRA which were used to assist in the design of EAP systems, use was also made of mainframe text processors, minicomputer word processors, standard proforma and data base tools. For future projects, the extension to a comprehensive, centralised, computerised engineering database is considered to be highly desirable.

- In any project there is a need for effective configuration control throughout the design phase. On EAP this was handled by manual means supported wherever possible with computer aids. As the size of a project increases and the use of a centralised engineering data base with multi-user access is established, so automated configuration control tools must be available.

- It is considered that complex system requirements cannot be accurately described in plain "English" text. The use of tools such as CORE generate their own design language and introduce a need for training not only of the system design engineers but of all the personnel who will be associated with the project such as test engineers, support engineers and in particular managers and representatives of the customer. It is extremely important that the latter two groups of people fully appreciate the design approach being undertaken and its associated documentation.

- A major difference between the structured "top-down" approach compared to existing "bottom up" approaches, is the elapsed time before some functioning of the system can be seen. In the latter case it is usual for some areas of the system to become visible early in the programme, but in the former case the system tends to come together all at once, albeit on time. Management must be made aware of this difference from the start of the programme and must rely on the visibility of progress provided by the method, to justify their faith that the System Designers will meet their objectives.

In conclusion it must be admitted that there were doubts during the initial stages of the project as to whether the structured design approach was sufficiently developed to enable us to achieve our declared objectives. In retrospect the success of the project in achieving 35 flights in the first 30 days with no requirements for system changes, shows that the doubts were unfounded. In particular, the success of the structured design approach applied to the avionics system in terms of the quality of design, the timescale achieved and its supportability were beyond our expectations and indicates the way forward for the design of the more complex weapon systems of the future.

ACKNOWLEDGEMENTS

Figure 1

EAP SYSTEMS (SIMPLIFIED ARCHITECTURE)

Figure 2

BRITISH
AEROSPACE

System Life-Cycle

| Design | Implementation | Test |
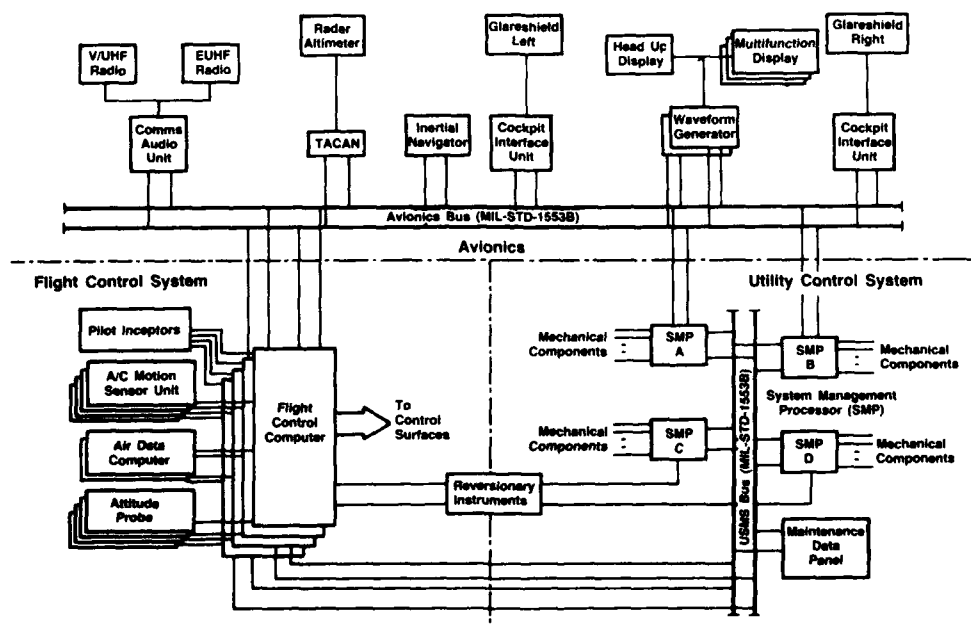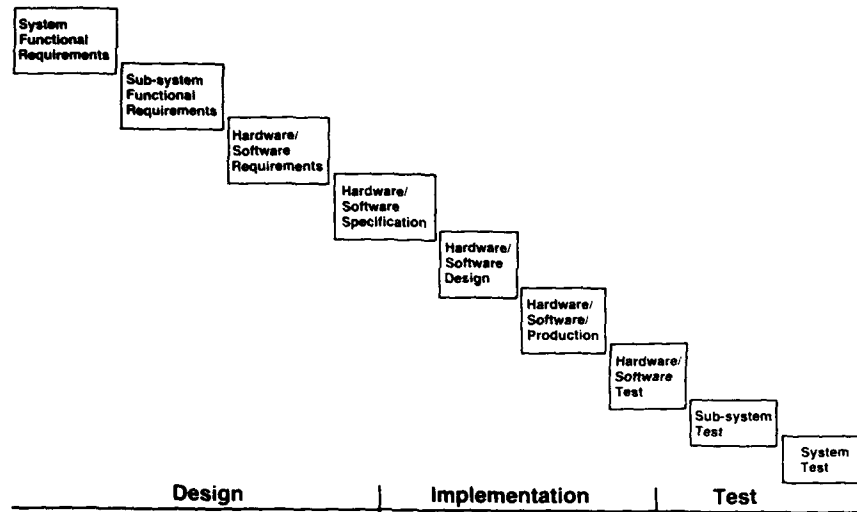
Figure 3

BRITISH
AEROSPACE

System Design Route Map

	Input	Activity	Output
Stage 1	– Air Vehicle Specification	– Define Functional Requirements	– Functional Requirements
	– Principles and Philosophies, etc.	– Partition to Sub-systems	– System Interfaces
Stage 2	– Functional Requirements	– Define Sub-system Functional Requirements	– Sub-system Functional Requirements
	– System Interfaces	– Partition to Equipments	– Sub-system Interfaces
Stage 3	– Sub-system Functional Requirements	– Partition to Hardware/Software	– Hardware Specifications
	– Sub-system Interfaces	– Define Processing Requirement	– Software Specifications
			– Equipment Interfaces

INTEGRATED AVIONICS - CONCEPTUAL DESIGN

Michael J. Breza
Directorate of Design Analysis
Aeronautical Systems Division
Wright-Patterson Air Force Base, Ohio   45433

ABSTRACT

Avionics of modern military aircraft is essential for maximizing performance realization of the total
aeronautical system. In the early conceptual phase, aeronautical systems designers give scant attention
to the interaction of avionics components. The aircraft design team generally provides weight, volume
and power considerations for the desired avionics functions and assumes that an avionics suite can
eventually be assembled. Even less attention is given to the potential synergistic effect avionics can
have with the aircraft design process. In contrast, the designers expend a large effort on finding the
best balanced combination of airframe and propulsion components which satisfy the design objectives.
This paper will attempt to show why avionics must be a co-equal member of the aeronautical system along
with airframe, propulsion, and armament. In becoming a co-equal partner, avionics must be an element of
the system design analysis, commencing with the early conceptual design phase of a new aeronautical system.

Introduction

During the conceptual design phase of military aeronautical systems, considerable attention is
placed on conducting tradeoff analyses with airframe and propulsion related parameters. The purpose of
these tradeoff studies is to find the best combination of wing size, wing shape (aspect ratio, sweep
angle, thickness), engine characteristics, and many other configuration parameters to best fulfill the
flight performance objectives.

Current conceptual design practices do not allow the avionics parameters to play in the design trade-
off and optimization process. Certainly, aircraft designers are aware that avionics systems are important
to the total aeronautical system. Thus, they establish allocations that account for the presence of the
avionics components within the aircraft. These allocations are specified in terms of volume, weight,
electrical power, and cooling requirements. Additionally, the designers do consider some of the very
obvious constraints imposed by certain sensors. For example, the fire control radar must be located to
provide a large, unobstructed field of view.

At the present time, the aeronautical system designer accepts these constraints. Their acceptance
may come with reluctance, but nonetheless, these constraints are accepted like other constraints imposed
by flight crew and payload provisions.

The Need

Avionics need a co-equal partner with the airframe, propulsion, and armament components, and sub-
jected to design iterations along with the other aircraft components. Why should the avionics system
parameters be subjected to iteration with airframe, propulsion, and armament parameters in the early con-
ceptual design tradeoff process? The answer to this question seems obvious. First, the avionics systems
costs are a major portion of the total system costs. For a modern fighter or attack aircraft, the avion-
ics system will probably cost more than either the airframe or propulsion systems. The second and
probably more important reason is that avionics are essential elements in the effectiveness of an aero-
nautical combat system. Simply stated, modern combat aircraft cannot accomplish its task without an
avionics complement.

Since the avionics suite in itself is critical to the mission accomplishment, it also has profound
interactions with the airframe, propulsion, armament, and, of course, very significant interactions with
the pilot. To examine these interactions, there is a need to make avionics conceptual design an integral
part of the concept formulation process of advanced aeronautical systems.

Due to ever increasing avionics complexity, these components of modern military aircraft now signif-
icantly impact the cost and utility of the total weapon system. Avionics requirements are usually defined
in operational terms (e.g., detection range, field of view/regard) and not in terms of design parameters
(e.g., band width, prf, NEI, etc) which the design engineers can relate to. Typically an "allocation" in
terms of weight, size, power and cooling needs is made to account for the presence of the avionics suite
inside the aeronautical system. As to what subsystems make up the avionics suite and how they are related
to the operational task is left for later definition. Much of the aircraft configuration is frozen. The
drawback with this approach is that it gives inadequate attention to the interacting effects avionics have
with the airframe/propulsion systems. Thus, the process of arriving at the "best" total design for the
mission is fundamentally incomplete.

To incorporate avionics into the aircraft design process, the aircraft design team has to integrate
the inputs of all participants involved with the system development process. These participants include
the designer, the operational user, and the technologist. Each of the participants' concerns must be
properly balanced within the framework of the overall system to insure that an effective and affordable
advanced aeronautical system can be produced with low risk. How these participants do their particular
job and how they interact in arriving at a system solution will be discussed below.

Avionics in the Aircraft Design Process

Currently, in the early stages of the aircraft design process, aircraft design teams spend considerable effort determining the best balanced set of airframe and propulsion parameters that achieve the design objectives. To accomplish this, the design team formulates an aircraft design synthesis and analysis procedure. Typically, the design synthesis and analysis procedure is formulated within the organizations' computer-aided design system. Once the procedure is established, the design team will utilize the procedure to systematically generate an array of aircraft design possibilities. Within the design array, the major configuration design variables such as wing size, wing shape, sweep angle, fuselage length and diameter, number of engines and engine thrust size will be examined. For the aircraft performance analysis, different measures of merit are used depending on the aircraft's intended mission. Generally, the aircraft design team is interested in flight range, maximum speed, maximum turn rate, etc. Aircraft cost is also computed based on data derived from the design synthesis process.

The inclusion of avionics design in the conceptual design process appears feasible. Obviously, the new conceptual design system would have a number of new input parameters. Also, different measures of merit are required to assess avionics performance. These new measures of merit must be clearly defined. There are a myriad of measures of merit used to characterize avionics performance that can also be linked to the aircraft performance. For example, aircraft velocity and altitude impact the performance of some sensors and the aircraft's configuration may limit a sensor's field of view. Also, avionics performance is strongly coupled to some newer aircraft performance measures such as infrared signature and radar cross-section. A reduced signature airframe would, for example, allow a reduction in ECM output power and, hence, weight necessary to defeat a threat.

To merge avionics requirements into the total aeronautical system design process, suitable design methods are needed. In the conceptual design phase, where there is little or ill defined avionics hardware, it will be necessary to construct mathematical models that synthesize the avionics suite within the overall system. These models need not be extremely elaborate, but must have sufficient detail to provide the definition needed for the avionics performance and cost analysis. Also, these models must provide a preliminary assessment of the volume, weight, power and cooling requirements that impact the airframe and propulsion systems.

The avionics system performance can then be evaluated within the total aeronautical system design process. Many of the computational tools required to conduct the avionics performance analysis already exist. It is just a matter of utilizing these tools earlier in the aeronautical system design process.

Once the new integrated design synthesis and analysis procedure is established, numerous trade studies can be conducted. Then it will be possible to examine the interactions that exist between the avionics suite and the other major components of the aeronautical system. The question of which combination of sensors should be used in a given weapon system could then be examined in quantitative terms. This will aid greatly in preventing the over-specification of subsystem requirements and consequent high cost.

Interaction with the User

Another important participant in the process of avionics into the aeronautical system conceptual design process is the user of the aircraft. The aircraft users are generally the military combat commands like the Tactical Air Command, Strategic Air Command, etc. In the Air Force, the commands make their requirements known by issuing Statements of Need (SON). The level of detail of the SON can vary from stating specific performance levels like detection range to very general statements like "it shall be able to detect." A problem frequently arises when the user specifies a level of performance that is difficult or very expensive to achieve with available technology. When this situation occurs, it is necessary to work with the using command to evolve a set of design requirements that are technically viable, reasonable in cost, and still provide the margin of military utility. In establishing this dialogue with the user, it is necessary to examine the effect that user requirements have on the total aeronautical system's cost and performance.

Interaction with the Technologist

Another important participant in the process of introducing avionics into the aircraft design process is the technologies. The technologist as defined here are the scientists and engineers who mature relevant technologies for transition into new systems. They are the people who provide new system performance opportunities. In the course of maturing new technologies, they build experimental articles to conduct both laboratory and field tests. Frequently, prototype articles are flight tested in a "real world" environment. The prototype equipment and tests conducted , x   ery valuable database for interpretation and application by the aircraft design team.

This database can provide part of the foundation for the mathematical models used in the integrated aircraft/avionics design synthesis and analysis methodology. The avionics specialist on the design team works closely with the technologist to insure that the database contains all the information needed. The test hardware provides the necessary physical data such as size, weight, power required, etc. The test results provide the performance data and all conditional information which affects the performance. The designers need to know the environment and other conditions which degrade or restrict the device's performance. Having a sound and complete database, the design team will then be able to evaluate not only the proposed device, but also synthesize variations of that device.

Armament Considerations

Armament technology is another area that the avionics conceptual designer and integrator must be intimately aware of. The accuracy and effectiveness of weapons delivery systems—fire control for both guns and missiles, both short and medium range—depends on the appropriate matching of the fire control sensor with the weapon capability. Here also, the avionics suite's performance requirements

have to match the anticipated capability of the armament. A reactive threat that responds to our system improvements by improvements in kind (such as those involving lower signatures, emissions, or tactics) must also be considered as part of the design process. This would require a capability to conduct sensitivity analysis on threat reaction to identify those attributes that could drive our avionics performance towards unacceptability.

An Avionics/Aircraft Design Trade Example

The issue of one versus two crew members in a fighter/attack aircraft has been long standing. The argument centers on the subject of the pilot work load versus the capability of the electronics aids that can be made available to reduce the pilot's work load. The topics of mission effectiveness and the aircraft's life cycle cost strongly impact this issue. Revolutionary changes that are occurring in information processing and displays will undoubtedly cause the issue of one versus two crew members to continue. If we go with one pilot, the inclusion of a FLIR with a field of regard (FOR) equivalent to that of a pilot's view is of current interest. A location in the nose of the aircraft would be a necessary constraint to satisfy this FOR. However, to obtain reasonable sensor recognition performance, the aperature size and the associated housing could create aerodynamic problems. Hence, aperature/sensor performance/aerodynamic impact trades must be made.

An integrated aircraft/avionics design synthesis and analysis procedure will give the conceptual design team an ability to quantify design issues that are currently the subject of much conjecture.

Conclusions

Much work needs to be done to insure that avionics are satisfactorily considered in the aeronautical system design process. It is technically feasible to make avionics an equal partner with the airframe, propulsion and armament components even in the early conceptual design phase of a new aeronautical system. Current aircraft design synthesis and analysis procedures can be modified (and/or expanded) to incorporate an avionics suite synthesis and analysis procedure that is integral to the overall aeronautical system design methodology. The new integrated design system could then be utilized to define the best set of aeronautical system design parameters, including the avionics suite. By incorporating avionics considerations into the aeronautical system design process with quantifiable measures of merit, the performance-to-cost ratio of our new aeronautical weapon systems will improve dramatically.

THE EVOLUTION OF DIGITAL AVIONICS ARCHITECTURES/SYSTEMS

AND

THE IMPACT OF TECHNOLOGY

JOHN C. OSTGAARD AND D. REED MORGAN

U. S. AIR FORCE WRIGHT AERONAUTICAL LABORATORIES

AVIONICS LABORATORY

WRIGHT-PATTERSON AIR FORCE BASE, OHIO, USA   45433-6523

SUMMARY

This article encompasses the evolutionary design/development of modern-day digital information systems. Included are the initial attempts at using digital technology in the early 1970s, the system integration thrusts of the 1980s and the continued system technology revolution of the 1990s.

DAIS - 1970s

The Digital Avionics Information System (DAIS) was the first attempt at defining a system architecture for an avionic system utilizing digital technology to reduce life cycle costs. The strategy was to develop hardware and software core elements and standardized interfaces which could be configured and applied to many aircraft.

The DAIS architecture consisted of federated processors communicating with each other and the other system elements (sensors, weapons, and controls and displays) through a standardized multiplex data bus. Centralized system single-point control was performed by a processor resident software executive that could be relocated for redundancy. Application software was structured to provide modularity, reliability, and transferability. This system architecture was flexible to accommodate a wide variety of avionics configurations, missions, and sensors, which provided redundancy to improve availability, and accommodate changes in technology.

The basic architecture was designed for a broad class of configurations where the number of processors could be reduced or enlarged depending upon the avionics and mission requirements. Standardization, commonality and application independent executive software allowed adaptability of this architecture to a broad class of different applications as well as to making mission-to-mission changes in a particular aircraft.

PAVE PILLAR - 1980s

The PAVE PILLAR program has established an avionics architecture for tactical and strategic aircraft using revolutionary new technology. The trend in avionics system design is to increase system integration thereby improving system performance, increasing system availability, and decreasing cost of ownership. The PAVE PILLAR design philosophy, which permits resources to be shared across subsystems, requires a highly coupled system-wide management and control program (operating system) supported by a wide-band data distribution network, high speed processors, and extensive mass memory. The ability to consider integration of this magnitude is dependent on recent advances in microelectronics technology, most notably the advent of VHSIC and fiber optics technology.

The PAVE PILLAR avionics architecture is functionally divided into three distinct areas: mission management, sensor management and vehicle management. The three areas define the enclosing boundaries for resource sharing, sparing and substitution. Unique characteristics of each of these areas preclude the utilization of resources across areas for the purpose of function recovery or reconfiguration. This does not imply that the areas do not contain many common hardware elements, but that the organization, connectivity and control of these components restrict their practical use in functional system-wide reassignment. Each of the three areas have associated with them a logical processor type and varying interface requirements. The implementation concept is for system elements to be built from a set of common modules supporting programmable processing, I/O and memory storage functions. The interfaces between system elements are standard high-speed time division multiplex buses and data links, all utilizing fiber optics technology.

PAVE PACE - 1990s

An avionics technology base of historic proportions is currently being sponsored by United States government R&D agencies. If these technologies are properly exploited, matured and integrated during the 1990s, 21st century avionics will undergo a change that may be as dramatic as the introduction of the transistor. Increasing avionics

capabilities while decreasing avionics acquisition and support costs will be a continuing challenge. This program is a systematic and disciplined approach to the technology maturation process and will provide the system definition for future development and demonstration of highly capable and affordable avionics technology for new/retrofitted aircraft in the 21st century. Complete compatibility with the PAVE PILLAR/ATF Advanced Avionics Architecture will be maintained. New Capabilities to be provided include: machine intelligence to provide better aircrew situation awareness and decision aiding in complex scenarios, ultra-reliable electronics for surge/austere support conditions, and system level computer-aided development and support tools to reduce embedded computer software costs.

INTRODUCTION

One of the principal trends in avionics has been and continues to be from analog to digital electronics. The rapid growth of solid state micro-electronics and its general commercial availability has been responsible for this trend. Applications have progressed from simple replacement of "conventional" analog components like vacuum tubes and discrete components by solid state equivalents, to function replacement by solid state logic arrays, finally to total implementation of system functions by solid state, integrated circuits, including full computer processing capability.

Another dominant trend has been to integrated system architectures where digital subsystems are closely coupled under software control, exchanging digital data, to perform a total weapon system function.

These trends have greatly increased the performance and capabilities of our weapon systems, although at the costs of increased system complexity. The gains we have achieved in improved system performance, decreased weight and volume, lower power consumption, increased reliability and lower cost per function by the use of high density solid state micro-electronics have been offset, to some degree, by penalties we now have to pay in longer and more complex system integration programs, software design and management and generally more sophisticated logistical support.

A relatively recent concern is the short "lifetime" that many micro-electronic products exhibit, as technology continues to rapidly evolve and companies move on to broader, more profitable markets. A five year product life cycle is not very compatible with a 25-year weapon system life cycle.

Given that most people will admit that avionics systems employing modern micro-electronics and real-time software carry their own unique class of problems, there is still a natural resistance by the system designers to any attempt by the customer to specify system characteristics beyond required performance. That is why there has been some rocky going as we cautiously implement a number of standardization concepts and actual architecture related standards. However, there has been steady progress.

Progress has been fastest and easiest when we have stayed in the domain of interface standards. The implementation of the MIL-STD-1553 multiplex bus standard is an obvious success story. Major weapon system designs like the F-16 and the B-1 broadly used this standard definition of multiplex signal interfaces and protocols. Even cross-service standardization has occurred through the Navy's commitment to MIL-STD-1553 on their F-18 program.

We have and are now implementing even more interfacing types of standards, such as the MIL-STD-1750 instruction set for avionics computers, the Ada Higher Order Language (HOL), specified by MIL-STD-1815, for avionics Operational Flight Programs (OFP), and the High Speed Data Bus standardization activity currently underway. Note that these are not "pure" interface standards, but begin to impinge on the area of system architecture, both from a hardware and software sense. Similar efforts are going on in the area of 32-bit micro-processors, an especially urgent task, since these devices may be pervasive in all of our systems in the near future to support such rapidly developing disciplines as artificial intelligence. But things start to get interesting when we move out of the area of simple interface standards into the world of system integration - when we talk not only of system interfaces but of system topology. Included in this level of system architectural considerations are concepts which have evolved from the Digital Avionics Information System (DAIS) program through the PAVE PILLAR program and are projected for the PAVE PACE program.

ERA OF THE 1970s

The decade of the seventies was marked by a rapid development in digital technology. Integrated circuit technology evolved from the 7400/5400 dual-inline packages (DIP) to the early rudimentary designs for bit-slice and microprocessor on a chip. Data communication evolved from point-to-point connection to multiple device interfaces on a single/dual redundant multiplex data bus. This increased use of digital technology also permitted an increase in programming capability and assembly language programming began to "give way" to the use of Higher Order Languages (HOLs) such as JOVIAL and FORTRAN. Likewise, display technology and the use of programmable displays began to take place with the development of the "all glass cockpit" concept.

DAIS

The *Digital Avionics Information System was formulated specifically to address the* System Architecture issues.

The *Digital Avionics Information System (DAIS) has been characterized as a system* architecture which can be applied and configured for a broad class of avionic applications and missions utilizing digital technology to reduce life cycle costs by defining and developing hardware and software core elements and standardized interfaces which can be configured and applied to many aircraft.

The DAIS approach reflected a total system concept rather than a functional subsystem or hardware oriented system. For example, a "navigation sub-system" in DAIS did not refer to a dedicated set of hardware and software which performed only the navigation function. DAIS architecture and elements were not dedicated to any one of several avionic functions, but used to perform many of the avionic functions with the avionic sensors and subsystems. The DAIS concept, therefore, proposed that the processing, information transfer, and the control and display functions be common and service the avionic application functional areas on an integrated basis. The DAIS system architecture consisted of hardware and software core elements and standard interfaces which could be structured for various avionics configurations. The architecture consisted of processors communicating with each other and other system elements (sensors, subsystems, weapon stations, controls and displays) through a standardized multiplex data bus. Application-independent executive software allowed adaptability of the architecture as well as performing system control. Additional key support software elements provided the tools for development, integration, and test of the core elements for a specific application. The basic attributes of the DAIS system which supported these system characteristics include flexibility, ease of use, ease of modification, and portability.

SYSTEM ARCHITECTURE OVERVIEW

The DAIS architecture consisted of federated processors communicating with each other and the other system elements (sensors, weapons, and controls and displays) through a standardized multiplex data bus. This standardized multiplex data bus provided dual redundant information paths between the system resources (each computer and other system elements). Centralized system single-point control was performed by a processor resident software executive that could be relocated for redundancy. Application software was structured to provide modularity, reliability, and transferability. This system architecture was flexible to accommodate a wide variety of avionics configurations, missions, and sensors, and provide redundancy to improve availability and accommodate changes in technology.

The basic elements of the DAIS architecture which can be restructured for various aircraft avionic configurations were called DAIS core elements (or building blocks) and were composed of the DAIS multiplex (bus controllers, remote terminals, and data buses), DAIS processors (with an associated memory), DAIS mission software, and DAIS controls and displays as shown in Figure 1. Additional elements were the support software elements, namely the JOVIAL Compiler (J73), and Partitioning, Analyzing and Linkage Editing Facility (PALEFAC). Sensors, weapons, and other subsystems were selected as required for the particular mission and connected to the interface modules of the Remote Terminals of the multiplex system or connected directly to the multiplex bus if the subsystem was compatible with the bus protocol.

The mission software or Operational Flight Program (OFP) resided in the memory of the DAIS processors. The mission software was structured in a modular form to allow easy mission-to-mission sensor/weapons changes, provide flexibility for major modifications, and provide transferability of portions of the software to other aircraft applications. The OFP was partitioned into the executive software and the application software. The latter was modularly separable into the mission avionic functional capabilities such as navigation, guidance, weapon delivery, communication, vehicle defense, target track and acquisition, subsystem management, stores management, autopilot, and pilot interface. The OFP application software also supported the on-board functional test capability. In addition, each processor contained a ROM program to control the start-up (cold start) and load of the OFP from the system mass memory.

INFORMATION TRANSFER SYSTEM

The DAIS multiplex system provided information transfer between the elements within the system, including DAIS processors, controls and displays, and other subsystems. It consisted of the bus controller interface unit (BCIU), remote terminal units (RT), and the multiplex cable assembly (data bus). The elements all were designed in conformance with what was later to be MIL-STD-1553B.

The system was a time division multiplex (TDM) data bus with a command/response protocol having one BCIU controlling the bus traffic at any given time. Each multiplex cable assembly consisted of a twisted, shielded wire pair. The information transfer on the bus consisted of messages composed of command, data, and status words. The information transfer consisted of three modes: bus controller to terminal, terminal to controller, and terminal to terminal transfer.

The BCIU was the interface between a processor and two data buses and operated in either of two modes: master or remote. In master mode, the BCIU operated under control of the Master Executive, issued all bus commands, and received all status words. In the remote mode, the BCIU monitored both buses for command words and responded to valid commands containing its own address, provided transfer of data in both directions between the processor and either of two data buses, provided status replies on the appropriate data bus in response to command (and special internal operations), and interrupted the associated processor upon receipt of certain mode commands or detection of exception conditions.

The remote terminal (RT) provided the interface between the subsystem and the two data buses. The RT transferred data in both directions between the multiplex bus and subsystem via the interface modules, based upon commands received from either data bus, and provided status replies on the appropriate bus in response to commands. The RT performed special operations upon receipt of a mode command. The RT was composed of a Multiplex Terminal Unit (MTU), Timing and Control Unit (TCU), and Interface Modules (IM). The Multiplex Terminal Unit (MTU) interfaced to the data bus, and transmitted and received information between the data bus and the Timing and Control Unit (TCU), under control of the TCU. The Timing and Control Unit (TCU) performed all of the timing, control, buffering, decoding and checking required to receive information from the data bus and transfer that information as outputs from the RT, as well as to accept inputs to the RT. The Interface Modules (IM) converted the data to the appropriate interface signal required by the subsystem. Any RT was capable of handling several signal interface types in various mixes and numbers.

PROCESSING SYSTEM

The DAIS processors were general purpose digital computers of the type normally referred to as mini-computers. They were specially engineered for airborne use and implemented the MIL-STD-1750 instruction set architecture (ISA). Operational features included a vectored priority interrupt system interval timers, and floating point arithmetic. All memory was directly addressable. Indexing as well as single level indirect addressing was available. A small read only memory (ROM) was enabled during the start-up sequence. A separate port to memory was provided for the BCIU via a direct memory access (DMA) channel. In a federated processor configuration each processor/BCIU could only address its own memory unit. Functionally, the BCIU was viewed as an extension of the processors input/output capability, very similar to the high speed DMA channels available on many mini-computers. The integration of the BCIU into the processor box was later accomplished with no external functional changes. Other elements attached to the bus saw no difference in the operating characteristics of this integrated unit as compared to the separate processor and BCIU.

EXECUTIVE SOFTWARE

The executive was organized into the master executive and the local executive. Each processor contained a local executive; the master processor also contained the master executive.

a. Master Executive - The master executive software performed the following functions:

1. Bus Control - Allocated time segments on data bus for synchronous communication and for asynchronous messages.

2. Systems Error Management - Monitored and analyzed errors relative to the operation of the processors and data bus communications, and provided control for error recovery.

3. Configuration Management - Initialized multiple computer system at start-up and after severe system errors.

4. Mass Memory Management - Provided for the retrieval of information from mass memory.

5. Monitor Management - Provided for monitoring of the master processor by the monitor processor.

b. Local Executive - The local executive software performed the following functions.

1. Task State Control - Used a task table to activate and deactivate periodic or non-periodic tasks when appropriate conditions were met. These conditions were based on a logical setting of real-time events.

2. Event Control - Used a table of real-time events to communicate conditions signalled between processes whether in the same or different processors.

3. Data Control - Guaranteed interlocks between shared data, provided mechanism for transmission and reception of data over the multiplex data bus.

4. CPU Fresh Start, Restart - Used to initialize CPU, to recover from transient failures, and to perform self-test.

CONTROLS AND DISPLAYS

The DAIS controls and displays were an integrated set of state-of-the-art control and display equipment incorporating the features of flexibility to accommodate changes in control and display equipment, redundancy where the CRT displays could serve as backup to each other, and an efficient design for system management by one pilot. The units of the DAIS controls and displays included:

Modular Programmable Display Generators
Display Switch/Memory Units
Integrated Multi-Function Keyboards
Data Entry Keyboards
Multi-Function Keyboards
Master Model Panel
Multi-Purpose Displays
Sensor Controller Unit
Head-Up Display
Armament Panel
Processor Control Panel

REDUNDANCY

DAIS provided redundancy in the system architecture which could be employed to provide backup and recovery to complete mission functions in spite of hardware failure. The areas which could be used for redundancy were:

a. Dual redundant data bus including redundant bus interface modules in the BCIU and RT.

b. System could employ dual redundant RTs for a specific subsystem.

c. Multi-Function Keyboard and associated Data Entry Keyboard could be used as a backup to the Integrated Multi-Function Keyboard and associated Data Entry Keyboard.

d. Raster displays could serve as backup to each other.

e. A processor/BCIU could be designated as monitor, serve as backup to the master processor/BCIU.

ARCHITECTURAL EVOLUTION

As a result of the DAIS program, standardization was actively pursued in the multiplex arena (MIL-STD-1553B), in the Higher Order Language field (MIL-STD-1589B) and in the avionics processor instruction set (MIL-STD-1750A). However, in order to meet the goals of true transportability (both hardware and software) it is necessary to pursue further standardization efforts in the areas of System level Control Procedures, Multiplex Bus Protocol, Executive Software Function and Application/Executive Software Interfaces. In addition, the advent of highly integrated subsystems and the evolving requirements for system level capabilities have indicated the need for advancements in the basic architecture. These requirements deal with multiple multiplex bus structures, increased system information processing and distribution capacities, improved fault detection, isolation and tolerance, and improved ability to provide system functions under degraded architectural conditions. In order to meet these objectives, the technologies in each area have been pursued (bus structures, processing, executive software, etc.), and the overall system architecture has been refined to accommodate those new capabilities. This is the subject for the decade of the eighties.

ERA OF THE 1980s

The decade of the eighties showed that use of advanced electronics has given modern combat aircraft phenomenal levels of performance, but at a stiff price in initial cost, maintenance workload and aircraft availability. Hence, aircraft design is shifting to give equal emphasis on performance, affordability, maintainability, and reliability in the development of avionic systems.

With the adoption of MIL-STD-1553B, Aircraft Internal Time Division Command/ Response Multiplex Data Bus, some of the newer U.S. Air Force strategic and tactical aircraft have begun the task of integrating beyond the subsystem level. However, even these newer system integration activities have addressed only the basic avionics functions of navigation, weapon delivery, communication, and to some extent, controls and displays. Analysis of a current tactical aircraft shows that the avionics suite consists of approximately 58 line replaceable units with 437 different sub-assembly spare types. The 1553 data bus and dedicated cables alone require 254 harnesses to connect the external line replaceable units. The combined number of internal and external interconnects explodes to more than 86,600 connections. Forty percent of the maintenance actions on the flight line are concerned with these cables and connectors.

Analysis shows that by applying high speed fiber optic multiplexing and Very High Speed Integrated Circuits (VHSIC) technologies, the number of cables and connectors can be reduced by as much as thirty-five percent. However, this does not imply that one can just insert these technologies and receive the improvements. One must structure or partition the technologies into a system architecture which will permit maintenance personnel to easily isolate faults and to replace failed avionics components. Also, mission analysis has shown that the system architecture must be flexible enough to support air-to-air and air-to-ground mission requirements. This leads one to conclude that the system architecture must adapt during the life time of the system to constant changes and new mission requirements.

It follows then that the hardware and software which form the system architecture must have a modular basis which will permit the maintenance personnel to remove and replace the components, yet allow system designers to adapt the avionics suite to new requirements. PAVE PILLAR was created to solve these problems.

PAVE PILLAR

The trend in avionics system design is to increase system integration thereby improving system accuracy, increasing system immunity to failures, and decreasing system reliance on multiple redundant sensors. This design philosophy which permits resources to be shared across subsystems requires a highly coupled system-wide management and control program (operating system) supported by a wide band data distribution network, high speed processors, and extensive mass memory. The ability to consider integration of this magnitude is dependent on recent advances in micro-electronics technology, most notably the advent of VHSIC technology.

The PAVE PILLAR avionics architecture is functionally divided into three distinct areas:

> Mission Management
> Sensor Management
> Vehicle Management

The three areas define the enclosing boundaries for resource sharing, sparing, and substitution. Unique characteristics of each of these areas preclude the utilization of resources across areas for the purpose of function recovery or reconfiguration. This does not imply that the areas do not contain many common hardware elements, but that the organization, connectivity, and control of these components restrict their practical use in functional system-wide reassignment. Each of the three areas have associated with them a logical processor type and varying interface requirements. Figure 2 illustrates the top-level organization of the PAVE PILLAR avionics architecture, system elements are built from a set of common modules supporting programmable processing, I/O, and memory storage functions. The interfaces between system elements are standard high speed time division multiplex buses and data links, all utilizing fiber optics technology.

MISSION MANAGEMENT AREA

The Mission Management area consists of Mission Data Processors, Mission Avionics Multiplex Bus, Block Transfer Multiplex Bus, system Mass Memory, Stores Management System, and a collection of interfaces to the Mission Avionics Bus. This area provides the resources to perform the mission and system management functions such as fire control, target acquisition, navigation management, defense management, stores management, TF/TA/OA functions, and crew station management. The Mission Management Area contains identically configured Mission Data Processors all connected to the Mission Avionics Multiplex Bus and loadable from the System Mass Memory via the Block Transfer Multiplex Bus. All Mission Processing control and data exchange outside of loading operations is performed using the Mission Avionics Multiplex Bus. The Mission Management Area controls job assignments for the Signal Processors and determines the connection paths from Signal Processors (such as target tracks, CNI reception results, threat descriptions) and moding and control data is provided back to both the Signal Processors and the sensor front ends.

The Mission Management Area also interfaces with the Vehicle Management Area to receive navigation state information and to supply route and trajectory commands. Other interfaces are to cockpit multi-function switches, Stores Management System, and miscellaneous avionics control devices not directly interfaced to the Mission Avionics Bus (helmet-mounted sight, voice recognition system, data recorders/readers). The Mission Management Area collects the health and status of all core elements and sensor/subsystem components for maintenance history and to maintain mission functional capability.

SENSOR MANAGEMENT AREA

The Sensor Management area consists of a set of common Signal Processors, a sensor data distribution network, a sensor control network, a data exchange network, and a video distribution system. The sensor management area provides the signal processing functions and interfaces necessary to convert conditioned data from multiple sensors via a sensor network into processed information suitable for distribution to other avionics systems. The sensor management area accepts encrypted data from a TRANSEC/COMSEC controller(s) and processes the data for transmission. This area also distributes
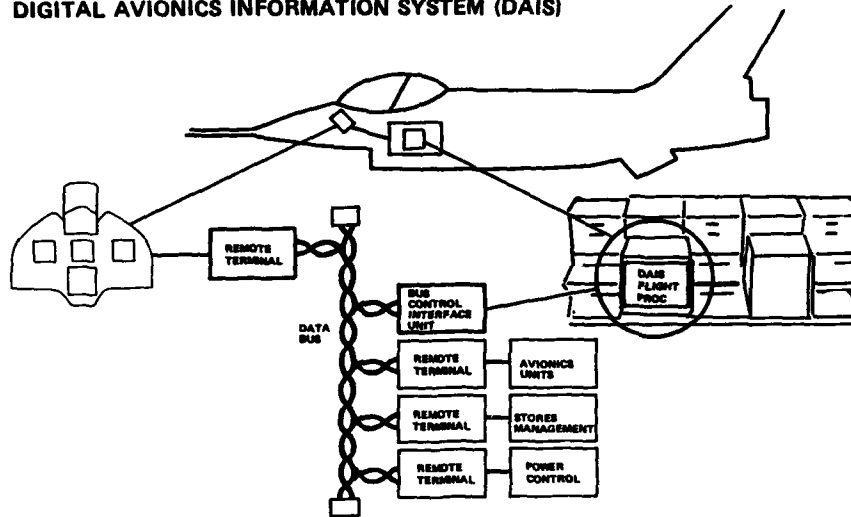
DIGITAL AVIONICS INFORMATION SYSTEM (DAIS)

Figure 1

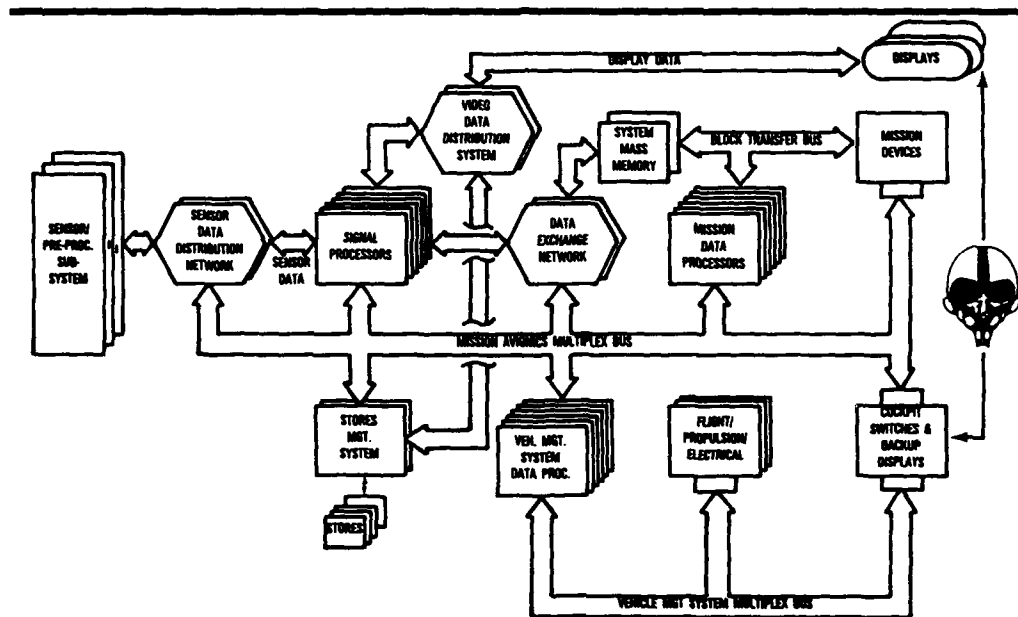# ADVANCED AVIONICS ARCHITECTURE

Figure 2

processed digital video to crew displays and distributes sensor control commands to sensors via a sensor control network. Signal Processor task assignment, sensor data distribution network control, and Signal Processor resource reconfiguration is managed by the Mission Data Processors.

VEHICLE MANAGEMENT AREA

The Vehicle Management System (VMS) is an independent subsystem supporting the fundamental flight and airframe related control functions. The VMS Management area is physically isolated from the rest of the architecture for safety of flight reasons and contains a higher degree of physical redundancy, usable only within the VMS area. The VMS management area contains VMS Data Processors, Controls/Displays interfaces, Flight Sensor/Actuator interfaces, Electrical Power Control interfaces, Engine Control interfaces, and Utility Systems interfaces. The VMS Data Processor is essentially identical to the Mission Data Processor except for memory configuration, multiplex bus interfaces, and reconfiguration methods. The VMS Data Processor has Read Only Memory for all program storage and does not perform any program loading. The VMS Data Processor also has six High Speed Bus interfaces. Two dual redundant interfaces to the Mission Avionics Multiplex Bus and four simultaneously active bus interfaces to the VMS Multiplex Bus. Each VMS Data Processor contains the entire program load to perform all VMS functions. Reconfiguration is accomplished by activating dormant tasks in available spare VMS Data Processor resources. The flight essential nature of VMS processing necessitates a very high degree of functional reliability (Fail-Op/Fail-Op/Fail-Safe). This is accomplished by physical quad redundancy of sensors, buses, processors, and actuators in the VMS processing area. The partitioning of processing functions among VMS Data Processors retains the quad redundant, simultaneously active characteristic of the VMS area.

SYSTEM CONTROL

Control of the core processing system is provided by a distributed software architecture providing for commonality of control software across the mission processors, flight control processors, and signal processors. Major control functions include:

- Initialization and system start-up and restart.

- Assignment of application software task to processing resources (software configuration and reconfiguration computing resources management).

- Sequencing and synchronization of related software tasks.

- Management of sensor and other device resources with respect to mission objectives, mode and task management, and software parameters.

- Interpretation of response to, and integration of, human control into the system functionality.

- Collection, maintenance, and reporting of system hardware and software status, and operational functionality.

- Response to hardware and software failure detection to preserve mission effectiveness.

- Flight control change management and response.

- Reintegration of recovered hardware or software functions.

- Assurance of a distributed data base consistency and integrity, management of access to that shared data.

- Preservation and collection of data required for continuity of system functionality across failure recovery points.

- Management of communications access to ensure optimal use of communication resources and correct addressing of data messages.

- Assurance of the security of classified data.

The operating system is partitioned into three elements: (1) the kernel executive which provides those functions common to all processors, (2) the distributed executive which provides for decentralized system control in each processor, (3) system executive which provides the monitoring of system state and reconfiguration based on mission requirements and detected system failures. Figure 3 depicts the interrelationship of the three elements.

OPERATIONAL CONCEPT

The PAVE PILLAR architectural concept was developed to support aircraft operations from deployed locations with a minimum of support. This architecture supports the resource sharing of core data and signal processing resources and is constructed of a
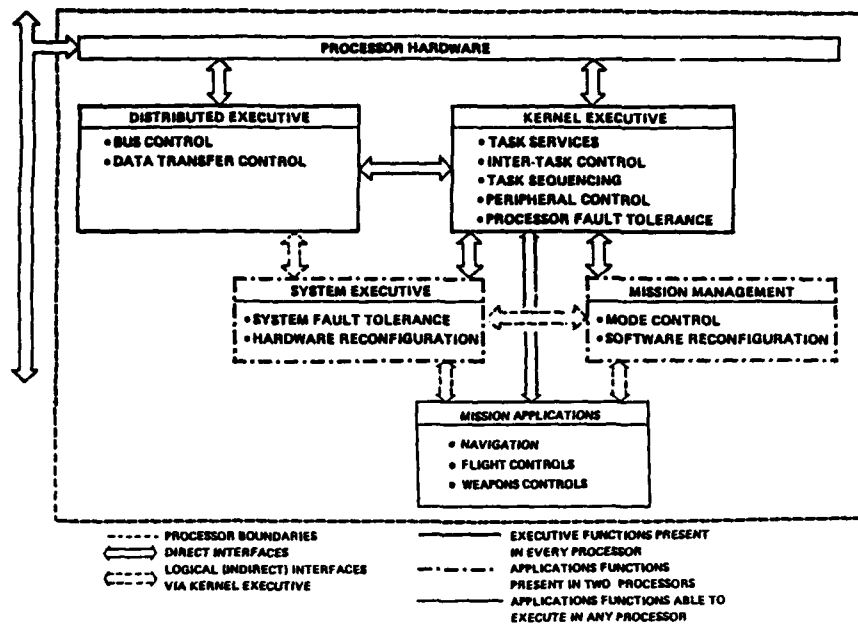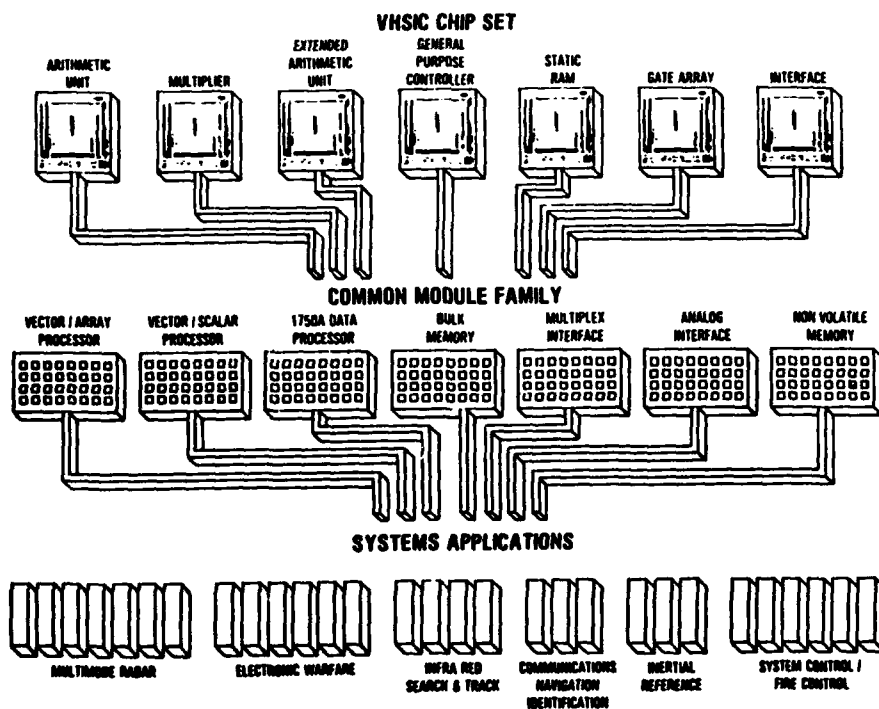
FUNCTIONAL PARTITIONING OF SYSTEM SOFTWARE

Figure 3

VHSIC CHIP SET

COMMON MODULE FAMILY

SYSTEMS APPLICATIONS

Figure 4

set of common modules that specifically support a two-level maintenance *concept*. This architecture supports high degrees of systems availability and reliability. This is accomplished through the application of spare signal and data processing resources at the system level so that backup services are provided when the primary sources fail. In addition, the architecture supports graceful degradation in that when spare resources are exhausted remaining resources can be assigned to the highest priority functions on a *mission basis*.

COMMON MODULES

The PAVE PILLAR architecture is physically comprised of a number of "building blocks" called *common* modules. A common module is sized to contain the circuitry to perform a complete digital processing function including interface control and health diagnosis. The approach for PAVE PILLAR is to *develop common modules* from a limited VHSIC chip set and then develop systems/subsystems which utilize the common modules. The exact composition of the VHSIC chip set will depend on the evolution of that technology program,while the members of the common module family will be subject to PAVE PILLAR avionic system design considerations.

A number of common modules can be built up from a family of VHSIC chips which, in turn, can be grouped to form the basis for any one of the avionics subsystems depicted in Figure 4. Certain *non-common* modules will undoubtedly be required for some specific subsystem implementation; however, the reduction in numbers of spares types required as a result of common module usage will provide a significant cost and effectiveness improvement.

The Avionics Laboratory has undertaken the design and development of two common module sets: the VHSIC 1750A Data Processor and the VHSIC Common Signal Processor. These modules are being designed with stringent requirements for both extremely low failure rates and high fault detection and fault isolation capabilities. Fault isolation to the single module level will be accomplished by on-board Built-In-Test (BIT) circuitry at the chip level and multi-tiered self test software. The use of a modular concept will permit the maintenance personnel to perform on-board diagnosis and replacement of the avionics at the module level with no auxiliary ground equipment.

This capability then leads one to conclude that a two-level avionics maintenance concept might be realizeable. Figure 5 shows the impact of various maintenance concepts. The current three-level maintenance approach consists of flight line, Avionics Intermediate Shop, and depot/factory diagnosis and repair. To illustrate the benefits of changing from a three-level to a two-level maintenance concept, an in-house study to examine the relative life cycle cost of several combinations of maintenance concepts and technology integration was conducted. As shown here, the costs consist of operations and support plus the Avionics Intermediate Shop and spare parts to support 1000 aircraft for 20 years at a flying rate of 300 flight hours per aircraft per year. All costs are stated in 1984 dollars. The first column r sents today's technology - F-16 A/B - utilizing the standard three-level maintenance with removal/replacement of LRUs at the flight line. In column B, the three-level maintenance concept is retained and LRUs are still used, but VHSIC technology has been incorporated and a fault tolerant design through extensive built-in-test has been introduced. These steps reduce our cost for operating and maintaining the 1000 aircraft fleet by nearly 20%. In the next step, where the Avionics Intermediate Shop is eliminated and everything else is kept the same as B, there is a further reduction in cost of over $200M. Finally, introduction of standard modules provides a further reduction in cost to approximately one-half of the current ownership bill. While these numbers are not hard and fast they do provide an indication of the sizeable cost gains which can be realized through the application of PAVE PILLAR technologies.

TECHNOLOGY TRANSPARENCY

The building block approach espoused in this section permits not only the initial development of a highly flexible avionics suite, but also the continued development and integration of Pre-Planned Product Improvement (P^3I). This is accomplished within the architecture by the concept of standard data buses (Parallel Interface (PI), Test and Maintenance (TM), High Speed Data Bus (HSDB)) and networks (SDDN, VDDN, DFN, Data Flow Network). This concept is implemented by Form, Fit, Function and Interface (F^3I), specifications for each module type thereby permitting different designs by vendor for each module type and specific vendor module design modification dependent upon technological improvements.

Due to the open architecture, as new building blocks are identified they can be readily integrated into the avionics system, thus permitting the performance upgrade to an existing aircraft at a relatively low cost. New investigations have already begun in the areas of parallel processing, artificial intelligence processing, and optical processing. The intent is to integrate these advanced processing elements into the PAVE PILLAR architecture if/when they become viable both operationally and logistically.

In summary, a PAVE PILLAR avionics architecture will result in dramatic improvements in availability, mission effectiveness, and cost of ownership. For reasons of mobility, logistics cost control, and austere maintenance of aircraft at forward sites, a two-level maintenance concept with line replaceable modules is endorsed by the

AVIONICS COST IMPACT OF MAINTENANCE CONCEPTS
(PRELIMINARY RESULTS)

O&S + AIS+ SPARES COSTS

RELATIVE
COST (%)

100%
1446M

81%
1174M

64%
931M

53%
770M

1000 AIRCRAFT
20 YEARS 84$
300 FH/YR

A B C D

COST SAVINGS

A → CONVENTIONAL F-16 A/B, 3 LEVEL/LRU — —

B → VHSIC, FAULT TOLERANCE, BIT, 3 LEVEL/LRU — $272M

C → 2 LEVEL/LRU — $515M

D → STANDARDS MODULES, 2 LEVEL/LRM — $676M

Figure 5

AIR FORCE SOFTWARE TRENDS

COST OF AIR FORCE MISSION CRITICAL COMPUTER RESOURCES*

$ BILLIONS — 5, 7.2, 9.9, 12.3, 14.5

TIME 85 87 89 91 93 95

- SOFTWARE IS THE TROUBLE IN 7 OUT OF 10 TROUBLED SYSTEMS**
- SHORTFALL OF SOFTWARE PERSONNEL
  - DEMAND – 12% /YEAR
  - PRODUCTIVITY ~ 4% /YEAR
  - PERSONNEL ~ 4% /YEAR
- COMPLEX, INTEGRATED SOLUTION REQUIRED
  - MANAGEMENT
  - TRAINING
  - TECHNOLOGY
- CULTURE MUST CHANGE TO MEET CHALLENGE

BREAKDOWN OF MCCR COSTS**

PERCENT OF COST

HARDWARE COST

SOFTWARE DEVELOPMENT

SOFTWARE SUPPORT

TIME 1965 1975 1985

* EIA DEFENSE ELECTRONICS MARKET TEN YEAR FORECAST

** HQ AFSC/PLR "BOLD STROKE" BRIEFING
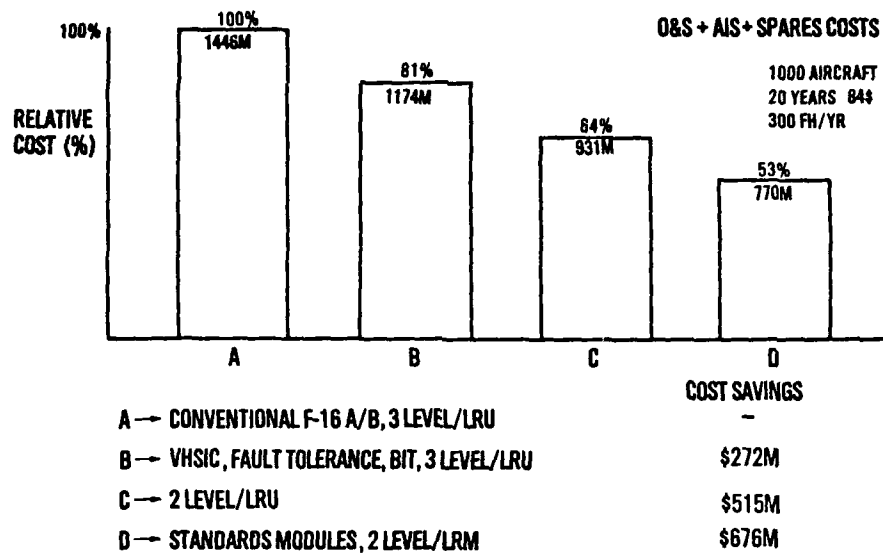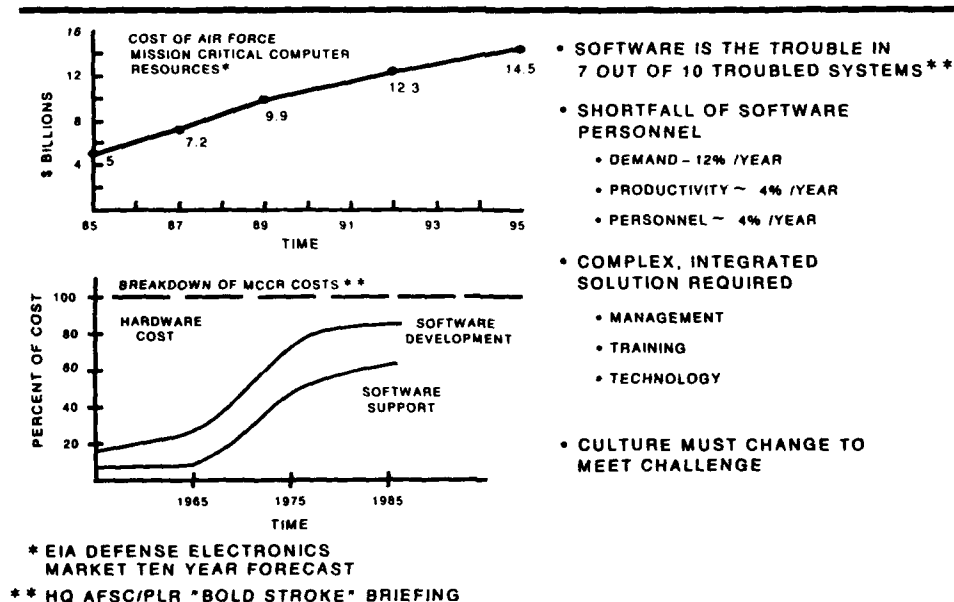
Figure 6

PAVE PILLAR system. PAVE PILLAR plans to achieve these benefits and goals through advances in technology and a systems approach to avionics integration. PAVE PILLAR is expected to provide the generic integration approach and architecture that will act as the foundation for avionics development in next-generation Air Force aircraft.

ERA OF THE 1990s

The decade of the nineties brings with it an avionics technology base of historic proportions which is currently being sponsored by government R&D agencies. For example, work in microelectronics is yielding processor chips that can be driven at a 100 megahertz clock rate. Sophisticated parallel processor networks are being developed in an attempt to achieve even higher processor speeds. Wafer scale integrated circuits have been built which could potentially reduce the size of large parallel processor networks to hand-held units. Researchers are also developing robust AI algorithms for the first time which will assist aircrews in decision making. Neural network theory and actual device construction is making rapid progress and adaptive learning systems on aircraft can now be realistically projected.

Advanced packaging and cooling technologies are under development which will allow us to dramatically reduce circuit connections and improve thermal management to enhance reliability. Also, powerful new CAD/CAM capabilities are under development that will revolutionize how RF and digital hardware will be designed and manufactured and how avionics software will be designed, developed and supported in an affordable manner.

If the above set of technologies are properly exploited, matured and integrated during the 1990s, 21st century avionics will undergo a change that will be as dramatic as the introduction of the transistor. The PAVE PACE initiative is being planned to affect these changes.

With the 21st century being only a decade away, we now should be able to begin performing forecasts to answer the following questions relating to avionics systems capabilities in that time frame: (1) What new performance capabilities will early 21st century avionics enable relative to improving situation awareness and crew workload reduction?; (2) Will we be able to finally make significant improvements in avionics system reliability? Is there hope that new technology developed during the 90s will allow scheduled maintenance for avionics? Will we be able to build avionics that is expected to have a 30-day operation with minimal repair and replacement?; (3) Will we be able to finally curtail the escalating cost spiral for avionics, particularly development and support cost?; (4) What new forms of functional integration can we expect to see as a result of the new technologies and the need to fuse information?; and, finally, (5) What types of avionic system architectures will be needed to support strides in reliability, performance and cost savings?

FORCES THAT WILL SHAPE THE 21ST CENTURY AVIONICS

The characteristics of early 21st century avionics will be shaped by irresistible forces operating on available technologies. Implications derived from how these forces and constraints will generally shape future avionic requirements will be made.

DESIRED OPERATIONAL CAPABILITIES

As scenarios become more complex, the quality of time-compressed decisions by the aircrew can be expected to decrease, while the number of required actions will likely increase unless proper steps are taken. Clearly, more information derived from sensors and weapon status information will be required and much larger amounts of external "C³I" data will come into the cockpit from other sources. Given that the basic skeletal framework for acquiring and distributing the data is being put in place (i.e. PAVE PILLAR), the next step is to develop the technology that translates the artificial world of sensors into an easily understood world that can be exploited by our human abilities. This "avionic technology missing link" is viewed as pivotal in determining the future of avionics. We must develop means to achieve full spherical situation awareness (SA) of threats, targets and terrain in a volume of space and time of interest to the aircrew. We must enable selected automation that allows the crew's intent to be expressed by direct, simple actions. Further, we must implement "simple to use" avionics in extremely complex scenarios that involve coordinated intra-flight netting of resources for both offensive and defensive tactics.

In order to affect needed information presentation and control improvements, several new technologies will be required. These include: (1) large full color high brightness, displays for "Big Picture" presentation of the situation; (2) extremely high speed graphics processors (tens of MOPS) will be required to present real-time panoramic graphical information for head-down and helmet mounted displays; (3) the use of real-time AI algorithms (and ultimately neural network implementations) will be needed to determine how information is to be presented, provide tactics recommendations, system health status, assist in mission replanning, sensor management, determine pilot intent, etc. High speed processing in the order of a few MOPs ("simple" AI applications) to a few BOPs (complex cooperating expert system) will be needed; (4) advanced signal processing techniques to enable continued operation across the RF spectrum in jamming environments will require extremely high speed processing which could reach a few BOPs; (5) "Smart" sensors are needed to detect and identify targets and threats. Again, several BOPs will be required; (6) selectable automation of integrated vehicle and

weapon control functions will be necessary in many situations to allow pilots to
concentrate on high level management decisions and communicate with support forces. For
example, the selected integration and automation of flight control, fire control, engine
control, thrust vector control and weapons control may be highly beneficial in aerial
combat, missile evasion and low altitude maneuvering-flight profiles. This integrated,
automated system (which also would likely employ AI algorithms) would again require high
speed, fault tolerant processing of the order of several tens to hundreds of MOPs; and
(7) finally, robotic air vehicles (RAVs) will likely make their presence known in a
significant way during the early 21st century to complement manned vehicles. The
authors envision these vehicles to serve as a force multiplier in extremely hazardous
scenarios across the range of tactical missions. Since RAVs represent the need for
massive automation, including the robust use of AI, most of the above comments relating
to the high speed processing apply here.

In summary, desired operational capabilities for the early 21st century require
extraordinarily high speed processing. These speeds are generally not achievable by one
or even a few of the fastest processors we will have available. We will need to utilize
parallel networks to achieve these speeds. We will need to find a way to solve several
problems before the capabilities cited in the above discussion become reality. Some of
these problems include: (1) achieving an affordable approach to the development and
support of massive amounts of numeric and symbolic software; and (2) packaging the
parallel networks into aircraft compatible dimensions while achieving affordable
hardware configurations that are reliable and fault tolerant. As challenging as these
problems appear to be, there are technologies under development that appear to offer
workable solutions.

COST CONSTRAINTS

Avionics usually represent about one-third the fly-away cost of a fighter and
usually a higher fraction of the maintenance cost. As missions become more complex and
as more robust suites of avionics are added, these fractions will likely increase unless
fundamental processes are reversed. Cost containment and even cost reduction of
avionics is expected to become a significant issue during the 1990s. It is increasingly
apparent that expensive avionics reduce the number of aircraft that can be afforded.

Avionics cost is justified if commensurate improvements in survivability and force
multiplication are realized. Over the past two decades, we have seen dramatic,
cost-effective improvements in weapon system flexibility, capability and survivability
due to the use of avionics, particularly programmable computers. These strides were
enabled by impressive increases in hardware speed and significant decreases in processor
and memory cost. Because the use of programmable computers is so pervasive, the
escalating cost of software design, development and support has become a central problem
in cost containment for the entire weapon system.

A software cost crisis is building. If it is not contained, fewer capable aircraft
will be built or we will have less capable aircraft. Strides in software productivity
have not begun to keep pace with hardware strides despite significant progress in
software standards and tools.

For example, Figure 6 shows the dramatic growth of software costs for mission
critical Air Force computer resources. Some observations are: (1) through 1985, most
computing hardware in the force structure consisted of data processors. The steady
percentage decrease in the cost of data processing hardware, in spite of its more robust
use on aircraft, can be seen; and (2) rising support software costs through 1985 was
caused by several factors: (a) a bow-wave effect where an increasing number of software
systems must still be supported (machine code, assembly code and HOL); (b) standard-
ization of languages and instruction sets have helped slow down the rate of cost growth
but has not been in place long enough to dramatically effect overall costs; and (c)
software design and development productivity has not kept pace with the opportunities
(and temptations) afforded by hardware strides.

Although the past decade has witnessed significant cost growth in data processing
software, new factors are at work which could further escalate software cost growth.

These new factors include: (1) steady movement towards automating aircraft
functions to relieve crew workload and stress, thereby increasing demands on data
processing software; (2) the growth of programmable signal processing, with an immense
software burden, is becoming necessary to meet complex threat environments; (3) a trend
towards functional integration on aircraft is resulting in a dramatic increase in
programmable data and signal processors; and (4) during the 1990s, real-time machine
intelligence should be demonstrable in the laboratory in flyable configurations.

Processors executing billions of operations per second may be flying by the year
2000. Exceedingly expensive software could result. Advanced software support tools
will be needed to help develop software for parallel processors. Advanced AI, image
processing and signal processing algorithms will be massively large and complex.

There are several factors that must be considered in containing processing hardware
and software cost: (1) we must adhere to standards that allow technology growth,
maximize competition and promote reusability of designs, design tools and support
hardware and software (i.e. computer instruction sets, HOLs, compilers, etc.); and (2)

we must stress the replicated use of common hardware modules across the entire force structure, with the ultimate goal of having common modules across DOD avionics, (3) we must make fundamental changes in the way software is designed, developed and supported.

MANPOWER CONSTRAINTS

The availability of skilled maintenance personnel in the Air Force is another factor which will drive early 21st Century avionics. An Air Force Studies Board report investigating the isolation of faults in Air Force weapon systems concluded that there would be a growing demand for competent Air Force electronics technicians due to increasingly complex avionics, increased demand by the private sector and a shrinking manpower pool among 18-24 year old people in the U.S. One conclusion of the report was that we must move toward decreasing functional specialization at the flight line. In the authors' opinion, manpower constraints will accelerate the use of a standard line replaceable avionic modules which has been espoused by the PAVE PILLAR program. In addition to modules for data and signal processing, various data bus and I/O interface and power supply modules, the module concept will be extended to encompass additional electronic functions include RF circuitry, displays, gyros, etc. (However, with different packaging and cooling designs.)

AVIONICS SUPPORT ENVIRONMENT FORCES AND CONSTRAINTS

The projected support environment for Air Force Avionics around the year 2000 will force fundamental changes in electronics design, packaging and testing. The "Air Force 2000 R&M" report portrays an immensely challenging situation for the avionics support community. The document states that vulnerability of our entire weapon system support structure to hostile action will be the most critical support issue by 2000. The report emphasizes the need to plan for austere support conditions, Dramatically improved avionics reliability and simplified maintainability with reduced flight line personnel are required, thereby reducing dependencies on the supply pipeline. The author believes that the "R&M 2000" scenario implies that the following characteristics are needed for 21st century avionics: (1) we must extend the use of modular electronics across a wide spectrum of avionics applications; (2) the number of different module types must be kept to a minimum to allow a full complement of spares to be carried on a small ground-based vehicle; (3) highly reliable on-board testing of circuitry will be mandatory. Pervasive use of BIT/SIT with AI programs is expected, along with more extensive use of fault tolerance; the concept of deferred or scheduled maintenance for avionics will need to be accomplished, where graceful degradation concepts are built into virtually every module; and (4) the reliability of avionics must be improved in a revolutionary manner.

SUMMARY OF DESIRED 21ST CENTURY AVIONIC CHARACTERISTICS

It is the authors' views that four fundamental (i.e. non-evolutionary) changes or developments will be required to meet the forces and constraints that will drive 21st century avionics. These changes, along with proposed goals are:

(1) Pervasive use of machine intelligence across virtually all electronic functions ranging from "smart sensors" to cooperating expert systems.

(2) Parallel processor networks which will execute numerical and heuristic algorithms at speeds 10-100 times faster than the fastest available uniprocessors.

(3) Software design and development tools that will allow productivity to improve by at least a factor of ten over current practices.

(4) The use of advanced materials, packaging and cooling techniques, along with fault tolerant techniques that will enable electronics to operate for say a thirty day period with minimal flight time support.

Two evolutionary trends in avionics reflecting a continuation of recent concepts are also seen. They are:

(1) an expanded use of a standard family of line replaceable modules across a broader spectrum of avionic applications to enable flight line maintenance under austere conditions.

(2) an acceleration of functional integration and sensor fusion design concepts across a broad array of electronics.

ADVANCED ENABLING TECHNOLOGIES FOR 21st CENTURY AVIONICS

PARALLEL PROCESSING

Many force multiplier improvements in smart sensors, vehicle control, situation awareness and crew decision aiding will be made possible if affordable, flyable supercomputers and associated software can be developed for next-generation military aircraft. New functional architectures will emerge because dramatic improvements in processing speed can be implemented through tightly coupled networks. Unconstrained system architectures can be developed where the system designer will have the capability to fuse together needed logical functions irrespective of previous boundaries.

Several types of parallel processor network architectures have been constructed under DARPA sponsorship as well as through private ventures. Researchers are programming these machines to develop an understanding of how best to match the algorithmic features of the application to specific architecture features in order to maximize computer network speed up.

The single most important issue in parallel processing is the equivalent speed up gained by the use of "n" processors. In general, a speed up of "n" cannot be expected since real world problems are not perfectly parallel. If α is defined as the fraction of the algorithm that can be parallelized, the relation between speed up achieved by n processors and α is shown in Figure 7. Figure 7 points out a fundamental property of parallel computation: unless α is greater than 0.9, parallel networks will not provide a significant speed increase. It is imperative that the algorithm be investigated for parallelism before proceeding to parallel network implementation. Since the slope of the curves in Figure 7 vary as a quadratic relationship with n as α approaches 1, new algorithms emphasizing parallelism and/or the use of compiler tools that reformat an existing algorithm into its most robust parallel form will be mandatory. Given that an algorithm lends itself to parallelism, then its significantly parallel portions must be efficiently mapped onto the proper network architecture if "real-time" processing of complex algorithms is to be achieved. The architecture selected must exhibit: (a) the proper "grain size" for each processing element; (b) highly efficient communications between processing elements; (c) high speed memory accessibility (particularly for large sized applications); (d) good load balancing for the processing tasks over time; and (e) real-time, fault-tolerant operation.

Dozens of different parallel network architectures are conceivable, each having different memory access approaches, communication schemes, network control schemes, node size (i.e. processor/memory pair), etc. A thorough description of the range of choices is beyond the scope of this paper. The interested reader should consult References 3-7 for brief descriptions of some of the more popular parallel architectures. Both multiple instruction multiple data (MIMD) and simple instruction multiple data (SIMD) network control schemes have been built. SIMD-controlled networks have single, broadcasted instruction streams to all nodes during a given cycle. A third general class of networks is the systolic array which is a SIMD machine that pipelines computations between several nodes in a serial fashion.

Figure 8 shows a general comparison of various network architectures. Note that a bus structured architecture, such as PAVE PILLAR is excellent for interfunctional integration where bus latencies do not present problems. Although this is a parallel architecture, most parallel nets will consist of switch-based circuits. These "domain processor networks" will be connected to the PAVE PILLAR bus. Such a concept of hierarchical or tiered system networks is perfectly compatible with PAVE PILLAR.

The cost effective choice of using a SIMD or MIMD network architecture depends on the characteristics of the application.

In general, the SIMD-based network is preferable when the algorithm is well structured, and has a regular pattern of control for the cooperating nodes. If the algorithm displays these properties, the network control hardware that issues the highly synchronized single instruction stream can be shared by all nodes, which also will result in simpler software programming. Example algorithms that generally possess these properties include matrix calculations, certain types of artificial intelligence applications (e.g. semantic networks), image processing and signal processing.

A MIMD network architecture will likely be preferred if the algorithm control flow is highly complex and data dependent. For such algorithms, much of the SIMD network nodes would sit idle much of the time waiting for the proper instruction.

Example MIMD-oriented algorithms include certain types of artificial intelligence applications (e.g. tactics planning, mission planning, system health management, situation assessment) and executive/system control of complex processing sites.

It must also be observed that a mixture of the SIMD and MIMD architectures may be the preferred design for some applications.

AIRBORNE PACKAGING OPPORTUNITIES

Use of VLSI circuitry applied to wafer scale integration and hybrid wafer integration, will ultimately lead to three-dimensional, stacked wafer computers to dramatically reduce weight, volume and size. Techniques for on-surface interconnection of processing microcircuit cells are being developed and techniques for interconnection of stacked wafers in a "3-D" array are under development.

The resulting "3-D computer" will consist of about 90% silicon (as compared with 1-15% with today's military processors). Reliability is substantially enhanced by deleting the use of printed wiring boards with its thousands of chip-to-board solder connections. Also, total system speed of the multi-cell wafer is enhanced due to the absence of high capacitance wires which inhibit chip-to-chip high speed data transfer.

(FRACTION OF PARALLELIZED ALGORITHM)
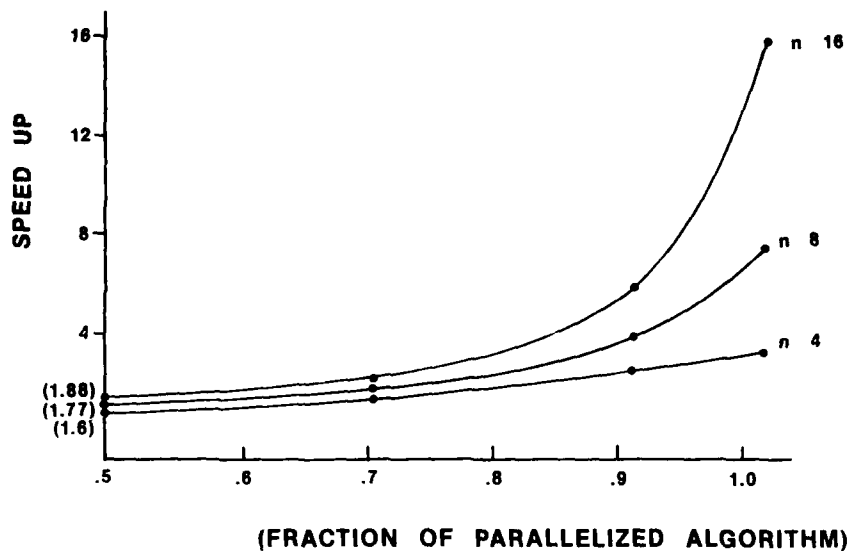
SPEED UP FACTORS FOR PARALLEL NETWORKS RELATIVE TO FRACTION OF PARALLELIZED ALGORITHM [ ]

Figure 7

PROCESSING ARCHITECTURE

Figure 8

The early 21st century designer can seriously contemplate wafer-based designs. Simply stated, a hand-held (e.g. 10 cm x 5 cm) parallel processor capable of executing billions of operations per second is reasonable by 1995.
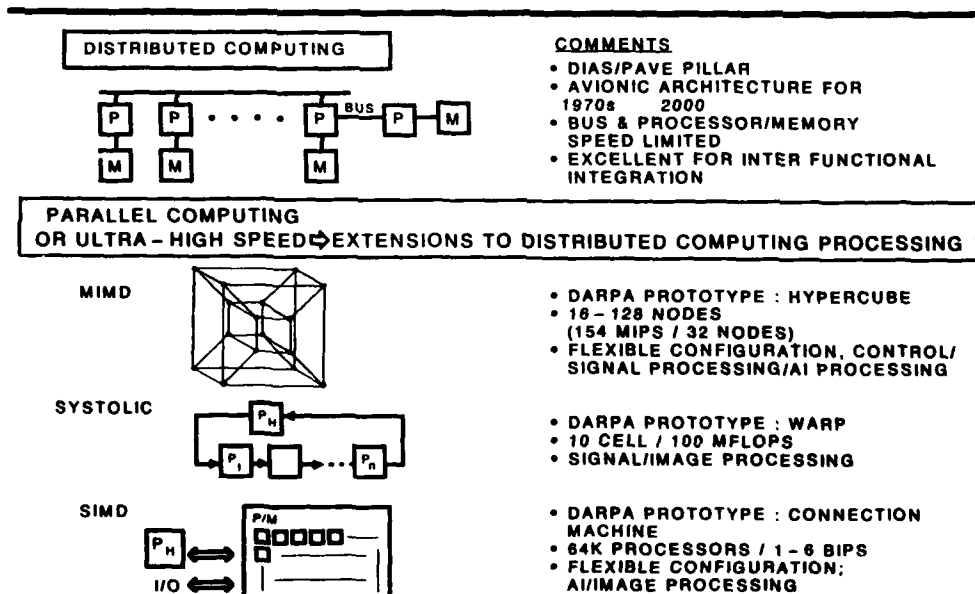
In achieving a "hand-held supercomputer," a few basic caveats must be stated. First, there is the question of achieving high yield of the cells across the wafer. The minimum feature size (and hence speed) of the separate cells and whether a monolithic or hybrid approach should be used must be determined. Secondly, and always very important, is the algorithm-to-architecture matching problem. There are two basic design choices: (1) use customized, stacked wafer processors having unique wafers for every high speed parallel network application on the aircraft, or (2) use a family of wafer modules, and replicate their common use across various applications.

CAN AVIONICS RELIABILITY BE DRAMATICALLY IMPROVED?

Achieving high avionics reliability will require combined use of (at least) the following:

(1)   carefully controlled manufacturing processes with extensive testing

(2)   avoidance of high voltage and mechanical componentry

(3)   improved thermal management to not only prohibit high-end temperatures from being reached, but temperatures cycling needs to be kept to a minimum.

(4)   a significant reduction in electrical connections between chip to printed wiring board (PWB), PWB to backplane and backplane to cabling/ data buses.

(5)   robust use of fault tolerance for mission critical functions.

Many exciting technology developments are responding to item (2) above. These include the use of laser gyros, phased-arrays and solid state RF circuitry. The accelerated use of RF micro-electronics will result from the tri-service MIMIC program.

Item (3) above requires the application of advanced technologies which, although they are yet to be proven across all of avionics, offer significant promise. These technologies include: improved means to reduce thermal resistance and stabilize temperature excursions through the use of direct immersion of electronics in liquids, phase-change cooling and use of heat pipes.

Items (4) and (5) can be accomplished by the use of new microelectronic packaging technology and the use of photonics. Electronics constructed at the wafer level or using hybrid surface mount techniques appears to be required. Interconnecting cells reliably, using surface-mount approaches, will be the largest challenge to wafer scale integration. Figure 9 shows a very preliminary sketch of an avionics module that combines the features of liquid cooling and WSI, along with estimates associated with module parameters. Although much work remains to be done in determining how the packaging, cooling and interconnections are to be accomplished most reliably, the significant observation is that the face of avionics packaging may substantially change by the early 21st century.

NEW AVIONICS DESIGN & DEVELOPMENT CAPABILITIES

In order to affordably design, develop, test and support electronic systems of the future, fundamental changes must be made. The resulting new design and development era for avionics will be dominated by the use of CAD/CAM technologies which will enable simulation, design, development, testing, and documentation from the system to the component level.

Figure 10 shows an example of the general flow of this highly automated process. Once requirements are generated, top level system network simulations will be accomplished to establish overall functional partitioning and proper system control operation. Software tools will then be used to ensure the proper hardware/software match for all processing resources, including parallel networks. These tools include simulators, compilers, linkers, assemblers and code generators.

Software application program development and support will be approached with the same hardware-oriented philosophy of using CAD tools and reusable modules. Here, libraries of reusable application software modules are tailored to fit specific application needs (these modules will be designed to have extensive modularity and I/O flexibility). In order to further enhance programmer productivity, "fourth-generation" programming techniques will be employed. Figure 11 shows the relative impact of various factors that drive software development cost and points out the need for fourth generation language programming. Note that whereas the programming team's experience with the language, the size of the data base and other factors are very important cost drivers, the numbers of source instructions is the dominant cost driver. Therefore, cost will be substantially reduced if software modules can be reused or modified in a simple manner, and if custom software can be developed with a minimum of source instructions.

PROCESSING IMPACT OF TECHNOLOGY INTEGRATION

FIBER OPTIC PINS

316 ELECTRICAL PINS

STATE-OF-THE-ART FOR 1990s
MODULAR PACKAGING
(VHSIC CHIPS)

HEAT EXCHANGE RIB

EARLY 21ST CENTURY
MODULAR PACKAGING
(VHSIC WSI)

4 F.O. PINS

COOLANT
QUICK-DISCONNECTS

AIR COOLING / RIB HEAT EXCHANGE
MODULE SIZE: 6.40" X 5.88" X 0.58"
POWER: 27 WATTS
WEIGHT: 1.5 LBS
DOUBLE SIDED
  # BONDED
CONNECTIONS: ≈ 2500
  CHIPS: 40-50
RELIABILITY: ≈ 10000 HRS
(55°C RIB TEMP)
CLOCK RATE: 20 MHz
3 MIPS / 256 KW RAM
(DAIS MIX)

FLOW THROUGH IMMERSION
MODULE SIZE: 6.40" X 5.88" X 0.58"
  POWER: 10 - 400+ WATTS
  WEIGHT: 1 LB
MULTIPLE WAFERS (2-15)
  # BONDED
CONNECTIONS: 10's - 100's
  CHIPS: 400 - 10's OF THOUSANDS
RELIABILITY:≈>100,000 HRS
CLOCK RATE: 8 - 100 MHz
200 MIPS - 1000's / FEW 1000 KW RAM

Figure 9

PAVE PACE SYSTEM DESIGN
ENVIRONMENT

MISSION
REQUIREMENTS

STANDARD SYSTEM
HARDWARE MODULES

STANDARD OPERATING
SYSTEM SOFTWARE MODULES
• PAVE PILLAR
  • PAVE PACE

AVIONIC SYSTEM,
HARDWARE & SOFTWARE
DESIGN

• NAV   • W.D.
• FLT CONTROL • CNI CONTROL
        ⋮
STANDARD APPLICATION
MODULES

FOUNDRY

• AFFORDABLE
  DESIGN &
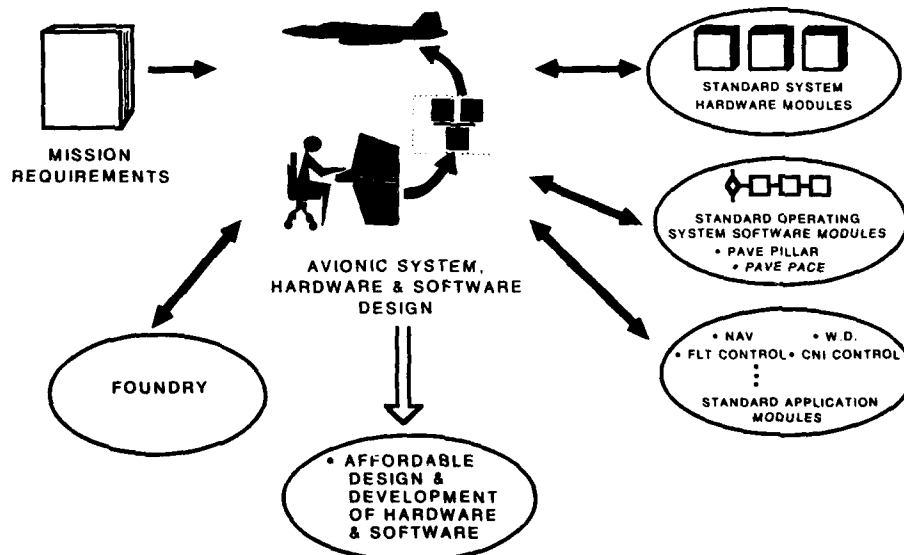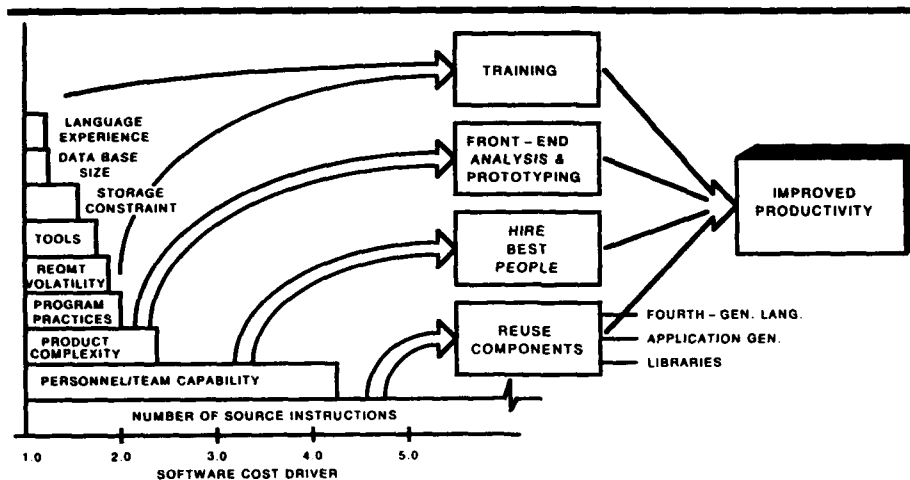 DEVELOPMENT
  OF HARDWARE
  & SOFTWARE

Figure 10

HOW DO WE IMPROVE
SOFTWARE PRODUCTIVITY ?*

COCOMO SOFTWARE LIFE – CYCLE PRODUCTIVITY RANGES (1985)

* BOEHM, BARRY, SEPT 87
  COMPUTER MAGAZINE

Figure 11
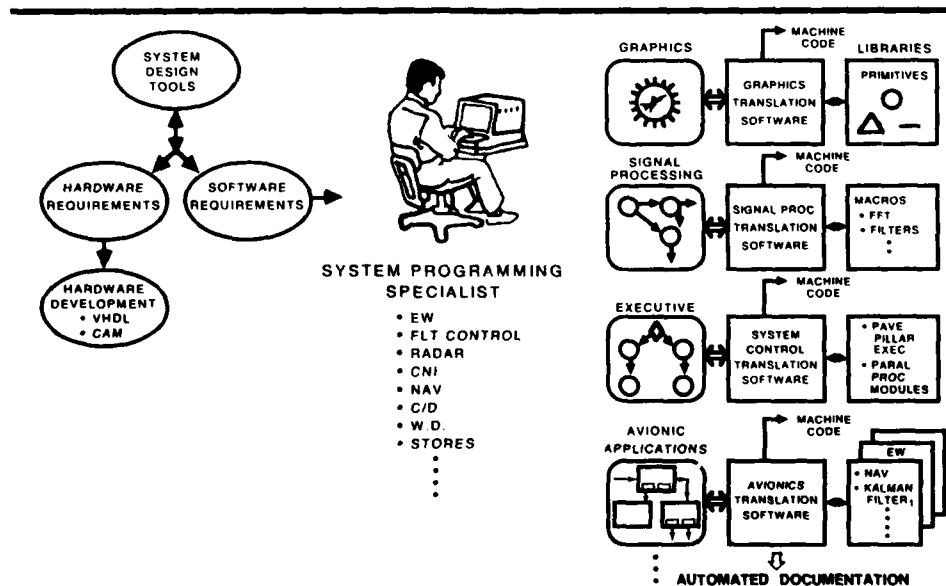
REUSABLE SOFTWARE/FOURTH
GENERATION PROGRAMMING

Figure 12

Figure 12 shows an example of how the concept of fourth generation programming and reusable software might be accomplished. Here support software for specific functional areas is used. This code translates the "pictorial" commands of a system programmer or functional specialist. The inituitive programmer/designer communicates at a level most familiar to established design practices (e.g. pictorial formats if a display generator is being programmed, flow graphs for signal processing, feed back loops/multipliers/ gains for flight controls), etc. Initially reusable application or executive modules will be developed using these tools. Subsequently, the same tools will be used to tailor reusable code to the specific application. A front-end, Ada-based compiler is assumed, with loadable machine code developed using these four tools. Automated documentation is also produced by the software.

ADVANCED ARCHITECTURE AND INTEGRATION TRENDS

If technology will allow us to build flyable supercomputers and fast switching networks, how will system architecture be affected? Will new ways of integrating functions develop, given that technical constraints will no longer be a serious factor in determining how the system is partitioned? These two questions must be answered together since architecture enables integration and integration drives architecture.

It was only slightly more than a decade ago when electronic systems began to be integrated by the DAIS program. Loosely coupled offensive and defensive sensors and controls/displays were integrated across a multiplex bus, along with the bus-oriented integration of flight controls and stores. (See Figure 1) Here, the thrust was on establishing digital standards, easing integration (e.g. reduce cabling) and reducing support costs.

PAVE PILLAR further enabled integration within and across avionic functional elements. The PAVE PILLAR architecture allows for improved fault tolerance and commonality of line replaceable modules. Three basic integrated functional areas are emerging from this architecture (Figure 2). These areas are: (1) sensor/pre-processors/ signal processors; (2) data processing for system control/reconfiguration and system level computations; and (3) vehicle management (flight, propulsion, electrical power). Redundancy of processing elements and buses is selectable, based on system requirements. This architecture allows the designer to select whether processing and information exchange should be done at a local level, as part of a functional affinity group or at the mission avionics level.

The authors believe that integrated affinity groups of functions, or "meta functions" will emerge. For example, an "RF meta-function" would utilize needed portions of the RF spectrum to transmit, receive or share information as an integrated system (thereby improving situation awareness, stealth, fault tolerance and reliability of processed results). Similarly, other metafunctions, shown in Figure 13, are expected to emerge in the areas of EO, vehicle control,crew station electronics and system management. It is of interest to note that the overall architectural framework of PAVE PILLAR is preserved. Intercommunications between meta functions can still be supported by the PAVE PILLAR high speed data buses. Intercommunications within meta functions are envisioned to be a hierarchial mixture of high speed data buses connecting tightly coupled parallel networks.

PAVE PACE

The PAVE PACE initiative has been established to validate the concepts described above. This advanced development initiative is envisioned as the means to select the appropriate technologies, mature them as necessary and conduct a series of integrated test bed (ITB) demonstrations that show how avionics availability, performance and cost can be significantly improved. Further, the applicability of the resulting hardware, software and tools for the modernization of the force structure and new aircraft systems would be projected. New avionic modules will be built and validated for reliability enhancement. A modular building block family approach to hardware and software for airborne parallel processing would be demonstrated. Advanced CAD/CAM tools would be exploited/developed and used to demonstrate improvements in hardware and software design, development and support. Advanced algorithms would be developed and used in a real-time ITB configuration using an advanced crew station in order to demonstrate improvements in situation awareness, fault tolerance and maintenance strategies. Advanced technologies such as neural networks would be investigated in an integrated system context.

CONCLUSIONS

The Air Force has faced and will continue to face an industrial, political, and military environment which will result in important constraints that drive avionics architectures/systems.

- High development, acquisition, and support costs.

- Long development times; 7-9 years typical.

- Post deployment reliability and maintenance.

- A history of change over the life cycle of an airplane.
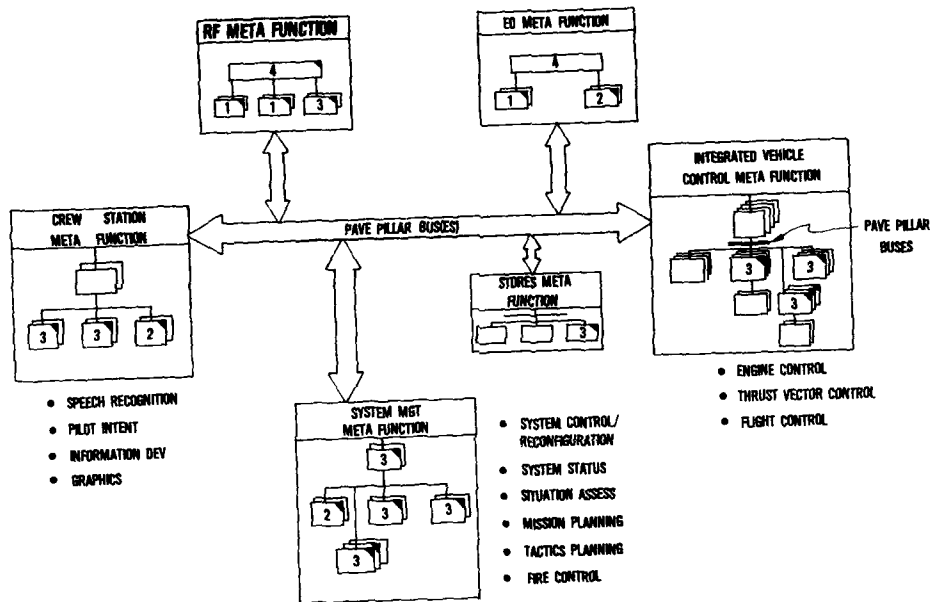
Figure 13

- A limited budget--continued pressure to reduce the budget.

- A developing critical shortage in technical people--engineers, maintenance technicians, computer programmers.

- "Computational plenty" from generally more powerful and available microcomputers which threatens to swamp the Air Force with software.

- An increasingly more capable and technically sophisticated enemy.

The technical implications arising from these constraints shape the design decisions and investment strategies for technology development.

- A history of avionics change over the life cycle of an airplane due to technological pressures and/or operational requirements pressures demand that avionics architectures be flexible. Flexibility may imply the heavy use of modularity concepts and clearly defined system interfaces that allow system upgrade without massive perturbation of the logistics system.

- High costs, budget constraints and limited personnel availability dictate concepts like reusable hardware and software--avoiding the costs associated with "reinventing the wheel" syndrome. If an avionics architecture can support the use of previously developed satisfactory components like standard subsystems or standard software modules, item development time can be reduced by minimizing the number of completely new components and integrating those with the standards.

- The continued influx of digital systems into the Air Force inventory, compounded by the microprocessor explosion and the developing critical shortage of qualified software people, dictate the need for focus on standardized software support concepts to minimize the capital investment required at our Air Logistics Centers as well as minimizing the training needed to qualify support people on new software programs.

- None of us are satisfied with the performance, cost, or reliability of the bulk of our current avionics. The ultimate has certainly not been reached. The required improvements will principally come from technology; therefore, a standardization approach in avionics architecture must not stifle this needed technological evolution, but should provide a framework of standardized interfaces which can support insertion of new technology, again without necessitating a massive change in system support.

REFERENCES

[1] Mason, R. C. and Thullen, M. J. "Integrated System Design - DAIS." AIAA/IEEE Digital Avionics System Conference, 1979.

[2] Ostgaard, J. C. and Szkody, R. "PAVE PILLAR Avionics Designed for Dependability." Defense Electronics. May 1986.

[3] Buzbee, William. "Parallel Processing Makes Tough Demands." Computer Design. September 1984, pp 137-140.

[4] Hillis, W. Daniel. "The Connection Machine." MIT Press. Cambridge, Mass. 1987.

[5] Wiley, Paul. "A Parallel Architecture Comes of Age At Last." IEEE Spectrum. 1986, June 1986, pp 46-50.

[6] Bond. "Parallel Processing Concepts Finally Come Together in Real Systems." Computer Design. 1 June 1987, pp 51-74.

[7] Fortes, A. and Band Wah, B. "Systolic Arrays - From Concept to Implementation." Computer Magazine. July 1987.

[8] Grinberg, J., et al. "A Cellular VLS I Architecture", Computer Magazine. January 1984, pp 69-80.

[9] Young, Jeremy L. "The Software Foundry: Almost Too Good To Be True" and "QTC Makes It Easy to Design Custom Processors." Electronics. 21 January 1988, pp 47-51.

[10] Boehm, Barry. Computer Magazine. September 1987.

[11] "Air Force Studies Board, Committee on Isolation of Faults in Air Force Weapons and Support Systems." National Academy Press. Washington DC. 1986.

[12] "USAF R&M Action Plan Final Report, Volume II." 1 February 1985.

[13] Morgan, D. R. "PAVE PACE: System Avionics for the 21st Century." 1988 NAECON Proceedings.

Avionic System Design Methodology

M. J. Tooze

Systems Consultant

GEC Avionics Limited, Airport Works, Rochester
Kent ME1 2XX, England

This paper describes an approach to avionic system design and its application to modular avionic architectures.

The approach is to test various candidate architectures using a common functional requirement. The method commences with a requirement analysis carried out in a top-down fashion to arrive at a full functional description. A parallel phase is to determine the technological base and define a number of candidate architectures and corresponding component sets (module sets in the case of modular architectures). Thus technological performance and in-place equipment limitations are included at an early stage independent of the requirement. Hence "top-down meets bottom-up", by taking various architecture candidates and corresponding modular sets and applying the functional description of the requirement, so each architecture may be investigated for its capability to cope with the trial or application system. Assessment of reliability and performance objectives is discussed. Also included is reference to the areas of operating system and BITE which may form part of the system but are not necessarily directly represented at the boundaries of the system.

The paper deals with philosophy of the approach and does not extend to application of the various CASE design tools which exist (or may be specified) in order to carry out such a project in practice.

1. Background

Firstly I shall start with a little background. Our industries have, for a number of years now, successfully delivered equipment to customer requirements which include reliability and maintainability considerations, in addition to performance requirements. Also we have operated schemes, such as reliability improvement warranty, which have been aimed at reducing lifecycle costs. Other customers have operated under conditions where their prime interest has been to minimise start-up and initial purchase costs for given performance. However, things are changing, customers now, at last, understand that R & M factors can be built into equipments from the beginning along with performance requirements.

So by adopting the advances promised by new technology and by planning sufficiently ahead, new systems architectures are being devised which will enable minimisation of lifecycle cost. In order to contain the potentially increased start-up costs of such systems, these architectures are aimed at serving more than one application.

Because the lifecycle cost implications of design are often not obvious and because this design task is additional and very much complicates the normal activities that are required for design for performance, we need a system design methodology which will enable us to sort through the plethora of technology and architectural choice in an ordered way to present us with a single, or at least a reduced, set of options with which to go forward into the detailed design or development phase for the system.

Key factors affecting the new modular approach to avionics design are:

. connector reliability

. maximum power per module

. distribution of power between processing and I/O on a module

. number of module types

. policies for safety, security, mission reliability, maintenance, operation without maintenance, survivability

. reconfiguration boundaries

- bus connectivity/bandwidth

- number of bus physical layer types

- level of BIT performance

- growth potential

2. Introduction to a System Design Method

This paper describes an approach to avionic system design and its application to modular avionic architectures. The study methodology goes through the phases of analysis, synthesis and assessment in a structured way.

2.1 Requirements for System Design Methodology

Because future goals have now moved to include reduction of lifecycle cost and architectures applicable to many requirements and vehicle types, one thing is certain, start-up development costs of such architectures will not be low. Also gearings of design parameters into LCC are more difficult to determine which makes potential design or architecture evaluation much more complex and certainly less obvious than design requirements driven mainly by performance. Therefore it is essential that system design methods used to determine such systems yield, for the limitations and goals given, the near optimum solution rather than a solution. Such a method must be able to show a visible, traceable derivation of the solution showing fully all arguments and assumptions made in reducing the many infinities available down to the chosen solution.

The system design method must be manageable, that is it should follow a clearly defined structure, able to be broken down into separate well-defined elements to enable experts to be brought into the programme appropriately and to enable mapping onto a project management aid. It is, for example, essential that areas of unexpected difficulty or need for additional resource be shown up quickly as this field is an area of potentially high risk, in that it combines the prediction of future technology with new or novel architectural concepts.

The system design method must be able to combine available technology performances to form candidate architectures for evaluation. Even though the selected modular architecture may be intended for a range of applications it is important that it be tried on a real, detailed, requirement in order that it be thoroughly tested and sized.

In pursuit then of the architecture, rather than a solution as it would seem from the viewpoint of a particular designer, it is essential to separate the requirement and requirement analysis from the technology study. By the technology study I mean the prediction of available and necessary technology for the period to be considered.

What this means is that top-down design is not enough in itself. A major feature for this method is that "top-down meets bottom-up" - both are essential aspects in the reduction of the solution.

As you are aware top-down refers to the structured analysis of the requirement(s) broken down hierarchically into greater detail the further the process is continued.

Bottom-up traditionally refers to the use of in-place equipment or technology - clearly an important part of the design method is the analysis of what technology will be or could be made available in time for inclusion in the design.

It is vital therefore that a modern system design method provides for both of these areas of influence upon the overall design result.

2.2 Method Structure

Fig. 1 shows the overall structure of the study and the main activities. The activities proceed chronologically top to bottom of the diagram.

2.2.1 Initial Phase

The method starts with an Initial Phase in which the project is set-up, the limitations and goals are set or confirmed with the customer and one or more applications or missions are defined at a user level. Also this phase includes certain vital support activities such as literature searches and determination of metrics to be used during the study/design period. However the prime activity during the initial phase is the mission(s) analysis. A timeline analysis showing the main functioning of the intended system by phase of mission(s) is the minimum necessary to ensure support of the follow-on design method and to maintain traceability back to the originating requirement statement(s). The time line analysis leads directly into definition of the top level functional representation of the system.

2.2.2 Requirements Analysis

More detailed analysis of the requirement may now be undertaken. This is based on functional decomposition of the top level requirement. Why Functional? The reason is that the system design does not exist at this time and complex modern systems are high on functionality but low on in-place objects. The output of the requirement analysis is a set of functional primitives at the base of the hierarchy forming the functional decomposition.

2.2.3 Technology Study

The Technology Study is aimed at determining what technology will be available to fit the required development, IOC, and lifetime of the projected system. This phase of the design method is less closely coupled to the mission requirement than the Initial and Requirement Analyses Phases and so may be run in parallel with these two. The objective of this phase, as applied to modular avionics, is to arrive at a number of candidate architectures and associated module sets for evaluation in the following phases.

It is necessary here to emphasise the clear distinction between the Technology Study and the Requirement(s) analysis phases of the programme. Technology is about actual achievement of component performances and integrated performance able to be achieved in the timescale. The functional analysis is totally without reference to need for hardware, except where it is 'given' at the boundaries of our projected system. So the difference is that the technology phase is about determination of maximum achievable performances whereas the requirement analysis phase is about determination of minimum required performance.

2.2.4 Synthesis

The Synthesis phase is to take the candidate architectures output from the technology phase, and in turn subject them, or load them, with the requirement as expressed in the function primitives listed from the requirements phase. Thus each candidate is 'loaded' with the set of functional primitives so sizing the architecture for the particular application(s). It is only when this phase is complete with competing designs defined and sized that the final major phase, that of parametric studies, may be commenced.

2.2.5 Parametric Studies

Parametric studies is the area in which quantified assessment of the remaining candidate(s), or hopefully the selected candidate, takes place. A division of the parametric studies is shown. The area of Lifecycle Cost Factors includes other cost related components. The assessment of Performance is a cross-check to see that the required performance is achieved and that performance margins are identified. Finally 'ilities, this is a term which we use for availability, reliability, maintainability, etc. These all have a gearing into the paramount LCC but objectives set at the beginning of the project may include certain intermediate goals and assessments rather than call for LCC reduction alone.

2.2.6 Reviews and Conclusions

The final phase provides for review of the results thus far and gives an opportunity to go back over the decisions that have been made and revise or determine points which may have been left undetermined. For example, only at this stage will loadings and margins be known, and if a finally checked figure occurs near a technology threshold, then implementation will be reviewed. In addition to defining the architecture based on the supplied data recommendations may identify development areas required to achieve enabling technology items. Thus the project will yield a means of focussing future development and support of future applications.

So that is an outline of a system design method which accommodates design of systems to objectives which may contain opposing elements. A range of solutions is generated against which deterministic evaluation of each candidate is applied to one or more well-defined requirements. This enables evaluation leading to selection of a particular candidate, or more realistically, a variation of that candidate as the recommended solution.

I will now take each phase in turn and explain the objectives and means of achievement of each phase along with the sort of difficulties which may be encountered in practice.

6-4

3. Initial Phase

As the basis of the study method is to attain traceability from the required missions and other available input data through to the final design, it is essential that the study commence with an overall view of the objectives of the study and analysis of the input material, this will include a literature survey.

The literature survey will be required to cover aspects of the threat scenario and projections of both friend and foe system design and capabilities for the projected IOC dates for the required system. It will also require to research into the available, and able to be made available, technologies from which the system may be constructed. This technical aspect of the survey will range from projected silicon technology through to data transmission media and data bus techniques. A key element in research for the modular avionics is the performance achievable from these technologies and the ability to suitably combine them to provide candidate solutions for system applications envisaged during the lifetime of the required architecture.

The practical problems of the literature survey are that having generated a large and wide range bibliography, the difficulty remains in imparting the information contained to the members of the project team. It is therefore of great assistance for a literature survey summary to be prepared, giving a brief summary and findings of each item. This enables team members to quickly arrive at a common starting point and to pick out those items of literature which have particular relevance to their role in the design activity. The literature survey needs to include reference to future and projected relevant standards.

Time Line Analysis

The time line analysis to be used for a modular avionics design point will need to include all aspects of the avionic system use and will, therefore, include clear description of all ground environment activities in addition to the airborne performance related time lines. Omission of these could result in a design of seriously limited scope. The purpose of the time line analysis is to relate required mission functions to phases of flight, events, and overall objectives, thus this analysis will include necessary attributes of maintenance, replenishment and mission preparation and other operational information necessary to visualise the full lifecycle of the avionics system. Finally the output of the initial phase will provide a top level functional description of the system and a clear statement of the objectives.

4. Requirement(s) Analysis

The last part of the initial phase was to define the top level functional representation of the system. There are many ways that the functional representation of, say, an aircraft or weapon system may be represented at the high level, however, we have found the following guide lines both useful and practical from the viewpoint of those carrying on the functional decomposition to lower levels. The first requirement is to ensure that the functional diagram is able to be interpreted back to the time line analysis output from the initial study phase. Also at the next level down, it would be convenient if the functional breakdown were to be along lines of areas of expertise available to the study team in order that experts be available for consultation as level of detail is increased as the decomposition progresses.

This may be seen as upsetting the parity of the approach but such is the mobility of placement of function in such architectures that in retrospect the outcome is not affected.

Functional decomposition is the functional representation method which is suggested as the means of providing a hierarchical diagrammatic representation of the system function and operation. The diagrammatic method should be of a nested nature as indicated. This means that a block at a given level in the structure will appear expanded as a complete functional diagram at the next level down and so on. At every stage the individual diagrams should be kept as simple as possible. Blocks in each diagram refer to functionality, the normal rectangular block referring normally to a system function, the ⬡ blocks referring to crew action and the ⬡ blocks referring to system display function. References, annotations, lists and supporting text include supporting data or message dictionaries and control tables. In this way system functionality may be represented by a connected set of system functions, system display functions and resultant crew actions. Representations of system operation therefore includes the crew's activity as part of the system.

The degree to which functional breakdown should proceed, i.e. the number of levels of functional decomposition, is dependent on the complexity of the system, but should extend to the condition where no elementary (lowest level) function would be expected to straddle a hardware boundary, i.e. every function should be containable totally within a single hardware unit. The lowest functional element resulting from the decomposition process takes the form of an elementary functional description or "functional primitive" and is represented in a standard format specification. Due to

the proliferation of blocks even after relatively few levels of functional decomposition, it is to be expected that the number of functional primitive specifications will be quite large. I have already referred to the disruptive effect of "loose ends". One advantage of functional decomposition is the early exposure of such potential errors.

As the top levels of functional decomposition are completed design reviews, including customer "walk through", may be carried out in order to verify the system in all its modes so that customer satisfaction be sought at an early stage.

Clearly the sheer volume of documentation and the need for consistency and other checks as the requirement is decomposed make this an area ideally aided by a computer driven system design tool or via integration of a number of such tools.

The extent of the detail of the decomposition of the requirement will be varied according to the form of the project being undertaken. Feasibility or an investigative study would require decomposition to three or four levels with reasonable accounts of the functions and the data flows between them. The detail required for a full project would need to be more rigorous and precise in its definitions but would not in fact need to be taken far beyond that of the feasibility study in terms of the level of detail examined, this is because the decomposition in both instances would be required to be taken to a level of assumed Functional Primitives. Remember that a functional primitive is a function which, given the level of technology anticipated could be contained in a module, either as hardware, as software, or as a mixture of both, but which does not span more than one of the intended system units or modules. More than one functional primitive may be contained within a module. Now at this point I need to reiterate that the requirements analysis is investigating the minimum necessary requirements for the system to perform the missions or system requirement. Because the functional primitives are to be packed into the, as yet candidate, module we need to express their attributes in terms common to those used in the technology study to describe the capabilities of the module. For example, a functional algorithm may be expressed by the number of lines of Ada code or the number of operations of a particular micro-processor order code. A statement of the iteration rate of the algorithm will also be required. This permits the required number of MIPS (Million Instructions Per Second) to be estimated. Other maximum and minimum requirements will also be laid down in order to specify the minimum performance required in order that the functional primitive meets the system need. These include latency, data storage, task state data (in support of reconfiguration) and estimates of the total number of lines of code required. State data may significantly contribute to data flows within a reconfiguration boundary.

Sources and sinks for input and output to the function along with estimated data/message bit rates are vital in the following evaluations of the candidate architectures.

For a large system it will be appreciated that the number of functional primitives received as a result of the expert teams endeavours will be large and time will be required in order to normalise these returns as deviations from the laid down rules always appear to be made by experts for their own good reasons. Thus in summary, we now have a pile of functional primitives which express the required characteristics needed to achieve our system function. These functional primitives address both the contained functions and estimates of the required IO data and control flows between functions. It should be emphasised that although there is some degree of design involved by the way in which the approach to the functional decomposition is made, as yet no prime design work has been carried out in terms of architecture selection or implementation decisions. By way of illustration, one could group the functional primitives such that all the high computational functions which contained a high level of computation were grouped within a single unit. This would represent a centralised computer solution. Alternatively functions could be grouped so that the computational functions were grouped with their prime sensors, this would represent a federated and sensor-based solution. Thus the prime importance of functional decomposition to our system design methodology is that we are able to break down and assess the functionality required, largely independent of any particular design implementation.

Thus the functional primitives are now ready to be distributed into the candidate architectures emerging from the technical study according to the rules and structures defined to characterise each candidate.

5. Technical Study

There have been a number of programmes carried out which address the question of advanced avionic systems for the future and which have tackled the reduction in lifecycle cost along with achievement of high performance by exploring the application of combinations of common modules, advanced data transmission media and protocols, integration of high levels of BIT, fault tolerance and reconfigurability. Therefore,

for this modular application of this study method key areas of technology were
identified which include topology, electronic technology for modules and operating
system.

The purpose of studying these key areas is to, firstly, identify the range of
technologies available, and then to combine these to provide a range of candidate
architectures intended to span the range of solutions but without encumbering the
designers at this stage with any more than an outline of the functionality of the
target system except as required to perceive the level of complexity which may be
entailed. Thus the candidate architectures are created without reference to the
detailed requirement analysis part of this methodology.

As stated before the objective of the technology study is to produce a number of
candidate architectures along with associated module sets. This entails the projection
of technology in terms and different stages of availability over the years leading to
the required in service date i.e. development availability of components, availability
of standards, availability for production. The creation of logically distinct
candidates is therefore an important output from this part of the method.

I should point out that it is just as important to close-down options as good
traceable reasons arise, one problem that will be encountered is that people will
continue with a 'dead' candidate as there are generally always some good aspects to a
line of approach even though there exist compelling reasons to kill it off.

Thus the main areas of the technology study are:

Topology

This part of the study is intended to reveal suitable topologies for the future
avionics architecture. It will therefore need to take into account the various
available electrical and opto technologies and bus standards. The factors which will
need to be taken into account in deriving these candidate topologies include
connectivity bandwidth v power and receiver technology, effects of failure for ring,
star and other bus configurations. The increase in mission availability may be
achieved in part by increasing the time between maintenance actions and providing the
system with fault tolerant characteristics so that the system will remain operational
in the presence of fault conditions. The topology study will also tackle LCC
contributions directly by improvements in reliability both in the data transmission
system and again in its means of tolerating fault conditions when they arise. Thus
minimisation of pin counts and reduction of juction temperatures within the data
transmission media are considerations made during this stage. The output of the
topology study is to provide topologies options from which the candidate architectures
may be drawn having regard to the parallel outputs from the electronic technology and
operating system investigations.

The important point here is to get some numbers down describing basic data
transmission building blocks so that these may be built into systems. At this stage it
is all about bandwidth, connectivity and power budgets, protocols and other nice points
come later.

Electronic Technology

The prediction of relevant electronic technology for the period for a future
system is clearly an enormous subject. This further bottom up aspect of the study
approach is to provide a view of what could be available in terms of module performance
under the primary constraints of size, weight and power consumption. A key issue is
the amount of silicon that can be affixed to a given size of module and the
functionality that can increasingly be compressed into a given area of silicon. The
primary considerations in the module in determining the future performance is the
realisation of certain trade-offs, for example the power bandwidth trade-off was
identified such that for a given constant power within an avionics common module so
processing power could be traded-off against IO bandwidth, thus over much allocation of
power to a CPU will limit module IO capability and alternatively overly much power
assigned to communication would limit the effectiveness of the CPU contained on the
module. Another trade-off area is that of power dissipation within the module versus
module reliability. Here cooling arrangements for the avionic modules limit the power
dissipation of the module for given junction temperatures. Thus given the critical
relationship between device reliability and junction temperature, so reliability/power
consumption options are provided against the variously available cooling arrangements,
some of which suit avionic applications, others because of their complication and
maintenance disadvantages detract from lifecycle cost that the apparently improved
cooling arrangment had promised. Another area of investigation is to determine module
capabilities such that the total number of module types may be limited. For example a
threshold technology which only permits a processor to be carried on a module would
entail support modules of storage and I/O as necessary members of the module set. With
higher IC technology so more self-contained functions may be conceived so limiting the
module types required.

Operating System

Software cost is another area where opportunity for LCC reduction is potentially very large. It has long been my contention that so-called software problems are more often than not, not the fault of the software programmer but are in fact rooted in poor system design and in poor system requirement specifications. Programmers have been expected to pick-up these "loose ends" i.e. an undetermined condition for which no outcome is defined, and knit them into the system. It is only when the operator comes upon this, possibly obscure, event that the system specification fault is revealed. At the other end of the programmers' task we see that they have been called upon to "hook" their programs into the hardware interface further adding to the complications of their immediate task. This may explain the apparent obsession these days with object orientated design techniques rather than with functional design techniques.

Thus, in conjunction with the use of modern CASE tools for system detailed design and configuration control, the long overdue avionics operating system is considered a necessary adjunct for modular avionics.

The operating system is required to provide a common transparent interface between the application software and the avionics system. Where a candidate architecture is reconfigurable so this would be supported by the operating system.

The transparency I talk of enables a software task programmer to communicate with other tasks by reference to the other task name and name of the data only, i.e. message passing without reference to the location of the other task within the system. This permits large systems to be composed of separately compiled code and, of course, aids the support of reconfigurable, load balancing system candidates. To software the importance of reconfiguration is not so much after a failure, but at start-up with a new application task aboard, it will be accepted with whatever priority it has been assigned, just as if it had always belonged to the club.

Candidate Architectures

Candidate architectures are derived as the output of this bottom up technology driven part of the study by covering topology and electronic technology considerations to provide maximum module processing performance while minimising IO bottle-necks while the operating system considerations help shape the topologies by providing insight into the way in which fault tolerance and reconfigurability will be available within the system and hence the way that these factors influence the design of the common modules. An example here is that if a module set were to be designated such that a processor module always had to work in conjunction with a memory module and an IO module then upon the need for reconfiguration it can be seen that reconfiguration would have to be carried out by reconfiguring in groups of three such modules rather than assigning reconfiguration to just one. Thus a leaning towards self-contained modules that are able to stand alone on the bus structure is preferred. Architectures based around such modules will considerably aid reconfiguration and this is the key to achieving the high availability low LCC system.

6. Synthesis

We have now reached a stage where we have candidate architectures and associated module sets defined by the technical study and the functional requirements of the system defined by the functional decomposition diagrams and the functional primitives. The process of building the system now takes place whereby the design team will take each of the candidate architectures in turn and load the functional primitives in accordance with the approach defined for each architecture. In an architecture which includes reconfiguration these module quantities will be based on an initial, assumed, configuration and may not correspond to the actual real time distribution. Thus we can treat architectures which use a "virtual machine approach", i.e. where an operating system assigns tasks to modules as part of a reconfiguration basis. So the total number of modules required for each candidate can now be determined. It is only when this stage is completed that the design team can stand back and appreciate the initial relative merits of the candidate architectures when loaded in their different ways with the requirement(s) represented by different distributions of the functional primitives. This stage provides bus capacity for basic functionality, and reconfiguration as a result of failures. To this will need to be added considerations for growth and survivability.

After a period for review the method now moves into the parametric study phase in order that quantitative assessment of the candidates may be made.

7. Parametric Studies

With the completion of the synthesis task, we now have our remaining candidate systems expressed in ways and in enough detail so that experts may now be made available in order to evaluate the key parameters necessary for final selection and/or verification of the selected architecture. This is important as such experts are not

able to contribute these skills for a system expressed only in high level terms; they require the problem to be reduced to technology and functions of which they have *knowledge or expressed in terms with which they are* familiar in order to extrapolate from their experience/database. Once this is achieved they are able to envisage hardware and/or software implications of the required modules or candidate system components. Hence confidence is added to the design data and the discriminants *between* options.

I hinted in my introduction to the method that suitable division of the parametric studies areas may be along the lines of a *lifecycle cost study, a performance study* and an 'ilities study. Whatever the breakdown of the parameters which are indeed chosen and which will therefore contribute to *selection and verification of* the selected architecture, it is most important to realise that these teams cannot start from day one with the loaded candidate architectures alone. It is vital that the metrics and methods by which these parameters are to be determined are ascertained in parallel with the other main phases of the work *in order that they are* ready and in-place for use from day one of the parametric study phase.

8. Conclusions

We have used this methodology applied to a study investigating the advances attainable in avionics systems around the turn of the century, and included in this the effect of modular techniques. Overall objectives were a reduction in LCC while meeting the predicted threat, with an ability to track the changing threat incrementally. It was shown that such objectives could be met. Initial design costs could be defrayed by applying the system elements across a wide range of future applications. A significant factor arising from the results is the increase in availability and the ability to operate for extended periods without need for maintenance.

The number of *module types was reduced even below our first objective figures and* very significantly we feel that we have at last capped the *ever growing software* problem by evolving an approach in which good system design and system design tools are used to produce clear bounded software tasks simply *interfaced through* an operating system. This supports message passing, reconfigurability, including load balancing, and maintenance to provide a transparent interface at *system and vehicle* component level operation. Simplicity of operation at all levels has been achieved while investigating application options which *include,* AI and *advanced* mission management aids leading into future intelligent vehicle/weapon systems.

It is a relief that so early in man's development of systems, that his ability to conceive developments *will* not be limited by his ability to implement them.

With a *reconfigurable* system margins may be reduced. If further facilities are required then these may be added to the system, thus avoiding the heavy contingency *margins associated with* LRU solutions.

These advances are primarily made possible by the new architectures, module MTBF's are only moderately better, the technology contribution is to lift module capability over the thresholds that make this possible.

The most gratifying conclusion was the ability of the system to achieve mission success over protracted periods with zero maintenance support.

Thus we have used this methodology and found it workable *and capable of producing* the results by ensuring that contributors make the correct consideration and contributions in time and at the right point in the *study, enabling diverse and* opposing elements to be weighed and worthwhile conclusions drawn.

System Design-A Structured Approach

OPERATIONAL
REQUIREMENT

DETERMINE
LOGISTIC
OBJECTIVES

DETERMINE
PERFORMANCE

MISSION
ANALYSIS

FUNCTIONAL
ANALYSIS

TECHNOLOGY
STUDY

TOP LEVEL FUNTIONAL
BREAKDOWN

FUNCTIONAL
DECOMPOSITION

TOPOLOGY
(DATA TRANS)

IC TECHNOLOGY

SOFTWARE
(OPERATING
SYSTEM)

FUNCTIONAL
PRIMITIVES

PROJECT SCHEDULE

ARCHITECTURE
CANDIDATES
AND
MODULE SETS

ALLOCATION OF FUNCTIONS
TO CANDIDATE ARCHITECTURES

SYNTHESIS

REVIEW

DETERMINE
EVALUATION
METRICS

PARAMETRIC
STUDIES

REVIEW

OPTIMISATION

90460

RECOMMENDATIONS
SELECTED ARCHITECTURE

RAPID PROTOTYPING OF COMPLEX AVIONICS SYSTEMS

by

L. Berardi

Aeritalia – Gruppo Sistemi e Teleguidati
10072 Caselle (Torino) – Italy

Abstract

The use of a rapid prototyping approach in the initial stages of complex avionics system design can complement some traditional computer design method. In fact most of the computer aids in engineering and design is aimed to a better, coherent and, as far as possible, complete description of the project, but not too much is done on the verification of the proposed concept implementation.

This paper discusses the advantages to have available in short time during the early design a software prototype of the system to highlight undesirable characteristics or possible improvements when the system has an high degree of complexity.

Then a design tool called ECATE (Expert Consultant for Avionics System Transformation Exploitation), developed by Avionics Systems Group of Aeritalia, is described. ECATE is an expert system that prototypes the information handling architecture of an avionics system.

The use of knowledge engineering and, in general, artificial intelligence approach for the rapid prototyping has been proven very effective, because of the high flexibility, complex domain mastering capability, and heuristic methods typical of these techniques.

Finally a description of a complete, integrated environment for the rapid development of prototypes of avionics systems, by using artificial intelligence and computer tools, is given.

1. Introduction

The increasing complexity of the systems, and a greater consciousness that "components" make "systems", has pushed toward a larger effort in enhancing the effectiveness of the system engineering science.

System engineering has been defined the study of a system concept in order to define its effectiveness and performance and to document results in specifications, from which hardware/software can be designed, developed and verified [1].

More in general, the system engineering can be considered the technical methodology that allows to develop a system in an organized way with minimum effort and maximum performance.

Pioneers in the field were the avionics engineers, who were forced to new and more effective approaches to the design and development of the electronic suite of the military aircraft. In fact the rapid development of electronics caused an increase in the demand of performances, and consequently cost and complexity in a rapidly growing spiral.

This is specially true in military avionics where the environment is extremely competitive and the challange very high.

On the other side if the causes are identified in the technology and the competition, the solutions can be found in the technology and in the scientific environment in which the actors of the system engineers move. In other words, if computer and digital circuits are the major factor in the electronics development, they can also be used to aid the system engineering; the results of the scientific views of the system can be used in the practice of the system engineering.

However, whilst the tools are the most up-to-date (supercomputer, CAE, CAD Work Station etc.) there is still a lack of integration among them because only now an holistic methodology is becoming popular for the system development.

Let us recall what are holism and reductionism by quoting Douglas Hofstadter [2]:

"Crab: HOLISM is the most natural thing in the world to grasp. It is simply the belief that the whole is greater than the sum of its parts. No one in its right mind could reject holism.

Anteater: REDUCTIONISM is the most natural thing in the world to grasp. It is simply the belief that a whole can be understood completely if you understand its parts, and the nature of their sum. No one in her left brain could reject reductionism".

To break the system in subsystems and components, each with the minimum possible interaction with the others is an example of reductionism in the design. To conduct performance simulations, specification preparation, installation studies etc. almost independently is an example of reductionism in development.

To make a prototype to check how the complete system works, is an holistic approach to the development. To design a system as an integrated network of sensors, displays and computers, with the maximum possible flexibility can be said holism.

In the last years the holistic approach has been more widely used in the design, thanks to the higher performances it can ensure, specially at the highest complexity levels: the development is still essentially reductionism, being prototypes build only in the latest stages of it.

It is clear that an approach in design and development only holistic or reductionistic cannot be "the answer", but a correct mix works effectively.

The factor limiting a more comprehensive and integrated development has been, up to now, the lack of suitable and widely available tools and methods to treat large and complex systems and theferore it was convenient to break the system into smaller, simpler units.

However it is our belief that the Artificial Intelligence tools and methods now available can extend the holistic approach to the earliest stages of the development with evident benefit in terms of result and performance.

The rapid prototyping is the way to consider the system as a whole also during the concept definition, feasibility or definition studies and to get better understanding, using also heuristics, of what will be the system when made.

The following paragraphs, illustrating some theoretical considerations upon the system project life cycle, intend to explain the need for a rapid prototyping methodology.

2. Theoretical Considerations

2.1. The Project Life Cycle

In order to establish the need for a rapid prototyping it is first necessary to have a brief look at the avionics system life cycle. The project phases may be described by means of a "waterfall" block diagram, a graphic representation which was and is widely used for software projects; it does not take into account explicitly for feedbacks but it is clear and simple and it will be integrated by other symbols later. For the purpose of this paper it is sufficient to consider seven major steps, see fig. 1; on top of the figure are also indicated other current denomination of the project phases.

It is clear that each project can have its own names for the phases, established according to Government/Customer or internal rules; the diagram of fig. 1 is only an example that will be used in this paper to locate the rapid prototyping methods. The project phases of fig. 1 establish a base for the understanding of the project evolution, but their definition does not intend to be complete or final; hereafter the complete life cycle of the project will be named "development", while the Design, Build, Integrate and Test phase will be called "full scale development".

The Operational Requirement Analysis. This phase groups everything before the actual definition study of the avionics system and includes many activities ranging from conceptual studies of the weapon system, aerodynamics simulations, to the feasibility study, all seen from the avionic point of view. All these activities are aimed to establish "what the avionics system shall do". The duration of this phase is probably the most sensitive to political and technical considerations not only depending upon the size of the project, it can span from few months to several years.

An important factor to consider is that many activities run in parallel and extend also to the next phase in a continuous refinement of the requirements.

The System Definition. After the first general idea of the system functional architecture was given in the previous phase, now the avionics is more deeply studied to precisely define what are their major subsystems, components and what performance, in general terms, are to be expected by them and by the whole system. A precise, functional, HW/SW architecture is established, the functional performances are computed, overall installation, electrical interface are described, general standards and procedures prepared and project design and management tool chosen.

The aim of this phase is to show "how the system looks like and works".

Also this phase is sensitive to other factors than the size of the project and the available resources, although this occurs less frequently than in the previous phase, its duration can range from few months to years.

Design. All system components defined in the previous phase need to be bought or made and therefore they shall be specified or designed and all supporting procedures and methods shall be made available. Today it means to specify each equipment and, often, to standardize or specify some important sub components, like microprocessors, connectors etc. and to choose the software development tools. Design include always the system software and in particular the mission relevant software, which is usually designed specifically for the project either by the airframe manufacturer or by the avionic system responsible company or under their direct control, because it is a major, highly complex component of the system and a key factor for its success.

The aim of this phase is therefore "to describe in all detail the system, its components and their interactions".

)

It can be said that, roughly, the duration of this phase is directly proportional to the size of the project and inversely proportional to the available resources.

Build. All components specified or designed in the design phase need to be build. The difficulty of this phase clearly depends not only by the size of the system but mostly by other factors, like the technological background of the suppliers in respect to the requirements and, more in general, it depends upon how advanced is the system, in absolute and relative terms. Aim is "to provide the components of the system". Excluding all research necessary to form the tecnology background, usually this phase lasts from one to three years.

Integration and test. Each system component need to be verified, to check if it complies with the specification prepared by the system designer; a deviation from the requirements always results, in some way, in a system performance modification. Therefore it shall be known before to pass to the other project phases and to the second part of this phase that consists of the integration of all components to build the whole system, including the integration of the mission software in the relevant avionics computers.
Aim of this phase is "to verify if the system and its components were build and interact as specified".
Duration is depending not only by complexity and resources by also by the effectiveness of the tools and methods used.

Validation. When it has been verified that all component correspond to their specification and they work together as predicted, it shall be demonstrated that they make the system that was conceived, or, in other words, the design shall be validated. Validation of the design means to prove that its specification corresponds to its definition and the definition to the requirement. More than other this phase requires independent interpretation of the work done in the previous phases, in order not to fall in the same mistakes already done by the designers from definition to design. Most of the previous phases are carried out by independent teams, but it is a special need of the validation to be prepared and performed by different people than the design team.
Aim is "to check if the system looks like and performs how intended".
More than the previous phase the validation depends upon the tools and the methods, that include also flight test; its duration is usually some years and often if overlaps in time with the integration and test.

In Service. The avionics system is not a self standing product but it is part of a weapon system for the military ones and of a transport system for the civil ones. The in service phase regards the use by the Customer of the avionics integrated in the larger system; for the purpose of this paper it is considered up to the reaching of the final operational clearance, when the Customer explored in all major aspects the system behaviour. Now it can be proven "if the system operations corresponds to its requirements".
Again this is a very complex activity that lasts in many cases some years and it tends to have a large overlap with the previous one.

It is worth to highlight that not only there are overlaps, often large, between the project phases described above but also there are forward and backward loops that largely influence the actual project development and the system engineering practice. When a modification in a component specification occurs after its construction begun it is an example of forward loop; when a modification in a specification is needed as result of an integration test it is a feedback. Clearly forward and backward loops can occur between each phase, for instance there is a feedback when a review meeting held during the design induces some change in the system architecture, and consequently it effects propagate down into the following stages. However only some loops take a major role in the project because, either they tend to be excluded from the main path of the project (some operational requirements may change also when the system is under validation but this is usually treated as an "addition" to the original scope of the project) or they are rare or not very relevant from the point of view of the resources required to fix the problem. Another important aspect is the correspondance between the various project phases, that is the fact that certain characteristics are established in a phase and demonstrated in another one (for instance the design and integration/test phases).
Both characteristics of the life cycle, loops and correspondances, have important consequences upon the effectiveness of the development methodology and the tools required; they will be illustrated in the following paragraphs; to conclude fig. 2 illustrates graphically the two characteristics above said.

2.2. Tools and Methods

The project life cycle illustrated in the previous paragraph is implemented by means of the methods of the system engineering supported by proper tools.

Each phase has its own problem, approached by a proper methodology and the tool that implments it; in the past almost all tools were independent, one tool and one method for one problem, only in the last few years, come out some "integrated development environment" that trays to integrate many tools across some of the project phases or provide a set of dedicated utilities designed according to a common guideline of development.

Great impulse toward that direction was given by the software engineering, the software and particularly the mission software of the avionics system, has a life cycle parallel to that of the avionics and corresponding to it almost from the beginning. The rapid growth of the avionics software in terms of complexity, cost and importance for the success of the development has pushed to an higher use of automated tools and methods for its development.

From the minimum set of compiler, assembler, linker/loader, necessary to generate the executable from the source code, the tools, again all based on the use of digital computers, grew to an integrated development environment that covers from the design to the integration/test, including configuration control, quality assurance, resource management etc. Tools such as the Program Development Language (PDL) or the Interactive Symbolic Debugger (ISD) gave new perspectives to the development methods, previously based on experience and skill only.

The importance of the influence of software engineering upon system engineering, as far the new methodologies are concerned, is highlighted by the fact that one of the first applications of rapid prototyping was the software requirement preparation, which is not well supported by the conventional development environments. The success of the new approach to the software development, which complexity can be sometimes comparable or higher then an entire avionics, lead to its use also in the system development.

This had the consequence to boost the use of digital computers and to extend those methodologies to the earliest phases of the development which previously were left mainly to experience and skill. The computers set the standard for the Flight Test Instrumentation, Ground Stations for on-line and off line data reduction and elaboration, integration and validation ground rigs, Test Equipment and so on. Computer Aided Engineering (CAE), Design (CAD), Manufacturing (CAM) workstations are integrated into the Computer Integrated Manufacturing (CIM) facilities, which take care of some important aspects of the avionics design. Also important are the simulation facilities and the operational research tools, specially in the requirement analysis phase. This short list of tools does not want to be exhaustive, but it is worth to recall everybody's experience in this field; however for the purpose of this paper it is necessary to mention the tools that can be used in the definition and design.

The tools known to the author are aimed to the description of the system at the lower convenient level by means of a formal language and/or graphic tools, they are examples of a methodology that can be said reductionistic, in fact the system and its functions shall be broken into sub-systems, major assemblies, part and components, each isolated as unit and connected to the other by means of a clearly defined interface. The operation to divide in parts the system is guided and coherency check can be automatically performed at the end of the operation. A formal description of the system is very important for the definition and the design to minimize the redundancies. It greatly helps the designer in clarifying the requirements and the corresponding implementation, particularly from the point of view of the functional characteristics. The formal description of a system previously or originally described only by means of plain english words avoids, in most of the cases, to forget some required characteristics, to generate a design which is not coherent with the requirements or to forget something in the design of the architecture. The overall result can be summarized saying that the use of a tool that formally describe the system reduce the forward loops in the life cycle as defined in the previous paragraph and illustrated in fig. 2.

However it shall be pointed out that an important aspect of the early phases of the development is not taken into account by a formal descriptor.

Certainly it is not the scope of these means but nothing advices the designer when the system he defined or designed, works or how good is its behaviour. Moreover their use shows insufficient results the earlier are the phases, clearly because the data are insufficient for a complete description. Again those considerations and results are left to the experience and skill of the design team.

2.3. The system engineering

Many are the needs for an organized way to proceed with the system development: minimization of time and cost, configuration control, quality assurance, producibility and so on: no question can be raised on the need for the system engineering in the avionics system development

However, specially when the complexity in high, an organization is not enough, but a_l effort shall be made to minimize the forward and backward loops in the life cycle (para. 2.1.).

In fact one of the greater, if not the greatest cause of increase in time and cost of complex avionics project can be found in the modifications induced by mistakes or omissions made in the first stages of the development.

There are many examples of project in which the need for architectural changes or more computing power, was discovered only when the system was at the validation stage, with dramatic impact on the project itself. Even if that level is not reached, it is clear that every change in the system has a cost, in terms of money, time and resources, which increases more than linearly with the stage of development in which it is introduced (see fig. 3).

A typical distribution of the modification in a system developed in a conventional way is shown in fig. 4; the modifications do not include those generated by the current development phase but only the consequences of errors made in all previous phases, excluding significant Operational Requirements changes. It is evident that most of the modifications came out when the system was assembled and tested in its global functioning; the corresponding cost in very high (data from some software development projects, which can be considered a good example of complex system development, show that the above described situation can more than double the overall cost).

The errors cannot be avoided but they can be minimized and their occurrence and the corresponding changes shifted to the left, i.e. toward the earliest stages of development; it means to reduce, as far as possible, the loops in the life cycle and, above all, the feedback loops, which are the most dangerous, because they induces modifications when the system is build and assembled.

The methodology that requires a formal description of the project down to the smallest suitable scale (see previous paragraph) is a significant step toward that objective, because it greatly reduces the forward loops in the first development stages and avoids some of the changes arising during integration/test. A complete, formal and coherent description of the system, its requirements and its interfaces avoids some unpleasant findings when the various components are put together and tested during the integration phase. The above methodology has been already employed in a number of projects and preliminary results known to the author call for a reduction of the cost of the order of 15%, which is anyway a significant result. However, assuming that the system, when integrated, work as required by its specification, there are other classes of problems that come out during the validation.

The validation of an avionic system is a complex and multistaged activity, it starts with tests on ground, in the lab and on the aircraft, and continues with flight tests. As said in para. 2.1, its purpose in "to check if the system looks like and works as intended".

In general terms, there are two types of malfunctions made evident by the validation activity, first it may happen that the implementation, although formally correct, does not correspond to the thinking of the designer. Example could be a synthetic display not replicating satisfactorily the corresponding conventional instrument. Secondly the solution chosen by the designer, although correctly implemented and theoretically satisfactorly, is not adequate when proven. Example could be a data transmission based on a hierarchy of several data bus that propagate unacceptably the transmission errors. Both categories of problems are typical of an insufficient study of the critical characteristics of the system architecture considered by an overall point of view and it is generated by the attitude of the designer to segment its design problem into "vertical" slices, that is to consider almost only the "equipment" aspect rather then the "architecture" aspect of the system. This methodology tends to allocate the functional characteristics of a system to its components rather than consider it a global attribute.

For example in a system based on a STANAG 3838 data bus the data transmission function does not reside only in the bus controller but also in the remote terminals of each equipment and in the host subsystems, a complete design of the data transmission system cannot ignore it.

Of course the formal description methodology cannot be disregarded, but on the contrary, shall be pursued in all its implications, increasing as necessary the level of description, but it is also worth to have an "horizontal" view of the system. It means that can be very fruitful, in reducing most of the annoying feedbacks of the validation, to consider some aspects of the design from an overall point of view early on the development.

Early prototypes of the criticals aspects of the system architecture and design can help very much in avoiding unwanted side effects by clarifying the component interaction, they improve the design and reduce induced costs, and, in general, carry on the objective of the system engineering.

2.4. The rapid prototyping

Primary scope of the rapid prototyping methodology is to provide the designers with means to improve their understanding of the system architecture they are conceiving. As the nomination says, the understanding is gained by exercising prototypes dealing with some important aspects of the system, like data transmission, expected development risk and so on; to be effective the prototypes shall be prepared in short time and shall not suffer too much of the incomplete and generic information available of the early stages of the development.

Although the use of rapid prototyping may be proven useful in several stages of the system development, its main application can be found in the steps between the operational concept definition and the build of the system. Rapid prototyping is intended to fill the gap between the weapon/overall system concept and requirement definition and the actual design; moreover it can complement the formal description methods in the design.

In the concept exploration, feasibility and definition steps the design team shall relay almost only on its experience and skill, which guarantees in most cases the success of the product. However that approach is not an organized and well supported way to proceed and sometimes has not given the best results; this in our opinion, was mainly due to the lack of development tools suitable to provide clear advices on problems cluttered in an highly complex environment. The conventional approach does not help very much, because of need a high number of information to work effectively and the available operations research tools are focused mainly on other aspects of the system.

The rapid prototyping can be very effective also in the final stage of the definition and design, when the information can feed a formal description of the system.

In fact on overall view of the most important aspects of the architecture and functioning can highlight some interaction problems and requirement misunderstandings. An early correction of those errors can avoid a large resource expediture to correct them later when the hardware is already build and the software coded. In short, rapid prototyping reduces the forward and backwards loops in the development life cycle (see para. 2.1).

Let us consider now the implementation of the method; key features are the rapidity in preparing the prototype, the flexibility in modifying it, the heustic approach in treating the available data, a "requirement oriented" description of the system.

The rapidity is needed because of the quick reaction time usually required during the early development stages and the flexibility allows for a comparison of several possible solutions. Heuristics helps when only few basic information are available, specially if they are expressed only as requirements.

All above characteristics cannot be found in the conventional simulation computer tools, the basic toolset for the preparation of the project-dedicated prototypes can be better derived, in our opinion, from the Artificial Intelligence machines and software. The Artificial Intelligence, as a scientific discipline has been developed about thirty years ago in the University . It created many expectations and resulted in some disillusionments, sometimes due to the lack of the necessary technology background.

However in the last few years the tools created expressely for this new discipline, its techniques and methods (in particular expert systems) gained popularity and favour in the industry as an applied technology suitable for interesting applications. Certainly that is due to the results obtained but also to the holistic approach and the intrinsic capability to organize the complexity, featured by the Artificial Intelligence tools and techniques.

The AI methods have all characteristics required for the rapid prototyping and more, the capability to dominate the complexity. To dominate the complexity means to have structures, methods, techniques apt to organize the knowledge, which is often heuristic, multivariate, sparse and non homogenous, unstructured.

That capability is intrinsic because AI is based on the knowledge and it consists in knowledge manipulation. This extra feature is not unnecessary but, on the contrary, could be a need because most of the applications of rapid prototyping are required by large and complex systems, where the conventional simulation and the model created by the designer in his/her mind are no more sufficient for a successful design.

Based on the above it can be inferred that the most challenging applications of rapid prototyping methodology shall be based on the AI tools, like LISP machines, knowledge engineering environments, which can ensure the necessary flexibility, complexity treatment capability and, being "rule based", can offer a "requirement oriented" description of the system.

In the following chapter it will be described a design tool for a specific aspect of a system, the information flow architecture. The tool, called ECATE (Expert Consultant for Avionics System Transformation Exploitation), has been developed and it is used by the Avionics System Group of Aeritalia and it is based on a computer and a software expressely conceived for the Artificial Intelligence. The example will highlight an important aspect of the rapid prototyping, which making use if AI tools, can be easily associated with expert systems, another useful approach to fuzzy problems. In fact ECATE not only allows to prototype architectures but also can give advices on its optimization.

The example has been already presented in similar form to the AGARD/AVP Symposium on "The design, development and testing of complex avionics systems" |3|.

Following it will be illustrated a more complete prototyping environment, that can be used at a later development stage, when more information are available and a prototype closer to the final system can give better results.

3. An example of rapid prototyping

3.1. General

A state-of-the-art avionics system shall be fully integrated with the other systems of the aircraft and shall take full advantage from the features of the microelectronics, to provide the crew with the highest mission success probability.

It means to find a real implementation for concept like distributed processing, sensor data fusion. adaptive reconfiguration, expert pilot assistant, synthetic world displaying, now made possible by the advancemement of the technology, specially the data processing and transmission.

But in such a system the performance increase is not simply due to the higher performances of the microcircuits, on the contrary it derives primarly by an increase of the overall system complexity. In fact the "black box", a large unit with well defined interface and function allocation, is no more the basis for the advanced system design but is being substituted by lower scale units, which changing combinations provides the best adptation of the system to a changing environment.

It is difficult to establich a metric for the system complexity (see for example ref. |4|), however it could be said that it is reflected by the amount of memory used for operational software storage, which today is increasing to a rate at least an order of magnitude higher than the number of other microcircuits in an avionics system.

The increasing complexity, while can allow for dramatic improvements in terms of reduced pilot workload and mission success probability, has also some important drawback.

It is evident that a complexity which is mainly software implies a design, development and testing process and a management of it much more difficult than in a conventional system.

3.2. The problem of the system architecture

The problem considered concerns the establishing of a correct data flow architecture. There are several interpretation of the term "architecture" in the avionic system design; it can be applied to the physical structure, the topology, the software organization and so on. All these are aspects of the same characteristic, the way in which the system components are organized and work together in order to create a system.

The architectural aspect chosen for the application described in this chapter is the information handling within the avionics, i.e. the characteristics of the data flow and processing among the various system components, considered from the point of view of the information treatment.

Therefore the following definition of architecture will be used:

Definition

System architecture is the organization of the information generation, distribution, processing and utilization within the boundaries of the avionics system.

A pictorial view of the above definition is given in fig. 5.

The boundaries of the avionics system are intended to define the meaning of generation and utilization of the information.

In other words if the boundary identifies the world outside the aircraft all information coming from it corresponds for the avionis system to a generation of information for the avionics system; on the other side the data are utilized when they are provided to the crew on a display or to the external world via an antenna.

Such an architecture is relatively easy to describe by means of few building blocks with a limited number of peculiar characteristics; but a correct design of it has relevant influence on the overall performance of the system, because it is usually established in the very early stages of the design and it is difficut to be drastically changed during the development process.

Therefore it is clear that a serious error in the data flow architecture design impairs the achievement of the design objectives in terms of time, cost and performance.

For that reason the architecture of the avionics system is usually designed by highly experienced people with support of the operations research tools (see ref. |5|): nevertheless the work of these people is difficult to quantify and to describe analitically, being often result of empirical knowledge and heuristics.

Rapid prototyping of a complex architecture helps to easily evaluate many alternatives while an expert system directs the search for the best design. A dedicated tool combining together the two techniques can organize and manage the overall complexity of an architecture, requiring from the operator higher level decisions only.

A tool like that sketched above, described in the following paragraphs and developed in our laboratory, can be of effective use for the purpose and can demonstrate the advantage of the Artificial Intelligence approach in the rapid prototyping of complex avionics systems.

3.3. System description

The building blocks that shall be used for the construction of an object oriented data flow architecture have characteristics that describe mainly their attitude with respect to the information handling.
Four types of objects represent the building blocks.

1. **Generators**, the sensors of the system, the controls available to the crew and the interface to other systems.

2. **Processors**, signal processors, mainly associated with sensors and displays and data processors to elaborate information at an higher level.

3. **Utilizers**, displays for the crew, interfaces to other systems, emitters or weapon which stimulate the external world.

4. **Channels** transmission means that link together all above objects when not directly interfaced (aggregation of objects).

Table 1 lists an example of the typical characteristics associated to the objects.
It shall be pointed out that the characteristics may vary in relation with some peculiarities of the described system.
The processors and the channels are possibly multiport devices, while equipment like a monostatic radar may be described by a signal processor, a generator and an utilizer, that is an aggregation of objects.
Although not directly related to a technology solution, the objects that form a system architecture from the point of view of the information handling, shall nevertheless take into account the state-of-the-art to avoid a design perfect but not feasible.
The building blocks shall be combined to form the information handling architecture corresponding to the functional architecture to model.
The architecture is characterized by some features, i.e. system descriptors which are listed in table 2.
Some descriptors need explanation on its definition, while the calculation methods are embedded into the tools and will be described in para. 3.6.

Risk The development risk take into account how much each object is close to its technology limit and how the the combination of objects influence the development.

Integration level It takes into account how good is the processing within the system. An higher integration level is a merit.

Growth Capability Represents the dual of the resource utilization of processors and channels.

It shall be noted that the descriptors can be computed also for a limited portion of the system, a subsystem.

3.4. Rapid prototyping and expert system design

A rapid prototyping tool shall assist the user to convert from the functional/performance requirements to a description that uses the object and connections illustrated in para. 3.2.
But an easy means to prototype many alternate design solutions is not sufficient because the knowledge behind the architectural design is not totally conveied by analytical descriptors.

Therefore an expert system, a tool that allows to acquire, use, modify and make available a type of knowledge which is complex, difficult to transfer, empiric, incomplete and heritage of a limited number of people is the most appropriate supplement for the rapid prototyping tool.

The expert system shall direct the search for a better architecture and provide advice on solution that may also not have different descriptor values but are known to guarantee an higher confidence of success.

The operational flow of an architectural design carried out by means of a rapid prototyping expert system is skectched in fig. 6.

3.5. The tool, ECATE

3.5.1. The environment

The tool, foreseen by para. 3.4 and called ECATE (Expert Consultant for Avionics system Transformation Exploitation), has been developed by means of KEE (Knowledge Engineering Environment, TM by Intellicorp), running on a dedicated LISP workstation (EXPLORER, TM by Texas Instruments).

KEE is a development environment prepared for Expert System construction, it could be considered an hybrid tool built on a range of state-of-the-Art Artificial Intelligence techniques utilized to combine different type of knowledge.

The knowledge is organized in frame/units associated to which are their peculiar characteristics, that structure is particularly suitable for the description of our problem because it implements a programming oriented to object linked by relations.

By means of KEE it has been implemented the following:

1. The user interface

2. The collection of objects and relations that represent the system

3. The algorithms and procedures for the descriptors computation

4. The expert knowledge

5. The knowledge handling structure.

The knowledge about the system is acquired via a graphic interface and processed by the inference mechanisms embedded in the KEE environment, according to the set of rules describing the expert knowledge.

3.5.2. The structure

The structure of the tool can be described by means of the block diagram shown in fig. 7. Hereafter follows a brief description of the main components of the structure.

User Interface The user interface assists the user to represent his system in accordance with convention of the formal description and is formed by:

    a) graphic utilities using icons, representing the objects, with associated menus for describing their characteristics.

    b) indicators of the system descriptors of the terminated system.

    c) menu sensitive "pushbuttons", i.e. means to activate a "method" (see below).

Methods The methods are procedures codified in LISP to execute algorithms, object interaction and reasoning/control stategies (see table 3 for example of methods).

Permanent data base It contains the description of the four types of objects and their classes . It contains, moreover the expert system rule base, unmodifiable by the user.

Working area It is formed by the units, which characteristics, called "slots", describe all information about the system under development.

Inference structure This structure is formed by an inference engine operating on the rules.

The structure and the development environment allow for the maximum flexibility; to change the object and system descriptors, inference rules or methods is extremely easy for the expert system designer. That feature is of capital importance and is used currently because the tool shall evolve with the available knowledge.

3.5.3. Consulting ECATE

The steps of a consulting session are summarized in fig. 8 and briefly explained hereafter.

Configuration Insertion The user, with assistance of the tool graphic facilities inserts the a configuration of objects, aggregations and relations he wants to prototype (see for example fig. 9).

Compatibility verification The tool verifies after each input its compatibility with the objects related to it.

Overall Compatibility When the configuration is complete the activation of the "terminated system" push-button starts a verification of the overall architecture compatibility.

The results of the step above can be:

1. Request for more information (for example some relation is lacking or some data are not available)

2. Display of incompatibility warnings at system level (for example a multipoint channel of insufficient capacity).

3. Satisfactory compatibility

Descriptors computation *When the system compatibility is not violated a method is activated to compute all system descriptors for which sufficient information is available*

Result display *The results of the previous step can be displayed (see fig. 10) in either the numerical or hystogram format.*

Optimization ECATE, by means of rules activated in forward chaining inference presents to the user advices on possible architecture problems and suggested changes involving objects, aggregations, subsets or the overall system.

Assistance request The user can ask for assistance in optimizing the architecture, advices are given in this sense.

Configuration change In case the user wants to follow one or more of the advices he can, by means of a pushbutton, modify the configuration and restart the consulting.

Explanation The user can ask at any time information about the facts that have activated the rules and generated the advices.

The tool accepts at any step not only numerical values in response to its queries but also generic indications, like high, low etc.
The consulting session can be terminated at any time and the results saved in the library.

3.5.4. Validation

The validation of a tool like ECATE shall answer to two kind of questions:

a) Is the tool conform to its specification?
b) Is the tool suitable for its purpose?

For the first check ECATE has been submitted for evaluation to the experts who concurred in its development and to foreign experts, *exercizing it by means of test cases.*
The second check is much more difficult to perform, because it is supposed to require a demonstration of the development of different architectures, accepted or rejected by ECATE.

The validation has been completed, the results available, show a good agreement with the predictions.
Nevertheless it shall be pointed out that the high flexibility allowed by the development environment and the tool structure stimulate a continuous refinement to adapt ECATE to new situations or to increase its knowledge. It is in fact current practice to introduce new object descriptors, computing methods or inference rules.
Therefore also the validation is continuing to follow the evolution of ECATE.

3.6. Example of application

In this paragraph it is shown an example of an architecture to illustrate how looks like the system results.
Fig. 11 shows how the tool allows to represent the sketch of the system prepared by the designer.

3.7 A Future Avionics System

At the moment we believe that the knowledge available on future avionics system architecture, (see ref. |6|), is not sufficient to effectively use ECATE.
Reason is mainly because, although enough data on sensors and processors can be found, the knowledge lacks in the isplay and control area and specially on standardized multipoint channels. switch, backplane and .igh speed data bus. Insufficient is also the knowledge of the rules that regulates the overall s; stem functioning.
Nevertheless data are gathered and trials are performed with reference to experimental data to allow the specific ECATE knowledge to improve.

4. An environment

4.1. General

The example described in the previous paragraphs deals with a particular aspect of the system definition activity, the tool used for the prototyping is a LISP machine, a computer for the Artificial Intelligence applications.
By means of the above said computer a software prototype of the system can be prepared and submitted to the designer's judgement, supported by an expert system.
However, when a more complete view is needed, a deeper definition is required, for instance in the design phase, or the results shall be examined and commented not only by the design team, but also by other people, the Customer for example, a software prototype only is not adequate.
Other resources are required to be able to develop and validate in short time the prototype and, above all, to use it effectively and to present the result in a suitable way.
Main purpose is to show how the avionics will operate in its environment, internal and external to the aircraft, when designed and implemented according to its requirements and defined architecture.
From the above it arises the need to have a displays and controls simulation, a real time capability and an evaluation mechanism embedded into the prototype.
The result of the integration of those resources in a rapid prototyping development environment (RAPIDE) will be described in the following paragraphs.

4.2. The Tool

The components of a rapid prototyping development environment are the following:

- man/machine interface simulating the displays and controls system of the aircraft.
- real time aircraft and scenario simulation
- object/rule based description of the system
- evaluation and consultation facilities.

Main requirements are:

- full integration of the components
- user friendly interface
- modularity and flexibility
- easy expandibility and interfacing to other computers.

The above listed characteristics may resemble an aircraft flight simulator, but the intent in different, more complete in the scenario and system simulation, less in cockpit and flight dynamics.

A typical implementation of the tool sketched above is shown in the block diagram of figure 12.
Major components are:

- the interconnecting network
- the computing nodes

they are described in the following paragraphs.

4.2.1. The interconnecting network

The interconnecting network is the backbone if an integrated, highly modular tool. It is a fundamental choice because it shall have at the same time an high modularity and expandibility and the capability to manage the data in real time. A suitable solution is a mix of a bus system, in a widely available standard, plus dedicated high speed links where an higher data transfer rate is required. This choice ensures an high compatibility and expandibility by the use of a local area network system and an excellent real time capability where and if required, to obviate to the data transfer rate limits of the most populare LAN implementations. All hardware and most of the software is available on the market, only part of the software is dedicated.

4.2.2. Computing modes

4.2.2.1. Artificial Intelligence workstation (Central Node)

The central node that coordinates the environment functioning is the AI workstation (one or more for complex applications).
On that computer the operator manages all the resources, controls the development of the prototype and make it work. On the workstation itself resides the kernel of the system prototype, i.e. the rules and the objects, and the real time data base of the network. The reason for the use of the AI tools and techniques has been already explained in the previous paragraphs; furthermore it allows the compatibility with less demanding applications. The workstation provides also an easy way to develop and evaluation an consulting tool, for example an expert system, that based on experience and heuristics can give useful advices on the system.

4.2.2.2. Graphic system

In order to provide a simulation of the displays and controls of the weapon system it is used a graphic system, formed by a workstation and satellite units.
The display formats and control logic are developed in a graphic form on the workstation and activated under control of the central node on the satellite units arranged to simulate the aircraft cockpit. A visual simulation of the scenario is not strictly necessary but often can improve very much the understandability.

4.2.2.3. Simulation facilities

The system and its environment, the weapon system and the scenario, shall be simulated by means of dedicated computers. The "intelligent" part of system simulation resides in the central mode, but for a specific implementation the simulation of an aircraft and its weapons resides on a powerful real-time computer. The scenario simulation can be implemented on a different computer because, usually it does not require stringent real time performances but it needs a large data base.
The real time computer can be used also for data acquisition and stimulation in case part if the system be available in hardware and its use more convenient that the corresponding simulation.

4.2.3. An Implementation

An implementation of the environment described above is under development by the Avionics System Group of Aeritalia. The main component are the following:

- an Ethernet network with TCP-IP protocol
- T.I. Explorer LISP workstation
- IRIS graphics system
- Gould 32/97 and VAX 8500 computers.

All component of the environment has been already connected and tested, the basic software in ready and a limited clearance has been given to start with the first application.

5. Conclusions

The benefits of rapid prototyping methodology do not need to be demonstrated. An early availability of a prototype of the system avoids late modifications of the implementation, in particular when the application is so complex to make difficult its complete and comprehensive understanding.

Current issues in achieving the goal of rapid prototyping are mainly related to the treatment of the complexity and its description. It is our belief that a suitable solution can be found by using the Artificial Intelligence tools abd techniques that are now widely available for industrial applications.

Those methodologies cannot be used for every aspect of the prototyping, conventional simulation is still useful, but they shall be the key point of the application because they can treat knowledge structures, heuristics and uncertainly, which characterize the complexity of large systems iun their early design stages.

Important is moreover the fact that an A I based prototyping can easily evolve in an expert system, that can assist the design team in his work.

The above said approach has been demonstrated effective by realizing a rapid prototyping and expert system dealing with the data flow architecture of the avionics system, a tool that is evolving in a full development environment for the prototyping of all major aspects of an avionics system.

We are confident that the implementation of the rapid prototyping by means of the knowledge management techniques can be a significant step in the system engineering.

References

1. Military Aircraft System Engineering. Leo Mehler, Boeing Military Aircraft Co. SAE Tech. Paper 861690.

2. Goedel, Escher, Bach.An eternal golden braid. Douglas R Hoftstadter, New York 1979.

3. Rapid prototyping of complex avionic system architectures L. Berardi, N. Giorgi, W. Mellano, A. Valente, E. Zucco. AGARD Conference Proceedings No. 417.

4. System Complexity : its conception and Measurerent in the Design of Engineering Systems. Sahal. IEEE Transaction on Systems, Man and Cybernetics. June 1976.

5. Critical Factors and Operational Research in tactical fighter avionics system development. L. Berardi, AGARD Conference Proceeding No. 343.

6. New Technology impacts on future avionics architectures - R.S. Majzak, Naval Air Development Cente, PA, USA. Paper No. 10 of the 51st AGARD AVP Symposium.
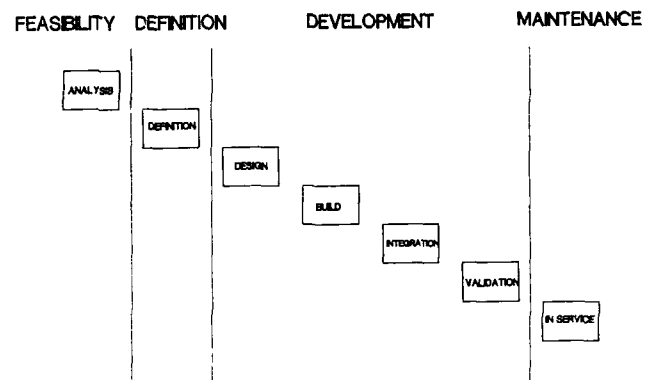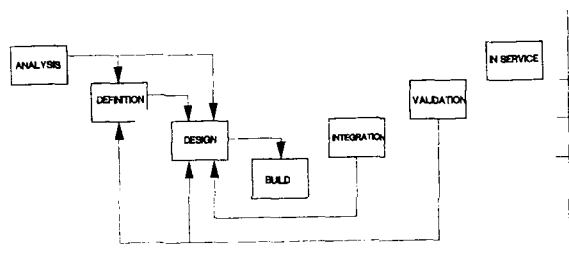
FEASIBILITY   DEFINITION   DEVELOPMENT   MAINTENANCE

ANALYSIS

DEFINITION

DESIGN

BUILD

INTEGRATION

VALIDATION

IN SERVICE

Fig. 1

ANALYSIS

DEFINITION

DESIGN

BUILD

INTEGRATION

VALIDATION

IN SERVICE

Fig. 2

cost   time   resources

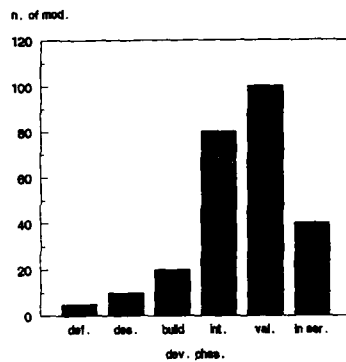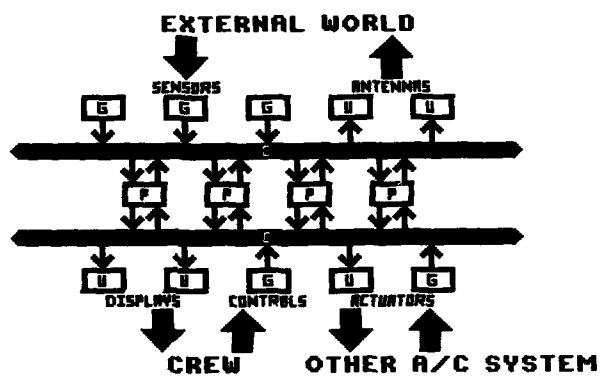design   build   test   validation

Fig. 3

n. of mod.

Fig. 4

EXTERNAL WORLD

CREW   OTHER A/C SYSTEM
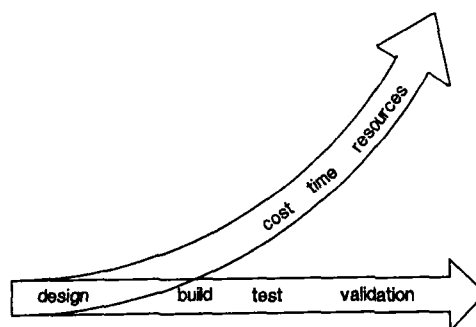
Fig. 5

RAPID PROTOTYPING DESIGN

Fig. 6

BLOCK DIAGRAM OF ECATE

Fig.7

A CONSULTING SESSION

ICONS → CONFIGURATION INPUT

OBVIOUS INCOMPATIBILITY CHECK → DISPLAY

PUSH-BUTTON → COMPLETED CONFIGURATION: COMPATIBILITY CHECK ← → DISPLAY

PERFORMANCE DETERMINATION ← PROCEDURE

PUSH-BUTTON → RESULT DISPLAY and/or HISTORY DISPLAY → DISPLAY

ICONS → OPTIMIZATION ADVICES ← → DISPLAY

PUSH-BUTTON → FURTHER CONFIGURATION CHOICE

Fig. 8

ECATE AERITALIA

Fig. 9

CREATE.OBJECT

Fig. 10

Fig. 11

Fig. 12

GENERATORS slots	CHANNELS slots
member	member
CONNECTED-TO-OBJECT	CHANNEL-TYPE (local)
COST	CONNECTED-TO-OBJECT
GENERATOR-TYPE (local)	COST
MINIMUM-LATENCY	INFORMATION-FLOW-CAPACITY (local)
MTBF	INPUT-INFORMATION-FLOW (local)
OUTPUT-INFORMATION-FLOW (local)	MAX-NUMBER-OF-CONNECTIONS (local)
OVERHEAD	MINIMUM-LATENCY
REDUNDANCY	MTBF
RISK-FACTOR	OUTPUT-INFORMATION-FLOW (local)
SIGNAL-CRITICITY	OVERHEAD
SIGNAL-TYPE	REDUNDANCY
	RISK-FACTOR
	SIGNAL-CRITICITY
	SIGNAL-TYPE (local)
	TYPE-OF-CONNECTION (local)

UTILIZERS slots	PROCESSORS slots
member	member
CONNECTED-TO-OBJECT	CONNECTED-TO-OBJECT
COST	COST
INPUT-INFORMATION-FLOW (local)	INPUT-INFORMATION-FLOW (local)
MINIMUM-LATENCY	MAX-THROUGHPUT (local)
MTBF	MEMORY (local)
OVERHEAD	MINIMUM-LATENCY
REDUNDANCY	MTBF
RISK-FACTOR	OUTPUT-INFORMATION-FLOW (local)
SIGNAL-CRITICITY	OVERHEAD
SIGNAL-TYPE	PROCESSOR-TYPE (local)
UTILIZER-TYPE (local)	REDUNDANCY
	REQUIRED-THROUGHPUT (local)
	RISK-FACTOR
	SIGNAL-CRITICITY
	SIGNAL-TYPE
	TRANSMITTED-INFORMATION-FLOW (local)

Tab. 1

SYSTEM DESCRIPTORS

Tab. 2

* Total data flow
* Total Throughput
* Total memory capacity
* Growth Capability
* Latency
* Overall efficency
* Redundancy
* Mean time between failure
* Integration level
* Total cost
* Development Risk factor
* Max Risk factor

```
(DEFUN TOT-COST (SYS)
   "Total cost computation function."
   (PUT.VALUE SYS 'TOTAL-COST
            (SUMMA 'COST (FIND-CHILDREN 'COMPONENTS SYS)))
   (BREAK-LIST (FIND-CHILDREN 'COMPONENTS SYS) 'COST 'TOTAL-COST SYS))


(DEFUN TOT-INLEVEL (SYS)
   "System integration level determination."
   (LET ((LIST-PROC (LIST-CONTROL (FIND-CHILDREN 'PROCESSORS SYS) NIL)))
   (PUT.VALUE SYS 'INTEGRATION-LEVEL
            (//
             (+ (SUMMA 'OUTPUT-INFORMATION-FLOW
                      (LET ((BIGLIST (FIND-CHILDREN 'GENERATORS SYS))
                            (PASSED-LIST NIL))
                        (LIST-GEN-CONTROL BIGLIST PASSED-LIST)))
                (SUMMA 'INPUT-INFORMATION-FLOW
                      (LET ((BIGLIST (FIND-CHILDREN 'UTILIZERS SYS))
                            (PASSED-LIST NIL))
                        (LIST-UTI-CONTROL BIGLIST PASSED-LIST))))
             (FLOAT (+ (SUMMA 'OUTPUT-INFORMATION-FLOW LIST-PROC)
                      (SUMMA 'INPUT-INFORMATION-FLOW LIST-PROC)
                      (GET.VALUE SYS 'TOTAL-DATA-FLOW)))))))
```

Tab. 3

MAQUETTAGE DES SPECIFICATIONS FONCTIONNELLES

DU LOGICIEL EMBARQUE

Monsieur Patrick SCHIRLE
Responsable méthodes à la division systèmes avioniques

AVIONS MARCEL DASSAULT
78, Quai Marcel Dassault
92 SAINT-CLOUD - FRANCE

RESUME

Les systèmes avioniques militaires représentent aujourd'hui la moitié du coût de l'avion d'armes moderne et atteignent de très hauts niveaux de complexité et d'intégration. Leur développement nécessite la mise en place d'une méthodologie, supportée par des outils, précisant notamment les tâches et produits associés aux différentes étapes de spécification et de conception.

En particulier, le développement du logiciel de ces systèmes requiert une documentation de spécification fonctionnelle abondante et souvent contractuelle.

Le maquettage de ces spécifications permet :

- d'améliorer la qualité formelle des spécifications
- de réaliser, très tôt dans le cycle de vie, une validation ___ unelle de ces spécifications
- de fournir des éléments de recette pour les sous-ensembles ___ me
- de disposer d'une référence fonctionnelle commode lors de ___ gration et de la mise au point des équipements réels.

Le maquettage est une technique de validation prévue dans la méthodologie et utilisée à l'étape de spécification fonctionnelle détaillée selon le scénario suivant :

- écriture des spécifications détaillées (langage semi-formel)
- analyse critique des spécifications contrôle de forme
- génération du code maquette et implémentation dans l'outil de maquettage
- tests fonctionnels sur maquette : contrôle de fond
- fourniture aux réalisateurs d'équipements de spécifications validées et de jeux de tests associés

La communication abordera de façon détaillée :

1 - Le contexte méthodologique dans lequel s'inscrit le maquettage (rappel des étapes du développement des systèmes, description des ateliers systèmes et ateliers logiciels, etc...)

2 - Les expériences réalisées au sein de la société Avions Marcel Dassault - Bréguet Aviation, par l'analyse et la synthèse des travaux de maquettage effectués dans le cadre des systèmes avioniques Rafale A et MIRAGE 2000 NC. l'accent sera mis d'une part sur la mise en oeuvre pratique du maquettage des spécifications et les résultats obtenus, d'autre part sur les utilisations connexes de la maquette en tant que :

  - support d'analyse de pannes
  - banc d'essais pour les évolutions
  - générateur d'éléments de recette
  - référence fonctionnelle pour tous les intervenants

3 - Les tendances pour les maquettages futurs (particulièrement pour le développement du Rafale D) en ce qui concerne notamment la représentativité de la maquette (temps réel, ergonomie, etc...), la généralisation du maquettage aux différentes couches de spécifications et l'évolution des outils utilisés.

1 - INTRODUCTION

Les systèmes avioniques militaires atteignent aujourd'hui un très haut niveau de complexité et représentent la moitié du coût de l'avion d'armes moderne.

L'évolution de ces systèmes au cours des deux dernières décennies peut s'analyser sur les plans opérationnels, technologiques et méthodologiques

- L'évolution opérationnelle est liée à l'accroissement de la polyvalence des systèmes qui se traduit par une plus grande richesse opérationnelle, et par une intégration de plus en plus serrée des fonctions au sein d'un même système et entre systèmes avion ou sol. (Le système est alors la somme de plusieurs systèmes embarqués ou systèmes sol). A l'origine limités à des fonctions opérationnelles traditionnelles telles que navigation, bombardement air/sol et/ou interception air/air, les systèmes se sont enrichis peu à peu d'une panoplie de fonctions spécifiques d'armements sophistiqués, de dispositifs de guerre électronique ou d'équipements de reconnaissance.

L'intégration de ces fonctions est réalisée dans le but d'obtenir une *efficacité opérationnelle maximale* par :

. l'optimisation *des ressources physiques* (capteurs, actionneurs, organes de traitement de l'information) grâce à la fusion de données et aux réseaux d'échanges d'informations entre avions et/ou infrastructures terrestres et maritimes.

. l'optimisation des ressources humaines, grâce à une ergonomie particulièrement soignée de l'interface homme/machine, assurant un dialogue *de haut niveau avec les pilotes*, le système sélectionnant lui-même les informations utiles à chaque phase de la mission et les présentant sous la forme synthétique la plus appropriée.

Le caractère hautement évolutif des systèmes s'affirme de plus en plus. l'enveloppe opérationnelle doit pouvoir évoluer facilement : intégrer de nouvelles fonctions sans modifier la mise en oeuvre opérationnelle des précédentes ou améliorer les fonctions pré-existantes au travers des évolutions technologiques.

- L'évolution technologique des systèmes avioniques se traduit principalement :

. par des architectures fonctionnelles et matérielles de plus en plus complexes intégrant des systèmes historiquement indépendants tels que moteurs, commandes de vol, carburant ou freinage et systématisant l'emploi de liaisons numériques multiplexées entre équipements.

. par l'introduction massive du logiciel, apportant une souplesse *et une ouverture* considérables mais induisant des problèmes spécifiques dont la maîtrise s'avère encore aujourd'hui difficile.

Les systèmes avioniques embarqués actuellement développés par les Avions Marcel Dassault comportent plus d'une centaine d'équipements, dont la moitié est fortement numérisée et dont la majeure partie est fonctionnellement dépendante de logiciels. Le volume de logiciel système embarqué se chiffre en méga-octets, le nombre d'informations échangées entre équipements et/ou modules fonctionnels dépasse 30.000 et le débit d'information sur les bus numériques est de plusieurs méga-bits par seconde.

- L'évolution méthodologique est la conséquence nécessaire des évolutions opérationnelles et technologiques dans le but de conserver la maîtrise du développement de ces grands systèmes. Les grands axes de cette évolution concernent :

. l'approche "système" de la conception
. l'assurance de la qualité en conception
. l'organisation industrielle pour le développement
. les spécificités du logiciel
. l'outillage informatique d'aide au développement

. <u>Approche système</u> : dès les étapes les plus amont de la conception, les systèmes anciens pouvaient être découpés a priori en entités autonomes telles que radio-communication, radio-navigation, commandes de vol, contrôle moteurs ou telles que radar, centrale aérodynamique, etc... chacune de ces entités faisant l'objet d'une conception séparée. L'intégration, la décentralisation et la complexité en ont modifié les règles du jeu. La conception du système, tâche amont des tâches de conception d'équipements, s'affirme comme étant la plus lourde et la plus délicate. Elle s'appuie en particulier sur des techniques d'analyse fonctionnelle aboutissant à des architectures fonctionnelles distinctes des architectures matérielles, le rôle opérationnel de chaque "boîte noire" ne se dégageant plus de façon évidente.

. *<u>Assurance de la qualité en conception</u>* : La qualité du produit (particulièrement la sûreté de fonctionnement) du produit est davantage démontrée par la qualification des méthodes utilisées pour son développement qu'au travers du produit lui-même. les documents d'assurance de la qualité des systèmes ou des logiciels décrivant pour l'essentiel des méthodologies, sont les témoins de cette évolution.

. <u>Organisation industrielle</u> : la taille des systèmes actuels implique *la mise en commun des ressources et des compétences* réparties dans de nombreuses sociétés industrielles. (Le nombre d'intervenants dans le développement d'un système avionique de MIRAGE 2000 dépasse 25.000...). Une méthodologie rigoureuse sert de support aux nouvelles organisations industrielles et permet de définir les tâches et responsabilité de tous les intervenants en renforçant la visibilité.

. <u>Les spécificités du logiciel</u> : Les travaux de conception du logiciel sont de même nature que les travaux de conception du système, l'ensemble devant donc être le fruit d'une démarche *méthodologique continue.* En conséquence, les méthodologies de développement des logiciels doivent être cohérentes de la méthodologie de développement du système.

. <u>L'outillage informatique d'aide au développement</u> : Les méthodologies de développement sont supportées par des outils informatiques d'aide à la conception ou à la validation, regroupés en ateliers. L'émergence de l'approche système cree le besoin d'un atelier système, chapeautant les ateliers logiciels et assurant l'aide à la conception amont des systèmes.

2 - METHODOLOGIE DE DEVELOPPEMENT, ATELIER SYSTEME ET ATELIERS LOGICIEL

Dans ce contexte particulièrement évolutif, la société des Avions Marcel Dassault - Bréguet Aviation a dû, à partir de son expérience des systèmes et en consentant de lourds investissements, s'adapter rapidement pour conserver sa maîtrise des systèmes avioniques complexes :

- En créant une méthodologie de développement de systèmes qui définit les tâches, produits et moyens associés à chaque étape et sert de base à l'assurance qualité.
- En définissant et en réalisant l'atelier système correspondant constitué d'un ensemble cohérent d'outils interconnectés multi-utilisateurs et multiversions.
- En participant à la définition et à l'évaluation des ateliers logiciel.

La branche de définition étapes de gauche de V se révèle comme étant la plus critique. En effet, s'inscrivant dans la partie amont du cycle de vie, toute erreur ou imperfection à ce niveau est amplifiée au cours du cycle et se traduit par des conséquences aval coûteuses et difficilement maîtrisables. De plus, les produits issus de cette phase étant encore essentiellement des documents papiers, la perception du produit final à travers ces simples documents *est délicate et problématique.*

ETAPES DE LA METHODOLOGIE
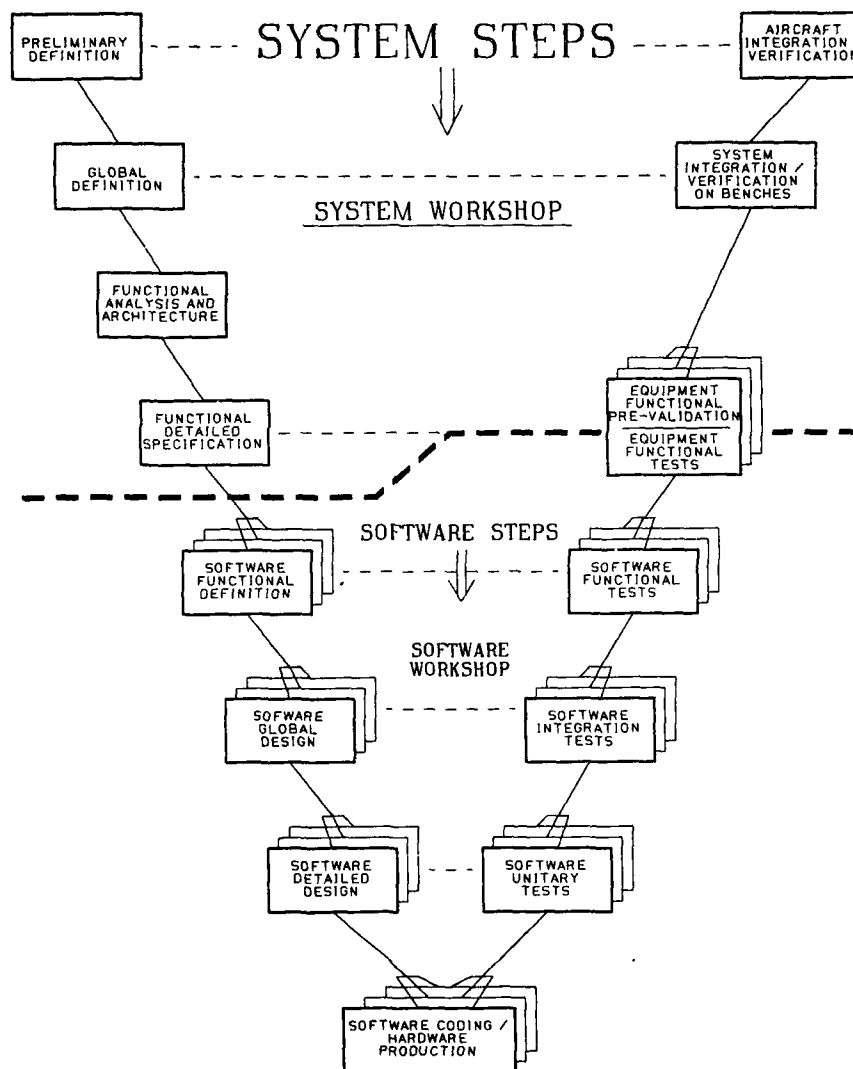
1 - Définition préliminaire

Le rôle de cette étape est de définir le cadre et les hypothèses du développement grâce à une analyse des missions et une première étude de l'organisation du système.

L'analyse des missions s'appuie sur diverses études qui permettront de s'assurer de la faisabilité globale du système. Elle se concrétise par une liste des fonctions opérationnelles nécessaires à la réalisation de l'enveloppe des missions, et fait ressortir un ensemble de besoins concernant particulièrement les principaux capteurs.

L'étude d'organisation du système conduit à une première définition de l'architecture matérielle du système (liste d'équipements et d'emports, organisation matérielle du poste de pilotage, etc...).

2 - Définition globale du système

L'étape de définition globale consiste à définir précisément les services que le système doit rendre (et non pas comment le système sera construit).

Le résultat de l'étape est une description, en terme de scénario opérationnel (donc vu de l'utilisateur), de la mise en oeuvre et du fonctionnement nominal du système pour toutes les fonctions qu'il assure. Cette description se traduit par deux types de documents :

- Documents de règles générales :

Ces documents décrivent de façon unique, les règles et philosophies d'utilisation applicables à toutes les fonctions opérationnelles dont est et sera doté le système. Ils garantissent ainsi une mise en oeuvre cohérente du système dans tous les modes. (Ex. règles générales de dialogue homme-machine, de signalisation des pannes, de superposition des fonctions, etc...).

- Documents de spécification globale

Chaque document de spécification globale décrit le scénario d'utilisation du système relativement à une fonction opérationnelle donnée. Chaque fonction opérationnelle fait donc l'objet d'une spécification qui est écrite dans le respect des règles générales. Ces spécifications étant indépendantes, elles peuvent être élaborées de façon autonome et asynchrone. La cohérence et l'indépendance des spécifications globales sont assurées par les règles générales.

Les règles générales et les spécifications globales sont validées grâce à un outil d'aide à la spécification mettant en jeu l'aspect dynamique. Cet outil, construit autour d'une maquette représentative du poste de pilotage, permet de simuler les séquences, commandes et visualisations permettant ainsi une validation plus efficace des scénarios par les pilotes utilisateurs.

2.3 - Analyse fonctionnelle et architecture

Le rôle de cette étape est de procéder à l'analyse fonctionnelle du système et d'en déduire son architecture fonctionnelle. Le dossier d'architecture fonctionnelle (produit de l'étape) décrit la solution apportée aux besoins exprimés à l'étape de définition globale.

L'étape se décompose en deux phases.

- La construction du graphe d'architecture fonctionnelle

La méthode utilisée consiste en une décomposition hiérarchique progressive du système en éléments fonctionnels, selon des critères précis. Chaque niveau successif de décomposition entraîne :

. une justification du découpage réalisé
. une collection systématique des interfaces induites entre éléments
. un affinage progressif de la définition de ces interfaces par rapport au niveau de décomposition immédiatement supérieur.

Le graphe fonctionnel décrit le système selon une arborescence cohérente supportée par outil. L'ensemble des éléments terminaux de la décomposition, appelés modules fonctionnels, et de leurs interfaces, représentent l'architecture fonctionnelle du système.

Les contraintes prises en compte pour l'établissement du graphe sont :

. des contraintes de qualité (particulièrement l'évolutivité) qui imposent des règles d'indépendance entre modules fonctionnels, ou des contraintes opérationnelles exprimées dans les documents de règles générales. L'analyse de ces dernières permet de dégager la structure du "coeur fonctionnel", ensemble de modules de gestion des ressources du système.

- Projection de l'architecture fonctionnelle sur l'architecture matérielle

Cette phase consiste à intégrer l'architecture fonctionnelle préalablement définie et l'architecture matérielle proposée au cours de la définition préliminaire. les modules fonctionnels sont alors distribués dans les équipements et identifiés (matériel ou logiciel).

Cette projection définit le compromis de réalisation tenant compte de plusieurs facteurs :

. des facteurs technologiques : selon la nature des traitements à effectuer, choix de l'équipement le plus adapté.
. l'optimisation de la connectique : on recherche une distribution des modules minimisant le flux d'échange entre équipements et privilégiant "le plus court chemin" pour les informations critiques d'un point de vue temps réel.
. des facteurs de qualité particuliers : par exemple, les modules supportant des fonctions critiques d'un point de vue sécurité seront regroupés dans un même équipement et/ou redondés dans plusieurs équipements.
. des facteurs déterministes de savoir faire ou d'organisation industrielle

2.4 - Spécification fonctionnelle détaillée

L'étape de spécification fonctionnelle détaillée consiste à établir la spécification des fonctions de transfert de chaque module fonctionnel identifié dans l'architecture.

Ces documents représentent la dernière étape de la phase de définition, directement du rôle et de la responsabilité de l'avionneur. Ils représentent la référence contractuelle vis à vis des coopérants pour la réalisation du logiciel des équipements. Il faut noter que la modularité de cette documentation (un document autonome par module fonctionnel) induit une contrainte de réalisation aiguë, l'architecture logicielle de chaque équipement devant respecter le découpage imposé par l'analyse fonctionnelle du système.

Idéalement, ces documents doivent contenir "tout ce qui est nécessaire et seulement ce qui est nécessaire" à la réalisation des logiciels et des matériels.

C'est à ce niveau de spécification que nous ressentons un besoin constant d'amélioration de la qualité des documents, sur les plans de la forme et du contenu.

Les méthodes et outils classiques utilisés pour l'écriture et la validation des spécifications fonctionnelles détaillées sont :

- L'utilisation de canevas stricts et de langages semi-formels
- La standardisation de la terminologie par le biais de dictionnaires
- L'organisation de relectures croisées systématiques
- L'utilisation d'outils documentaires (traitement de texte (texte + graphique) multiversion).

Depuis 1985, l'éventail des méthodes s'est enrichi de la technique de maquettage des spécifications fonctionnelles détaillées qui présente l'avantage d'assurer à la fois la qualité formelle de la spécification (par nécessité) et sa qualité "fonctionnelle" par effet de miroir.

Le maquettage fonctionnel a été utilisé pour la première fois dans le cadre du programme RAFALE A à l'étape de définition fonctionnelle détaillée.

Il a été reconduit pour le développement du système MIRAGE 2000 New Cockpit et sera appliqué en vraie grandeur grâce à un ensemble d'outils idoines, pour le développement du système ACE/RAFALE D.

2.5 - Développements des logiciels et matériels

Les développements spécifiques des équipements et des logiciels sont des tâches avales et conséquentes des étapes de développement du système résumées plus haut. Ils sont confiés à de multiples sous-traitants et concernent les étapes de développement suivantes :

- Définition fonctionnelle du logiciel
- Conception globale du logiciel
- Conception détaillée du logiciel
- Codage / réalisation des équipements
- Tests unitaires du logiciel
- Tests d'intégration du logiciel
- Tests fonctionnels du logiciel
- Tests fonctionnels de l'équipement

Remarque : les méthodologies de développement de logiciel à partir des spécifications système ainsi que les ateliers logiciels correspondants peuvent être sensiblement différents selon les réalisateurs.

Néanmoins, l'architecte industriel du système doit s'assurer de leur adéquation à la méthodologie de développement du système par des audits ou à travers les normes appliquées par chaque réalisateur.

2.6 - Prévalidation fonctionnelle des équipements

Le but de cette étape est de procéder à une *évaluation fonctionnelle séparée* de chaque équipement développé afin d'aboutir à un niveau suffisant de qualité avant intégration du système.

Le principe consiste à dérouler des scénarios fonctionnels à l'échelon de l'équipement pour vérifier les fonctions implantées dans l'équipement et mesurer *diverses marges de fonctionnement* (temps de tracé pour un équipement de visualisation, charge de calcul et occupation mémoire pour un calculateur, etc...).

Les scénarios d'essais utilisés peuvent être issus de sources variées comme :

- Résultats du maquettage des spécifications
- Scenarios générés au moyen du banc d'intégration du système
- Scénarios enregistrés en vol
- Simulations d'environnement

Les moyens utilisés pour la prévalidation fonctionnelle sont multiples : moyens de développements système, moyens de traitement de l'information et outils spécifiques ou non de génération d'environnement.

2.7 - *Intégration et vérification sur banc d'intégration système*

Le but de cette étape est de s'assurer, avant intégration sur avion, de la conformité du système à sa définition globale, de la cohérence des configurations successives, et d'évaluer le comportement du système dans tout son domaine d'utilisation (domaine avion, cas de pannes, tolérance aux pannes, etc...).

Les bancs d'intégration permettent la mise en oeuvre aisée des équipements constituant le système et de la panoplie d'outils d'intégration. Ils possèdent :

- l'ensemble des *équipements réels* du système intégré
- des moyens de surveillance analogiques et numériques
- des *simulateurs d'interfaces numériques et analogiques*
- une implantation réaliste des équipements de dialogue homme/machine

L'environnement du système avionique embarqué est restitué sur le banc grâce à un complexe informatique permettant les fonctions de stimulation et de simulation. Ces deux fonctions ont pour but de générer un environnement réaliste évoluant avec la même dynamique de paramètres que sur avion.

- La stimulation consiste à rejouer sur le banc un *scénario enregistré* sur avion de façon à mettre le système dans un état identique à celui rencontré en vol
- La simulation consiste à générer un scénario de façon *interactive*, *permettant ainsi de "piloter"* le système et d'étudier ses réponses dans tout son domaine d'utilisation.

Les bancs d'intégration permettent aujourd'hui de procéder à la plupart des tâches de vérification et de qualification des systèmes, réservant les tests en vol aux quelques essais critiques pour lesquels le banc n'est pas suffisamment représentatif.

Leurs avantages sont leurs coût et souplesse d'utilisation (comparés à ceux d'un avion), leur disponibilité, leur facilité d'évolution, leur actuelle représentativité et enfin leur puissance d'analyse sans cesse croissante.

2.8 - *Intégration et vérification sur avion*

Le but de cette étape est de vérifier et de garantir le fonctionnement opérationnel du système embarqué. Les tâches se composent d'essais au sol et d'essais en vol.

Au sol sont conduits des *essais d'intégration* nécessitant l'avion complet, des essais de maintenance, ou des essais de comportement electromagnetique réalisés dans une chambre anéchoïde.

En vol sont conduits des essais d'ouverture de domaines, des essais de séparation d'emports, les essais fonctionnels complémentaires des essais réalisés sur banc d'intégration et enfin des *évaluations opérationnelles particulières* à la demande de l'avionneur lui-même ou de ses client .

Les moyens utilisés sont nombreux et variés, soit non exhaustivement : les avions prototypes, les installations d'essais, les dispositifs de télémesure, les salles d'écoute, les complexes informatiques *temps réel*, les moyens de préparation et restitution de mission et les différents matériels de mise en oeuvre en piste.

3 - <u>APPLICATION AU RAFALE A</u>

3.1 - <u>Contexte</u>

Le système avionique de l'avion démonstrateur RAFALE présente un certain nombre de nouveautés et de spécificités par rapport aux avions de la génération précédente, telles que :

- L'intégration très poussée incluant les systèmes avion
- L'extension du nombre de fonctions nécessaires dès le premier vol
- La décentralisation des fonctions système (réparties dans plusieurs équipements)
- La généralisation de l'emploi des techniques numériques dans des domaines où l'expérience en était faible
- Les délais très courts et très tendus de l'opération.

Pour faire face à cette situation, il a alors été décidé de réaliser un travail de maquettage des spécifications avec les objectifs suivants :

a) Améliorer la qualité des spécifications fonctionnelles détaillées
b) Permettre une validation fonctionnelle de ces spécifications
c) Fournir aux coopérants des jeux d'essais cohérents pour valider aussi tôt que possible leurs développements
d) Disposer d'un banc d'essai pour tester a priori les évolutions.

Ce travail a débuté en Mars 85 avec les étapes suivantes :

a) Analyse critique des spécifications
b) Réalisation de la maquette
c) Exploitation de la maquette

3.2 - <u>Analyse critique des spécifications</u>

1 - <u>But</u>

Les spécifications fonctionnelles détaillées représentent la charnière entre la conception (travail AMD-BA) et la réalisation des logiciels (travail des coopérants). Il est donc important qu'elles soient à la fois :

- Entièrement représentatives des besoins opérationnels du concepteur (aspect fonctionnel)
- Compréhensibles et réalisables par les coopérants (aspect formel)

Le but de l'analyse critique des spécifications est d'améliorer leur qualité pour couvrir l'aspect formel, c'est-à-dire s'assurer que les spécifications sont :

- lisibles
- complètes
- cohérentes entre elles
- sans ambiguïté
- réalisables informatiquement

2 - <u>Principe et organisation</u>

Un spécificateur étant naturellement satisfait de son document grâce à sa connaissance du contexte opérationnel, pour que l'expérience soit rentable il a fallu isoler les lecteurs critiques de ce même contexte en limitant les explications fournies sur les spécifications. Le mot d'ordre a été : <u>ne pas juger de ce que doit faire la spécification</u> (fonctionnel) mais juger uniquement la manière dont elle est écrite et sa faisabilité.

L'équipe de relecture <u>n'a pas eu connaissance du besoin opérationnel</u> exprimé à travers les spécifications détaillées (spécifications globales non fournies).

La totalité des documents de spécification détaillée (3000 pages) a été soumise à l'équipe de relecture critique. Toute fiche d'évolution, quelle que soit son origine, s'est vue appliquer la même procédure de relecture.

3 - <u>Résultats de l'étape</u>

Le nombre de critiques (justifiées) a été extrêmement important : 250 pages de remarques, représentant de l'ordre de 1000 points précis.

Le nombre de critiques par page de spécification (donc in fine la qualité formelle de la spécification) varie <u>considérablement</u> en fonction :

- du <u>rédacteur</u> (rigoureux/non rigoureux, précis/général, structuré/collectionneur de détails)
- du <u>module spécifié</u> (logique/algorithmique)

Les erreurs relevées par la critique entrent toutes dans les trois catégories :

- erreurs de rigueur
- erreurs de généralité
- inadaptation de la spécification à une réalisation informatique.

a) Erreurs de rigueur

- <u>Interface manquante</u> : l'information utilisée par les traitements n'est pas déclarée à entrée de la spécification
- <u>Interfaces incohérentes</u> : les interfaces déclarées en entrée du module spécifié n'existent pas dans le système (non calculées par d'autres modules)
- <u>incomplétude des traitements</u>
  . Le traitement relatif à une sortie déclarée du module n'est pas spécifié
  . Dans une combinaison logique, tous les cas ne sont pas renseignés ; il est à noter que le cas est très fréquent lorsque la logique est exprimée au moyen de phrases (si, alors, sauf, quand) et pratiquement inexistant si la logique est décrite sous forme de tableaux de vérité.
  . Les conditions d'initialisation, d'activation, d'enchaînement des traitements ne sont pas spécifiées.
- <u>Terminologie floue ou ambiguë</u>
  Exemple :
  "on déterminera ..."
  "dans la plupart des cas..."
  "dans certaines conditions..."
  "l'information existe dans les cas suivants..."

b) Erreurs de généralité

- <u>Caractéristiques d'interfaces non spécifiées</u>

  Ne sont pas précisés l'unité, le type (logique, booléen), les valeurs possibles d'une information.

- <u>Traitement decrit trop globalement</u> :

  Ce genre d'erreur est fréquent lorsque le spécificateur surestime le savoir faire (ou l'intuition) des réalisateurs de logiciel.

c) Inadaptation de la spécification à une réalisation informatique

- Choix de la solution informatique : le spécificateur, dans un louable souci de rigueur impose la <u>façon</u> dont doit être réalisé le traitement : connaissant peu les critères de "programmation", le choix est parfois non astucieux et peut déboucher sur un logiciel démesuré ou non évolutif.

4 - <u>Remarques et commentaires sur l'étape d'analyse critique</u>

L'étape d'analyse critique des spécifications a été extrêmement fructueuse et révélatrice. Nombre de problèmes de toute importance ont pu être ainsi résolus a priori, évitant de les reporter à la phase d'intégration. Par contre, l'énergie mise en oeuvre a été également importante : équipe de relecture, temps "volé" aux spécificateurs, lourdeur de mise à jour de la documentation.

La plupart des erreurs recensées sont des erreurs <u>évitables</u> qui ne remettent pas en cause le profil actuel des spécificateurs. En effet, cette phase de relecture a conduit à améliorer une spécification existante, non pas à créer une couche de spécification plus détaillée.

Enfin, l'expérience a été vécue par les spécificateurs comme un contrôle qualité supplémentaire, donc ressentie de façon très mitigée...

3.3 - <u>Réalisation de la maquette</u>

1 - <u>Elaboration des programmes-maquette</u>

La nécessité de coder les spécifications a mis en évidence trois impératifs d'écriture de celles-ci. Les deux premiers sont d'ordre général et concernent toute spécification devant être codée, le troisième est lié à la structure doublée du système RAFALE.

Premier impératif   : complétude des interfaces
Deuxième impératif : définition du type de chaque variable
Troisième impératif : définition pour chaque variable du type de liaison entre module émetteur et module(s) récepteur(s)

D'autre part, l'opération de codage a confirmé la nécessité de description d'un logiciel enveloppe pour chaque module implanté dans un équipement doublé.

<u>Complétude des interfaces</u>

Cette étape est indispensable avant toute opération de codage. Le renseignement <u>complet</u> des interfaces a donc été nécessaire, avant codage du logiciel initial, comme avant chaque passage d'une version à l'autre.

Définition du type de chaque variable

Cet impératif a conduit à enrichir la base de donnée d'interfaces avec la définition, pour chaque variable, d'un type analogue aux déclarations de variables FORTRAN, à savoir :

- Booléen, tableau de booléens, logique, réel, entier.

Nota : La phase d'analyse critique des spécifications avait déjà fait apparaître la nécessité de définition du type de variable.

Codage des modules fonctionnels

Les programmes maquette ont été écrits en FORTRAN, directement à partir de la spécification (après phase d'analyse critique) et en utilisant la base de donnée d'interfaces comme référenciel des variables.

Produits

Le résultat de la phase d'élaboration des programmes-maquette se décompose en :

- Un produit intermédiaire sous la forme d'un fichier de variables de 8 caractères extrait de la base de données d'interfaces. Ce produit constitue en fait un complément de spécification, indispensable pour la génération du code.

- Un produit final composé des programmes-maquette (ou modules) d'une chaîne de calcul. L'obtention de l'ensemble des modules représentatifs des deux chaînes devant être faite grâce à la duplication de ces programmes-maquette.

2 - Adaptation de la console de visualisation des échanges (CVE)

La CVE était, à l'origine, un outil de visualisation des échanges entre équipements numériques. Pour les besoins du maquettage, il a fallu faire tendre cet outil de visualisation vers un outil de validation de spécifications. Les travaux d'adaptation ont porté sur :

Au niveau conversationnel

- Le pilotage de la simulation (choix du mode simulation, des modules à activer, etc...)
- Le développement de procédures d'entrée de valeurs à la CVE
- L'amélioration de la gestion des chaînes mémorisées (concaténation de chaînes, appel nominatif de celles-ci).
- Le développement des procédures de tests automatiques

Au niveau système

- La génération des lexiques CVE

  Ces lexiques sont nécessaires au bon fonctionnement de la simulation et à l'exploitation des résultats de celles-ci.
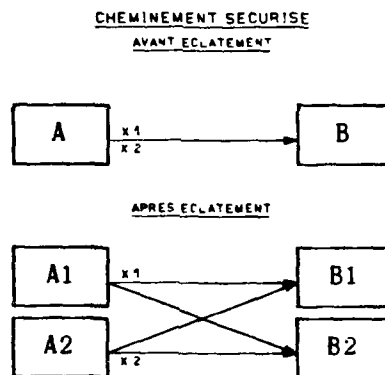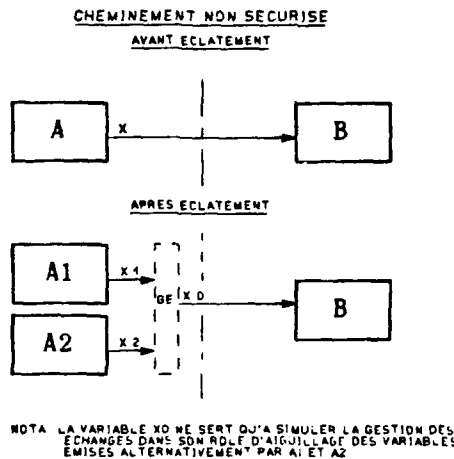
  Ils comprennent :

  . la liste des modules et leurs adresses
  . la correspondance des fichiers 8 caractères par rapport aux fichiers de la base de donnée d'interface initiale (40 caractères)
  . la liste des pavés récepteurs d'une information donnée

3 - Mise en oeuvre de la simulation

Les caractéristiques essentielles de la simulation mise en place sont les suivantes :

- Simulation mono-fréquence
- Cycle de simulation correspondant à l'activation séquentielle des modules maquettés
- Prise en compte de l'architecture double-chaîne du RAFALE :
  Cette prise en compte se résume à l'établissement de deux procédures :
  . Eclatement des variables
  . Duplication des programmes

Afin de générer l'ensemble des variables émises et reçues dans les équipements des chaînes 1 et 2, il a été nécessaire, à partir du fichier standard 8 caractères, d'éclater les variables pouvant être émises par deux équipements symétriques (voir schémas ci-dessous).

CHEMINEMENT NON SECURISE

AVANT ECLATEMENT

```
┌─────┐                    ┌─────┐
│  A  │ x                  │  B  │
└─────┘                    └─────┘
```

APRES ECLATEMENT

```
┌─────┐                        ┌─────┐
│ A1  │ x1  ┌─ ─┐              │  B  │
└─────┘     │GE │ x0          └─────┘
┌─────┐     └─ ─┘
│ A2  │ x2
└─────┘
```

NOTA LA VARIABLE X0 NE SERT QU'A SIMULER LA GESTION DES
      ECHANGES DANS SON ROLE D'AIGUILLAGE DES VARIABLES
      EMISES ALTERNATIVEMENT PAR A1 ET A2

CHEMINEMENT SECURISE

AVANT ECLATEMENT

```
┌─────┐ x1              ┌─────┐
│  A  │ x2    ────────▶ │  B  │
└─────┘                 └─────┘
```

APRES ECLATEMENT

```
┌─────┐ x1          ┌─────┐
│ A1  │     ╲    ╱   │ B1  │
└─────┘      ╲  ╱    └─────┘
┌─────┐       ╳
│ A2  │ x2   ╱  ╲    ┌─────┐
└─────┘              │ B2  │
                     └─────┘
```

3.4 - 1ère exploitation maquette - tests unitaires et manuels

Définitions :

Tests unitaires : tests portant sur les variables d'Entrées/Sorties d'un module unique
Tests manuels   : tests pour lesquels les valeurs des variables d'Entrées doivent être modifiées
                  manuellement par le spécificateur.

Le déroulement de ce type de test est le suivant :

1) Constitution d'une chaîne
2) Entrée des valeurs à la CVE
3) Déclenchement d'un pas de simulation et lecture des résultats

1 - Constitution d'une chaîne

Cette opération consiste à sélectionner un certain nombre de variables parmi le total des variables d'E/S d'un module donné et à les regrouper dans un ensemble appelé chaîne. Suivant la taille de ce module ce choix a été fait :

Pour les modules de taille importante

En sélectionnant toutes les variables d'Entrées/Sorties se rapportant à une entité donnée.

Pour les modules de taille réduite

La taille des sous-modules de ces spécifications permet d'utiliser toutes les variables de ces sous-modules pour constituer les chaînes. Chaque chaîne est l'image d'un sous-module.

Cette méthode permet :

- de retrouver chaque variable de sortie du module dans une chaîne
- de disposer dans chaque chaîne de toutes les informations utilisées pour élaborer les variables émises par le sous-module.

On peut ainsi se rapprocher de la démarche visant à valider les pièces de spécification de niveau le plus bas avant de passer au niveau supérieur.

2 - Entrée des valeurs à la CVE

Cette opération est réalisée en désignant la variable à modifier dans le liste des variables de la chaîne, en sélectionnant le mode "modification de valeur", puis en entrant la nouvelle valeur.

Bien que la modification manuelle de valeur de n'importe quel type de variable soit possible, la plupart des tests ont porté sur des modifications de variables booléens VRAI/FAUX, et de variables logiques.

Quelques tests ont porté sur des modifications de variables numériques soit pour vérifier des logiques (déclenchement de seuils, temporisations...), soit plus rarement, pour valider des fonctions de transfert numériques.

3.5 - 2ème exploitation maquette - tests automatiques

1 - Intérêt des tests automatiques

L'utilisation des tests manuels a révélé plusieurs limitations de ces derniers :

1) Difficultés de manipulation des chaînes comportant un nombre important de variables.
2) Inadaptation de ces tests pour la recherche de dépendances entre variables.
3) Difficultés de validation des mécanismes mettant en jeu des transitions ou des mémorisations.

Ces limitations ont amené à envisager le développement de tests automatiques qui permettraient à la fois :

- de générer automatiquement des combinaisons de valeurs d'entrée pour les chaînes à tester
- de faciliter l'exploitation des résultats par des éditions appropriées sur listings.

Génération automatique de valeurs d'entrée

Les objectifs visés correspondent aux limitations mentionnées plus haut pour l'exploitation des tests manuels :
- Pouvoir balayer toutes les combinaisons des variables d'entrée choisies et observer leur impact sur toutes les variables émises par le module pour détecter d'éventuelles dépendances anormales entre variables.
- Mettre au point des scénarios nominaux permettant de simuler différentes configurations d'initialisation, des enchaînements de phases de vol mettant en jeu des transitions ou des mémorisations.

Les premiers tests ainsi mis en place appelés tests mono-variables ont permis de tester toutes les combinaisons déduites d'une combinaison initiale en faisant varier une variable booléenne de la chaîne.

La validation des transitions et mémorisations a donné lieu à la création du type de test "SCENARIO DE PANNE". Ce type de test permet d'introduire à des intervalles de temps traduits en nombre de pas de simulation des jeux de valeurs d'entrée préparés à l'avance.

La vérification exhaustive de tous les cas possibles de valeurs d'entrée d'une chaîne de variables booléennes donnée a été rendue possible grâce aux tests combinatoires.

En tout 7 types de tests automatiques ont été développés :

a) Combinatoire statique
b) Combinatoire séquentiel
c) Mono-variable statique
d) Mono-variable séquentiel
e) Aléatoire statique
f) Aléatoire séquentiel
g) Scénario de panne

Nota : Un test statique se distingue d'un test séquentiel par la remise de toutes les variables à la valeur d'initialisation après chaque pas de simulation.

Facilités d'exploitation des résultats

Parallèlement au développement de ces types de tests, les possibilités de sortie des résultats ont été étendues :

- Possibilités d'impression ou non de la "référence" (c'est-à-dire de la chaîne avec ses valeurs d'initialisation).
- Possibilité d'éditions sélectives des variables ayant varié par rapport au cycle précédent.
- Possibilité d'éditions sélectives des variables ayant varié par rapport à la combinaison de référence.

Ces possibilités ont permis en particulier de manipuler des chaînes constituées d'un grand nombre de variables, sans pour autant avoir à rechercher les résultats significatifs dans la liste de toutes les variables constituant la chaîne.

2 - Exploitation des tests automatiques

La phase d'exploitation des tests automatiques a démarré en automne 85 et a comporté deux parties :

- Tests unitaires automatiques
- Tests globaux automatiques

Tests unitaires automatiques

Suivant la nature de la spécification, deux types de tests ont été principalement utilisés :

- Tests combinatoires
- Scénarios de panne

Pour les spécifications décrivant des mécanismes de logique sans mémorisation ni problèmes d'initialisation, la majorité des tests a été du type combinatoire statique (spécifications de visualisations notamment).

Pour les spécifications décrivant un nombre important d'états, de transitions et de temporisations, des scénarios de panne ont été généralement employés (spécification de signalisation des informations moteur ou commandes de vol notamment).

Tests globaux

Définition
- Test global : test consistant à activer l'ensemble des modules maquettés.

Nota : Ce type de test, disponible également en mode manuel n'a été utilisé pratiquement qu'en mode automatique.

Intérêt

L'intérêt des tests globaux est de valider le comportement de l'ensemble des modules maquettés.
En effet chaque spécification peut être vérifiée en théorie, à la seule lecture du document de spécification lui-même.

Par contre, il n'existe pas de document décrivant la répartition précise des traitements entre les différentes spécifications et assurant ainsi que la mise bout à bout des différents modules conduise au respect de la spécification globale.

3.6 - Bilan

1 - Ressources spécifiques mises en oeuvre

L'analyse des spécifications et leur codage dans la maquette ont requis 50 Homme-Mois, dont près de 50 % pour la phase d'analyse critique des spécifications. Le développement de la chaîne d'outils ainsi que la mise en oeuvre de la maquette ont nécessité 22 Hommes-Mois.

Sur le plan matériel, la maquette a été *réalisée* sur un *système* informatique GOULD (SEL 32/27) dont il a été nécessaire d'augmenter la puissance au cours du développement en raison de la dégradation du temps de réponse. Il faut souligner que la *réalisation de la maquette a* pu se faire dans les temps impartis grâce à l'utilisation de l'expérience acquise lors de développements antérieurs dans le domaine de la simulation.

2 - Répercussions sur la méthode de travail

L'introduction du maquettage a modifié la méthode suivie pour développer le logiciel avionique RAFALE par rapport aux habitudes des programmes précédents. Cette modification concerne *essentiellement* :

- la déclaration au niveau fonctionnel des interfaces entre modules (y compris modules d'un même équipement)
- l'écriture d'une spécification avec une <u>contrainte de maquettabilité</u>
- la possibilité de voir "vivre" une spécification

Déclaration des interfaces entre modules

La cohérence imposée au niveau des interfaces par la saisie de celles-ci sur l'outil de base de données (première étape du maquettage) a induit une plus grande rigueur dans le dialogue entre spécificateurs. En effet l'obligation de rentrer chaque interface dans une référence unique et regroupant l'ensemble des interfaces a évité la plupart des redondances *d'informations (ou les origines d'informations* inconnues) que la dispersion des interfaces aurait risqué d'entraîner.

Ecriture des spécifications avec une contrainte de maquettabilité :

La nécessité de décrire des traitements pouvant être transcrits sans ambiguïté en code exécutable (en l'occurrence FORTRAN) a permis d'éviter des retards dus aux *difficultés rencontrées par les* coopérants dans la lecture de spécifications détaillées trop "générales".

Possibilité de voir vivre une spécification

Lors d'application de fiches de *modifications*, la possibilité de valider presque immédiatement celles-ci a permis de "resserrer" le lien entre le spécificateur et son produit. En effet, bien qu'il soit pratiquement toujours possible de *tester une modification* (ou un *mécanisme* de façon générale) mentalement ou sur le papier, l'utilisation d'un outil accomplissant lui-même l'effort nécessaire à l'exécution de la fonction de transfert a déchargé d'autant le *spécificateur*, lui permettant ainsi de se consacrer à l'interprétation des résultats.

Par contre le bilan de l'opération maquettage fait apparaître des contraintes ayant limité l'intérêt de celle-ci, en ce qui concerne le programme *RAFALE. Ces contraintes* sont de deux types :

- Ergonomie de la maquette
- Délais et disponibilité des utilisateurs

Ergonomie de la maquette

Malgré les efforts importants d'aménagement du conversationnel de la maquette, la présentation très désincarnée des informations a rapidement modéré l'*empressement* des utilisateurs pour ce nouvel outil. Il est en effet difficile de valider le comportement d'un ensemble de réticules (à plus forte raison d'une page complète de réticules), à l'*aide de VRAI/FAUX ou de* variables logiques codées. Cette présentation quelque peu rébarbative n'a pas permis d'exploiter complètement les possibilités pourtant importantes de la maquette.

Délais et disponibilités des utilisateurs

Les périodes de disponibilité des utilisateurs de la maquette n'ont pas toujours coïncidé avec les périodes où il aurait été le plus profitable d'exploiter celle-ci. En particulier les *tests globaux*, intérêt majeur de la maquette, n'ont pas eu l'importance qu'ils auraient dû avoir, en raison de l'exploitation tardive de ceux-ci.

3 - Répercussion sur la qualité du logiciel avionique

L'analyse de l'impact du maquettage sur la qualité du logiciel fourni aux bancs d'intégration est rendue difficile pour les deux raisons suivantes :

- Le maquettage est l'une des composantes de la méthode de travail, au niveau des spécifications *fonctionnelles détaillées* ; le résultat obtenu est donc inhérent à l'ensemble de la méthode, l'impact purement maquette étant quasiment impossible à extraire.
- La nouveauté des fonctions *assurées* par le logiciel, la nouveauté de l'architecture et le caractère d'avion démonstrateur font du Système Avionique RAFALE un cas particulier, rendant toute comparaison délicate par rapport aux systèmes précédents.

Néanmoins, l'analyse des erreurs rencontrées à l'étape d'intégration (vérification du système permet de mettre en évidence un impact majeur du maquettage :

A nombre donné d'erreurs, la part due aux erreurs de spécification est tombée de 2/3 (pour les systèmes précédents), à 1/3 dans le cas du RAFALE.

4 - APPLICATION AU 2000 NC

4.1 - Contexte 2000 NC

Ce contexte est sensiblement différent de celui du RAFALE A tant sur le plan des caractéristiques du système avionique que sur le plan méthodologie.

- Le système avionique 2000 NC représente une évolution des systèmes avioniques M2000 par un changement de l'interface homme-machine aux niveaux matériel et fonctionnel (systématisation des écrans de visualisation multiplexés).

Par contre les quelques 40 fonctions opérationnelles supportées par le système (particulièrement les conduites de tri) sont les mêmes que celle du M2000 classique.

- Les "plus" méthodologiques 2000 NC

. au niveau de l'analyse fonctionnelle

Elle a été réalisée grâce à un outil d'aide à la conception (OCS) qui permet de supporter la décomposition hiérarchique et d'assurer la collection cohérente des interfaces fonctionnelles entre modules.

. au niveau de la spécification fonctionnelle détaillée

a) les documents ont été écrits puis gérés en configuration grâce à un outil permettant :

. la composition (textuelle et graphique) standard des documents
. une gestion centralisée de l'ensemble de la documentation
. une édition de fiche de modification et une remise à jour instantanée automatique de tous les documents.

b) Chaque document a été rédigé en équipe intégrée, c'est-à-dire que les coopérants ont participé (sous responsabilité de l'avionneur) à l'écriture des spécifications les concernant directement, avec les conséquences suivantes :

. La présence des coopérants dans l'équipe de spécification permet d'éviter de spécifier des fonctions non réalisables.
. De plus, les coopérants, dans un souci de sauvegarde de leur savoir-faire poussent à une spécification en termes de besoin et non en termes de solution informatique, ce qui conduit à la recherche d'un compromis, qui a atteint sur le programme 2000 NC par les décisions suivantes :

* les spécifications seront autant que possible écrites en "bon français" afin de rester lisibles et de permettre leur compréhension globale par le réalisateur, qui sera ainsi en mesure d'apporter une certaine "expertise" lors de la réalisation. De plus, une fonction donnée pouvant souvent être décrite comme un traitement générique s'appliquant sur toute une collection de données diverses, on s'efforcera dans les SDF de séparer physiquement les traitements (Mécanismes) des données qu'elles manipulent. Cette manière de procéder permet d'améliorer encore la lisibilité, tout en assurant la complétude : l'analyse nécessaire pour bien séparer traitement et données amène en effet un recensement complet de celles-ci.
* les interfaces des "pavés" devront être complètement répertoriés, afin de permettre la "visibilité" du logiciel. Par contre, la nature "informatique" de celles-ci est prise en compte en dehors des SDF.

4.2 - Décisions et objectifs du maquettage

Compte tenu du contexte MIRAGE 2000 NC, un maquettage global n'a pas été envisagé pour des raisons évidentes de redondance, une grande partie des modules fonctionnels étant "récupérés" du développement précédent.

Le maquettage a donc porté sur les modules fonctionnels relatifs aux fonctions de dialogue homme-machine, facilement identifiables au niveau de l'architecture fonctionnelle et représentant des traitements entièrement nouveaux, soient :

. le module "affectation des commandes" :

Cette fonction aiguille les diverses commandes vers les fonctions utilisatrices, en tenant compte des choix du pilote et de l'état du système.

. le module "affectation des visualisations"

Cette fonction choisit les visualisations présentées sur les divers écrans de la cabine, en tenant compte des choix du pilote et de divers critères opérationnels.

. le module "activation des fonctions"

Cette fonction choisit les fonctions à activer, en tenant compte des demandes du pilote et de critères d'exclusion mutuelle des fonctions.

L'objectif étant de vérifier que dans tous les cas opérationnels envisageables, les fonctions correspondant aux besoins de la mission seraient activées, qu'elles recevraient correctement les commandes pilotes et que les visualisations leur seraient affectées conformément aux besoins.

Le processus de maquettage a été déclenché en Janvier 87 pour être terminé fin Avril, soit suffisamment tôt pour que les éventuelles erreurs de spécifications puissent être corrigées avant la fourniture du premier logiciel réel.

4.3 - Analyse critique des spécifications

A été soumis à l'analyse critique un ensemble constitué par les SDF des trois fonctions à maquetter, soit environ 500 pages de spécification. Y a été adjoint un document décrivant les spécificités de la maquette, en effet, les SDF fournies décrivaient la première livraison du logiciel, livraison qui ne disposait pas de toutes les fonctions possibles ; en conséquence, le document de spécificités décrivait le traitement à réaliser pour les fonctions non intégrées dans la première livraison. On notera toutefois que ce document se limitait à la description de tables de données, les mécanismes des traitements ayant été décrits dans les SDF et ne devant pas être affectés par l'introduction de nouvelles fonctions.

L'étape d'analyse critique a été aussi fructueuse que dans le cas du RAFALE A mais beaucoup plus rapide. En effet, l'utilisation systématique du traitement de texte pour l'écriture des SDF, l'équipe intégrée et l'attention toute particulière apportée à la cohérence des interfaces ont permis de limiter l'importance du travail de relecture.

De plus, l'expérience a été mieux ressentie par les spécifications pour les raisons suivantes :

* Le principe même de l'équipe intégrée suppose que les spécifications soient écrites en collaboration, avec lecture croisée et critique pendant l'écriture.
* Les spécificateurs concernés avaient bénéficié de l'expérience RAFALE et étaient intimement convaincus de la nécessité de spécifications complètes, cohérentes.

4.4 - Réalisation de la maquette

Idem RAFALE A.

4.5 - Exploitation maquette

La procédure utilisée pour le RAFALE A a été aménagée de la façon suivante pour limiter le temps "volé" aux spécificateurs.

1) Prise de contact avec la maquette et test manuel de quelques cas dans les 3 pavés maquettés

2) Ecriture de scénarios de pannes pour tester les mécanismes du pavé d'affectation des visualisations de manière aussi complète que possible. Ce pavé est en effet, du fait des impératifs opérationnels qu'il doit satisfaire, une collection de cas particuliers, dont l'interaction pouvait poser problème.

3) Test global des mécanismes de l'ensemble des pavés maquettés. Simultanément, un ensemble de tests automatiques était lancé pour valider complètement les tables de valeurs utilisées.

Cette procédure a été justifiée par les idées suivantes :

1) Le test manuel étant très lourd en charge de travail, il ne doit normalement pas être utilisé.

2) Si un traitement est décrit sous la forme d'un mécanisme agissant sur une table de données, ce qui était souvent le cas, et si ce traitement n'est pas satisfaisant, deux cas se présentent :

- une des données est mal choisie : il suffit de la changer pour revenir à un fonctionnement correct, ce qui peut être rapide.
- le mécanisme lui-même est en cause : il faut le réanalyser et en tirer les conséquences sur les données traitées, ce qui est forcément assez long.

Il est donc judicieux de valider au plus tôt les mécanismes.

3) L'affectation des commandes et l'activation des fonctions se résument à des mécanismes très simples agissant sur de nombreuses données. Il est donc judicieux de commencer le test par l'affectation des visualisations.

4) Les commandes et l'état de sélection des fonctions se traduisent par des visualisations caractéristiques. En validant l'affectation des commandes et l'activation des fonctions en test global avec l'affectation des visualisations, on profite de la mise en forme effectuée par celle-ci pour un plus grand confort d'utilisation.

4.6 – Bilan

1 – Ressources spécifiques mises en oeuvre

L'analyse des spécifications et leur codage dans la maquette ont requis 9 Hommes-mois (équipe de relecture et réalisation), dont 25 % pour la phase d'analyse critique des SDFO, 25 % pour le codage et 50 % pour la mise au point, la validation et le suivi de la maquette.

Le suivi de la chaîne d'outils ainsi que la mise en oeuvre de la maquette ont nécessité 1 homme-mois.

Les tâches de coordination et de suivi de la maquette ont nécessité 1 homme-mois.

La validation des spécifications par les spécificateurs a nécessité 2 hommes-mois.

Sur le plan matériel la maquette a été réalisée sur un système informatique GOULD (SEL 32/27). Il faut souligner que la réalisation de la maquette a pu se faire dans les temps impartis grâce à l'utilisation de l'expérience acquise lors des développements de la maquette RAFALE A.

2 – Méthode de travail

L'introduction du maquettage a permis de confirmer la validité de la méthode suivie pour développer le système avionique MIRAGE 2000 NC. En particulier :

- La conjonction de l'écriture des spécifications sous la forme mécanismes + tables et de la description des traitements en bon français a prouvé sa valeur. Elle a en effet permis d'atteindre un consensus avec les coopérants sur le niveau de spécification et d'obtenir des spécifications lisibles mais susceptibles d'être réalisées directement par un informaticien sans expérience aéronautique.
- L'attention apportée à la définition des interfaces entre pavés a porté ses fruits : on ne retrouve pas les erreurs du RAFALE (oublis d'interfaces, interfaces inutiles...).
- Enfin, l'utilisation systématique du traitement de texte pour l'écriture des spécifications, et la correction de ces spécifications par les spécificateurs eux-mêmes a permis de livrer des documents de meilleure qualité dans un meilleur délai.

Par ailleurs, l'équipe intégrée, en apportant une vision critique dans l'écriture des spécifications, a permis d'éviter un certain nombre d'erreurs, ce qui a pu être constaté au maquettage (pas de spécification irréalisable sur le plan informatique).

3 – Répercussion sur la qualité du logiciel

Un des objectifs majeurs du maquettage étant de s'assurer a priori de l'adéquation des spécifications relatives à l'interface homme/machine à toutes les fonctions opérationnelles dont serait à terme doté le système, il convient d'attendre la qualification du dernier standard système pour établir un bilan définitif.

Néanmoins, nous avons pu noter à ce jour :

- une confirmation de la réduction des erreurs de spécification découvertes en phase d'intégration (moins d'un tiers des erreurs découvertes).
- la stabilité de la spécification obtenue (et donc l'évolutivité du système réalisé) : l'adjonction continue des fonctions nouvelles, standard par standard n'a conduit à ce jour à aucune modification des mécanismes maquettés.

5 – CONCLUSIONS ET ENSEIGNEMENTS DU MAQUETTAGE DE SPECIFICATIONS FONCTIONNELLES DETAILLEES

5.1 – Ce que le maquettage permet

a) Amélioration des spécifications

Contrôle de forme

La relecture critique permet de corriger et de compléter la spécification pour l'amener à un niveau de qualité autorisant son codage direct par des informaticiens (équivalent manuel des futures spécifications formelles où l'outil se charge de la relecture et permet une compilation).

Le maquettage agit à ce niveau comme un puissant révélateur d'erreurs et un contrôle qualité efficace.

L'outil maquette permet une réelle validation fonctionnelle de la spécification au niveau module, équipement ou système : validation horizontale (tests exhaustifs d'un module) ou verticale (test d'une chaîne fonctionnelle donnée).

b) Génération de jeux de test ou d'éléments de recette

Les tests statiques et dynamiques réalisés pendant la phase de maquettage sont réutilisables dès les étapes de vérification (branche remontante du V).

A titre expérimental dans le cadre du RAFALE A et de façon plus systématique dans le cadre du 2000 NC, ces tests ont été utilisés comme aide à la mise au point des équipements (fourniture aux coopérants). De plus, ils ont servi de base à la prévalidation fonctionnelle des équipements, en amont de l'étape d'intégration.

En effet, la maquette était représentative de l'architecture du système réel, elle permet de générer des jeux d'essais fonctionnels (combinaisons d'entrées/sorties) au niveau désiré (module, équipement ou système).

c) Disposition d'une référence fonctionnelle validée

Une fois validée, la maquette est entretenue de toutes les modifications apportées au système et reste donc une référence à laquelle on peut se reporter. Il est alors possible de l'utiliser (sous réserve d'une ergonomie suffisante) comme une "documentation vivante" et représentative du système, portable et ne nécessitant pas de matériel spécifique ; ce à des fins pédagogiques, didactiques ou trivialement pour analyse et contrôle de cas de fonctionnement non nominaux du système.

d) Support de test "a priori" des évolutions

La qualité des spécifications autorise l'analyse et le choix des évolutions, directement en terme de solution, permettant ainsi une connaissance a priori de leur impact sur le logiciel. La maquette elle-même peut servir de banc d'essai pour tester fonctionnellement les-dites évolutions et s'assurer de leur adéquation aux besoins avant leur mise en chantier. Enfin, le correspondance document de spécification/module de logiciel permet de prévoir et de planifier la récupération de logiciel. Elle permet en outre, après analyse, de disposer de gabarits en matière de volume de logiciel, de charge calcul,... ou même coût.

e) Efficacité de la spécification

La conséquence directe du maquettage est de disposer de spécifications autonomes et auto-suffisantes pour réaliser et tester le logiciel, module par module ou équipement par équipement.

Pour des systèmes très décentralisés de type RAFALE, ceci permet de fournir à chacun des intervenants les éléments précis, nécessaires et suffisants à la tâche qu'il doit conduire. Ce point, qui concourt à la simplicité et à la clarté des documents, permet également une bonne protection des informations confidentielles.

f) Amélioration de la qualité du produit final

Outre le gain de qualité obtenu pour l'étape de spécification fonctionnelle détaillée, l'exploitation de la maquette, particulièrement l'utilisation des tests automatiques, permet et a permis d'améliorer la confiance dans le produit final. En effet des centaines de tests ont été réalisés relativement à des configurations avion non nominales, permettant ainsi de découvrir (et de remédier) à des situations non prévues a priori dans la spécification. De même la recherche d'un profil de pannes donné, par balayage combinatoire systématique de toutes les entrées a permis de conforter les analyses de panne et les solutions choisies.

5.2 - Ce que le maquettage implique

a) Une gestion rigoureuse et outillée des interfaces (18000 dans le cas du RAFALE)

La collection des interfaces entre modules et entre équipements représente le cœur de la "machine" ; c'est à partir de ces données que vont être créées automatiquement les variables logicielles de la maquette, lesquelles pourront être stimulées et scrutées en phase de validation.

Il faut donc disposer, avant de commencer le maquettage, d'une description complète et cohérente de ces interfaces, ce qui demande, à une extrême rigueur, un gros travail de synthèse et de rapprochement.

Cette base de donnée a été fournie directement par l'outil support de l'analyse fonctionnelle.

b) Une description précise et complète des fonctions de transfert

Pour les modules que l'on désire valider, la réalisation du logiciel maquette correspondant consiste à décrire par du code exécutable (manuellement dans le cadre du RAFALE) les fonctions de transfert extraites du document de spécification. Il est donc indispensable que cette description soit suffisamment fine pour pouvoir exprimer chaque sortie du module selon une combinaison mathématique des entrées ; dans le cas où la description est trop générale, il y a valeur ajoutée entre la spécification et la maquette, ce qui est contraire au principe de validation des spécifications.

A noter que dans le cas du 2000 NC, la forme des spécifications (mécanismes + tables de constat) a grandement facilité leur analyse.

c) Une gestion de la documentation et de la configuration

L'abondance et la variété de la documentation, la nécessité de sa remise à jour systématique (toute évolution ne peut être décrite que par modification cohérente et immédiate de la documentation concernée), exigent un support de documentation souple et efficace.

Le parallélisme indispensable entre les états de définition :

- documentation (à tous niveaux)
- logiciel maquette
- logiciel réel

est obtenu grâce à une méthode de travail intégrant l'usage d'outils de gestion de la documentation et de la configuration.

5.3 - Ce que le maquettage enseigne

1 - Sur le maquettage lui-même

a - Le maquettage n'est qu'un composant d'une méthode de travail

Il est clair qu'un maquettage isolé de son contexte n'a aucun sens. En effet, si une méthode de travail appropriée n'est pas utilisée pour l'écriture des SDF, celles-ci ne seront pas écrites sous une forme autorisant leur traduction directe sous forme informatique. Il devra alors y avoir une "valeur ajoutée" lors de la réalisation de la maquette, ce qui ne garantit plus la qualité des spécifications.

b - Le maquettage doit être précoce

Un maquettage réalisé après que les SDF aient été fournies aux coopérants ne permet pas de retirer tous les bénéfices attendus. En effet, toute erreur détectée ne pourra être corrigée que par une procédure de fiches d'évolution, qui n'est pas gratuite dans le cas général.

De plus un maquettage réalisé a priori permet dans les cas difficiles de tester plusieurs hypothèses afin de choisir la meilleure (cf "un test a priori des évolutions").

c - Le maquettage doit être intégré dans une lignée d'outils

Le maquettage ne nécessite une charge de travail raisonnable que s'il est intégré dans une lignée d'outils informatiques. Il n'est en effet envisageable et profitable que si les interfaces sont définies a priori à l'aide de l'outil approprié, et si un outil de documentation ad hoc garantit un strict parallélisme entre les documents et le logiciel.

d - L'ergonomie de la maquette doit être soignée

Afin que le bénéfice retiré de la maquette soit maximal, il faut que les utilisateurs n'aient à se préoccuper que de la validation, et ne se perdent pas dans le dialogue avec la machine.

Pour les futurs développements il semble nécessaire, tout en conservant les principes de la CVE, qui sont sains, de développer une interface utilisateur plus "intuitif", permettant de désigner directement les variables à l'écran, de changer leur valeur plus simplement, voire de simuler à l'écran les faces avant des postes de commande ou de visualisation de l'avion, permettant ainsi à des personnes non strictement spécialistes d'une fonction, mais ayant une vision opérationnelle, de tirer également bénéfice de la maquette.

e - La réalisation de la maquette est un travail d'informaticien

Les maquettes RAFALE A et 2000 NC ont été réalisées par des informaticiens sans expérience aéronautique particulière, issus de sociétés de services.

Par contre, le profil des "maquetteurs" doit comporter une expérience des méthodes de développement de logiciel afin de garantir la qualité des programmes maquette.

2 - Sur la méthode d'écriture des spécifications

a - La forme des SDF influe sur leur qualité

La maquette a permis de constater que la forme mécanismes + tables utilisée pour les spécifications apporte une amélioration de la qualité formelle des SDF, d'une part en forçant l'auteur à systématiser ses traitements plutôt que d'en faire une collection de cas décrits successivement, d'autre part en fournissant une vision globale des traitements, permettant ainsi de détecter plus facilement les "trous".

b - Doit-on utiliser un langage formel de spécification ?

L'expérience du maquettage du MIRAGE 2000 NC prouve que des spécifications "en bon français" complétées par des tables de constantes peuvent être traduites facilement bien que manuellement, en langage exécutable.

Néanmoins, de façon à réduire le temps de réalisation des programmes maquette, il serbie souhaitable à terme de mettre en œuvre un langage formalisé de spécification, qui seul garantira le maquettage automatique. Dans ce cas, ce langage devra être un composant intégré d'un système d'outils gérant également les interfaces à tous les niveaux (fonctionnel, bus,...).

6 - CONCLUSION ET AVENIR DU MAQUETTAGE

Les systèmes avioniques complexes sont aujourd'hui soumis à des exigences contraignantes en matière de qualité et de performances.

La prise en compte et la réalisation des objectifs assignés aux systèmes passent par une maîtrise du processus de développement.

Cette maîtrise est obtenue par le respect d'une méthodologie rigoureuse, recouvrant de façon continue la totalité du cycle de développement et qui est à la base de toutes les actions Qualité.

Cette méthodologie est supportée par une panoplie d'outils constituant l'atelier système et les ateliers logiciel.

La technique de maquettage des spécifications fonctionnelles détaillées a vu son efficacité démontrée au travers des expériences RAFALE A et 2000 MC. Elle est donc à ce jour partie intégrante de la méthodologie de développement et sera utilisée pour les futurs programmes, en particulier RAFALE D et HERMES.

Les travaux entrepris à ce jour pour en tirer le maximum de bénéfices sont articulés selon quatre axes :

1) Intégrer le maquettage dans une stratégie globale de simulation.

   Les techniques de simulation sont couramment utilisées pendant de nombreuses étapes du développement.

   - en phase avant-projet des modèles permettent de dégager les concepts opérationnels et de fixer des hypothèses réalistes pour le développement.
   - à l'étape de spécification globale, une simulation de comportement du système sert d'aide à la conception et de moyens de validation des spécifications de besoin opérationnel.
   - à l'étape de spécification fonctionnelle, les simulations réalisées à l'occasion du maquettage permettent de valider l'architecture fonctionnelle et les spécifications détaillées des constituants du système.

   Le but est d'établir une continuité de tous ces travaux et l'utilisation de ressources communes et/ou couplables.

2) Utiliser les produits issus de maquettage pour les étapes de vérification dans un centre de simulation hybride.

   Le travail consiste à compléter le maquettage fonctionnel pour le rendre capable de s'interfacer avec les équipements réels, donc de simuler le comportement physique et dynamique des dits équipements . Ceci permettra une vérification pragmatique du système, chaque équipement recetté venant "remplacer" sa maquette correspondante au sein de la simulation hybride.

3) Améliorer l'interface utilisateur de la maquette en la couplant à des simulateurs d'interface homme/machine représentatifs du système et en développant des outils de test plus souples d'emploi et plus performants.

4) Utiliser des langages formels pour la spécification fonctionnelle détaillée, permettant ainsi une génération automatique du code maquette par compilation de la spécification.

SELECTIVE BIBLIOGRAPHY

This bibliography with abstracts has been prepared to support
AGARD Lecture Series No. 164 by the Scientific and Technical
Information Division of the U.S. National Aeronautics and Space
Administration, Washington, D.C., in consultation with the
Lecture Series Director, Dr. Fred I. Diamond, Rome Air Development
Center (AFSC), Griffiss Air Force Base, New York.

UTTL: Cooperating knowledge based systems' development issues
AUTH: A/PAYNE, J. ROLAND; B/MARSH, JAMES P.
PAA: B/(Advanced Decision Systems, Mountain View, CA) IN: AAAIC '87 - Aerospace Applications of Artificial Intelligence; Proceedings of the Third Annual Conference, Dayton, OH, Oct. 5-9, 1987 (A88-52226 22-66). Dayton, OH, AAAIC Conference Secretariat, 1988, p. 31-41.
ABS: Three avionics projects that illustrate a progression of 'small' to 'large' systems of cooperating knowledge-based subsystems are considered: INS-FAAMS, IFFN, and D-1. Lessons learned are enumerated, spanning the areas of knowledge acquisition (KA), KA to code interaction capabilities, breadth vs. depth, the importance of I/O formats, the use of more than one machine, and the combination of AI and systems analysis techniques. Implications of the lessons learned are identified as requirements or desirable features for future development environments. 88/00/00 88A52230

UTTL: A rapid prototyping tool for pilot vehicle interface design
AUTH: A/MILLER, BRAD S.; B/SHELNUTT, JACK B.; C/FARRELL, RICHARD
PAA: C/(Boeing Military Airplane Co., Seattle, WA) IN: AAAIC '87 - Aerospace Applications of Artificial Intelligence; Proceedings of the Third Annual Conference, Dayton, OH, Oct. 5-9, 1987 (A88-52226 22-66). Dayton, OH, AAAIC Conference Secretariat, 1988, p. 7-14.
ABS: The Pilot Vehicle Interface Development Tool (PVIDT) is an integrated group of computer work stations used for (1) rapid prototyping of pictorial format displays and cockpit automation concepts, (2) identifying avionics software requirements, and (3) developing a knowledge base for an expert system to assist the pilot with the operation of an aircraft. The expert system automates pilot-vehicle interface management (PVIM) tasks which the pilot would otherwise have to perform. It controls the amount of information displayed on each format, determines how each format is configured before it is brought up on a cockpit display and merges pilot inputs from different sources such as cursor, voice, touch screen and hard switch into integrated commands to the avionics and display systems. The rules for the expert system are developed by experienced pilots, human factors experts and computer scientists acting in concert to maximize the utility of the expert system as it is evaluated by testing on the PVIDT. 88/00/00 88A52228

UTTL: Integrated diagnostics for a joint integrated avionics architecture
AUTH: A/BRIDGES, ALAN L.
PAA: A/(Lockheed Aeronautical Systems Co., Marietta, GA) AIAA, AHS, and ASEE, Aircraft Design, Systems and

ABS: Operations Meeting, Atlanta, GA, Sept. 7-9, 1988. 14 p. The integrated diagnostics (ID) product requirements for the Joint Integrated Avionics Plan are discussed. The ID process, ID technology for integrated avionics, the development of ID for weapons systems that employ advanced integrated circuitry, and the integration of preventive maintenance with the overall diagnostic system are examined. The integrated rack packaging concept, advanced avionics architecture, and the Generic Integrated Maintenance Diagnostics process are illustrated. It is suggested that automating the diagnostic design process will result in a system with a high probability of mission success, which can be maintained at high levels of readiness, with due consideration to cost and human resources.
RPT#: AIAA PAPER 88-4432 88/09/00 88A51947

AUTH: UTTL: Modular avionics system architecture decision support system
A/DICKMAN, THOMAS J.; B/ROBERTS, THOMAS M.
PAA: A/(Analytic Sciences Corp., Fairborn, OH); B/(USAF, Aeronautical Systems Div., Wright-Patterson AFB, OH) IN: NAECON 88: Proceedings of the IEEE National Aerospace and Electronics Conference, Dayton, OH, May 23-27, 1988. Volume 4 (A88-50926 22-01). New York, Institute of Electrical and Electronics Engineers, 1988, p. 1549-1552.
ABS: A methodology for evaluating modular avionics applications in terms of life-cycle cost and system effectiveness is reported. The methodology is to be used to compare alternative modular architecture strategies as well as conventional strategies for introducing avionics into new or existing weapon systems. The process described is intended to be a flexible means of evaluating postulated alternatives for US Air Force-wide implementation of modular avionics. The analysis is expected to be performed using data normally available during the conceptual phase of a development program. Functions performed by avionics have been limited to navigation, communications, and some portion of fire control. The role of avionics, however, has steadily increased with each new generation of aircraft. It is argued that commonality of avionics elements within and among weapon systems may reduce both development cost and development risk. 88/00/00 88A51050

UTTL: A cooperative expert system architecture for embedded avionics
AUTH: A/NG, ANDREW
PAA: A/(Honeywell, Inc., Technology Dept., Albuquerque, NM) IN: NAECON 88: Proceedings of the IEEE National Aerospace and Electronics Conference, Dayton, OH, May 23-27, 1988. Volume 4 (A88-50926 22-01). New York, Institute of Electrical and Electronics Engineers, 1988, p. 1282-1286.
ABS: An underlying requirement for artificial intelligence

software in avionics environments is real-time throughput and behavior. By using multiple processors and efficient communication mechanisms, the concept of cooperative expert systems can be used effectively in real-time applications. A hardware and software environment for supporting cooperative expert systems with multiple streamlined set (RISC-influenced) processors and a real-time shared memory multicomputer operating system executive is presented and analyzed. By using the parallelism of cooperative expert systems and RISC-influenced real-time shells in conjunction with conventional preemptive multitasking, the execution speed of artificial intelligence programs can be increased to meet real-time requirements while also maintaining the predictable response characteristics expected of avionics systems. 88/00/00 88A51034

UTTL: Digital avionics susceptibility to high energy radio frequency fields
AUTH: A/LARSEN, WILLIAM E.
PAA: A/(NASA, Ames Research Center; FAA, Moffett Field, CA) CORP: National Aeronautics and Space Administration. Ames Research Center, Moffett Field, Calif.; Federal Aviation Administration, Moffett Field, Calif. IN: NAECON 88: Proceedings of the IEEE National Aerospace and Electronics Conference, Dayton, OH, May 23-27, 1988. Volume 3 (A88-50926 22-01). New York, Institute of Electrical and Electronics Engineers, 1988, p. 1098-1107.
ABS: Generally, noncritical avionic systems for transport category aircraft have been designed to meet radio frequency (RF) susceptibility requirements set forth in RTCA DO 160B, environmental conditions and test procedures for airborne equipment. Section 20 of this document controls the electromagnetic interference (EMI) hardening for avionics equipment to levels of 1 and 2 V/m. Currently, US equipment manufacturers are designing flight-critical fly-by-wire avionics to a much higher level. The US Federal Aviation Administration (FAA) has requested that the RTCA SC-135 high-energy radio frequency (HERF) working group develop appropriate testing procedures for section 20 of RTCA DO 160B for radiated and conducted susceptibility at the box and systems level. The FAA has also requested the SAE AE4R committee to address installed systems testing, airframe shielding effects and RF environment monitoring. Emitters of interest include radar (ground, ship, and aircraft) commercial broadcast and TV station, mobile communication, and other transmitters that could possibly affect commercial aircraft. 88/00/00 88A51024

UTTL: Controlling large cyclic avionics software systems written in Ada
AUTH: A/ELLIS, JOHN R.; B/VON EDWINS, STEVEN A.
PAA: A/(Harris Corp., Government Aerospace Systems Div.,

Melbourne, FL); B/(Integrated Software, Inc., Palm Bay, FL) IN: NAECON 88: Proceedings of the IEEE National Aerospace and Electronics Conference, Dayton, OH, May 23-27, 1988. Volume 2 (A88-50926 22-01). New York, Institute of Electrical and Electronics Engineers, 1988, p 727-731.
ABS: Although Ada was created to standardize programming for real-time embedded computer applications, especially in defense systems, it has inherent limitations when applied to large numbers of concurrent periodic processes. Modern integrated avionics systems, such as are in use on the AGUSTA A-129 Mangusta Helicopter, involve as many as 140 concurrent periodic tasks running at a variety of frequencies from 180 Hz to once every 30 seconds. Published Ada solutions to managing concurrent periodic tasks revert to a 1960s 'cyclic executive' technology, thrust control logic into the applications code, or introduce significant system overhead. When software systems get very large and involve tasks of varying duration and execution frequencies, these solutions become unmanageable. The problem domain and several of these approaches are outlined, and alternate solutions are presented. Central to the issue is the preservation or determinism in the execution control of integrated avionics software systems. 88/00/00 88A50991

UTTL: PAVE PACE: System avionics for the 21st century
AUTH: A/MORGAN, D. REED
PAA: A/(USAF, Wright Aeronautical Laboratories, Wright-Patterson AFB, OH) IN: NAECON 88: Proceedings of the IEEE National Aerospace and Electronics Conference, Dayton, OH, May 23-27, 1988. Volume 1 (A88-50926 22-01). New York, Institute of Electrical and Electronics Engineers, 1988, p. 146-157.
ABS: The rationale is presented for the continuation of the advanced architecture development, established by the PAVE PILLAR initiative, under the PAVE PACE program. The goals are to achieve: (1) practical and affordable airborne versions of modular parallel-processing network architectures for a large array of new applications (currently beyond real-time implementation); (2) highly available avionics for use across all avionics; and (3) dramatically improved techniques to reduce the cost of software development and support. A novel approach to the overall design structure for future avionics is also presented. Continued use of the PAVE PILLAR high-speed data bus and operating system is recommended as the means to integrate and control the data input and output of physically and functionally separate parallel networks. 88/00/00 88A50941

UTTL: Real-time operating system for advanced avionics architecture
AUTH: A/BENNING, STEPHEN L.; B/EVANS, R. SCOTT

PAA: A/(USAF, Wright Aeronautical Laboratories,
Wright-Patterson AFB, OH); B/(TRW Dayton Engineering
Laboratory, OH) IN: NAECON 88: Proceedings of the IEEE
National Aerospace and Electronics Conference, Dayton, OH,
May 23-27, 1988. Volume 1 (A88-50926 22-01). New York,
Institute of Electrical and Electronics Engineers, 1988.
p. 138-145.

ABS: A real-time operating system, the Ada Avionics Real-Time
Software (AARTS) Operating System (AOS), is discussed,
which under development for the US Air Force. The AOS is
intended to mitigate problems with Ada executing in real
time on 16-bit data processors in a distributed
architecture configuration. The AOS consists of a three
part executive: system executive, kernel executive, and
distributive executive. Each is described along with AOS
operation and system functions. 88/00/00 88A50940

UTTL: Avionics for transport aircraft - Current
development status
AUTH: A/WESTPHAL, GUSTAV
PAA: A/(Interflug Gesellschaft fuer Internationalen Flugverkehr
mbH, Berlin, German Democratic Republic)
Technisch-oekonomische Information der zivilen Luftfahrt
(ISSN 0232-5012), vol. 24, no. 2, 1988, p. 76-80. In
German.

ABS: The technology and capabilities of present and
next-generation avionics for transport aircraft are
reviewed. Systems for status monitoring, flight control,
navigation, communication, flight safety, and special
missions are considered, and particular attention is given
to computer integration of different onboard systems, the
display of systems information in the cockpit, new demands
on ATC and the technologies being developed to meet them,
the application of satellite-based navigation and
emergency position-finding systems, and the maintenance
and repair problems posed by the introduction of advanced
avionics. Block diagr... and drawings of cockpit displays
are provided. BF.../... 88A41098

UTTL: An intelligent maintenance aid for portable ATE
AUTH: A/HICKEY, KEITH A.; B/KIEU, THANH C.; C/ADAMS, NANCY L.
PAA: C/(Emerson Electric Co., B/KIEU, THANH and Space Div., Saint
Louis, MO) IN: AUTOTESTCON '87: Proceedings of the
International Automatic Testing Conference, San Francisco,
CA, Nov. 3-5, 1987 (A88-36528 14-59). New York, Institute
of Electrical and Electronics Engineers, Inc., 1987. p.
383-386.

ABS: A description is given of the development approach taken
to embed a diagnostic expert system within an existing
avionics portable automated test equipment (ATE) system.
The objective of the project was to enhance the
built-in-test (BIT) data by relating the BIT data to user
observations, maintenance and test documentation, and
failure history. The result was the dual effect of

simultaneously increasing the sophistication and
capability of the testing system and maintaining the
compactness of a portable ATE system. 87/00/00
88A36575

UTTL: ESATE - Expert system ATE
AUTH: A/KNAPP, PHILLIP M.
PAA: A/(AAI Corp., Hunt Valley, MD) IN: AUTOTESTCON '87;
Proceedings of the International Automatic Testing
Conference, San Francisco, CA, Nov. 3-5, 1987 (A88-36528
14-59). New York, Institute of Electrical and Electronics
Engineers, Inc., 1987, p. 177-180.

ABS: ESATE is a study effort to evaluate the potential
advantages of using expert systems in conjunction with
existing ATLAS test program sets (TPSs). The predicted
advantages of expert systems as they apply to automatic
test equipment (ATE) are reviewed. In most cases, these
advantages counter the shortfall of ATLAS and data-driven
diagnostic programs, whose structural rigidity and
precomputed testing strategy create limitations for the
automated testing community. The author addresses the
question of whether a knowledge base can be extracted
automatically from existing TPSs so that a database
automatically created from these TPSs can work with expert
systems. Analysis has shown the need for a dependency
structure overlay for these existing TPSs, which would
define which tests within the ATLAS test program are
independent of other tests, and which were really just one
step in a sequential series of dependent tests. The lack
of this dependency data in current analog and hybrid TPSs
presents the major obstacle in achieving the desired goal.
The author examines this issue and presents some early
conclusions and recommendations. 87/00/00 88A36548

UTTL: A PC based expert diagnostic tool
AUTH: A/BURSCH, PAUL M.; B/MEISNER, JOHN W.; C/WINEGAR, KEITH
F.
PAA: C/(Honeywell, Inc., Minneapolis and Golden Valley, MN)
IN: AUTOTESTCON '87; Proceedings of the International
Automatic Testing Conference, San Francisco, CA, Nov. 3-5,
1987 (A88-36528 14-59). New York, Institute of Electrical
and Electronics Engineers, Inc., 1987, p. 73-77.

ABS: Trends in future military avionic systems indicate that
advanced maintenance techniques will be required. To
better facilitate advanced diagnostics, a knowledge-based
system for a personal computer (PC) has been developed by
porting knowledge bases from a symbolic computing
environment to a conventional PC system. This system
demonstrates the feasibility of advanced maintenance
diagnostics techniques using a frame system and
object-oriented programming to implement the diagnostic
inference mechanism. 87/00/00 88A36540

UTTL: A modern Tower of Babel - Integration, test, and evaluation of inertially aided avionics

AUTH: A/WALTON, NORMAN; B/ZISKIND, STEVEN

PAA: B/(Boeing Military Airplane Co., Seattle, WA) IN: Institute of Navigation, Annual Meeting, 43rd, Dayton, OH, June 23-25, 1987, Proceedings (A88-35551 14-06). Washington, DC, Institute of Navigation, 1987, p. 172-177

ABS: Important considerations for the integration of inertially supported avionics are outlined. Attention is given to the following specific applications of inertial systems: (1) flight control, (2) SAR motion compensation, (3) IR sensor stabilization, and (4) coordinate frame alignment among sensors and transfer of data between different vehicles having similar sensor suites. Case studies are used to illustrate the problems which can arise when insufficient attention is paid to critical details of the integration task. 87/00/00 88A35562

UTTL: Instrumentation of advanced avionics suites using real time data compression techniques

AUTH: A/SQUIRES, THOMAS; B/RIGGINS, BRUCE

PAA: B/(Boeing Military Airplane Co., Wichita, KS) IN: Rotorcraft flight controls and avionics; Proceedings of the National Specialists' Meeting, Cherry Hill, NJ, Oct. 13-15, 1987 (A88-35366 14-01). Alexandria, VA, American Helicopter Society, 1987, 9 p.

ABS: Advanced avionics architectures will be incorporated into future aircraft designs employing computational systems far more complex than current systems. In order to provide instrumentation support during the design, development, and test of these avionics, a number of technical issues must be addressed. The purpose of this paper is to address these issues and define an approach for developing advanced avionics instrumentation. Typically, a data system must have at a minimum, the same communications capabilities as the system under test. For example, in advanced avionics architectures, fiber-optic and electrical data buses will communicate at data rates from 1 to 1000 megabits/second. Instrumentation and analysis systems will have to collect, correlate, and compress the data in order to integrate and validate the avionics system designs. Real time data compression will be required to reduce the storage device requirements and simplify the data analysis process. 87/00/00 88A35390

UTTL: Knowledge based system concepts and techniques applied to integrated diagnostics

AUTH: A/POOLE, JERRY; B/FOWLER, EDDIE R.; C/HIGHTOWER, RON; D/HEFTY, KEITH

PAA: A/(Boeing Military Airplane Co., Wichita, KS); D/(Kansas State University of Agriculture and Applied Science, Manhattan) IN: Rotorcraft flight controls and avionics; Proceedings of the National Specialists' Meeting, Cherry Hill, NJ, Oct. 13-15, 1987 (A88-35366 14-01). Alexandria,

ABS: VA, American Helicopter Society, 1987, 14 p. The generic methodology presented for the design of knowledge-based diagnostic systems gives attention to the airborne and ground diagnostics of avionics systems. The methodology encompasses three major steps; its substeps, however, also furnish insights into the design concepts with which a knowledge engineer must be familiar and adept throughout the diagnostic system development cycle. A proof-of-concept project review is given for an avionics diagnostics system. 87/00/00 88A35384

UTTL: Common module implementation for an avionic digital map

AUTH: A/MAHTERA, JOHN C.; B/GRACIA, JIM A.; C/FARLEY, PAUL E.

PAA: C/(Harris Corp., Melbourne, FL) IN: Rotorcraft flight controls and avionics; Proceedings of the National Specialists' Meeting, Cherry Hill, NJ, Oct. 13-15, 1987 (A88-35366 14-01). Alexandria, VA, American Helicopter Society, 1987, 11 p.

ABS: The modular implementation alternatives investigated for next-generation avionics' digital moving map subsystem uses the Pave Pillar architecture as the baseline from which the common modules for the U.S. Army's LHX, USAF ATF, and Navy Advanced Tactical Aircraft will be developed. Three common module conceptual implementations for the moving map are discussed: the vector processor module set, the array processor module set, and the postprocessor module set. Custom-designed, state-of-the-art postprocessors are found to offer substantial size, weight, and power advantages over the two other alternatives. 87/00/00 88A35380

UTTL: Rotorcraft flight controls and avionics; Proceedings of the National Specialists' Meeting, Cherry Hill, NJ, Oct. 13-15, 1987 Meeting sponsored by AHS. Alexandria, VA, American Helicopter Society, 1987, 305 p. For individual items see A88-35367 to A88-35394.

ABS: The present conference on rotorcraft cockpit systems discusses digital flight control technology for advanced combat helicopters, far-field mission planning for nap-of-the-earth flight, terrain following/avoidance for helicopters, cognitive engineering for novel cockpit designs, V-22 crew station design, advanced augmentation and split-axes control systems for rotorcraft, and helicopter side arm-integrated controllers. Also discussed are a common module implementation for an avionic digital map system, the rotorcraft application of DARPA's 'pilot's associate', AI's application to diagnostics prognostics of flight control systems, an X-wing rotorcraft flight control system test, counterair applications of AI/knowledge-based systems, and the application of a tilt-rotor flight simulator. 87/00/00 88A35368

integrated performance and supportability analysis methodology. Although the methodology was originally set up for supportability design and analysis only, the capability and structure for integration of performance issues developed as the result of evaluating the impact of weapons systems constraints on the system design and support issues. The methodology and analysis provided a program-directed focus breaking across traditional organizational and communications barriers in avionics design and analysis. 87/00/00 88A34193

UTTL: Standard air-vehicle equipment (SAVE) - Bringing transport aircraft avionics one step closer to the twenty first century
AUTH: A/FELDGAUER, BENJAMIN
PAA: A/(Lockheed-Georgia Co., Marietta) IN: NAECON 87; Proceedings of the IEEE National Aerospace and Electronics Conference, Dayton, OH, May 18-22, 1987. Volume 4 (A88-34026 13-01). New York, Institute of Electrical and Electronics Engineers, Inc., 1987. p. 1289-1295.
ABS: The author addresses the issue of standardized avionics and suggests three characteristics of such systems: (1) modular architecture with standardized interfacing; (2) unified diagnostics for high fault tolerance; and (3) simplified logistic support (i.e., one- and two-level maintenance). Design guidelines are presented for smart built-in test, and for the fault data recorder, the protocol configuration adapter, the autothrottle computer, and the fuel interface unit. Directions in which this work may proceed are noted. 87/00/00 88A34192

UTTL: Avionic standard module development
AUTH: A/MAKI, STANLEY C.; B/CORMIER, EDMOND P.; C/PISZKIN, THOMAS A.
PAA: C/(General Dynamics Corp., Space Systems Div., San Diego, CA) IN: NAECON 87; Proceedings of the IEEE National Aerospace and Electronics Conference, Dayton, OH, May 18-22, 1987. Volume 4 (A88-34026 13-01). New York, Institute of Electrical and Electronics Engineers, Inc., 1987. p. 1274-1281.
ABS: Avionics standard modules with redundancy offer substantial economic benefits compared to special-purpose processor units for the orbital transfer vehicle and advanced launch vehicle programs. A fiber optic, serial vehicle bus provides high throughput with modest hardware. A bistage, split tapered, star optical coupler uses a token-pass/token-demand protocol. It is reported that a standard module implementation of the above is a feasible cost-effective approach to avionics design using standard buses and standard packaging. The VHSIC integrated package readily accommodates higher-speed VLSI chips as they become available. 87/00/00 88A34190

UTTL: STEP: A tool for estimating avionics life cycle costs
AUTH: A/CURRY, ERNEST E.
PAA: A/(USAF, Wright Aeronautical Laboratories, Wright-Patterson AFB, OH) IN: NAECON 87; Proceedings of the IEEE National Aerospace and Electronics Conference, Dayton, OH, May 18-22, 1987. Volume 4 (A88-34026 13-01). New York, Institute of Electrical and Electronics Engineers, Inc., 1987. p. 1472-1474.
ABS: Estimation of life cycle costs (LCC) for avionics hardware and software presents a difficult challenge. The US Air Force has had significant success with the standardization evaluation program (STEP) model for estimating operating and support costs. The author describes the capabilities of the existing STEP model as well as ongoing enhancements to provide a total LCC estimating methodology. The paper discusses reliability sensitivity analyses, impact of retest OK, impact of fault isolation, acquisition cost enhancements, an automated database, analogy selection capability, and parametric estimating capability. 87/00/00 88A34217

UTTL: Avionics expert systems: The transition to embedded systems
AUTH: A/PILET, S. C.; B/STENERSON, R. O.
PAA: B/(Boeing Military Airplane Co., Seattle, WA) IN: NAECON 87; Proceedings of the IEEE National Aerospace and Electronics Conference, Dayton, OH, May 18-22, 1987. Volume 4 (A88-34026 13-01). New York, Institute of Electrical and Electronics Engineers, Inc., 1987. p. 1388-1393.
ABS: An approach to the transition of expert systems into an embedded environment has been described. This approach improves reliability while reducing integration costs. However, the issues of fault tolerances, applications interfaces, and validation require further investigation. The ability to insert current expert systems into prototype avionics environments was necessary to research these issues as. It is noted, they can only be solved within a systems context. 87/00/00 88A34207

UTTL: Integrated design of modular avionics for performance and supportability
AUTH: A/FLEMING, RANDALL; B/JOSSELYN, JILL V.; C/BOYLE, PAUL
PAA: B/(Systems Control Technology, Inc., Palo Alto, CA); C/(Northrop Corp., Aircraft Div., Hawthorne, CA) IN: NAECON 87; Proceedings of the IEEE National Aerospace and Electronics Conference, Dayton, OH, May 18-22, 1987. Volume 4 (A88-34026 13-01). New York, Institute of Electrical and Electronics Engineers, Inc., 1987. p. 1298-1303.
ABS: A case study is presented of a generic complex modular avionics system during system and subsystem design phases. The case study demonstrates setup and application of an

UTTL: Built-in-test software for an Ada avionics hot bench

AUTH: A/MOODY, BEVERLY L.; B/HENSLEY, DAVID A.; C/BENNETT, JAMES W.

PAA: C/(USAF, Aeronautical Systems Div., Wright-Patterson AFB, OH) IN: NAECON 87: Proceedings of the IEEE National Aerospace and Electronics Conference, Dayton, OH, May 18-22, 1987. Volume 3 (A88-34026 13-01). New York, Institute of Electrical and Electronics Engineers, Inc., 1987. p. 820-824.

ABS: The built-in-test (BIT) software project is part of the Ada avionics hot bench (AAHB) project at Wright-Patterson Air Force Base. The BIT software project, written entirely in the Ada programming language, provides a means of testing all the avionics hardware on the hot bench. BIT tests all systems and subsystems connected to the MIL-STD-1553B data bus. BIT software is described, including error detection and exception handling, the error log file, useful tasking features, message processing, and enumeration types. 87/00/00 88A34133

UTTL: Methods for evaluating integrated airframe/propulsion control system architectures

AUTH: A/COHEN, GERALD C.; B/LEE, CHARLES W.; C/PALUMBO, DANIEL L.

PAA: B/(Boeing Military Airplane Co., Seattle, WA); C/(NASA, Langley Research Center, Hampton, VA) CORP: Boeing Military Airplane Development, Seattle, Wash.; National Aeronautics and Space Administration. Langley Research Center, Hampton, Va. IN: NAECON 87: Proceedings of the IEEE National Aerospace and Electronics Conference, Dayton, OH, May 18-22, 1987. Volume 2 (A88-34026 13-01). New York, Institute of Electrical and Electronics Engineers, Inc., 1987. p. 569-575.

ABS: Methodology, analysis, and detailed design of integrated control system architectures suitable for high-performance aircraft of the 1990s are presented. A methodology, with supporting analytical tools, has been developed to provide system designers with the capability to specify candidate architectures and accurately predict their reliability and performance in the early stages of design development. The authors address the methodology and supporting tools. 87/00/00 88A34102

UTTL: System design and avionics integration of a takeoff performance monitor

AUTH: A/BALDWIN, STEVEN F.

PAA: A/(E-Systems, Inc., Greenville, TX) IN: NAECON 87: Proceedings of the IEEE National Aerospace and Electronics Conference, Dayton, OH, May 18-22, 1987. Volume 2 (A88-34026 13-01). New York, Institute of Electrical and Electronics Engineers, Inc., 1987. p. 538-543.

ABS: The system design and development of a prototype takeoff performance monitor (TOPM) and its avionics integration into a Falcon 10 jet aircraft are described. The design maximizes the use of color graphics for quick and discernible interpretation by the flight crew for the computations of weight and balance, takeoff performance requirements, and to display real-time takeoff acceleration. An Intel 310 computer is integrated with the antiskid system, the air data computer, and the color radar indicator. All necessary weight, balance, and performance charts were taken from the Falcon 10 flight manual and digitized for increased speed and accuracy. An FAA supplemental type certificate has been issued for the Falcon 10 that incorporates the TOPM, which has a patent pending. 87/00/00 88A34098

UTTL: An avionics expert system for ground threat assessment

AUTH: A/LIND, HENRIK; B/STENERSON, RICHARD

PAA: B/(Boeing Military Airplane Co., Seattle, WA) IN: NAECON 87: Proceedings of the IEEE National Aerospace and Electronics Conference, Dayton, OH, May 18-22, 1987. Volume 2 (A88-34026 13-01). New York, Institute of Electrical and Electronics Engineers, Inc., 1987. p. 326-331.

ABS: A prototype avionics threat assessment system consisting of an interactive threat editor, a ground threat situation assessor, and a graphical user interface has been designed as a first step toward building an embedded situation assessor for flight systems. The interactive threat editor provides a menu system to create the threat knowledge base. The editor generates both the data structure displays and the code needed by the situation assessor. The situation assessor uses a blackboard inferencing environment to form threat hypotheses. These hypotheses drive the user interface to create area sensitive displays that can be interrogated to determine the nature of the threats. The threat hypotheses developed are also readily available for input to other candidate embedded avionics expert systems; particularly, the mission planner and the crew station information manager. 87/00/00 88A34073

UTTL: A review of traditional system reconfiguration techniques and their applicability to the unique requirements of digital avionics

AUTH: A/RUTENBERG, MARK R.

PAA: A/(ITT Corp., ITT Avionics Div., Nutley, NJ) IN: NAECON 87: Proceedings of the IEEE National Aerospace and Electronics Conference, Dayton, OH, May 18-22, 1987. Volume 1 (A88-34026 13-01). New York, Institute of Electrical and Electronics Engineers, Inc., 1987. p. 192-198.

ABS: The major approaches to reconfiguration for the achievement of fault tolerance are surveyed with a view toward identifying the underlying fundamental technical challenges. The implications on fault detection and isolation resulting from the fact that digital systems do

not have a representative continuous transfer function is stressed. It is shown that implementation of reconfiguration in digital avionics would require serious system reliability analysis and probably extend the state-of-the-art in reliable system design. This is because digital avionics falls directly between the two classical modes of application of reconfiguration to achieve fault tolerance. Suggestions are made as to how this gap can most effectively be bridged. 87/00/00 88A34058

UTTL: Laboratory facility for F-15E avionics systems integration testing
AUTH: A/SMITH, MARK B.; B/BUCHANTIN, GREGORY T.   IN: NAECON
PAA: B/(McDonnell Douglas Corp., Saint Louis, MO) IN: NAECON 87; Proceedings of the IEEE National Aerospace and Electronics Conference, Dayton, OH, May 18-22, 1987. Volume 1 (A88-34026 13-01). New York, Institute of Electrical and Electronics Engineers, Inc., 1987, p. 180-182. USAF-supported research.
ABS: A dedicated avionics integration laboratory facility at McDonnell Aircraft Company in St. Louis, Missouri is described. The facility consists of a closed laboratory area with forced cooling air, all necessary power, multiplex bus interconnections, discrete signal interfaces, and voice communication lines to other avionic lab areas. The major element of the facility is the avionic system integration bench (ASIB). The ASIB consists of two components; the central computer integration bench, and the controls and displays integration bench. The ASIB, with dedicated aircraft equipment installed, is the nucleus for all integration tests. The combinations of benches, systems, and software provides a flexible and capable integration tool. 87/00/00 88A34055

UTTL: Designing a master executive for a distributed multiprocessor avionics system
AUTH: A/CLAUSING, BRIAN   IN: NAECON 87; Mason, OH)
PAA: A/(Symbolic Systems, Inc., Mason, OH) IN: NAECON 87; Proceedings of the IEEE National Aerospace and Electronics Conference, Dayton, OH, May 18-22, 1987. Volume 1 (A88-34026 13-01). New York, Institute of Electrical and Electronics Engineers, Inc., 1987, p. 174-179.
ABS: The design of a master executive program for an avionics system with many conventional processors loosely coupled in a two-tier, high-bandwidth network must utilize the inherent advantages and avoid the disadvantages of the architecture, guarantee not only real-time performance but also failsafe operation, use Ada, and present a usable interface to application. It is possible to design a real-time master executive that is reusable and largely independent of hardware specifications. An analysis of some of the major issues in the design of a distributed processing operating system is presented and some

suggestions on the use of Ada are made. 87/00/00 88A34054

UTTL: Tradeoffs in avionic signal processing configuration
AUTH: A/WILLIAMS, KENNETH E.
PAA: A/(Texas Instruments, Inc., Dallas) IN: NAECON 87; Proceedings of the IEEE National Aerospace and Electronics Conference, Dayton, OH, May 18-22, 1987. Volume 1 (A88-34026 13-01). New York, Institute of Electrical and Electronics Engineers, Inc., 1987, p. 165-168.
ABS: A number of the tradeoffs required in configuring a comprehensive avionic signal processor are discussed. The advantages of processor centralization for economy of backup and system availability are indicated, along with the complications caused by the unique processing requirements of various sensors. Alternative solutions and their implications are briefly discussed. Also addressed are such factors as number of sensors served by each processor, distribution of modules by type in typical processor designs, and physical limitations. 87/00/00 88A34052

UTTL: Integrated avionics   Aerospace Engineering (ISSN 0736-2536), vol. 8, April 1988, p. 8-14.
ABS: Advances in avionics made due to the introduction of electronic flight instrument systems are discussed. The avionics systems in the cockpits of the Beech Starship and the Gulfstream IV are described.The feasibility of using electronic voice synthesis systems in cockpits is addressed. 88/04/00 88A32799

UTTL: Future avionics for Army aviation
AUTH: A/MAYER, FRANK H.
PAA: A/(U.S. Army, Aviation Center, Fort Rucker, AL) Vertiflite (ISSN 0042-4455), vol. 34, Mar.-Apr. 1988, p. 32-37.
ABS: Such next-generation military helicopters as the LHX will have to give crews very direct access to their avionics/electronics equipment package, as well as incorporate automatic exchanges of information among electronic subsystems without crewmember intervention and yield substantial improvements in life cycle costs. A hald-held 'electronic data loader' will be used to load operational and maintenance data in and out of the LHX airframe. Two or three VHSIC technology-incorporating computation hardware modules will be employed to furnish redundant computing capacity in the event of damage. 88/04/00 88A30935

UTTL: Perspective on intelligent avionics
AUTH: A/JONES, HAROLD L.
PAA: A/(Analytic Sciences Corp., Reading, MA)   SAE, Aerospace

Technology Conference and Exposition, Long Beach, CA. Oct. 5-8, 1987. 10 p.

ABS: Technical issues which could potentially limit the capability and acceptibility of expert systems decision-making for avionics applications are addressed. These issues are: real-time AI, mission-critical software, conventional algorithms, pilot interface, knowledge acquisition, and distributed expert systems. Examples from on-going expert system development programs are presented to illustrate likely architectures and applications of future intelligent avionic systems.

RPT#: SAE PAPER 871856 87/10/00 88A30813

UTTL: Performance evaluation of medium access control protocols for distributed digital avionics

AUTH: A/RAY, ASOK

PAA: A/(Pennsylvania State University, University Park) (ASME Winter Annual Meeting, Boston, MA, Dec. 14-18, 1987) ASME, Transactions, Journal of Dynamic Systems, Measurement, and Control (ISSN 0022-0434), vol. 109, Dec. 1987, p. 370-377. Research supported by the Allied Bendix Aerospace Corp.

ABS: The paper presents the results of an ongoing research project where the objectives are to evaluate medium access control (MAC) protocols in view of the requirements for distributed digital flight control systems (DDFCS) of advanced aircraft and to recommend a specific protocol for their prototype development. The selection of an appropriate MAC protocol is critical for the dynamic performance of an aircraft because the DDFCS, in addition to the sampling time delay, is subject to time-varying transport delays due to data latency of messages at different terminals of the control loop. The SAE linear token bus, SAE token ring and the conventional MIL-STD-1553B protocols have been analyzed using combined discrete-event and continuous-time simulation techniques. The impact of data latency on the dynamic performance of an advanced aircraft is illustrated by simulation of the closed loop DDFCS.

RPT#: ASME PAPER 87-WA/DSC-2 87/12/00 88A21269

UTTL: Electromagnetic compatibility modeling for future avionics systems

AUTH: A/LOCKYER, ALLEN

PAA: A/(Northrop Corp., Aircraft Div., Hawthorne, CA) IN: Avionics in conceptual system planning; Proceedings of the Eighth Annual IEEE Symposium. Dayton, OH, Dec. 3, 1986 (A88-16912 05-66). New York, Institute of Electrical and Electronics Engineers, Inc., 1986, p. 79-89.

ABS: Two computer programs have been developed which can model a weapon system's electromagnetic compatibility and can be used to identify potential electromagnetic interference problems in the conceptual phase of avionic system planning. The Intrasystem Electromagnetic Compatibility Analysis Program (IEMCAP) is designed for overall intrasystem electromagnetic compatibility prediction; the General Electromagnetic Model for Analysis of Complex Systems (GEMACS) can model detailed coupling problems that have been identified by IEMCAP. Some useful code enhancements illustrating the growing importance of computer graphics in electromagnetic analysis are also discussed. 86/00/00 88A16920

UTTL: Improving avionics acquisition and support from conceptualization through operations

AUTH: A/GEBMAN, JEAN; B/SHULMAN, HY

PAA: B/(Rand Corp., Santa Monica, CA) IN: Avionics in conceptual system planning; Proceedings of the Eighth Annual IEEE Symposium. Dayton, OH, Dec. 3, 1986 (A88-16912 05-66). New York, Institute of Electrical and Electronics Engineers, Inc., 1986, p. 69-76.

ABS: The problem of the supportability of avionics equipment is examined with emphasis on an approach to acquisition and support that begins with the concept formulation stage and follows through the equipment's full life of service. The basis for such an approach is summarized, and a broad strategy for enhancing avionics supportability is formulated. Some tradeoffs are proposed which should be made at the concept formulation stage to further enhance the benefits of the strategy for improving avionics supportability. 86/00/00 88A16919

UTTL: The design agent process as a strategy for future avionics competition enhancement and quality assurance

AUTH: A/DELANEY, WILLIAM J.

PAA: A/(Charles Stark Draper Laboratory, Inc., Cambridge, MA) IN: Avionics in conceptual system planning; Proceedings of the Eighth Annual IEEE Symposium. Dayton, OH, Dec. 3, 1986 (A88-16912 05-66). New York, Institute of Electrical and Electronics Engineers, Inc., 1986, p. 53-58.

ABS: The design agent concept of acquisition management is examined with particular reference to future avionics acquisition requirements. The activities, responsibilities, and competition enhancing benefits of the design agent approach to acquisition management are discussed and illustrated by several different application examples. It is claimed that the design agent's ability to uniquely establish and control multiple contractors for competition enhancement purposes has direct relevance to the need for improved acquisition strategies on select 6.3 programs. 86/00/00 88A16918

UTTL: The avionics acquisition process beyond the year 2000

AUTH: A/LAVOIE, R. P.; B/CULP, A. M. IN: Avionics in conceptual system planning; Proceedings of the Eighth Annual IEEE Symposium. Dayton, OH, Dec. 3, 1986 (A88-16912

05-66). New York, Institute of Electrical and Electronics Engineers, Inc., 1986, p. 45-49.

ABS: The current weapon system acquisition and support process is examined with emphasis on problems related to the useful life of microelectronic component technology. Requirements changes, and technology obsolescence. The need for changes in the present acquisition process is emphasized, and it is shown that a good solution should accept the reality of long development programs and adjust the process to deal with rapidly developing technology. Requirements changes, and obsolescence. The critical elements of the solution are long-term planning, sustained investment for improving systems, managed change, and incremental transfer of system responsibility. 86/00/00 88A16917

UTTL: System architecture design aid tools for a distributed avionics system

AUTH: A/BROCK, LARRY D.

PAA: A/(Charles Stark Draper Laboratory, Inc., Cambridge, MA) IN: Avionics in conceptual system planning; Proceedings of the Eighth Annual IEEE Symposium, Dayton, OH, Dec. 3, 1986 (A88-16912 05-66). New York, Institute of Electrical and Electronics Engineers, Inc., 1986, p. 37-42.

ABS: Avionics systems are becoming an increasingly important component of advanced aircraft and thus an important part of the planning for these new aircraft. Methods and tools are needed to support the definition of avionics systems architectures in the early planning stages. Techniques using a flexible distributed processing architecture concept based on generic building blocks are described to meet this need. A two stage process is outlined. The first stage uses structured analysis techniques to describe the functional design of the system. The second stage maps these functional processes into the physical components of the architecture. A prototype tool to aid these design techniques is described. 86/00/00 88A16916

UTTL: Integrated avionics - Watershed in aeronautical systems development

AUTH: A/BORKY, JOHN M.

PAA: A/(USAF, Aeronautical Systems Div., Wright-Patterson AFB, OH) IN: Avionics in conceptual system planning; Proceedings of the Eighth Annual IEEE Symposium, Dayton, OH, Dec. 3, 1986 (A88-16912 05-66). New York, Institute of Electrical and Electronics Engineers, Inc., 1986, p. 29-34.

ABS: Trends in avionics technology and systems for aeronautical weapon systems are reviewed, with attention given to functional integration, R&M, digital electronics, and modularity. Design ground rules which must be imposed to realize the full potential of avionics systems are considered; emphasis is placed on architecture, R&M design, standardization, and life cycle upgrading.

Finally, implications for system planning are discussed. 86/00/00 88A16915

UTTL: Integrating avionics into the conceptual design phase of aeronautical systems

AUTH: A/QUINN, GORDON F.; B/BREZA, MICHAEL J.

PAA: B/(USAF, Directorate of Design Analysis, Wright-Patterson AFB, OH) IN: Avionics in conceptual system planning; Proceedings of the Eighth Annual IEEE Symposium, Dayton, OH, Dec. 3, 1986 (A88-16912 05-66). New York, Institute of Electrical and Electronics Engineers, Inc., 1986, p. 15-18.

ABS: The role of avionics in the aircraft design process is examined, and it is shown why avionics system parameters should be subjected to iteration with airframe, propulsion, and armament parameters in the early conceptual design tradeoff process. It is noted that current aircraft design synthesis and analysis procedures can be modified or expanded to incorporate an avionics suite synthesis and analysis procedure that is integral to the overall aeronautical system design methodology. It is emphasized that the incorporation of avionics considerations into the aeronautical system design process with quantifiable measure of merit will significantly improve the performance-to-cost ratio of new aeronautical weapon systems. 86/00/00 88A16913

UTTL: Avionics in conceptual system planning; Proceedings of the Eighth Annual IEEE Symposium, Dayton, OH, Dec. 3, 1986 Symposium sponsored by IEEE. New York, Institute of Electrical and Electronics Engineers, Inc., 1986, 92 p. For individual items see A88-16913 to A88-16920.

ABS: The papers presented in this volume deal with various aspects of the problem of integrating avionics into total system design during the concept formulation stage, with particular attention given to impacts upon definition of requirements; future avionics concepts; tradeoffs between the vehicle, propulsion, and avionics; integration of supportability into the design; and acquisition strategies. Papers are included on system architecture design and tools for a distributed avionics system; the design agent process as a strategy for future avionics competition enhancement and quality assurance; the avionics acquisition process beyond the year 2000; and electromagnetic compatibility modeling for future avionics systems. 86/00/00 88A16912

UTTL: Software considerations for interfacing avionics computers and MUX buses

AUTH: A/PLEHATY, S. LLOYD

PAA: A/(Norden Systems, Inc., Norwalk, CT) IN: Aerospace Avionics Equipment and Integration Conference and Exhibit, Phoenix, AZ, Apr. 23, 24, 1986, Proceedings (A88-15575

ABS: 04-33). Warrendale, PA. Society of Automotive Engineers, Inc. 1986. p. 89-94.
The impact of avionics-system complexity on the design of interface bus terminals to link MIL-STD-1750A computers with MIL-STD-1553B data buses is discussed, with a focus on software aspects. The characteristics of typical state-of-the-art avionics are reviewed, including simple synchronous systems, complex multimode systems, and asynchronous systems. Consideration is given to bus features such as message chaining, selective interrupts, automatic retry, memory addressing, I/O buffering, buffer-access conflicts, terminal control, and I/O processing. The need for standardized smart bus terminals capable of handling asynchronous I/O is stressed, and it is suggested that such standards could facilitate the development of standardized software drivers and software tools.

RPT#: SAE PAPER 860853   86/00/00   88A15586

UTTL: Avionics, artificial intelligence, and embedded processing systems
AUTH: A/KOGGE, PETER M.
PAA: A/(IBM, Federal Systems Div., Owego, NY)   IN: AIAA Computers in Aerospace Conference, 6th, Wakefield, MA, Oct. 7-9, 1987, Technical Papers (A88-12526 02-61). Washington, DC, American Institute of Aeronautics and Astronautics, 1987. p. 236-245.
ABS: This paper addresses the characteristics that an embedded processing system must have to support AI functions for various avionics applications. The emphasis is on those kinds of functions that are unconventional in nature, computationally stressing, and of high value to future real-time avionics systems. The topics covered include a brief review of the major computational models for AI in use today, their appropriateness to several avionics AI applications, what their implementation means in terms of processor architectures, and how they might be supported in high performance avionics systems of the near future.

RPT#: AIAA PAPER 87-2818   87/00/00   88A12558

UTTL: Automated thermal and reliability analysis of avionics designs
AUTH: A/DIRE, JOHN D.;   B/FRANKS, DONALD E.;   C/CHOI, HOON S.
PAA: A/(General Dynamics Corp., San Diego, CA);   C/(General Dynamics Corp., Fort Worth, TX)   (AIAA, AHS, and ASEE, Aircraft Systems, Design and Technology Meeting, Dayton, OH, Oct. 20-22, 1986, AIAA Paper 86-2730.) Journal of Aircraft (ISSN 0021-8669), vol. 24, Aug. 1987, p. 499-502. Previously cited in issue 05, p. 595, Accession no. A87-17948.   87/08/00   87A50335

UTTL: Design and test of integrated sensors, avionics, and flight controls - Experiences in developing an

AUTH: A/HOWARD, JOHN D.;   B/SKOOG, MARK A.
PAA: B/(USAF, Flight Test Center, Edwards AFB, CA)   IN: Society of Flight Test Engineers, Annual Symposium, 17th, Washington, DC, Aug. 10-14, 1986, Proceedings (A87-49198 21-01). Lancaster, CA, Society of Flight Test Engineers, 1986, p. 6.1-1 to 6.1-8.
ABS: Methods for testing integrated avionics and flight control systems are described and analyzed. The AFTI/F-16 program, the air-to-surface Automated Maneuvering Attack System (AMAS), and the System Wide Integrity Management system are briefly described. The software test cycle and the use of hardware and software simulations are considered along with the use of an avionics laboratory at the test site. Finally, flight test techniques and results in testing the air-to-surface AMAS are reviewed.   86/00/00   87A49212

UTTL: Avionic integration in future battlefield helicopters
AUTH: A/THOMAS, PAUL
PAA: A/(Royal Aircraft Establishment, Farnborough, England)   European Rotorcraft Forum, 11th, London, England, Sept. 10-13, 1985, Paper. 8 p.
ABS: Programs undertaken by the RAE to develop fully integrated avionics and weapon systems for future battlefield helicopters are described. It is believed that optimum integration can be achieved by considering the following factors: (1) the identification of common avionics-system areas with a view to providing a single element to serve a number of functions; and (2) the division of responsibility between the crew and the avionics system. A schematic outline is presented of the integrated avionics and weapon system to be installed in a Lynx helicopter at Farnborough.   85/09/00   87A48917

UTTL: The impact of new technologies on the life cycle cost of avionic systems
AUTH: A/WILCOCK, G. W.;   B/OLDFIELD, D.
PAA: B/(Royal Aircraft Establishment, Farnborough, England)   IN: Cost-effective avionic and weapon systems; Proceedings of the Spring Convention, London, England, May 14, 15, 1986 (A87-48051 21-83). London, Royal Aeronautical Society, 1987. p. 13.1-13.17.
ABS: An assessment is made of the likely impact of foreseeable hardware and software advancements of future avionics systems. It is anticipated that VLSI will be a primary factor in the enhancement of performance and reduction of life cycle costs, through the generation of custom chips for specific applications. Software changes will be of greater significance to cost reduction than system performance, and demand that attention be given to software reliability. Future hardware will consist of a mixture of LRUs and LRMs. LRM-based subsystems are likely

to lead to a radical change in maintenance philosophies, and therefore in life cycle costs. 87/00/00 87A48064

UTTL: Cost-effective avionic and weapon systems; Proceedings of the Spring Convention, London, England, May 14, 15, 1986 Convention sponsored by the Royal Aeronautical Society. London, Royal Aeronautical Society, 1987. 165 p. For individual items see A87-48052 to A87-48067.

ABS: The present conference on the cost-effectiveness of avionics and weapons for military aircraft gives attention to the cost-effectiveness of research, methods for forecasting development and production costs, the control of procurement cost contracting techniques, the management of avionics development processes, and the maintenance of production cost targets. Also discussed are in-service maintenance, next-generation combat aircraft avionics, standards for avionics interfaces, avionics quality and environmental standards, the impact of new technologies on the life cycle of avionics systems, the system-impact of costs of sensors and sensor technologies, data bus systems, and integrated avionics systems. 87/00/00 87A48051

AUTH: A/HETZNER, W.; B/KOHL, W.
PAA: B/(Messerschmitt-Boelkow-Blohm GmbH, Munich, West Germany)
UTTL: Radar requirements for future avionics systems
IN: Radar Technology 1986; Symposium, 6th, Bremen, West Germany, Nov. 4-6, 1986. Reports (A87-45876 20-32). Duesseldorf, Deutsche Gesellschaft fuer Ortung und Navigation, 1986. p. C3.-C3.15. In German.

ABS: The performance demands on radar equipment for future aircraft are reviewed. General requirements include integrability in advanced avionics architectures with VHSIC components, high-speed data buses, speech-recognition and synthesis devices, and AI features: low probability of intercept; polarization agility; advanced conformal or phased-array antennas; automatic correction for radome-induced boresight errors; light weight; and reduced cooling and power demands. These requirements are discussed in detail for combat aircraft (air-air and air-ground operation), early-warning aircraft, and maritime patrol aircraft. 86/00/00 87A45879

UTTL: Standardization and logistic support cost effectiveness of advanced avionics systems
AUTH: A/BUONTEMPO, V.
PAA: A/(Selenia S.p.A., Pomezia, Italy) DGLR, European Rotorcraft Forum, 12th, Garmisch-Partenkirchen, West Germany, Sept. 22-25, 1986. Paper. 17 p.

ABS: Modular standardization which has already been adopted within avionic systems can also be used to optimize the logistic support in terms of performance (such as operative availability, maintainability, system reliability, and testing) and costs (purchasing, maintenance, spare parts, technical documentation, training, and ground support equipment). After a short description of the status of technological integration, hardware, and software standardization available to date on avionic systems, a demonstration of the effectiveness of the new maintenance philosophy and concepts (elimination of the second maintenance level) is given. The results derived can be extended to naval and ground defense systems. 86/09/00 87A43468

UTTL: Integrated communications navigation identification avionics moves into the next generation avionics
AUTH: A/CANANA, PETER
PAA: A/(TRW, Inc., TRW Military Electronics and Avionics Div., San Diego, CA) IN: PLANS '86 - Position Location and Navigation Symposium, Las Vegas, NV, Nov. 4-7, 1986. Record (A87-41351 18-17). New York, Institute of Electrical and Electronics Engineers, 1986. p. 510-514.

ABS: The design of the Integrated Communications Navigation Identification Avionics (ICNIA) is described. The ICNIA architecture is flexible, and the flexibility is implemented with RF switches and common buses. The ICNIA terminal contains 28 different module types all interconnected to support rapid reconfiguration for high Communications Navigation Identification (CNI) availability. These module types can be classified as either RF standard modules or digital standard modules. The software used in the ICNIA is examined. The information an applicant needs to include in the request, and methods for providing adequate security for the system are considered. 86/00/00 87A41404

UTTL: Ring laser gyro inertial and GPS integrated navigation system for commercial aviation
AUTH: A/DIVAKARUNI, SUDHAKAR P.; B/STENSLAND, RICHARD A.; C/BRENNER, MATS B.
PAA: A/(Honeywell Systems and Research Center, Minneapolis, MN) ; C/(Honeywell, Inc., Commercial Aviation Div., Minneapolis, MN) IN: PLANS '86 - Position Location and Navigation Symposium, Las Vegas, NV, Nov. 4-7, 1986. Record (A87-41351 18-17). New York, Institute of Electrical and Electronics Engineers, 1986. p. 73-80.

ABS: The ring laser gyro-based inertial reference and navigation systems presently discussed are integrated in order to take advantage of GPS in civil aviation use, in conjunction with the ARINC 700-series of avionics equipment. The basic feature of the system architecture employed is that it furnishes pure inertial, autonomous GPS and GPS-bounded inertial navigation outputs while preserving the inertial reference system's integrity. Attention is given to the navigation filter algorithms

that derive the autonomous GPS solution, the GPS/inertial hybrid solution, and the inertial sensor calibration data. 86/00/00 87A41364

UTTL: Helicopter systems integration
AUTH: A/BELMAN, HARRIS J.
PAA: A/(IBM Corp., Federal Systems Div., Owego, NY)
Vertiflite (ISSN 0042-4455), vol. 33, Mar.-Apr. 1987, p. 22-27.
ABS: Systems integration makes it possible to create different [...]-capable helicopters from one basic helicopter [...], each with varying levels of avionics and mission capability. Experience as a systems integrator on major weapons systems is reported, most importantly as system prime contractor for the U.S. Navy's LAMPS MK III (Light Airborne Multi-Purpose System) ship/air weapon system, as avionics prime contractor for the U.S. Air Force's HH/60A Nighthawk combat search and rescue helicopter, and as a participant in the U.S. Army's LHX (Light Helicopter Experimental) program. The status of the three programs from the avionics integration viewpoint is considered. The next generation of helicopters will include artificial intelligence-based systems, for such functions as maintenance/self-diagnostics, tactical planning, and sensor fusion. 87/04/00 87A35673

UTTL: Avionics systems integration - There is a better way
AUTH: A/JOHNSON, MERLIN G.
PAA: A/(Teledyne Systems Co., Northridge, CA) Vertiflite (ISSN 0042-4455), vol. 33, Mar.-Apr. 1987, p. 18-22.
ABS: An approach to minimizing the total cost of Special Test Equipment is presented, following the example set by the SH-60F CV/ASW Helicopter (CV Helo) mission avionics system development program. A single full-capability System Integration Facility was built in transportable form. For the CV Helo (served by a crew of four), the avionics system is integrated by a new Tactical Data Management Subsystem (TDMS) with the following elements: dual redundant MIL-STD-1553B data bus; dual redundant tactical data processors; four identical programmable control-display units; dual independent multifunction displays; dual electronic horizontal situation video displays; centralized TEMPEST-capable communication system controller; integrated armament system controller; and a dual redundant red/black bus isolator. Advantages of the approach - including savings in time and costs, ability to provide a complete closed-loop real-time simulation environment for system integration and software validation, and application to other aircraft - are noted. 87/04/00 87A35672

UTTL: Rotorcraft integration for improved mission cost effectiveness

AUTH: A/SCOUGHTON, CRAIG
PAA: A/(Allied-Bendix Aerospace, Flight Systems Div., Teterboro, NJ) Vertiflite (ISSN 0042-4455), vol. 33, Mar.-Apr. 1987, p. 12-16.
ABS: The integration of electronic systems on board modern military helicopters - to reduce the clutter and complexity of the proliferating black boxes commonly used to provide flight control, wide spectrum communications, precise navigation, threat protection, weapons guidance, cockpit instrumentation, obstacle/threat detection, and aircraft subsystem monitoring is examined. Basic system functions are noted that can be managed by integrated systems: the mission computer, remote terminal unit, display processor, crew station data entry, and crew station displays. System integration technologies currently under development include very-high-speed integrated circuitry, common modules, and the high-speed data bus. Examples of the more integrated approach are the advanced crew stations for the Boeing Model 360, the Air Force HH-60, the Army AHIP, and the joint service V-22. 87/04/00 87A35671

UTTL: System methods for avionics development and integration
AUTH: A/GARTZ, PAUL EBNER
PAA: A/(Boeing Commercial Airplane Co., Seattle, WA) IN: Digital Avionics Systems Conference, 7th, Fort Worth, TX, Oct. 13-16, 1986, Proceedings (A87-31451 13-01). New York, Institute of Electrical and Electronics Engineers, Inc., 1986, p. 784-792.
ABS: A set of life cycle methods were developed in 1980 and 1981 and used in the later phases of the 757/767 airplane programs. They have been used as a framework to establish and guide the introduction of a wide use of similar methods for the future avionics of Boeing's next airplane, the 7J7. The methods were designed to improve communication of the system's requirements, architecture, and implementation to a wide group of interested parties. A major aspect of these methods is that they were designed to be used for systems in general not just software systems. This paper describes the background, goals, and objectives leading to the need for systems engineering methods. It also describes the methods and gives an example of their use. 86/00/00 87A31548

UTTL: Digital avionics systems - Overview of FAA/NASA/industry-wide briefing
AUTH: A/LARSEN, WILLIAM E.; B/CARRO, ANTHONY
PAA: A/(NASA, Ames Research Center; FAA, Moffett Field, CA); B/(FAA, Technical Center, Atlantic City, NJ) CORP: National Aeronautics and Space Administration. Ames Research Center, Moffett Field, Calif.; Federal Aviation Administration, Moffett Field, Calif.; Federal Aviation Administration, Atlantic City, N.J. IN: Digital Avionics

Systems Conference, 7th, Fort Worth, TX, Oct. 13-16, 1986, Proceedings (A87-31451 13-01). New York, Institute of Electrical and Electronics Engineers, Inc., 1986, p. 731-737.

ABS: The effects of incorporating digital technology into the design of aircraft on the airworthiness criteria and certification procedures for aircraft are investigated. FAA research programs aimed at providing data for the functional assessment of aircraft which use digital systems for avionics and flight control functions are discussed. The need to establish testing, assurance assessment, and configuration management technologies to insure the reliability of digital systems is discussed; consideration is given to design verification, system performance/robustness, and validation technology. 86/00/00 87A31543

UTTL: Problems with failure modes and effects analysis for digital avionics
AUTH: A/MECHT, HERBERT
PAA: A/(SoHar, Inc., Los Angeles, CA) IN: Digital Avionics Systems Conference, 7th, Fort Worth, TX, Oct. 13-16, 1986. Proceedings (A87-31451 13-01). New York, Institute of Electrical and Electronics Engineers, Inc., 1986, p. 695-700.

ABS: The provisions of the MIL-STD-1629A standard for Failure Modes and Effects Analysis (FMEA) are discussed with respect to their applicability to digital avionics equipment, and problem areas are highlighted. It is noted that current practices usually circumvent rather than correct deficiencies, and that they introduce duplication and uncertainty into the application of FMEA-related information in the design of digital equipment. An approach in which an individual FMEA is restricted to one hierarchical level, and in which a built-in feedback mechanism identifies and corrects its own deficiencies by identifying FMEA problem areas as part of the normal reporting system, is proposed. 86/00/00 87A31539

UTTL: Fault-free performance validation of avionic multiprocessors
AUTH: A/CZECK, EDWARD W.; B/FEATHER, FRANK E.; C/GRIZZAFFI, ANN MARIE; D/SEGALL, ZARY Z.; E/FINELLI, GEORGE B.
PAA: D/(Carnegie-Mellon University, Pittsburgh, PA); E/(NASA. Langley Research Center, Hampton, VA) CORP: National Aeronautics and Space Administration. Langley Research Center, Hampton, Va. IN: Digital Avionics Systems Conference, 7th, Fort Worth, TX, Oct. 13-18, 1986. Proceedings (A87-31451 13-01). New York, Institute of Electrical and Electronics Engineers, Inc., 1986, p. 670-677.

ABS: This paper describes the application of a portion of a validation methodology to NASA's Fault-Tolerant Multiprocessor System (FTMP) and the Software Implemented Fault-Tolerance (SIFT) computer system. The methodology entails a building block approach, starting with simple baseline experiments and building to more complex experiments. The goal of the validation methodology is to thoroughly test and characterize the performance and behavior of ultrareliable computer systems. The validation methodology presented in this paper showed that the methodology is not machine specific and can be used in lieu of life testing approaches. By applying a building block approach at the system level, the machine complexity was broken down to manageable levels independent of system implementation. 86/00/00 87A31538

UTTL: Software reliability - Measures and effects in flight critical digital avionics systems
AUTH: A/DUNN, WILLIAM R.
PAA: A/(NASA, Ames Research Center; Southern Colorado, University, Moffett Field, CA) CORP: National Aeronautics and Space Administration. Ames Research Center, Moffett Field, Calif. IN: Digital Avionics Systems Conference, 7th, Fort Worth, TX, Oct. 13-16, 1986. Proceedings (A87-31451 13-01). New York, Institute of Electrical and Electronics Engineers, Inc., 1986, p. 664-669. FAA-supported research.

ABS: The paper discusses software reliability as it applies particularly to design and evaluation of flight-critical digital avionics systems. Measures of software reliability, measurement methods and reliability (macro-) models are discussed. Recent work assessing their accuracy in predicting software errors in 'fly-by-wire' Newtonian applications is presented. Additional, detailed topics are discussed including software error distributions (e.g. catastrophic vs. noncatastrophic) and the effects of system growth/maturity on reliability improvement. In practical flight-critical digital applications, software reliability improvement is sought through use of parallel, redundant software (i.e. N-version programming) or backup software that can be invoked in the event of (primary) software failure. Achievable reliability levels are however highly sensitive to common-mode specification and programming errors. Recent data correlating these errors with net software reliability are discussed. 86/00/00 87A31537

UTTL: Real-time fault tolerant software in distributed avionics systems architectures using digital data buses
AUTH: A/HITT, ELLIS F.
PAA: A/(Battelle Columbus Laboratories, OH) IN: Digital Avionics Systems Conference, 7th, Fort Worth, TX, Oct. 13-16, 1986. Proceedings (A87-31451 13-01). New York, Institute of Electrical and Electronics Engineers, Inc., 1986, p. 523-528.

ABS: Real-time fault tolerant multi-version and recovery block software techniques, and problems confronted in applying them to avionics system architectures using distributed multiprocessing networks that employ standard digital avionics data buses are described. Also described is the impact of emerging software engineering tools, resulting from application of knowledge based systems and software simulators, on avionics software development. 86/00/00 87A31517

AUTH: A/KEINER, WILLIAM L.
PAA: A/(U.S. Navy, Surface Weapons Center, Dahlgren, VA). IN: Digital Avionics Systems Conference, 7th, Fort Worth, TX, Oct. 13-16, 1986, Proceedings (A87-31451 13-01). New York, Institute of Electrical and Electronics Engineers, Inc.. 1986, p. 397-401.
UTTL: Testability management for digital avionics

ABS: MIL-STD-2165, 'Testability Program for Electronic Systems and Equipments,' was issued on Jan. 26, 1985. This standard may be applied to all electronics developments within the Department of Defense. This paper discusses the role of the new military standard as a management tool for the development of testable avionics systems and equipments. A framework is presented for managing testability requirements trade-offs, design tradeoffs and testability evaluations. 86/00/00 87A31500

AUTH: A/PITTS, JAMES F.; B/ZAGARDO, VINCENT S.
PAA: B/(Westinghouse Electric Corp., Baltimore, MD) IN: Digital Avionics Systems Conference, 7th, Fort Worth, TX, Oct. 13-16, 1986, Proceedings (A87-31451 13-01). New York, Institute of Electrical and Electronics Engineers, Inc.. 1986, p. 373-381.
UTTL: Digital processing for emerging avionics systems

ABS: This paper discusses the methodology utilized to derive a digital avionics architecture and the requirements utilized to derive the subsequent tradeoff analyses. A resultant digital avionics architecture is described, outlining its overall performance and the constituent signal and mission processing module set. The approach is highly synergetic with ongoing development programs with extremely high payoff (size, weight, and power) through effective utilization of high performance fixed point and floating point processor modules. 3/4 ATR packaging approaches are discussed along with reliability and maintainability characteristics. A distributed fault tolerant operating system is described along with a complete support software environment that facilitates programming at the Ada level. 86/00/00 87A31497

UTTL: A Hardware and Software Integration Facility (HSIF) for SH-60F CV-Helo
AUTH: A/DONOGHUE, PATRICK J.; B/JENSEN, PREBEN; C/PEABODY,

PAA: ROBERT M.
C/(Teledyne Systems Co., Northridge, CA) IN: Digital Avionics Systems Conference, 7th, Fort Worth, TX, Oct. 13-16, 1986, Proceedings (A87-31451 13-01). New York, Institute of Electrical and Electronics Engineers, Inc.. 1986, p. 212-218.
ABS: This paper discusses the requirements, development, and use of a mobile Hardware and Software Integration Facility (HSIF) which has been developed to support the mission avionics of a carrier based anti-submarine warfare helicopter, namely the SH-60F DV-Helo. The HSIF supports the full life-cycle of the avionics system including initial development, system integration, flight test, customer (Navy) test and evaluation, and fleet operations. The facility uses a DEC VAX 11/785 for central processing and Motorola 68000 Based VME Modules for avionics control. It has full capabilities for stimulation and testing at the box or WRA level, flight software development and download, stimulation of the mission avionics with simulated mission scenarios, and data reduction and analysis of collected information. 86/00/00 87A31478

UTTL: Avionics system development in a ground based laboratory environment
AUTH: A/WEEKS, RICHARD A.
PAA: A/(VERAC, Inc., San Diego, CA) IN: Digital Avionics Systems Conference, 7th, Fort Worth, TX, Oct. 13-16, 1986, Proceedings (A87-31451 13-01). New York, Institute of Electrical and Electronics Engineers, Inc., 1986, p. 204-208.
ABS: This paper describes the utilization of ground-based laboratories for the development of complex, highly-integrated avionics systems. Concept development laboratories, full-mission flight/mission simulator laboratories, and various avionics integration laboratories are discussed to substantiate the need of a systematic approach to avionics development in a ground-based laboratory environment. 86/00/00 87A31476

UTTL: A generic methodology for passive sensor avionics emulation in man-in-the-loop cockpit simulators
AUTH: A/ISAKSON, KYLE
PAA: A/(Bell Helicopter Textron, Fort Worth, TX) IN: Digital Avionics Systems Conference, 7th, Fort Worth, TX, Oct. 13-16, 1986, Proceedings (A87-31451 13-01). New York, Institute of Electrical and Electronics Engineers, Inc.. 1986, p. 190-195.
ABS: Ground based man-in-the-loop simulation is becoming increasingly important in the evaluation and testing of advanced cockpits and associated avionics suites. In the future the availability to prototype expert systems using simulations is a desirable goal. This paper addresses a real-time sensor emulation capable of interfacing with a target database containing up to 256 targets (up to 32 at

any one time). The generic approach taken in this
simulation: (1) increases the fidelity and relevance of
the simulation of the proposed design, (2) allows for
evaluation of the actual mission computer algorithms in
the flight simulator early in the development process, and
(3) can reduce the duplication of effort in the software
development task both for the actual aircraft and the
flight simulator.   86/00/00   87A31474

UTTL: Use of microprocessor elements in simulation of
digital avionic systems
AUTH: A/BANOUSSAKIS, E.; B/SVOBODA, J. V.; C/MCKINNON, G. M.;
D/PETRUZZIELLO, F.
PAA: D/(Concordia University, Montreal, Canada)   IN: Digital
Avionics Systems Conference, 7th, Fort Worth, TX, Oct.
13-16, 1986, Proceedings (A87-31451 13-01). New York,
Institute of Electrical and Electronics Engineers, Inc.,
1986, p. 185-189.
ABS: Design of microprocessor based avionic systems is the
current state-of-the-art in civil and military aviation
engineering. High reliability, reasonable development cost
and design flexibility are ideal features that enable
avionics manufacturers to produce powerful and very
reliable equipment. This paper presents the results of
experimental research in simulation of digitally
controlled avionic systems using microprocessors. This
research is part of a development effort toward an
advanced general aviation instrument flight rules (IFR)
simulation system. The current microprocessor technology
is supportive of such training equipment design and
development.   86/00/00   87A31473

UTTL: Advanced avionics display processor architecture
AUTH: A/KROFCHICK, BARRY
PAA: A/(Litton Systems Canada, Ltd., Toronto)   IN: Digital
Avionics Systems Conference, 7th, Fort Worth, TX, Oct.
13-16, 1986, Proceedings (A87-31451 13-01). New York,
Institute of Electrical and Electronics Engineers, Inc.,
1986, p. 177-182.
ABS: The changing role of the pilot in next generation fighter
aircraft is discussed. This leads to a new set of
requirements for next generation avionics display formats,
and hence display processors. The need for an avionics
graphics interface which can handle complex imagery and
associated picture dynamics without overburdening the
mission computer or the avionics bus is established. The
Litton Graphics System (LGS), an interface command
language, and also an advanced avionics Display Processor
functional specification, is presented. LGS picture
representation, and generation models are discussed, along
with multiple display facilities, geometric modelling and
special graphics features.   86/00/00   87A31472

UTTL: A customer's perspective of integrated CNI avionics
AUTH: A/HARRIS, ROBERT L.
PAA: A/(USAF, Avionics Laboratory, Wright-Patterson AFB, OH)
IN: Digital Avionics Systems Conference, 7th, Fort Worth,
TX, Oct. 13-16, 1986, Proceedings (A87-31451 13-01). New
York, Institute of Electrical and Electronics Engineers,
Inc., 1986, p. 76-84.
ABS: Aspects of the development of integrated CNI avionics
systems are discussed in the context of the Integrated
Communication Navigation Identification Avionics (ICNIA)
program. The ICNIA customer's major problems with today's
CNI avionics are examined, and the customer's views of
electronics technology as it bears on avionics are
summarized. The relationship between ICNIA and the Air
Force Reliability and Maintainability Program is
addressed, stressing the reliability and maintainability
goals that have been established to guide the development
of the Air Force full-function ICNIA system. ICNIA
software and fault tolerance and ICNIA technology and
logistics support are discussed.   86/00/00   87A31459

UTTL: Digital Avionics Systems Conference, 7th, Fort
Worth, TX, Oct. 13-16, 1986, Proceedings
Conference
sponsored by IEEE and AIAA. New York, Institute of
Electrical and Electronics Engineers, Inc., 1986, 831 p.
For individual items see A87-31452 to A87-31549.
ABS: Various papers on digital avionics systems are presented.
The general topics addressed include: software management;
standard modular avionics, communication, navigation, and
identification; space systems technology; rotorcraft
avionics; advanced control/display technology for crew
systems; software development and evaluation tools; data
buses in subsystem interconnections; digital map
techniques; commercial transportation avionics/collision
avoidance systems; and human factors in crew systems. Also
considered are: software verification and quality
assurance; sensor signal and data processing; design for
testability in system concept; fiber optics; Ada; fault
tolerance and reconfigurability in system concept; digital
flight controls; data link system applications; artificial
intelligence and expert systems; advanced digital
integrated circuits technology, design, and testability;
integrated flight/propulsion control; and general aviation
avionics.   86/00/00   87A31451

UTTL: V-22 avionics methodology and design
AUTH: A/KLIPPERT, R.; B/KEY, E.
PAA: A/(IBM Corp., Owego, NY);   B/(Boeing Military Aircraft
Co., Wichita, KS)   IN: American Helicopter Society,
Annual Forum, 42nd, Washington, DC, June 2-4, 1986,
Proceedings. Volume 2 (A87-19201 06-01). Alexandria, VA,
American Helicopter Society, 1986, 3 p.
ABS: Attention is given to the design features and performance
capabilities of the V-22's avionics system hardware and

software elements. Software encompasses communication-identification, mission management, and test module monitoring; hardware prominently includes four multifunction displays driven by two display processors, two control display units, and a dual helmet-mounted display. Attention is given to the two-company development program management aspects of V-22 avionics endeavors. 86/00/00 87A19289

UTTL: Management of utility systems in the Experimental Aircraft Programme
AUTH: A/MOIR, I.; B/SEABRIDGE, A. G.
PAA: A/(Smiths Industries, PLC, Cheltenham, England); B/(British Aerospace, PLC, Warton Div., Preston, England) Aerospace (UK) (ISSN 0305-0831), vol. 13, Sept. 1986, p. 28-34.
ABS: The design of an integrated data processing system for control and management of aircraft utility systems in the European Experimental Aircraft Program is described. Block diagrams are presented of the architecture which emerged for semiautomated control and management of systems such as the engine, fuel and hydraulic systems, environmental systems, secondary power system, etc. Finally, software development techniques which were implemented to realize the project goals in the new hardware are summarized. Demonstration flights are to begin in 1986. 86/09/00 87A18998

UTTL: Automated thermal and reliability analysis of avionics designs
AUTH: A/DIRE, J. D.; B/FRANKS, D. E.; C/CHOI, H. S.
PAA: C/(General Dynamics Corp., Convair Div. and Space Systems Div., San Diego, CA) AIAA, AHS, and ASEE, Aircraft Systems, Design and Technology Meeting, Dayton, OH, Oct. 20-22, 1986. 6 p.
ABS: This paper describes a new automated process currently being developed and implemented at General Dynamics Convair/Space Systems Divisions to enhance the reliability of printed circuit boards (PCBs). The process is entitled 'Convair-Computer Aided Reliable Design' (C-CARD) and is used to optimize the component mounting locations on a PCB for thermal reliability. PCB information contained in a computer aided design (CAD) data base is used to automatically produce a thermal mathematical model for use with the SINDA finite difference heat transfer code. Resulting temperature and reliability predictions are displayed and evaluated by the packaging engineer based on the design requirements. Once the PCB design is optimized for thermal reliability it is then released for manufacture. In this way, the heat-transfer analysis is integrated into the design process and provides the packaging engineer with a thermally safe design in minutes instead of weeks.
RPT#: AIAA PAPER 86-2730 86/10/00 87A17948

UTTL: Pilot's associate demonstration one - A look inside
AUTH: A/SHELNUTT, J. B.; B/STENERSON, R. O.; C/NELSON, P. C.; D/MARKS, P. S.
PAA: B/(Boeing Military Airplane Co., Seattle, WA); D/(Advanced Decision Systems, Mountain View, CA) IN: NAECON 1986; Proceedings of the National Aerospace and Electronics Conference, Dayton, OH, May 19-23, 1986. Volume 4 (A87-16726 05-01). New York, Institute of Electrical and Electronics Engineers, 1986, p. 1184-1189.
ABS: The software architecture used in the Pilot's Associate D-1 program for examining limited prototype expert systems for military avionics systems is described. Attention is also given to the knowledge representation and control techniques applied in a crew station information manager (CSIM). The system architecture comprised a situation assessor, planner, integration controller and the man-machine interface. The components provided mission status, areal IFF functions, battle damage assessment and threat response information. The CSIM was devised to facilitate the interactions between pilots and future avionics systems and to perform situation assessment, route planning and overall systems control, using data from a distributed system architecture. 86/00/00 87A16841

UTTL: Pilot's associate demonstration one - A look back and ahead
AUTH: A/POHLMANN, L. D.; B/PAYNE, J. R.
PAA: A/(Boeing Military Airplane Co., Seattle, WA); B/(Advanced Decision Systems, Mountain View, CA) IN: NAECON 1986; Proceedings of the National Aerospace and Electronics Conference, Dayton, OH, May 19-23, 1986. Volume 4 (A87-16726 05-01) New York, Institute of Electrical and Electronics Engineers, 1986, p. 1178-1183.
ABS: Progress in the programming of prototype expert systems for military aircraft avionics as part of the Boeing D-1 demonstration program is summarized. The D-1 effort exploits previous pictorial display format work. AI techniques and existing simulation facilities, and is directed at both air-to-ground and air-to-air combat modes. Implemented on a LISP processor. D-1 incorporates a specific sequence of mission situations considered amenable to expert systems applications. The prototype system architecture is described in terms of a crew station information manager, integration controller, situation assessor and a planner. 86/00/00 87A16840

UTTL: The B-1B Central Integrated Test System Expert Parameter System
AUTH: A/MONTGOMERY, G. J.
PAA: A/(USAF, Wright Aeronautical Laboratories, Wright-Patterson AFB, OH) IN: NAECON 1986; Proceedings

of the National Aerospace and Electronics Conference, Dayton, OH, May 19-23, 1986. Volume 4 (A87-16726 05-01). New York, Institute of Electrical and Electronics Engineers, 1986, p. 1170-1175.

ABS: The B-1B Central Integrated Test System (CITS) provides a comprehensive on aircraft diagnostic capability and records approximately 19,600 parameters. The B-1B CITS Expert Parameter System (CEPS) is an initiative to improve B-1B diagnostic capabilities by applying expert system and data analysis techniques to the in-flight recorded data. The manner in which CEPS enhances B-1B on and off aircraft diagnostic capabilities and reduces false alarm, can not duplicate and re-test okay occurrences will be presented. The CEPS capabilities will be discussed and an overview of the accomplishments and status of the CEPS program will be given. This paper will also illustrate the applicability of the B-1B CEPS concepts to other existing and future weapon systems. The ability to reduce future weapon system built-in test requirements through the use of on-aircraft expert systems will be discussed, along with the need for a ground based diagnostic system. 86/00/00 87A16839

UTTL: IESS - Dynamic test and evaluation facility for integrated CNI
AUTH: A/HALDEMAN, D. G.; B/WALLACE, E.
PAA: A/(TRW, Inc., TRW Defense Systems Group, Redondo Beach, CA); B/(USAF, Wright Aeronautical Laboratories, Wright-Patterson AFB, OH) IN: NAECON 1986; Proceedings of the National Aerospace and Electronics Conference, Dayton, OH, May 19-23, 1986. Volume 3 (A87-16726 05-01). New York, Institute of Electrical and Electronics Engineers, 1986, p. 1001-1005.
ABS: Features and applications of an integrated electromagnetic systems simulator (IESS) being developed to provide generic communications, navigation and identification (CNI) systems test capabilities for the Air Force are described. The IESS will generate, receive and respond with RF signals in the 2 MHz-1 GHz range in real-time for CNI units under test (UUT). Signals from satellites, stand-off jammers, ground stations, and other aircraft will be available for stimulating the dynamic environment of the UUTs. Account will be taken of the data rates and the flight paths of other signal-generating aircraft. The IESS is modular, which permits alteration to generate and receive new waveforms as they are identified. 86/00/00 87A16827

UTTL: Rapid prototyping of pilot interfaces as an aid to defining software requirements
AUTH: A/ARKUSINSKI, A.; B/HORNBACH, K.
PAA: B/(Lear Siegler, Inc., Instrument Div., Grand Rapids, MI), IN: NAECON 1986; Proceedings of the National Aerospace and Electronics Conference, Dayton, OH, May 19-23, 1986. Volume 3 (A87-16726 05-01). New York, Institute of Electrical and Electronics Engineers, 1986, p. 828-835.
ABS: Criteria are defined for a designing prototype user interfaces for evaluating digital avionics displays from the prospective user standpoint. The design goal is to limit the display to necessary data only, provide readable, comprehensible displays, and provide understandable transitions between screens. Design efforts must consider the size and shape of the screen and entry keys, selection of interaction mechanisms, display character and graphics, sound output, simulation of real-time external inputs, and ease of modification. It is also desirable that the system be portable, usable on a customer's computer, and allow for large-screen displays at conferences. 86/00/00 87A16807

UTTL: AFTI/F-16 Automated Maneuvering Attack System configuration development and integration
AUTH: A/RAMAGE, J. K.; B/BENNETT, W. S.
PAA: A/(USAF, Wright Aeronautical Laboratories, Wright-Patterson, AFB, OH); B/(General Dynamics Corp., Fort Worth, TX) IN: NAECON 1986; Proceedings of the National Aerospace and Electronics Conference, Dayton, OH, May 19-23, 1986. Volume 2 (A87-16726 05-01). New York, Institute of Electrical and Electronics Engineers, 1986, p. 538-549.
ABS: The AFTI/F-16 Advanced Development Program is developing and integrating advanced technologies that will improve mission capabilities through integration of task-tailored digital flight controls, fire control, attack sensors, weapon interfaces, mission avionics, and associated controls and displays into an Automated Maneuvering Attack System (AMAS). Phase II AMAS development expands the baseline systems in conjunction with a new FLIR-Laser Sensor/Tracker. This paper summarizes the Phase II AMAS configuration development and integration with emphasis on mechanization concepts, design rationale, and critical system-integration considerations. 86/00/00 87A16781

UTTL: Synthesis and test issues for future aircraft inertial systems integration
AUTH: A/BIEZAD, D. J.
PAA: A/(USAF, Institute of Technology, Wright-Patterson AFB, OH) IN: NAECON 1986; Proceedings of the National Aerospace and Electronics Conference, Dayton, OH, May 19-23, 1986. Volume 1 (A87-16726 05-01). New York, Institute of Electrical and Electronics Engineers, 1986, p. 279-285.
ABS: This paper surveys the potential benefits, possible pitfalls, and anticipated testing needs of integrating inertial guidance systems with systems dependent upon the availability for the electromagnetic spectrum. Advance preparations for testing integrated modules include upgraded and shared time reference standards on military aircraft, the development of expertise in modular

filtering, and the consideration of new test criteria for systems with embedded intelligence in its component subsystems. 86/00/00 87A16751

UTTL: OTV avionics and subsystems design utilizing life cycle cost techniques
AUTH: A/MAKI, S. C.
PAA: A/(General Dynamics Corp., Space Systems Div., San Diego, CA) IN: NAECON 1986; Proceedings of the National Aerospace and Electronics Conference, Dayton, OH, May 19-23, 1986. Volume 1 (A87-16726 05-01). New York, Institute of Electrical and Electronics Engineers, 1986, p. 136-143. Research supported by the General Dynamics Corp.
ABS: Traditional space vehicle avionics design trades consider factors such as weight, power, reliability, performance, software complexity, and hardware costs. For the OTV avionics study, a figure of merit (FOM) equation was developed that quantified the relationship between these factors and mission parameters (such as propellant delivery cost, and space station attendant manhour cost) for a total life cycle cost (LCC) comparison. The FOM equation was applied incrementally resulting in a curve having a minimal life cycle cost with a partial OTV redundancy. Some interesting results were obtained with this comprehensive approach. 86/00/00 87A16738

UTTL: The high speed interconnect system
AUTH: A/ANDERSEN, S. C.
PAA: A/(Sperry Computer Systems, Saint Paul, MN) IN: NAECON 1986; Proceedings of the National Aerospace and Electronics Conference, Dayton, OH, May 19-23, 1986. Volume 1 (A87-16726 05-01). New York, Institute of Electrical and Electronics Engineers, 1986, p. 100-110.
ABS: This paper describes a proprietary, second generation High Speed Interconnect System (HSIS) developed during 1985 as part of a cooperative Internal Research and Development project. The HSIS is a 100 MBPS token passing fiber optic bus which services up to 256 stations and utilizes an IEEE 802.4-like token passing protocol. During 1986 the SAE AE-9B linear token passing bus protocol is being incorporated into the HSIS and down-sizing through the use of VLSI/VHSIC components is being examined. The HSIS system architecture and design allows for technology insertion and supports a wide range of equipment types and capabilities by the incorporation of firmware personality modules. 86/00/00 87A16733

UTTL: Design for testability for future digital avionics systems
AUTH: A/SUBRAMANYAM, V. R.; B/STINE, L. R.
PAA: A/(TRW, Inc., San Diego, CA) IN: NAECON 1986; Proceedings of the National Aerospace and Electronics Conference, Dayton, OH, May 19-23, 1986. Volume 1 (A87-16726 05-01). New York, Institute of Electrical and Electronics Engineers, 1986, p. 21-25.
ABS: An integrated approach, referred to as the Maintenance and Diagnostics System (MADS), for testing the applicability of VHSIC, VLSI, and semicustom circuits in digital avionics systems is described. The MADS defines a module maintenance node (MMN) chip set for each line replaceable unit. The use of the MADS to test sequential circuits, to eliminate test vector/response storage, and to enhance fast coverage and reduce error escape is discussed; the integrated test system also provides a reduction in test time and false alarm rates. 86/00/00 87A16727

UTTL: Avionics systems for future commercial helicopters
BMFT, Statusseminar ueber Luftfahrtforschung und Luftfahrttechnologie, Munich, West Germany, Apr. 28-30, 1986. Paper. 23 p. In German.
ABS: The state of the art and future developments in avionics for commercial helicopters are discussed. Present systems of cockpit instrumentation with color displays, systems for testing these displays, and central control systems are described. Future sensors, cockpit instrumentation, and central controls are described, stressing the greater levels of control and information about the entire aircraft that will be concentrated in the cockpit. The use of simulators to test these future technologies before their installation in helicopters is discussed, and signal standardization, bus systems, and software that will be used in future commercial helicopters are considered. 86/04/00 87A14005

UTTL: Selection of media access protocol for distributed digital avionics
AUTH: A/RAY, A.; B/MCGOUGH, J.
PAA: A/(Pennsylvania State University, University Park); B/(Bendix Corp., Flight Systems Div., Teterboro, NJ) IN: 1986 American Control Conference, 5th, Seattle, WA, June 18-20, 1986, Proceedings. Volume 3 (A87-13301 03-63). New York, Institute of Electrical and Electronics Engineers, 1986, p. 1558-1562.
ABS: The paper presents the results of an ongoing research project where the objectives are to evaluate avionic network topologies and media access protocols in view of the distributed digital flight control systems (DDFCS) of high performance aircraft and to recommend a specific protocol for its prototype development. The selection of an appropriate protocol is critical for the stability of an aircraft because the DDFCS, in addition to the sampling time delay, is subject to the transport delay due to media access and message transmission in the network. The MIL-STD-1553B and SAE token ring protocols were analyzed and the analytical results have been verified by use of discrete event simulation techniques. 86/00/00

87A13436

UTTL: Smart probes for air data   Avionics (ISSN 0273-7639), vol. 10, June 1986, p. 8-11.

ABS: The characteristics of a distributed air data system (DADS) are described and compared to a central air data system. The DADS is an integrated system of sensors, transducers, and electronics, and is easy to expand and modify. The operation of the avionic computer utilized to process the air data is discussed. The output provided by the Rosemount instrument of the DADS is examined. 86/06/00   86A49498

AUTH: A/GIBSON, M. W.
PAA: A/(McDonnell Aircraft Co., Saint Louis, MO)   IN: Society of Flight Test Engineers, Annual Symposium, 16th, Seattle, WA, July 29-August 2, 1985 Proceedings (A86-47776 23-05). Lancaster, CA, Society of Flight Test Engineers, 1985, p. 4.4-1 to 4.4-6.

ABS: A high capacity digital data acquisition system for flight tests of high performance fighter aircraft must be small, yet capable of monitoring many parameters at high data rates in order to reduce the number of flights required in a given test program. Attention is presently given to a system answering these performance criteria, which will be used in F/A-18 flight testing. The system is structured in such a way that the user can assemble a data system architecture uniquely suited to the requirements of the given test, through the arrangement of an array of standard functional modules.   85/00/00   86A47790

AUTH: A/VIVET, M.
PAA: A/(Telecommunications Radioelectriques et Telephoniques, Paris, France)   ICAO Bulletin, vol. 41, March 1986, p. 22-24.

ABS: The growth of avionics is examined. The early development of avionics equipment (1910-1940) is discussed. The use of transistors, semiconductors, and solid-state circuits in airborne equipment is described. The advantages of digital technology as compared to analog technology are studied. The present utilization of electronics and computers in integrated avionics equipment is analyzed. The digitalization of information and data processing methods, decreases in avionics maintenance costs, improvements in communication systems, and an automatic data management concept are proposed for future avionics systems. 86/03/00   86A48141

UTTL: Avionics system architecture for Beechcraft Starship 1

AUTH: A/FENWICK, C. A.; B/SPENCER, J. L.
PAA: B/(Rockwell International Corp., Avionics Group, Cedar Rapids, IA)   SAE, Aerospace Technology Conference and Exposition, Long Beach, CA, Oct. 14-17, 1985. 19 p.

ABS: The architectural design of avionics to be used in the Beechcraft Starship 1 aircraft is presented. The avionics system consists of an integrated array of over 70 electronic line replaceable units organized to provide unprecedented levels of functional capability, fault tolerance, and reconfigurability. The digital data communication network connects all avionics and many nonavionics systems, including engine and fuel sensors, through the use of a dual-dual set of data concentrators. As compared with the current avionics equipment, the advanced avionics concept will result in weight and volume reductions, fewer wires and an integrated data network, simplified cockpit operation, and enhanced reliability.

RPT#: SAE PAPER 851851   85/10/00   86A38346

UTTL: The integrated digital avionics system for the F-20 Tigershark

AUTH: A/SMITH, W. L.
PAA: A/(Northrop Corp., Aircraft Div., Hawthorne, CA)   SAE, Aerospace Technology Conference and Exposition, Long Beach, CA, Oct. 14-17, 1985. 16 p.

ABS: Salient features of the avionics system to be discussed include: (1) a federated arrangement of the multiple processors in the system that maximizes mission reliability, (2) integration of the system using a MIL-STD-1553 multiplex data bus that minimizes the effects of changes and growth, and (3) a pilot-friendly cockpit using one head-up and two head-down displays and controls that allow combat with hands on stick and throttle.

RPT#: SAE PAPER 851850   85/10/00   86A38345

UTTL: Experience with flight testing integrated avionics systems

AUTH: A/PARLINI, F. E.; B/COMBLEY, J. R.
PAA: B/(Boeing Commercial Airplane Co., Seattle, WA)   AIAA, AHS, CASI, DGLR, IES, ISA, ITEA, SETP, and SFTE, Flight Testing Conference, 3rd, Las Vegas, NV, Apr. 2-4, 1986. 6 p.

ABS: The development and testing of the Boeing aircraft integrated digital Flight Management System (FMS) is discussed. On the 747, the INS was followed by the engine thrust limit computer, which led to the Performance Management System, which was then interfaced with the INS, autopilot and autothrottle to provide lateral/vertical navigation FMS capability. Digital avionics follow ARINC 700 and 429 specifications, and include the use of a Mark 33 Digital Information Transfer System. Integrated system testing involves both laboratory testing, including the employment of an engineering flight simulator, and inflight testing, including the use of the PADDS data

recording system for troubleshooting on customer aircraft. A flight test bus has also proved useful in providing outputs of internal signals, and a navigation performance program has been developed for data analysis purposes.
RPT#: AIAA PAPER 86-9804   86/04/00   86A32133

UTTL: The Crew Station Information Manager - An avionics expert system
AUTH: A/POHLMANN, L. D.; B/MARKS, P. S.; C/FEHLING, M. R.
PAA: A/(Boeing Military Airplane Co., Wichita, KS); C/(Advanced Information and Decision Systems, Mountain View, CA) IN: NAECON 1985; Proceedings of the National Aerospace and Electronics Conference, Dayton, OH, May 20-24, 1985. Volume 2 (A86-28326 12-04). New York, Institute of Electrical and Electronics Engineers, 1985. p. 1576-1582.
ABS: Progress to date is reported on the development of CSIM, the Crew Station Information Manager, which is a prototype avionics expert system being designed and developed under contract to the Air Force Avionics Laboratory. The function of CSIM is to manage the interface between the pilot and the avionics suite of a future tactical aircraft. A review is given of program motivation and objectives, and a set of avionics expert system candidates is identified. Preliminary development and early simulation are discussed, and near-term plans are outlined by DARPA.   85/00/00   86A28515

UTTL: Independent verification and validation (IV&V) of SMDPS (strategic mission data preparation system) software using microcomputer-based analysis tools
AUTH: A/SCHANE, R. N.; B/BENTLEY, M. R.; C/MACH, C. P.
PAA: C/(Logicon, Inc., Dayton, OH) IN: NAECON 1985; Proceedings of the National Aerospace and Electronics Conference, Dayton, OH, May 20-24, 1985. Volume 1 (A86-28326 12-04). New York, Institute of Electrical and Electronics Engineers, 1985, p. 834-842.
ABS: SMDPS is an interactive mission planning system that produces mission data on magnetic tape cartridges for strategic avionics systems. SMDPS employs a complex CODASYL-type DBMS (Data Base Management System) to manipulate a large data base of mission-related data. This paper discusses two major topics: (1) a general technical approach to software IV&V; and (2) development of an SMDPS requirements data base using a CODASYL DBMS on an IBM personal computer. One application of the requirements data base is to trace high-risk source requirements to the design and then to the code. This information is then used to select SMDPS subsystems and modules for in-depth analysis and testing.   85/00/00   86A28429

UTTL: Software management for integrated avionics systems
AUTH: A/EDWARDS, K. C.
PAA: A/(Harris Corp., Government Aerospace Systems Div., Melbourne, FL)   IN: NAECON 1985; Proceedings of the National Aerospace and Electronics Conference, Dayton, OH, May 20-24, 1985. Volume 1 (A86-28326 12-04). New York, Institute of Electrical and Electronics Engineers, 1985. p. 798-805.
ABS: Software engineering methodologies providing the level of control necessary to achieve cost effective software development for an integrated avionics software system are discussed with particular reference to the integrated software system for the Agusta A-129 Light Attack Helicopter. The high level of control is required to maintain the functional independence of the different software elements (e.g., Automated Flight Controls, Engine Monitor, Operator Interface, Navigation, and others) in order to avoid unnecessary couplings that will drive the cost of the system development and maintenance up while reducing the reliability of the system. As a result of an implementation of this approach, the software development costs for the Agusta A-129 project have been comparable to less complex ground-based systems.   85/00/00   86A28425

UTTL: Planning for cost/schedule control in a software development project
AUTH: A/MACK, O., JR.
PAA: A/(USAF, Aeronautical Systems Div., Wright-Patterson AFB, OH) IN: NAECON 1985; Proceedings of the National Aerospace and Electronics Conference, Dayton, OH, May 20-24, 1985. Volume 1 (A86-28326 12-04). New York, Institute of Electrical and Electronics Engineers, 1985, p. 778-785.
ABS: The transition from an estimate of the size, cost, and schedule of a software development project to a manageable contract is examined with reference to a case study of a hypothetical company's proposal. The discussion covers the implications of the Cost/Schedule Control Systems Criteria and the impacts of the Price-S and CDCDMO models. The processes presented here are based on the experiences of many current avionics software projects.   85/00/00   86A28422

UTTL: Designing into the unknown
AUTH: A/HACKMAN, M.
PAA: A/(Boeing Military Airplane Co., Seattle, WA)   IN: NAECON 1985; Proceedings of the National Aerospace and Electronics Conference, Dayton, OH, May 20-24, 1985. Volume 1 (A86-28326 12-04). New York, Institute of Electrical and Electronics Engineers, 1985, p. 764-769.
ABS: The concept of 'configurable software,' i.e., software generically constructed and table-driven to suit special contexts from tables read in at program initialization, is examined. Such an approach makes it possible to bypass many of the rigidities created by attempting to design

from top down and many of the uncertainties involved in designing from bottom up. General guidelines for developing configurable software are discussed with reference to an effort to design track-file software to form the core of a battle-management system intended for the next generation of tactical fighters.   85/00/00 86A28420

AUTH: A/ELLIS, J. R.
PAA: A/(Harris Corp., Government Aerospace Systems Div., Melbourne, FL) IN: NAECON 1985; Proceedings of the National Aerospace and Electronics Conference, Dayton, OH, May 20-24, 1985. Volume 1 (A86-28326 12-04). New York, Institute of Electrical and Electronics Engineers, 1985. p. 756-763.

UTTL: A new approach to ensuring deterministic processing in an integrated avionics software system

ABS: A software architecture providing mechanisms for highly reliable avionics systems which must integrate a variety of task execution rates and criticalities is presented. The executive supports single and multiprocessor systems for both cyclic and event driven tasks; foreground and background tasking with task suspension are implemented. Also covered are built-in-test features which ensure reliability. The architecture supports the use of high-level programming languages and is compatible with such advanced language features as 'tasking' and 'rendezvous' as defined for Ada. The architecture described here has been implemented in an integrated avionics software system for the Agusta A-129 Light Attack Helicopter.   85/00/00 86A28419

AUTH: A/ARCHER, H. S., III
PAA: A/(Lockheed-Georgia Co., Marietta) IN: NAECON 1985; Proceedings of the National Aerospace and Electronics Conference, Dayton, OH, May 20-24, 1985. Volume 1 (A86-28326 12-04). New York, Institute of Electrical and Electronics Engineers, 1985. p. 144-151.

UTTL: Digital avionics for modern aircraft - A case study into the problems and promise of aircraft electronics

ABS: With decreasing static stability margins and increasingly rigorous mission requirements, the role of aircraft electronics has become vital. A brief historical perspective of the evolution of avionics is given, and the harsh environmental constraints under which current avionics must operate are noted. The case for spatially redundant integrated racks composed of standardized modules is made. A state-of-the-art design approach is then presented, followed by a discussion of an advanced system that will be typical of aircraft in the 1990's. The topics discussed include: designing for thermal management, distributed processing, built-in-test requirements, and redundant systems for flight-critical applications.   85/00/00 86A28346

UTTL: Development testing of integrated avionics systems using dynamic environment simulation

AUTH: A/SHILLITO, P. L.
PAA: A/(Westland Helicopters, Ltd., Yeovil, England) Netherlands Association of Aeronautical Engineers and Technische Hogeschool te Delft, European Rotorcraft Forum, 10th, The Hague, Netherlands, Aug. 28-31, 1984, Paper. 21 p.

ABS: The evolution of sophisticated integrated avionics packages and their incorporation into civil and military helicopters has caused and will continue to cause significant perturbations to avionics and airframe manufacturers' integration plans. A coherent testing philosophy is discussed which co-ordinates avionic system design with its supporting development and validation tools. Highlighted is the use of inherent system performance data to maintain system integrity during validation of the developed system core whilst permitting full test automation. Facilities developed for dynamic simulation and performance assessment and presentation are described. These techniques are shown to reduce costs, improve effectiveness, and provide a high demonstrable level of confidence in the resultant integrated system.   84/08/00 86A26166

UTTL: MIL-STD-1553B 'networking' techniques for local area networks based on M.A.R.A. airborne computer

AUTH: A/DI GIOVANNI, A.; B/PADELLA, L.
PAA: B/(Selenia S.p.A., Pomezia, Italy) Netherlands Association of Aeronautical Engineers and Technische Hogeschool te Delft, European Rotorcraft Forum, 10th, The Hague, Netherlands, Aug. 28-31, 1984, Paper. 14 p.

ABS: The Time Sharing Multiplexed bus military standard MIL-STD-1553B stipulates an operating mode that compels such intelligent units as computer nodes (ultimately connected to the bus as remote terminals) to respect the communication sequence and timing embedded in the bus controller. The bus controller and remote terminal interface implementations presented allow any software application function to perform logical connections for data communication with any other software function of the network system, independently of physical allocation. An avionics software system may therefore be implemented without prior knowledge of hardware configuration. In addition, asynchronous software requests for the exchange of data between nodes may be generated in real time by the application programs embedded in the remote terminals.   84/08/00 86A26134

UTTL: Automated Electromagnetic Compatibility (EMC) testing of Naval aircraft and integrated avionics

AUTH: A/TETHER, D. E.

PAA: A/(Veda, Inc., Lexington Park, MD) IN: Society of Flight Test Engineers, Annual Symposium, 15th, St. Louis, MO, August 12-16, 1984, Proceedings (A86-23252 09-05). Lancaster, CA. Society of Flight Test Engineers, 1984, p. 2-1 to 2-7.

ABS: The Electromagnetic Compatibility (EMC) testing of Naval aircraft and their associated mission systems is characteristically as complex as the mission systems being tested and consumes most of the EMC engineers time performing repetitive observations. This paper discusses automation of the repetitive sequences of observations contained in Emission Control (EMCON) and Conducted Emission (CONEM) testing of Naval aircraft and their mission systems. The effects of automation on the continuing escalation in cost of EMC testing are addressed. 84/00/00 86A23254

UTTL: Implications of new aircraft avionics reliability performance

AUTH: A/MOORHEAD, M. E.

PAA: A/(Rockwell International Corp., Collins Div., Melbourne, FL) IN: Environmental integration technology today for a quality tomorrow; Proceedings of the Thirtieth Annual Technical Meeting, Orlando, FL, May 1-3, 1984 (A86-22176 08-38). Mount Prospect, IL, Institute of Environmental Sciences, 1984, p. 232-235.

ABS: An evaluation is made of MTBF values projected for avionics operating under environmental conditions typical of state-of-the-art 'digital' aircraft. Equipment reliability levels obtained during the fist year of operational experience with the B 767 and B 757 airliners are presented and compared with levels characteristic of previous avionic failure rates. Theoretical reliability levels based on existing MIL-HDBK-217 failure rate curves and acceleration factors are related to demonstrated performance levels, in order to identify points of correlation as well as discrepancies. 84/00/00 86A22178

UTTL: Helicopter cockpit avionics development or the advanced transport and next generation aircraft

AUTH: A/SCOUGHTON, C.; B/JOHNSTON, R.; C/COHEN, R.

PAA: C/(Bendix Aerospace Sector, Flight Systems Div., Teterboro, NJ) IN: Specialists' Meeting on Advanced Cockpit Design, Grapevine, TX, October 3, 4, 1984, Proceedings (A86-18451 06-06). Alexandria, VA, American Helicopter Society, 1984, 28 p.

ABS: Significant advances are currently being made in cockpit technology. One advanced cockpit design currently under development by Bendix provides a fully integrated avionics management approach for a new technology military transport helicopter. However, for next generation helicopters such as the Army's LHX, innovative new approaches are required to provide a tighter coupling

between pilot and aircraft. An overview of the avionics management system being developed for the Boeing-Vertol Model 360 is presented with emphasis on crew assist functions to reduce stress and increase overall crew efficiency. This is followed with a discussion of crew station drivers, requirements, and technologies for a next generation cockpit solution. 84/00/00 86A18461

UTTL: A reusable software system

AUTH: A/NISE, N.; B/DILLEHUNT, D.; C/GIFFIN, C.; D/MCKAY, C.; E/KIM, N.

PAA: C/(Rockwell International Corp., Downey, CA); D/(Houston, University, Clear Lake City, TX); E/(Rockwell International Corp., Anaheim, CA) IN: Computers in Aerospace Conference, 5th, Long Beach, CA, October 21-23, 1985, Technical Papers (A86-11401 02-59). New York, AIAA, 1985, p. 487-495.

ABS: Trends in the development of reusable software systems for avionics applications are considered. Emphasis is given to the growing need for lower maintenance costs which currently make up 50-80 percent of the total life-cycle cost of software engineering labor. Some recent breakthroughs in business practices and software design which have reduced maintenance costs are reviewed, including structured programming; Ada; and new software engineering tools such as DART and TESCT. A design strategy for the development of reusable software systems is described, and some areas of study that impinge on a reusable software system are discussed, including design, coding and documentation standards; training programs for software designers; and incentives for private industry to develop reusable software.

RPT#: AIAA PAPER 85-5067 85/00/00 86A11466

UTTL: Architectural concepts for integrated realtime multiprocessors

AUTH: A/GRUBE, G. W.; B/SCALES, W. W.; C/WILLIAMS, D. P.

PAA: C/(AT&T Bell Laboratories, Whippany, NJ) IN: Computers in Aerospace Conference, 5th, Long Beach, CA, October 21-23, 1985, Technical Papers (A86-11401 02-59). New York, AIAA, 1985, p. 474-477.

ABS: Methodology and architectural concepts at the system level for real-time microprocessor systems are examined with particular reference to multiprocessors for aerospace platforms. Data-flow multiprocessor architectures are illustrated by two examples, the Enhanced Modular Signal Processor and the AT&T Bell Laboratories Data Flow Architecture. It is pointed out that the system architecture is strongly influenced by such nonfunctional requirements as reliability, fault tolerance, programmability, and modularity.

RPT#: AIAA PAPER 85-5053 85/00/00 86A11464

AUTH: A/SCHMID, H.; B/GASKA, T. O.
PAA: B/(General Electric Co., Binghamton, NY) IN: Computers in Aerospace Conference, 5th, Long Beach, CA, October 21-23, 1985, Technical Papers (A86-11401 02-59). New York, AIAA, 1985, p. 185-194.

UTTL: Is there a risc in the avionic computer future?

ABS: It has been claimed that Reduced Instruction Set Computers (RISC) offer not only much smaller chips than Complex Instruction Set Computers (CISC) but also more throughput, shorter design time, better support for high level languages, and the ability to emulate other instruction sets. This paper examines these claims and evaluates the possibility of using RISC in real-time avionic control systems that are compatible with MIL-STD-01750. Several methods of translating from 1750 to RISC codes are discussed, and throughput predictions for a VHSIC MIPS are compared with those for the VHSIC 1750 CPUs.

RPT#: AIAA PAPER 85-5081 85/00/00 86A11427

UTTL: Development of complex avionics systems: Experience from French military programs

AUTH: A/COURSIMAULT, ANTOINE CORP: Service Techniques des Programmes Aeronautiques, Paris (France). In AGARD, The Design, Development and Testing of Complex Avionics Systems 15 p (SEE N88-23767 17-06)

ABS: The principles used to guide the development of the avionics systems for the Mirage 2000 and ATL2 aircraft are discussed. In addition, the anomalies discovered during these development experiences and improvements contributing to the current ACT program are examined. The various development, verification, and production stages are characterized. Emphasis is placed on the importance of requirements analysis and specifications development. 87/12/00 88N23804

UTTL: Workshops for the design of avionics systems and the development of onboard software

AUTH: A/SLISSA, MONIQUE; B/LAROCHE-LEVY, PHILIPPE CORP: Avions Marcel Dassault, Saint-Cloud (France). In AGARD, The Design, Development and Testing of Complex Avionics Systems 13 p (SEE N88-23767 17-06)

ABS: An avionics development system based on an open-ended suite of integrated computer tools is described. Additions and changes to the basic tool set can be implemented to suit the requirements of the specific project or individual contractor. A hosting structure tool that ties the whole system together is described along with three other basic tools: OCS, a system design aid; DLAO, a computer aided software definition tool; and SAO, a graphic, detailed specification aid. 87/12/00 88N23802

UTTL: Development of an airborne facility for advanced avionics research

AUTH: A/VANDRIEL, N. CORP: National Aerospace Lab., Amsterdam (Netherlands). In AGARD, The Design, Development and Testing of Complex Avionics Systems 14 p (SEE N88-23767 17-06)

ABS: Based on its flight test and simulator experience, the Netherlands National Aerospace Laboratory (NLR) has started the development of an airborne avionics research facility. In the framework of the Avionics Research Testbed (ART) project, NLR's Metro research airplane is being equipped with a number of advanced avionics systems, including programmable electronic flight instrument system, flight management computer, microwave landing system, global positioning system and secondary surveillance radar. 87/12/00 88N23801

UTTL: Avionics systems: Development method and computer tools

AUTH: A/LAROCHE-LEVY, P. CORP: Avions Marcel Dassault, Saint-Cloud (France). In AGARD, The Design, Development and Testing of Complex Avionics Systems 13 p (SEE N88-23767 17-06)

ABS: The avionics development system based on the use of computer automation tools is described. The problems posed in the development of avionics are defined and the various phases constituting the development life cycle for avionics systems are characterized. The capabilities of the computer tools and how they are used in various development stages are described. Finally, the benefits gained through the use of such a development methodology are discussed. 87/12/00 88N23799

UTTL: A look toward the future of complex avionics systems development using the USAF Test Pilot School's avionics systems test training aircraft

AUTH: A/BROOME, WILLIAM H., JR.; B/PARRAG, MIKE
PAA: B/(Calspan Corp., Buffalo, N.Y.) CORP: Air Force Test Pilot School, Edwards AFB, Calif. In AGARD, The Design, Development and Testing of Complex Avionics Systems 16 p (SEE N88-23767 17-06)

ABS: The Avionics Systems Test Training Aircraft (ASTTA) is a special configuration of the NC-131H Total In-flight Simulator (TIFS), and was developed to fill a significant gap in the education and experience of the avionics systems test community. It provides a cost effective means of quickly exposing both designers and testers to the key issues of systems development and in-flight testing, especially the operator to systems interface human factors issues. Its benign flight environment is conducive to both initial and advanced training in flight test techniques. 87/12/00 88N23797

UTTL: Testing complex avionics software: A practical experience

AUTH: A/MUENIER, M.   CORP: Electronique Serge Dassault, St.
Cloud (France).   In AGARD, The Design, Development and
Testing of Complex Avionics Systems 16 p (SEE N88-23767
17-06)

ABS: The methods used at Dassault for testing avionics
software, particularly the operational programs used on
the Mirage F1 and Mirage 2000 aircraft, are presented. The
MINERVE software development methodology is briefly
described. The objective of MINERVE is to facilitate
software production while ensuring control of quality,
costs and development delays. The various software tools
used to perform unitary, integration, functional and final
software validation and testing are described. These tools
provide hardware environment simulation, graphic output
and various operational modes. A trend towards the
integration of front-end tools (specification tools) and
back-end tools (test tools) is noted and a natural
progression from specification tool to semiformal
specification, to prototype, to test tools is
foreseen.   87/12/00 88N23794

AUTH: A/BEEVIS, D.   CORP: Defence and Civil Inst. of
Environmental Medicine, Downsview (Ontario).   In AGARD,
The Design, Development and Testing of Complex Avionics
Systems 9 p (SEE N88-23767 17-06)

ABS: Based on a review of human engineering activities in ten
major acquisition projects, some conclusions aimed at
facilitating the integration of human engineering
activities with the development of advanced avionics are
outlined. Conclusions are also drawn about the systems
design and human engineering processes, and the role that
mission, function, and task analysis can play in
integrating human engineering and systems development
activities. It is concluded that an approach that combines
the interaction of hardware, software, and human functions
is made especially necessary by the impact of advanced
technology on the roles of human operators and
maintainers, on the man-machine interface, and on the
system development process itself. Finally, it is argued
that there is a need to establish standardized approaches
to the application of human engineering in avionics system
design.   87/12/00 88N23793

AUTH: A/AUDONE, BRUNO   CORP: Aeritalia S.p.A., Caselle Torinese
(Italy).   In AGARD, The Design, Development and Testing
of Complex Avionics Systems 10 p (SEE N88-23767 17-06)

UTTL: The electromagnetic threat to future avionic systems

ABS: The electromagnetic threat to future aircraft is studied
and evaluated on the basis of the evolution of the avionic
systems. The high level of integration of these systems
combined with the increased number of electromagnetic
sources which may interfere with the performance of the

overall weapon system create the need to reexamine the
usual design and testing approach in order to reach an
adequate level of aircraft hardening. It is essential to
design and test at system level rather than at hardware
level. System design guidelines are discussed. Areas where
basic research studies need to be undertaken are
highlighted.   87/12/00 88N23790

AUTH: A/KOMOROS, GEORGE   CORP: Air Force Wright Aeronautical
Labs., Wright-Patterson AFB, Ohio.   In AGARD, The Design,
Development and Testing of Complex Avionics Systems 5 p
(SEE N88-23767 17-06)

UTTL: Design for interoperability (interchangeability)

ABS: Interoperability of the various elements used in a system
is the design property which allows the intermixing of
elements from various sources (manufactures) without any
impact on the performance of the system or the operational
hardware. Here, the line replaceable module approach is
discussed. This is a new approach to avionics where a
processor module is a 6 inch by 6 inch plug-in board with
processing power many times higher than that of older line
replaceable units.   87/12/00 88N23789

AUTH: A/BUUHERET, DANIEL; B/ROCH, JEAN LOUIS
PAA: B/(Crouzet Aerospace and Systems, Valence, France)
CORP: Societe Nationale Industrielle Aerospatiale,
Marignane (France).   In AGARD, The Design, Development
and Testing of Complex Avionics Systems 16 p (SEE
N88-23767 17-06)

UTTL: Design and development of an avionics system adapted
to helicopter missions

ABS: The development of the avionics system for the Dauphin 365
F configured for search and rescue operations is
discussed. A general description of flight missions and
system architecture is presented with particular emphasis
being given to the navigation and mission management
subsystems. The system is based on a Nadir Mk2
microprogrammable computer capable of an execution rate of
1 MOPS. The interactive relationship between the system
designers and integrators is described along with the
design support tools that aid in achieving successful
interaction.   87/12/00 88N23786

AUTH: A/SIMONS, PHILIP H.   CORP: Northrop Corp., Hawthorne.
Calif.   In AGARD, The Design, Development and Testing of
Complex Avionics Systems 8 p (SEE N88-23767 17-06)

UTTL: Managing advanced avionic system design

ABS: The critical requirements of advanced avionic system
design are overall system requirements, configuration item
requirements, interface requirements, and design process
management requirements. These requirements can be met
through the design process of abstract requirements
definition, requirements/functional decomposition,

functional recomposition, and detailed interface definition. This process can be greatly aided by computer automation, resulting in the design of more complex avionics systems in far less time than would be possible using older tools.   87/12/00   88N23783

UTTL: Crewstation Information and Development System (CIDS)

AUTH: A/ROWLAND, M. E.; B/WAGONER, W. R.   CORP: Boeing Military Airplane Development, Wichita, Kans.   In AGARD. The Design, Development and Testing of Complex Avionics Systems 11 p (SEE N88-23767 17-06)

ABS: The process by which requirements for an avionics system are translated into an integrated crewstation design is discussed. The Crewstation Information Development System (CIDS) was divided into three phases of activity. In Phase 1, methodology development, a comprehensive set of requirements, resource allocation, and an information utilization assessment is derived. Phase 2, crewstation development, focuses on deriving the most effective methods of utilizing required crewstation information, taking into consideration the impact of the operational environment. The final phase, design application, concerns the details of crewstation design and the development of a crewstation information manager.   87/12/00   88N23781

UTTL: Development and testing of a predictive methodology for optimization of man-machine interface in future avionics systems

AUTH: A/PARKS, ROGER E.   CORP: Textron Bell Helicopter, Fort Worth, Tex.   In AGARD. The Design, Development and Testing of Complex Avionics Systems 9 p (SEE N88-23767 17-06)

ABS: The trend toward increasing complexity and cost in emerging avionics systems, driven by requirements for increased functional capability, has created a need for a predictive analytical methodology which accurately forecasts system performance early in the design process, and treats the human operator and the equipment as a fully integrated man-machine system. A methodology that meets these needs has been developed and validated by Bell Helicopter Textron. The process is being used to provide early, accurate avionics system characterization, thereby reducing design costs.   87/12/00   88N23780

UTTL: The avionics software architecture impact on system architecture

AUTH: A/LOCKE, C. DOUGLASS   CORP: International Business Machines Corp., Owego, N.Y.   In AGARD. The Design, Development and Testing of Complex Avionics Systems 5 p (SEE N88-23767 17-06)

ABS: Technology developments that led to the problem of system vulnerability to software errors resulting from increased

avionic system complexity and the impact of these developments on the functionality and design of new systems are considered. The current sequence for performing the physical and software architectural design, including a definition of the software architecture is discussed. The likely consequences of using these methods for designing new avionics systems are also discussed.   87/12/00   88N23778

UTTL: Modeling of functional specifications for onboard software: Experience with the Rafale avionics system

AUTH: A/SCHIRLE, PATRICK   CORP: Avions Marcel Dassault, Saint-Cloud (France).   In AGARD, The Design, Development and Testing of Complex Avionics Systems 12 p (SEE N88-23767 17-06)

ABS: The avionics system of the Rafale aircraft entailed innovations such as the integration of aircraft systems and the acquisition of control information. These new functions resulted in a notable increase in, and qualitative evolution of, onboard computer software. For the improvement of quality and response characteristics of software, development was supported by an integral methodology for modeling software functional specifications. The model was constructed around a SEL computer supporting a FORTRAN listing of 3500 pages of system functional specifications. The model allows the direct validation of each individual specification and the whole ensemble of specifications. The model was used to support the analysis of the system failures, provide a base for the assessment of modifications, generate acceptance rules, and prepare a functional reference for integration.   87/12/00   88N23777

UTTL: Test philosophy of the EH101 integrated avionic system

AUTH: A/GALLI, E.   CORP: Agusta Sistemi S.p.A., Varese (Italy). In AGARD, The Design, Development and Testing of Complex Avionics Systems 5 p (SEE N88-23767 17-06)

ABS: The philosophy employed during the development and testing of the EH101 integrated avionic Naval helicopter is outlined. The avionics architecture is given in graphic form. Overall avionic integration, the aircraft management computer test, the sensor interface units test, and the aircraft management system test are described.   87/12/00   88N23775

UTTL: Development of a generic architecture

AUTH: A/BERGGREN, CHRISTINA   CORP: International Business Machines Corp., Owego, N.Y.   In AGARD, The Design, Development and Testing of Complex Avionics Systems 5 p (SEE N88-23767 17-06)

ABS: A new generation systems architecture is designed to bridge the gap between today's 1553-based systems and fault-tolerant, totally integrated systems of tomorrow.

the decision to specify a two-machine system, including an emulation computer connected to a general-purpose computer. There is also an evaluation of potential computers to serve as the emulation computer. 81/02/00 88N23472

RPT#: NASA-CR-181641 NAS 1.26:181641 81/02/00

UTTL: The use of rule induction to assist in the diagnosis of avionic circuit board defects

AUTH: A/SADLER, G. B. CORP: Strathclyde Univ., Glasgow (Scotland).

ABS: An expert system to assist in the diagnosis of avionic circuit board faults was developed using the rule induction package Intelligence1. The initial attempt at building an expert system failed but when the level of detail of attributes was altered an expert system was successfully built. The method was proved by building an expert system for a second circuit board using the same approach and by reproducing the same rules for the first board using a different rule induction package, IRIS. The expert system built for the first board was evaluated for accuracy by interrogation using data from additional historical examples and for worth by monitored trials. The former show that the expert system is accurate but not complete and the latter is inconclusive.

RPT#: ETN-88-92077 87/00/00 88N22899

UTTL: Rapid prototyping of complex avionics system architectures

AUTH: A/BERARDI, L.; B/GIORGI, N.; C/MELLAND, W.; D/VALANTE, A.; E/ZUCCO, E. CORP: Aeritalia S.p.A., Turin (Italy).

ABS: The Expert Consultant for Avionics System Transformation Exploitation was developed for rapidly prototyping different alternatives, and to establish the information flow architecture of the avionics system. The tool provides the user with an interface to assist in describing the avionics from the point of view of the data handling, and presents the results in a suitable format; it performs consistency checks and advises the user on possible architectural problems by means of the expert system techniques. The development environment of the tool and how it works in a consulting session are described.

RPT#: ETN-88-92275 87/00/00 88N22898

UTTL: Architecture specification for PAVE PILLAR avionics

AUTH: A/OSTGAARD, JOHN; B/SZKODY, RON; C/BENNING, STEVE; D/PURDY, KARL; E/MAUERSBERGER, GARY CORP: Air Force Wright Aeronautical Labs., Wright-Patterson AFB, Ohio.

ABS: This document establishes the functional requirements for the PAVE PILLAR Systems Architecture. This architecture is specifically targeted for advanced tactical fighters, and in general for all military aircraft applications. The PAVE PILLAR Architecture addresses those functions which could be implemented with common hardware and computer

---

Described here is a novel approach to system functional area partitioning and the design of this generic distributed real time architecture. The architecture incorporates new military standards in development. 87/12/00 88N23774

UTTL: Rapid prototyping of complex avionic system architectures

AUTH: A/BERARDI, L.; B/GIORGI, N.; C/MELLAND, W.; D/VALANTE, A.; E/ZUCCO, E. CORP: Aeritalia S.p.A., Caselle Tortnese (Italy). In AGARD, The Design, Development and Testing of Complex Avionics Systems 13 p. (SEE N88-23767 17-06) Sponsored in part by Unisys, Italy

ABS: A design tool called Expert Consultant for Avionics system Transformation Exploration (ECATE) is described. ECATE, rapidly prototyping different alternatives, helps the designer in establishing the information flow architecture of the avionics system, that is, the organization of the internal data handling. The tool provides the user with an interface to assist him in describing the avionics from the point of view of the data handling and presents the results in a suitable format. It performs consistency checks and advises the user on possible architectural problems by means of expert system techniques. 87/12/00 88N23771

UTTL: The Design, Development and Testing of Complex Avionics Systems CORP: Advisory Group for Aerospace Research and Development, Neuilly-Sur-Seine (France).

ANN: The design, development and testing of complex avionics systems are discussed. Design aspects for future avionics systems, managing the future system design process, and system design tools and integration are among the topics covered. For individual titles, see N88-23768 through N88-23804.

RPT#: AGARD-CP-417 ISBN-92-835-0437-2 AD-A198666 87/12/00 88N23767

UTTL: Digital avionics design and reliability analyzer

CORP: Martin Marietta Corp., Denver, Colo.

ABS: The description and specifications for a digital avionics design and reliability analyzer are given. Its basic function is to provide for the simulation and emulation of the various fault-tolerant digital avionic computer designs that are developed. ...as been established that hardware emulation at the g... level will be utilized. The primary benefit of emulation t... reliability analysis is the fact that it provides t... pability to model a system at a very detailed level. E ...ion allows the direct insertion of faults into th... ystem, rather than waiting for actual hardware failures to occur. This allows for controlled and accelerated testing of system reaction to hardware failures. There is a trade study which leads to

programs to allow adaptation to either air-to-air or air-to-ground missions. This document defines the major elements of the PAVE PILLAR Architecture, the mechanizations for inter-connecting those elements, a set of common modules from which those elements could be constructed, and the operation of the network constructed of these elements. The PAVE PILLAR core avionics exploits the commonality in air-to-air and air-to-ground missions. The PAVE PILLAR core avionics consists of the following functional areas: (1) Digital Signal Processing. (2) Mission Processing. (3) Vehicle Management Processing, and (4) Avionics Systems Control.

RPT#: AD-A188722 AFWAL-TR-87-1114 87/01/00 88N20303

UTTL: Reducing time required for the development of avionic systems
AUTH: A/SCARFE, W. R.; B/GRANTHAM, K. CORP: British Aerospace Public Ltd. Co., Lancashire (England). In AGARD, Flight Vehicle Development Time and Cost Reduction 12 p (SEE N88-20173 12-81)
ABS: The current rapid increase in avionic complexity causes difficulties in both defining and implementing the system requirement. Errors in the requirement and in its implementation lead inevitably to an extension in the time necessary for system development and consequently to an increase in the cost of the developed system. This paper describes the avionic system onboard the Experimental Aircraft Programme (EAP) and the design process used on that system. In particular, the effectiveness of: (1) rapid prototyping as a means to arrive at a sensible requirement, and (2) implementation via the use of Semi-Automated Functional Requirement Analysis (SAFRA), is outlined. A resume of the results from the EAP is used to show that this structured approach to design using both rapid prototyping and SAFRA does reduce the time required for avionic system development. 87/09/00 88N20185

AUTH: A/LONGBRAKE, RONALD B. CORP: Aeronautical Systems Div., Wright-Patterson AFB, Ohio. In AGARD, Flight Vehicle Development Time and Cost Reduction 11 p (SEE N88-20173 12-81)
UTTL: Avionics acquisition, trends and future approaches
ABS: The current and future direction of the U.S. Air Force avionics is discussed. While the paper discusses primarily tactical aircraft avionics, the findings and conclusions are applicable across USAF systems. The paper covers the acquisition methodology, the background and trends of avionics and future approaches. The basic influences are operational needs, availability, survivability, available technology, cost and schedules. The challenge is to provide effective avionics in a budget constrained world. To accomplish this requires emphasis on providing performance to counter the threat, flexibility for diverse use and basing, cost and schedule realism, and systems

capable of being upgraded through planned growth as the threat changes. It has been shown that the final 5 to 10 percent improvements in performance can increase the cost 20 to 50 percent, therefore, 'sufficient' and not 'best' performance should be the goal. While initial acquisition cost is of concern, life cycle cost is even more important. To keep life cycle costs down and have an effective system during combat, maintenance concepts need serious attention. To accomplish these objectives, the discrete avionics systems of the past must be replaced with integrated avionics responsive to crew needs, increasing threats and fiscal constraints. Future needs will cause continued increases in avionics cost. The use of new technologies, new avionics system integration and architecture techniques, use of common hardware, modular and reusable software and improving the environment in which the avionics must operate, can control the life cycle cost of avionics while meeting needs of future systems. 87/09/00 88N20184

UTTL: Computer architecture for efficient algorithmic executions in real-time systems: New technology for avionics systems and advanced space vehicles
AUTH: A/CARROLL, CHESTER C.; B/YOUNGBLOOD, JOHN N.; C/SAHA, AINDAM CORP: Alabama Univ., Tuscaloosa.
ABS: Improvements and advances in the development of computer architecture now provide innovative technology for the recasting of traditional sequential solutions into high-performance, low-cost, parallel systems to increase system performance. Research conducted in development of specialized computer architecture for the algorithmic execution of an avionics system, guidance and control problem in real time is described. A comprehensive treatment of both the hardware and software structures of a customized computer which performs real-time computation of guidance commands with updated estimates of target motion and time-to-go is presented. An optimal, real-time allocation algorithm was developed which maps the algorithmic tasks onto the processing elements. This allocation is based on the critical path analysis. The final stage is the design and development of the hardware structures suitable for the efficient execution of the allocated task graph. The processing element is designed for rapid execution of the allocated tasks. Fault tolerance is a key feature of the overall architecture. Parallel numerical integration techniques, tasks definitions, and allocation algorithms are discussed. The parallel implementation is analytically verified and the experimental results are presented. The design of the data-driven computer architecture, customized for the execution of the particular algorithm, is discussed.
RPT#: NASA-CR-182650 NAS 1.26:182650 BER-410-17 87/12/00 88N20016

UTTL: Columbus feasibility studies. Volume 3: Avionics systems   CORP: Messerschmitt-Boelkow-Blohm G.m.b.H., Bremen (West Germany).; Aeritalia S.p.A., Turin (Italy).; Dornier-Werke G.m.b.H., Friedrichshafen (West Germany). Sponsored by Bundesministerium fuer Forschung und Technologie/DFVLR and the PCN/WRST Prepared in cooperation with Aeritalia SpA, Turin, Italy and Dornier-Werke GmbH, Friedrichshafen, West Germany

ABS: The Columbus information management, electrical power supply and distribution, telecommunication, guidance and navigation, attitude control, propulsion; and rendezvous and docking systems are discussed.

RPT#: ETN-88-91075   86/00/00   88N16801

UTTL: Helicopters as test carriers for avionics systems (HETAS)

AUTH: A/BENDER, K.; B/DANNEBERG, E.; C/DIERKE, R.; D/HAEHNLEIN, G.; E/MANSFELD, G.; F/TERSTEEGEN, J. CORP: Deutsche Forschungs- und Versuchsanstalt fuer Luft- und Raumfahrt, Brunswick (West Germany).

ABS: The HETAS integrated test bed for avionics systems in human engineering, digital control engineering, and sensors is presented. The flight test system, including helicopter test carrier, systems concept, and integrated flight test system, is described. The ground test system, including helicopter simulation, cockpit, and redundant fly-by-wire system are described.

RPT#: DFVLR-IB-112-85/18 LVL-8302-I-2 ETN-88-91013   85/10/09   88N16692

UTTL: Digital system bus integrity

AUTH: A/ELDREDGE, DONALD; B/HITT, ELLIS F.   CORP: Battelle Columbus Labs., Ohio.; Lockheed-Georgia Co., Marietta. Prepared in cooperation with Lockheed-Georgia Co., Marietta

ABS: This report summarizes and describes the results of a study of current or emerging multiplex data buses as applicable to digital flight systems, particularly with regard to civil aircraft. Technology for pre-1995 and post-1995 timeframes has been delineated and critiqued relative to the requirements envisioned for those periods. The primary emphasis has been an assured airworthiness of the more prevalent type buses, with attention to attributes such as fault tolerance, environmental susceptibility, and problems under continuing investigation. Additionally, the capacity to certify systems relying on such buses has been addressed.

RPT#: NASA-CR-181446 NAS 1.26:181446 DOT/FAA/CT-86/44 AD-A189964 87/03/00   88N10030

UTTL: The development process for the space shuttle primary avionics software system

AUTH: A/KELLER, T. W.   CORP: IBM Federal Systems Div., Houston, Texas.

ABS: Primary avionics software system; software development approach; user support and problem diagnosis; software releases and configuration; quality/productivity programs; and software development/production facilities are addressed. Also examined are the external evaluations of the IBM process.

RPT#: NASA-CR-180425 NAS 1.26:180425   87/00/00   87N29530

UTTL: A flight evaluation of voice interaction as a component of an integrated helicopter avionics system

AUTH: A/LITTLE, R.; B/COWAN, R.   CORP: Royal Aircraft Establishment, Farnborough (England).

ABS: A Wessex helicopter was used to develop and evaluate an integrated avionics system which incorporates advanced displays and a flight management system for military and civil applications. The system has automatic speech recognition and synthetic speech output. Flight tests were conducted to establish guidelines for the integration of these devices with advanced avionics such as color displays, digital maps, and touch overlays. The tests show that data input and retrieval can be achieved quickly, simply, and easily.

RPT#: RAE-TM-FS(B)-637 BR100581 ETN-87-90266 AD-A187074 86/04/00   87N28547

UTTL: Systems Testbed for Avionic Research (STAR) automation concept development

AUTH: A/STRINGER, PAUL G.; B/RILEY, DANIEL D.   CORP: Essex Corp., Warminster, Pa.

ABS: The primary purpose of the work performed under this contract has been to develop an automation and avionics integration design concept permitting single-pilot operations. Recommendations for that design concept are provided in this report. These recommendations were derived through analyses of the U.S. Army Scout and Utility mission concepts. Identification of suitable automation-candidate functions for the performance of those missions, assessment of automation technologies available, and, finally, the establishment of system design option preferences. Ultimately, these recommendations will impact the definition of design specification requirements for the NUH-60 Systems Testbed for Avionics Research (STAR) aircraft.

RPT#: AD-A180658 AVSCOM-TR-86-E-2   87/04/00   87N28907

UTTL: Avionics systems for future civil helicopters

AUTH: A/INGELSPERGER, ANTON   CORP: Messerschmitt-Boelkow-Blohm G.m.b.H., Ottobrunn (West Germany). In its Research and Development: Technical-Scientific Publications 1986 p 167-174 (SEE N87-26826 20-99)

ABS: Civil helicopter cockpit instrumentation using color displays, and the testing of color displays are explained.

The central control unit principles are discussed. The main fields in cockpit instrumentation and central control unit which need further development for use in future helicopters are discussed. The use of development simulators, signal standardization and bus system, and software for display systems are presented.

RPT#: MBB-UD-473/86 86/00/00 87N26838

UTTL: An integrated aircraft navigation and display system utilising an on-board composite data base

AUTH: A/BUSBRIDGE, M. L.; B/PULESTON, D. J. CORP: Marconi Avionics Ltd., Rochester (England). In AGARD Advanced Computer Aids in the Planning and Execution of Air Warfare and Ground Strike Operations 4 p (SEE N87-24940 18-66)

ABS: It is likely that future aircraft will contain a central data base containing feature details, ground elevation, low flying obstruction data, intelligence, and mission data. Recent advances in technology have made it possible to implement such a data base within an aircraft. The existence of a central data base facilitates many functions for both low level ground attack and air defense aircraft. These functions include: improved moving map displays, precision navigation systems, covert radar shadowing, surface to air missile site intervisibility, and perspective terrain displays for use in adverse visibility conditions. The integration of such a system is currently being finalized and forms part of an ongoing flight trials program in fixed wing jet aircraft.
87/02/00 87N24955

UTTL: New technology impacts on future avionics architectures

AUTH: A/MEUZAK, RICHARD S. CORP: Naval Air Development Center. Warminster, Pa. In AGARD Advanced Computer Aids in the Planning and Execution of Air Warfare and Ground Strike Operations 7 p (SEE N87-24940 18-66)

ABS: An interpretation of avionics architecture is provided with respect to system components, organization, and design factors. Initially, general avionics architecture characteristics are addressed followed by discussions on emerging technologies and their impact on advanced systems. Information handling requirements are projected for future tactical aircraft. In addition, advanced avionics architecture design consideration and technical issues are addressed relative to achieving improved performance, reliability, survivability, flexibility, and low life cycle cost. 87/02/00 87N24949

UTTL: The impact of future avionics technology on the conduct of air warfare

AUTH: A/DASARO, JOSEPH A. CORP: Army Avionics Research and Development Activity, Fort Monmouth, N. J. In AGARD Improvement of Combat Performance for Existing and Future

---

Aircraft 9 p (SEE N87-22663 16-05)

ABS: A synopsis is given of the conclusions reached by the Systems Subpanel of the NATO AGARD workshop on the potential impact of development in electronic technology on the future conduct of air warfare. Avionic system integration technology, system architecture, data processing, data communication paths, computer programs, fault detection and isolation, and system design methodology are among the topics discussed. 86/12/00 87N22673

UTTL: Handbook. Volume 1: Validation of digital systems in avionics and flight control applications

AUTH: A/HITT, ELLIS F.; B/ELDREDGE, DONALD; C/WEBB, JEFF; D/LUCIUS, CHARLES; E/BRIDGMAN, MICHAEL S. CORP: Battelle Columbus Labs., Ohio. Sponsored by FAA

ABS: Techniques, methodologies, tools, and procedures are identified in a systems context that is applicable to aspects of the validation and certification of digital systems at specific times in the development, and implementation of software based digital systems to be used in flight control/avionics applications. The application of these techniques in the development of discrete units and/or systems will result in completion of a product or system which is verifiable and can be validated in the context of the existing regulations/orders of the government regulatory agencies. A systems engineering approach is used to implement and test the software and hardware during the design, development, and implementation phase. The handbook also recognizes and provides for the evaluation of the pilot workload in the utilization of the new control/display technology, especially when crew recognition and intervention may be necessary to cope with/recover from the effects of faults or failures in the digital systems or the crew introduces errors into the system under periods of high workload due to some inadvertent procedure or entry of incorrect or erroneous data.

RPT#: DOT/FAA/CT-82/115-VOL-1-REV 86/00/00 87N15204

UTTL: Integrated communication, navigation, and identification avionics resource allocation

AUTH: A/GRIFFITHS, B. E.; B/MILLER, D. E. CORP: Analytic Sciences Corp. Reading, Mass.

ABS: The Integrated Communication, Navigation, and Identification Avionics (ICNIA) architecture is being designed to replace a number of discrete avionics components with an integrated, modular system. In order to maximize the usefulness of ICNIA-supported functions, a new resource allocation technique is needed. This report presents a preliminary assessment of several methods of reallocation and describes measures of performance for all of these techniques.

RPT#: AD-A170357 AFHRL-TR-86-10 86/07/00 87N13439

UTTL: Investigation of an advanced fault tolerant integrated avionics system
AUTH: A/DUNN, W. R.; B/COTTRELL, D.; C/FLANDERS, J.; D/JAVORNIK, A.; E/RUSOVICK, M. CORP: University of Southern Colorado, Pueblo.
ABS: Presented is an advanced, fault-tolerant multiprocessor avionics architecture as could be employed in an advanced rotorcraft such as LHX. The processor structure is designed to interface with existing digital avionics systems and concepts including the Army Digital Avionics System (ADAS) cockpit/display system, navaid and communications suites, integrated sensing suite, and the Advanced Digital Optical Control System (ADOCS). The report defines mission, maintenance and safety-of-flight reliability goals as might be expected for an operational LHX aircraft. Based on use of a modular, compact (16-bit) microprocessor card family, results of a preliminary study examining simplex, dual and standby-sparing architectures is presented. Given the stated constraints, it is shown that the dual architecture is best suited to meet reliability goals with minimum hardware and software overhead. The report presents hardware and software design considerations for realizing the architecture including redundancy management requirements and techniques as well as verification and validation needs and methods.
RPT#: NASA-CR-176980 NAS 1.26:176980 86/03/00 86N28945

UTTL: Design considerations for flight test of a fault inferring nonlinear detection system algorithm for avionics sensors
AUTH: A/CAGLAYAN, A. K.; B/GODIWALA, P. M.; C/MORRELL, F. R.
PAA: A/(Charles River Analytics, Inc., Cambridge, Mass.)
CORP: National Aeronautics and Space Administration. Langley Research Center, Hampton, Va.
ABS: The modifications to the design of a fault inferring nonlinear detection system (FINDS) algorithm to accommodate flight computer constraints and the resulting impact on the algorithm performance are summarized. An overview of the flight data-driven FINDS algorithm is presented. This is followed by a brief analysis of the effects of modifications to the algorithm on program size and execution speed. Significant improvements in estimation performance for the aircraft states and normal operating sensor biases, which have resulted from improved noise design parameters and a new steady-state wind model, are documented. The aircraft state and sensor bias estimation performances of the algorithm's extended Kalman filter are presented as a function of update frequency of the piecewise constant filter gains. The results of a new detection system strategy and failure detection performance, as a function of gain update frequency, are also presented.
RPT#: NASA-TM-88998 NAS 1.15:88998 86/08/00 86N28933

UTTL: The impact of VHPIC on avionic system architecture, packaging and maintainability.
AUTH: A/SALMON, L. T. J.; B/OLDFIELD, D. E. CORP: Royal Aircraft Establishment, Farnborough (England). In AGARD The Impact of Very High Performance Integrated Circuits on Radar, Guidance and Avionics Systems 5 p (SEE N86-28337 19-33)
ABS: The advent of high performance integrated circuit technology will have a far reaching effect on avionic systems. Currently, the impact is being mainly felt at a high bandwidth, signal processing level where the benefits are self-evident, however, unless the effect of the technology is considered across the whole of the avionic system then there is little doubt that much of the improved mission effectiveness promised by the new, high bandwidth systems will not accrue. A wider, system level approach is therefore needed which not only encompasses the technical aspects of individual sensors but also considers the remainder of the airborne data processing system and more traditional topics such as maintainability, packaging and architecture. This paper highlights some of the areas in which very high performance integrated circuits (VHPIC) could have a major impact on avionics systems and indicates some of the factors that will need to be considered during the design of future systems. 85/08/00 86N28372

UTTL: The Impact of Very High Performance Integrated Circuits on Radar, Guidance and Avionics Systems CORP: Advisory Group for Aerospace Research and Development. Neuilly-Sur-Seine (France).
ANN: Advances with silicon integrated circuit technology have demonstrated the feasibility of very large scale integration with gate densities of 10 to the 5th power/sq cm and with functional throughput rates in excess of 10 to the 12th power gate Hz/sq cm. These advances offer the prospect of compact, low power consumption, high throughput processors in a wide variety of roles throughout military electronic systems. This symposium aimed to involve integrated circuit experts who reviewed their current and projected capabilities, circuit and subsystems designers who are exploiting the technology in implementing sophisted processing and data manipulation techniques and who reported on their progress, and system designers who described applications addressed by the advancing technology. For individual titles see N86-28338 through N86-28374.
RPT#: AGARD-CP-380 ISBN-92-835-0379-1 AD-A163151 85/08/00 86N28337

UTTL: Development experience with a simple expert system demonstrator for pilot emergency procedures

AUTH: A/VANNORMAN, M.; B/MACKALL, D. A.   CORP: National
Aeronautics and Space Administration. Ames Research
Center, Moffett Field, Calif.
ABS: Expert system techniques, a major application area of
artificial intelligence (AI), are examined in the
development of pilot associate to handle aircraft
emergency procedures. The term pilot associate is used to
describe research involving expert systems that can assist
the pilot in the cockpit. The development of expert
systems for the electrical system and flight control
system emergency procedures are discussed. A simple.
high-level expert system provides the means to choose
which knowledge domain is needed. The expert systems were
developed on a low-cost. FORTH-based package, using a
personal computer.
RPT#: NASA-TM-85919 H-1272 NAS 1.15:85919   86/02/00   86N23603

UTTL: Logistics engineering analysis techniques for
fault-tolerant avionics systems
AUTH: A/MCMANUS, J. C.   CORP: Air Force Human Resources Lab.,
Brooks AFB, Tex.
ABS: A technique which performs reliability, supportability,
and survivability (RSS) analysis of fault-tolerant.
dynamically reconfigurable systems during early design is
presented. Implemented in the Mission REliability Model
(MIREM) computer program, this method analyzes the
structure of functional components in a system. Use of
MIREM will allow design engineers to apply RSS analysis
before the design is fixed.
RPT#: AD-A161981 AFHRL-TR-84-60   85/11/00   86N23596

AUTH: UTTL: Dependable avionic data transmission
AUTH: A/POWELL, D. R.; B/VALADIER, J. C.   CORP: Laboratoire
d'Automatique et d'Analyse des Systemes, Toulouse
(France).; Crouzet Aerospace and Systems, Valence
(France).; Electronique Serge Dassault, St. Cloud
(France).   In AGARD Fault Tolerant Hardware/Software
Architecture 19 p (SEE N86-20399 11-08) Sponsored in
part by Direction des Recherches, Etudes et Techniques.
Prepared in cooperation with Crouzet Aerospace and
Systems, Valence, France, and Electronique Serge Dassault,
St. Cloud, France
ABS: This paper outlines the major constraints imposed on the
design of dependable local area networks for avionic
systems and underlines the essential differences in
requirements that exist with respect to those of
ground-based (civil) LANs. The different choices available
to the system designer are then discussed: technology
(electrical or optical), architecture (bus or loop).
general philosophy (centralized or decentralized). medium
access control (competition or consultation). The paper
then goes on to summarize two different and independent
avionic LAN research projects: one which focusses on fault
and damage-tolerance and the other on high-speed.

85/09/00   86N20405

UTTL: Description of an experimental expert system flight
status monitor
AUTH: A/DUKE, E. L.; B/REGENIE, V. A.   CORP: National
Aeronautics and Space Administration. Ames Research
Center, Moffett Field, Calif.
ABS: This paper describes an experimental version of an expert
sy tem flight status monitor being developed at the Dryden
Flight Research Facility of the NASA Ames Research Center.
This experimental expert system flight status monitor
(ESSFSM) is supported by a specialized knowledge
acquisition tool that provides the user with a powerful
and easy-to-use documentation and rule construction tool.
The EESFSM is designed to be a testbed for concepts in
rules. inference mechanisms, and knowledge structures to
be used in a real-time expert system flight status monitor
that will monitor the health and status of the flight
control system of state-of-the-art. high-performance.
research aircraft.
RPT#: NASA-TM-86791 H-1317 NAS 1.15:86791 AIAA-85-6042-CP
85/10/00   86N11195

REPORT DOCUMENTATION PAGE

1. Recipient's Reference	2. Originator's Reference	3. Further Reference	4. Security Classification of Document
	AGARD-LS-164	ISBN 92-835-0513-1	UNCLASSIFIED

5. Originator	Advisory Group for Aerospace Research and Development North Atlantic Treaty Organization 7 rue Ancelle, 92200 Neuilly sur Seine, France

6. Title	SYSTEMS ENGINEERING

7. Presented on	15—16 May 1989 in Kettering, OH, USA, on 22—23 May 1989 in Delft, The Netherlands and 25—26 May 1989 in Pratica di Mare (Rome), Italy.

8. Author(s)/Editor(s)	9. Date
Various	May 1989

10. Author's/Editor's Address	11. Pages
Various	134

12. Distribution Statement	This document is distributed in accordance with AGARD policies and regulations, which are outlined on the Outside Back Covers of all AGARD publications.

13. Keywords/Descriptors

Avionics	Design
Systems engineering	Requirements
Prototypes	

14. Abstract

Recent AGARD activities have indicated a strong need for more effective avionics system engineering. There is a growing need for reducing development time, effecting savings in costs of ownership, and in extending the life-time of avionics systems. This must be accomplished along with meeting needs of the user faced with a growing threat. With the growing complexity of avoinics systems (as well as other systems), it is important to develop and maintain expertise in system planning, architecture and management.

This Lecture Series addresses the important systems engineering aspects of Requirements, System Integration, Prototyping, and Design. In addition, the impact of technology on system architecture will be discussed. Methodologies will be described and actual case histories will serve as practical examples of modern system engineering.

This Lecture Series, sponsored by the Avionics Panel of AGARD, has been implemented by the Consultant and Exchange Programme.

AGARD-LS-164
AGARD Lecture Series No.164 Advisory Group for Aerospace Research and Development, NATO SYSTEMS ENGINEERING Published May 1989 134 pages Recent AGARD activities have indicated a strong need for more efficient avionics system engineering. There is a growing need for reducing development time, effecting savings in costs of ownership, and in extending the life-time of avionics systems. This must be accomplished along with meeting needs of the user faced with a growing threat. With the growing complexity of avionics systems (as well as other systems), it is important to develop and maintain expertise in system planning, architecture, and management. P.T.O

Avionics
Systems engineering
Prototypes
Design
Requirements

AGARD-LS-164
AGARD Lecture Series No.164 Advisory Group for Aerospace Research and Development, NATO SYSTEMS ENGINEERING Published May 1989 134 pages Recent AGARD activities have indicated a strong need for more efficient avionics system engineering. There is a growing need for reducing development time, effecting savings in costs of ownership, and in extending the life-time of avionics systems. This must be accomplished along with meeting needs of the user faced with a growing threat. With the growing complexity of avionics systems (as well as other systems), it is important to develop and maintain expertise in system planning, architecture, and management. P.T.O

Avionics
Systems engineering
Prototypes
Design
Requirements

AGARD-LS-164
AGARD Lecture Series No.164 Advisory Group for Aerospace Research and Development, NATO SYSTEMS ENGINEERING Published May 1989 134 pages Recent AGARD activities have indicated a strong need for more efficient avionics system engineering. There is a growing need for reducing development time, effecting savings in costs of ownership, and in extending the life-time of avionics systems. This must be accomplished along with meeting needs of the user faced with a growing threat. With the growing complexity of avionics systems (as well as other systems), it is important to develop and maintain expertise in system planning, architecture, and management. P.T.O

Avionics
Systems engineering
Prototypes
Design
Requirements

AG\RD-LS-164
AGARD Lecture Series No.164 Advisory Group for Aerospace Research and Development, NATO SYSTEMS ENGINEERING Published May 1989 134 pages Recent AGARD activities have indicated a strong need for more efficient avionics system engineering. There is a growing need for reducing development time, effecting savings in costs of ownership, and in extending the life-time of avionics systems. This must be accomplished along with meeting needs of the user faced with a growing threat. With the growing complexity of avionics systems (as well as other systems), it is important to develop and maintain expertise in system planning, architecture, and management. P.T.O

Avionics
Systems engineering
Prototypes
Design
Requirements

AGARD

NATO ⊕ OTAN

7 rue Ancelle · 92200 NEUILLY-SUR-SEINE
FRANCE

Telephone (1)47.38.57.00 · Telex 610 176

**DISTRIBUTION OF UNCLASSIFIED
AGARD PUBLICATIONS**

AGARD does NOT hold stocks of AGARD publications at the above address for general distribution. Initial distribution of AGARD publications is made to AGARD Member Nations through the following National Distribution Centres.Further copies are sometimes available from these Centres, but if not may be purchased in Microfiche or Photocopy form from the Purchase Agencies listed below.

NATIONAL DISTRIBUTION CENTRES

BELGIUM
Coordonnateur AGARD — VSL
Etat-Major de la Force Aérienne
Quartier Reine Elisabeth
Rue d'Evere, 1140 Bruxelles

CANADA
Director
Dept of I
Ottawa,

DENMARK
Danish I
Ved Idre
2100 Cc

FRANCE
O.N.E.R
29 Aven
92320 C

GERMANY
Fachinfc
Physik, I
Karlsruh
D-7514

GREECE
Hellenic
Aircraft Support Equipment Directorate
Department of Research and Development
Holargos, Athens, TGA 1010

ICELAND
Director of Aviation
c/o Flugrad
Reyjavik

ITALY
Aeronautica Militare
Ufficio del Delegato Nazionale all'AGARD
3 Piazzale Adenauer
00144 Roma/EUR

LUXEMBOURG
See Belgium

NETHERLANDS
Netherlands Delegation to AGARD

NASA

National Aeronautics and
Space Administration

Washington, D.C.
20546

Postage and Fees Paid
National Aeronautics and
Space Administration
NASA-451

Official Business
Penalty for Private Use $300

**SPECIAL FOURTH CLASS MAIL
BOOK**

ment

L2 001 AGARDLS1648906145002672D
DEPT OF DEFENSE
DEFENSE TECHNICAL INFORMATION CENTER
DTIC-FDAC
CAMERON STATION BLDG 5
ALEXANDRIA VA 223046145

ARGE Daire Başkanlığı (ARGE)
Ankara

UNITED KINGDOM
Defence Research Information Centre
Kentigern House
65 Brown Street
Glasgow G2 8EX

UNITED STATES
National Aeronautics and Space Administration (NASA)
Langley Research Center
M/S 180
Hampton, Virginia 23665

THE UNITED STATES NATIONAL DISTRIBUTION CENTRE (NASA) DOES NOT HOLD
STOCKS OF AGARD PUBLICATIONS, AND APPLICATIONS FOR COPIES SHOULD BE MADE
DIRECT TO THE NATIONAL TECHNICAL INFORMATION SERVICE (NTIS) AT THE ADDRESS BELOW.

PURCHASE AGENCIES

National Technical
Information Service (NTIS)
5285 Port Royal Road
Springfield
Virginia 22161, USA

ESA/Information Retrieval Service
European Space Agency
10, rue Mario Nikis
75015 Paris, France

The British Library
Document Supply Centre
Boston Spa, Wetherby
West Yorkshire LS23 7BQ
England

Requests for microfiche or photocopies of AGARD documents should include the AGARD serial number, title, author or editor, and publication date. Requests to NTIS should include the NASA accession report number. Full bibliographical references and abstracts of AGARD publications are given in the following journals:

Scientific and Technical Aerospace Reports (STAR)
published by NASA Scientific and Technical
Information Branch
NASA Headquarters (NIT-40)
Washington D.C. 20546, USA

Government Reports Announcements (GRA)
published by the National Technical
Information Services, Springfield
Virginia 22161, USA