P-191

NASA Conference Publication 10028

NASA-LaRc Flight-Critical Digital Systems Technology Workshop

Edited by
C. W. Meissner, Jr.
NASA Langley Research Center
Hampton, Virginia

J. R. Dunham and G. Crim
Research Triangle Institute
Research Triangle Park, North Carolina

Proceedings of a workshop sponsored by the
National Aeronautics and Space Administration,
Washington, D.C., and held at
NASA Langley Research Center
Hampton, Virginia
December 13-15, 1988

APRIL 1989

Date for general release April 30, 1991

National Aeronautics and
Space Administration

Contents

Executive Summary

The National Aeronautics and Space Administration's Langley Research Center continually reviews its research programs for their responsiveness to national needs, and the Langley program in flight-critical digital systems has been a part of this systematic process. Flight-critical digital systems are in transition from specialized use to pervasive use, and a more comprehensive program review is being undertaken to ensure Langley's program is responsive to the changing needs. The first step of this review is to ask industry for its view of the issues which must be addressed for the practical realization of flight-critical digital systems. This was the question posed to a significant sample of the U. S. Aerospace industry at a U. S.-only workshop held at Langley on December 13-15, 1988. The results of this workshop are relevant not only to NASA and the participants but also to the entire industry. This publication documents the workshop and presents the issues and recommendations found by the individual working groups.

Issues that generated the most consensus across the workshop were the lack of effective design and validation methods with support tools to enable engineering of highly-integrated, flight-critical digital systems, and the lack of high quality laboratory and field data on system failures especially due to electromagnetic environment (EME). EME is emerging from relative obscurity as a technical issue to become a pacing issue for flight-critical systems deployment. There were many important issues identified by individual working groups such as proving the effectiveness of multiversion software, the lack of test and certification standards, and the lack of effective on-line test. The space systems participants had a unique viewpoint since they come from a tradition where extreme weight sensitivity has forced the use of non-redundant systems; they remain to be convinced of redundant systems effectiveness and demand figures of merit before embracing their use in either space or launch vehicles.

There were 115 participants at the workshop; 85 of them were from off-site organizations. Fifty organizations, including 31 commercial organizations, were represented. An organizational center of gravity would lie near the commercial aircraft industry. This make up was partly intentional since commercial aircraft has been the primary focus of the fault tolerance work at Langley. The good turnout is a measure of the importance of the subject and is a tribute to the willingness of the industry to support industry-wide activities. One of the overview speakers observed that the audience contained

enough expertise to design an aircraft. Judging from the companies represented and the experience of the participants, many of the critical design decisions for future U. S. aerospace systems may actually be made by those in attendance.

Although not a strictly technical issue, most groups had difficulty communicating about flight-critical digital systems. Much time was spent seeking common ground on which to define the issues. Perhaps the most important task is to formulate a common understanding of the problems and a common language to express that understanding.

1 Introduction and Overview

The Flight-Critical Digital Systems Technology Workshop was conducted by Langley Research Center to elicit the aerospace industry's view of the technical issues facing those who will be applying digital systems to flight vehicle functions, where loss of function would cause loss of vehicle or unsafe vehicle operation. The Langley Research Center has for the past fifteen years been developing methods to design and validate flight-critical digital systems with the primary emphasis on systems that would be suitable for commercial air transport application. Most of the Langley research is generic since it has dealt with assessing system reliability, designing systems to be validated, proving designs correct, and other research areas that are generally applicable to a variety of real-time systems. The aerospace industry on the other hand is practical and product-oriented, and the views of the issues are not uniform over the aerospace industry because of the different mission requirements for different industry products. The workshop was conceived to use the generic elements of digital systems as a context for the identification of issues that are or will be of importance to industry. It was felt that the combination of the theoretical and practical would produce a lively interaction among the participants and result in not only the widest coverage of ideas, but also the greatest benefit to the participants from having been exposed to greatly differing views of flight-critical digital systems technology. The workshop was open to interested parties with U.S. citizenship. The workshop was divided into three main parts as shown in the agenda in Figure 1.

An overview session opened the workshop, and six speakers presented their views of broad research issues that apply to both commercial and military aircraft electronic systems, and to the still broader questions of electronic reliability. These presentations provided a framework for the more detailed working group deliberations that followed. The overview session opened with an introductory presentation by Dr. J. F. Creedon, welcoming the participants to Langley and reviewing the goals of the workshop in the context of the NASA- Langley research program. The following presentations addressed research needs that result from increasing use of digital systems for primary flight controls. This increased use is accompanied by an increase in the complexity of flight control systems that includes more integration of flight controls with other subsystems (e.g., propulsion control, stores management). New requirements for system-wide integrity management including coverage of generic failure modes and threats due to electromagnetic interference are being established. These newer requirements translate into a disproportionate increase in design effort making concurrent, multidisciplinary engineering teams and early analysis of computer system dependability properties necessary. The presentations challenged some commonly held notions by, for example, showing that system failures in a sample of fielded systems were not caused by coding errors but by other elements of system implementation. The presentations also emphasized the human side of the system development process where communication and understanding between engineering groups must be enhanced to effectively support the integration of large systems. The visual aids for these talks are reproduced in Appendix A.

The heart of the workshop was the second part where the working groups met in three half-day sessions. The participants were asked to preselect a working group, and there was enough self-selected participation in each working group to form a viable group for each working group topic. Each working group was chaired by an industry representative except for the software group which was chaired by a university professor. The first half-day session was assigned according to each participant's stated preference. The second and third half-day working group sessions were open to further attendance selection at the participant's prerogative. Although there was some movement between groups, most of the participants chose to stay with their initial selection. There were seven working groups which represented generic elements of flight-critical system design and validation. The seven working groups topics were as follows:

- Aeronautical Requirements
- Space Requirements
- System Design For Validation
- Failure Modes
- System Modeling
- Reliable Software
- Flight Test

The aeronautical and space requirement working groups addressed the levels of dependability (e.g., performance and reliability) that must be achieved in order that flight-critical digital systems can fulfill useful roles in their respective flight regimes. The system design for validation working group addressed how flight-critical digital system technology can be made a part of the initial vehicle design thus escaping the traditional "add-on" role of electronic systems. The failure modes working group addressed how the various failure modes impact the design of digital systems used in flight-critical applications. The system modeling working group addressed the modeling techniques and support tools that are required to permit designers to adequately judge the merits of different system designs. System reliability modeling formed the bulk of the modeling discussions which may reflect the emphasis that has been placed on that aspect of fault-tolerant digital systems. The reliable software working group addressed how software should be treated as a component of flight-critical digital systems. The flight test working group addressed the role of flight test in demonstrating the acceptability of flight-critical digital systems. The following section is devoted to the detailed reports of the working groups.

The third part of the workshop was a half-day summary of the research issues found by each group. This was presented by each working group chairman speaking for his group. A viewgraph style summary of the major findings is presented in Appendix B. The key recommendations were provided by each working group chairman. Appendix C contains a list of workshop participants. A further condensation of the issues and recommendations over all the groups is presented in Figures 2 and 3. The purpose of this document is to present the issues and recommendations, since they represent the dedicated labor of some very knowledgeable representatives from the aerospace industry and will be of value to the industry as it finds its way in the age of flight-critical digital systems.

December 13, 1988

9:00 a.m. — 12:00 noon: Opening Session (Overview Talks)

9:00	Dr. J.F. Creedon, NASA Langley Research Center
9:30	Dr. Thomas B. Cunningham, Honeywell Systems Research Center
10:00	Dr. Carl S. Droste, General Dynamics
10:30	Mr. Jim Treacy, Federal Aviation Administration
11:00	Mr. Larry J. Yount and Mr. Richard F. Hess Honeywell Commercial Flight Systems
11:30	Mr. Richard S. Ullman, ITT Defense Technology Corporation

1:00 p.m. — 5:00 p.m.: First Parallel Working Groups Session

- Requirements for Flight-Critical Digital Systems — Aeronautical
- Requirements for Flight-Critical Digital Systems — Space
- System Design for Validation
- Failure Modes
- System Modeling
- Reliable Software
- Flight Test

December 14, 1988

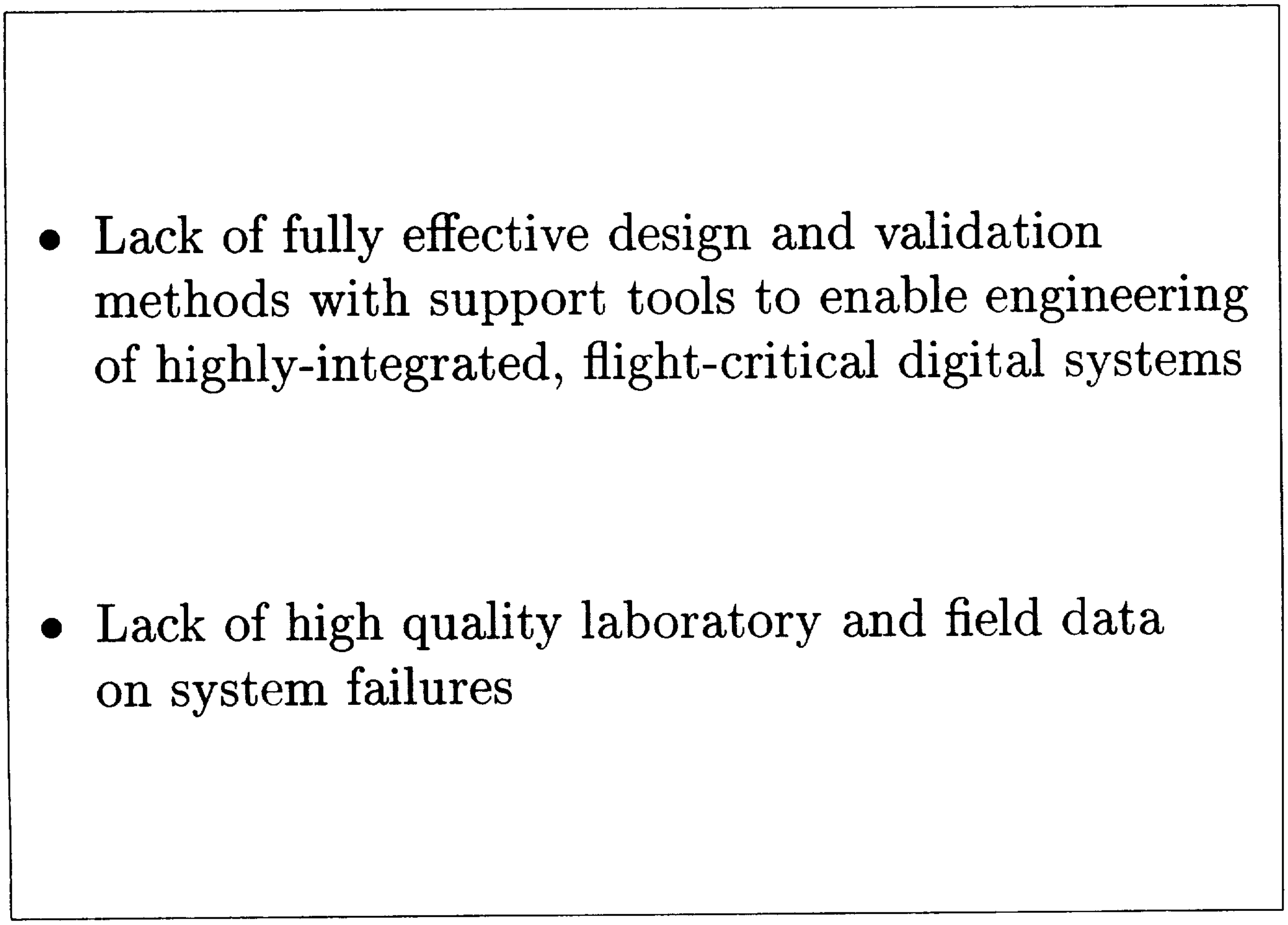
8:30 a.m. — 12:00 noon: Second Parallel Working Group Session

1:00 p.m. — 5:00 p.m.: Third Parallel Working Group Session

December 15, 1988

8:30 a.m.	Chairmen's Reports
12:30 p.m.	Workshop Adjourns

Figure 1: Workshop Agenda



- Lack of fully effective design and validation methods with support tools to enable engineering of highly-integrated, flight-critical digital systems

- Lack of high quality laboratory and field data on system failures

Figure 2: Summary of Issues Common to Many Working Groups

- Collect and analyze data for both operational and experimental systems
- Evaluate the cost-effectiveness of design and validation technologies
- Provide an easy-to-use, integrated, and validated environment of tools, guidelines, and results
- Establish criteria for EME validation

Figure 3: Summary of Recommendations Common to Many Working Groups

2 Summary of Working Groups Recommendations

2.1 Working Group Goals

The seven parallel working groups addressed the topics of aeronautical and space requirements, system design for validation, failure modes, system modeling, reliable software, and flight test. The participants in each working group represented industry, government, and academia. The relative representation is given in the viewgraph-style summary in Appendix B. To promote a lively examination of ideas and issues within each working group, the working group minutes were taken without attributing the items discussed to individuals or organizations. Each working group report is presented under the names of three participants who were responsible for both guiding and summarizing the working group discussions and for compiling the report provided in this section. This team consisted of a working group chairman, a representative from the Research Triangle Institute (RTI), and a NASA Langley Research Center (NASA-LaRC) sponsor. The chairman, who in the majority of cases was from industry, ran the session and ensured that the report captured the proceedings of the working group. The Research Triangle Institute representative captured the meeting minutes and drafted the working group report. In all except one case, the RTI representatives had actively participated in research and development related to the working group topic being addressed. The NASA-LaRC sponsor handled any problems that arose and ensured that the working group meeting and report met the objectives set forth.

Each working group addressed three groups of research issues. The first group of issues are associated with urgent problems where there is not time to mount a research program and for which some interim solution is the best result that can be expected. The government role would be supportive in providing results of previous/current research and perhaps demonstrations/evaluations of interim solutions. The second group is composed of research issues associated with longer term problems which are amenable to being addressed by deliberate research programs. The results of the research could be aimed at bettering current practice or at solutions to problems which are anticipated to become critical in the near future. The third group are research issues associated with problems that may become important at some more distant time because of a slowly developing technology or because of some foreseeable market demand.

The following subsections are the working group reports. Although there was a cross-group transfer of participants, the reports have been, for the most part, independently generated by separate groups. Issues and recommendations that appear in many reports can be judged to have a somewhat universal recognition by the total workshop.

2.2 Working Group Report on Requirements For Flight-Critical Digital Systems – Aeronautical

Chair: John Todd, Douglas Aircraft
Co-Chair: James Kelly, NASA-LaRC
Coordinator: Jill Hallenbeck, RTI

2.2 Requirements For Flight-Critical Digital Systems – Aeronautical

2.2.1 Introduction and Overview

The rapid introduction of digital avionics to jet transportation has been most profound. Almost every function associated with the operation of recently manufactured aircraft involves digital monitoring and processing techniques. The susceptibility of these newer digital systems to electrical transient effects appears to be higher than that of their older analog counterparts. Recently, the growing concern of upset to flight-critical, fly-by-wire (FBW) control systems in military aircraft has been highlighted in technical journals and the media by reports of high-energy radio frequency (RF) (HERF) fields insidiously inducing control-system failures that resulted in loss of aircraft and life. Additionally, lightning whose encounters are random and even less frequent, can produce more intense voltages and currents for a much shorter duration that, in turn, can also cause upset. Thus, effects of the electromagnetic (EM) environment produced by lightning could be even more insidious than effects that have been shown to be caused by man-made RF radiators (radar, microwave, television, radio, directed energy weapons).

The most dramatic news to hit the EM compatibility (EMC) community in some time is the recent revelation that a number of flight-critical, FBW control systems are highly susceptible to radiated EM energy. Conclusive proof is hard to come by as system upsets (i.e., nuisance disconnects, actuator movements, etc.) usually occur at significantly lower energy levels (lightning currents, RF field strengths) than energy levels that cause component failures, leave no trace, and are very often nonrepeatable. The problem of designing highly reliable, maintainable, and lightweight FBW flight controls is further complicated by the following technology trends.

Two recent trends in technology have increased the probability of digital system upset. First, commercial and military aircraft (including rotorcraft) are employing far greater percentages of composite materials, which inherently provide less low-frequency shielding within the Faraday cage provided by the airframe. Second, the increasing number of modern digital systems are turning to more densely packaged integrated circuits (ICs) that operate at lower powers and higher speeds and to more and more complex software. The susceptibility of these devices is usually evidenced by the response to an undesired transient voltage that creates any unwanted logic state which shows up on the system output. In general, IC susceptibility is dependent not only upon the incident source's amplitude and frequency, but also on the system's circuit values, clock rate, pulse width, pulse repetition frequency, bandwidth, loop gain, and flow rate of information processed by the device.

Despite these trends, the performance and weight requirements imposed on military aircraft necessitate the use of FBW flight and engine controls.

2.2.2 Critical Issues

The first issue before the working group was to come to a common understanding on a few important issues. The first of these is

2.2.2.1 What is a flight-critical digital system?

The working group discussed a wide variety of topics, including:

1. Reliability requirements
   (a) Commercial: 10^{-9} /Flight/Hour
   (b) Military: 10^{-7} /Flight/Hour
2. Commercial Aircraft Goal
   (a) Never in fleet lifetime will you lose an aircraft
3. The system is required for safe flight while engaged
4. Flight-critical systems are not necessarily required for the entire term of the mission
5. What/who drives the requirements?
   (a) government
   (b) liability
   (c) industry
   (d) consumers

6. If a flight-critical system fails, will you necessarily lose the aircraft?

7. Are development approaches different or the same for military and commercial aircraft?

   (a) Are the processes similar?

   (b) Are there different criteria?

   (c) Are different architectures emerging for each?

8. Doesn't the definition of flight-critical change dependent on

   (a) Is the system needed throughout the mission?

   (b) Is the system only needed during part of the mission? (e.g., landing system)

   (c) If you lose a system that wasn't critical, but its loss confuses the pilot and/or crew in charge of taking alternative measures, does the system then become flight-critical?

   (d) Is the system conditionally necessary only if it has been switched on? (e.g., automatic pilot)

   (e) Overall Mission Reliability

      i. Military - Peace time

      ii. Military - War time (e.g., may fly with failures if on a critical mission)

      iii. Commercial

2.2.2.2 Flight-critical digital systems

The working group lists the following digital systems as flight-critical.

1. Primary FBW/Fly-by-light (FBL) Flight Controls

   (a) actuators

   (b) signaling

   (c) computers

   (d) sensors

   (e) power supply

   (f) controllers, flight deck

2. Full-authority digital engine control (FADEC)/Electronic Engine Control (EEC)

3. Primary Flight Displays

2.2.2.3 Flight-critical digital functions

The working group lists the following digital functions as flight-critical in their opinion.

1. Aircraft stability and control (augmentation).

   (a) Enhance stability

   (b) Establish stability

2. Propulsion Control (Is this critical for multi-engine aircraft?)

3. Integrated Flight Propulsion Control

4. Flight Displays

5. Structure Load Limiting/Static & Dynamic Stability

6. Stores Management

7. Configuration Management

2.2.2.4 Flight-Critical Systems Implementation Requirements

To achieve a design of a "safe" flight-critical system, the following requirements must be met.

1. The designers must *prove* that their system will always recover from any and all non-hard faults reasonably quickly.

2. The pilot must be integrated into the system design process. He and the crew are the users of the system and are responsible for normal operation and for taking alternative actions in abnormal situations.

3. Well-formed and correctly implemented specifications are absolutely necessary.

4. A method to verify the completeness and consistency of specifications must be established.

5. Up-front analysis must be performed before functional specifications are written. How do you get the airlines/government to express their requirements adequately so that a functional specification can be written?

6. A requirements methodology, which outlines structured methods for building requirements suitable for simulation and designed for automatic testing of the requirements, must be developed. This approach also applies to the design methodology.

2.2.2.5 Discussion of issue areas

The working group decided to spend its time discussing the following eight topics and related subtopics. They are presented here in the agreed upon order of importance to the working group members.

1. Fault tolerance/redundancy management:

   (a) fault monitoring

   (b) fault detection

   (c) fault masking

   (d) fault isolation

   (e) fault types and specifications

   (f) system architecture

   (g) fault recovery

2. Functional specifications

3. Requirements methodology
4. Design methodology (for safety)
5. Prove/demonstrate reliability and survivability
6. Validation and verification
7. Certification basis
8. Maintenance (system availability is equivalent to the self-test result)

Discussion summaries of each of the issue areas follow.

2.2.2.5.1 Fault tolerance

As with other flight-critical systems, one of the primary requirements of FBW flight control systems is that they must be capable of sustaining hardware and software faults and still be fully operational. Over the past decade a number of approaches have been utilized to address this requirement. They range from very simple schemes where "sufficient" redundancy on critical components and extensive voting are used to "mask" faults to more complex schemes involving fault detection, isolation, and some form of system reconfiguration.

The simplest scheme is unacceptable for several reasons. First, the level of "sufficient" redundancy is very difficult to define due to the large number of potential failure modes possible in FBW systems. Second, a high level of redundancy is required to mask faults in all critical areas. Third, as undetected and uncorrected faults accrue, the level of redundancy and thus the reliability of the system falls off rapidly as a function of flight hours. Therefore, effective fault detection is a necessary first step in sustained FBW system reliability.

It is evident that any flight-critical fault-tolerant computer system must be able to handle software and hardware faults. Furthermore, it should be capable of dealing with coincident multiple faults to a reasonable extent. Except in the case of purely static redundancy where faults are masked via voting/signal selection, all fault-tolerant systems must be able to detect and isolate faults. To accomplish this task, sophisticated and overlapping fault detection techniques are required to provide near unity coverage. The extent of coverage overlap will depend on the importance of the operation and the redundancy of components in each portion of the computer system. Obviously the greater the coverage, the more complex our system becomes, so some trade-off must be made between system complexity and adequate fault detection. The trade-off criteria will depend to some extent on how the system is designed. Furthermore, coverage criteria will need to be further defined to better address generic faults issues.

Particular areas of discussion were

1. Types of Faults
2. Upsets
3. Lightening
4. "Hiccup"

It became clear from the discussion that everyone did not have the same understanding of terminology for describing faults and failures and rather than spend a lot of time not agreeing, the working group agreed to disagree, but to standardize the discussion to include the following two "definitions".

1. Fault - any condition which tends to degrade the operation of any part of the system.
2. Failure - the inability of a system to perform its intended function.

   Further threads of this discussion

   (a) What are the requirements for digital flight systems (from actuators to sensors)?
   (b) Probabilities are associated with requirements - if you can't prove it should you build it? How do you obtain statistics about faults (common mode - environment, design, power spikes, lightning, specification, non-stationary)? How do you *predict* fault statistics? What are the marginal areas which can be affected by research?
   (c) What is the probability of occurrence of generic hardware faults? Is this the bottom line?
   (d) Where do you put your redundancy?

(e) If you can't tell how reliable something is do you automatically add back-ups?

   i. Do back-ups only provide a psychological advantage?

   ii. Have back-ups ever paid their way?

   iii. Availability of back-up:

      A. Back-up is a form of redundancy

      B. Back-ups just point to the need for high reliability of primary systems

      C. Back-ups maybe shouldn't be automatically switched on - pilot should have switch - if he can't flip it - then failure.

      D. There is a need for reliability figures in the area of back-up systems.

(f) How long should recovery take?

(g) Requirements may need to state probabilities (occurrence of some kind of fault under some conditions) to use in design.

We draw on past experience. Each system is new, but past experience is used as a starting point. Each company has decided on reliability figures so that each design effort isn't a big research project, but these figures are company confidential.

NASA-LaRC needs to know experience of companies which is kept secret from the public, FAA, and other companies. They need a list of expected faults. These faults seem to vary in importance from company to company.

System reliability analysis output needs to drive testability requirements. Where is built-in test (BIT), self-test? How much?

2.2.2.5.2 Functional specifications

The proliferation of critical digital flight control systems has evidenced a number of new, potentially catastrophic failure modes not encountered with conventional mechanical and analog control systems.

Look at field data rather than other academic "studies" (e.g., multiversion). In general most companies will not release information except to NASA-LaRC, and then only under certain conditions.

FBW allow improvements in controls (e.g., sticks, input devices).

The pilot is a critical part of the system. His stress and workload vary throughout the flight/mission. He is the monitor of generic faults. In general, what should be assigned to the pilot? How should he be involved in the design process?

With specifications, it is hard to tell whether "all the bases are covered". Even if the specification is complete, how do you know if it is correct or not.

Standardized specification checkers, style sheets, and possibly a MIL standard are needed to validate specifications. Specifications can be a source of defects in the implementation.

2.2.2.5.3 Requirements methodology

If the output of the requirements phase of a design methodology is an executable specification, the designer could then build a model from the specifications and test the specifications. If the specifications are consistent, then you can test them and find any errors. Do available tools adequately support real-time systems?

Proof of correctness efforts and structured programming are two areas that aid the requirements methodology.

2.2.2.5.4 Design methodology

Today's systems have too many states for a designer to process by hand. A need for a structured approach to process a design specification is recognized. Tools to help designers through this process are an immediate need.

The use of design languages, graphics, and possibly English should be used in specifications to "cut out the middle man" and make specifications more readable and therefore more understandable to the designer. If the specifications are immediately understandable to the designer then errors will be found more readily. (Refer to *Communications of the ACM*, September 1988, Alan M. Davis, "A Comparison of Techniques for the Specification of External System Behaviors", pp. 1098-1115.)

There is a need for structured methodology tools, which can verify for correctness and which support traceability. These tools should support hierarchical representations, automate code generation/logic synthesis, and provide a library of reusable, reliable, and validated modules for both hardware and software.

2.2.2.5.5 Demonstration of reliability/survivability

Optimum protection of critical digital aircraft systems requires both survivability and recoverability from faults resulting from EM as well as other causes. For EM induced faults it appears that the use of many different prevention and tolerance techniques may be necessary to harden flight-critical digital systems to high confidence levels. If properly implemented, these high protection levels can be achieved with little or no weight and cost penalties while improving system reliability and maintainability.

The working group felt that the following topics were important in this discussion

1. Accelerated life testing of environmental effects
2. Synthesize
3. Analysis from experience of similar systems
4. Degradation of system over time
5. Reliability of software
6. Failure Modes and Effects Analysis (FMEA)

Accelerated life testing is important because of the large number of possible faults and the apparent random appearance of environmental factors that create a situation where these faults manifest themselves.

The ability to create testing facilities with control features is extremely important to our understanding of the environmental phenomenon. These facilities are extremely expensive. Unfortunately, specifications are vague because knowledge about unexpected (low probability) phenomenon is limited. The problem of modeling what is not understood was discussed briefly.

Designers depend a great deal on experience gathered from existing systems when designing a new system. Experience is what minimizes recurrence of the same mistakes.

Systems degrade with time, therefore systems need to be tested throughout their lifetime, adding to the life cycle expense, while also adding to the pool of knowledge from which designer's will draw.

How does one begin to show the reliability of software? How susceptible is software to random generic common mode faults.

How good are the models for FMEA analysis? Is physical fault insertion good enough? Can enough information be gained by this process? There are the problems of "building it" to "try it", a very expensive proposition.

2.2.2.5.6 Verification and validation (V & V)

The working group felt that this topic was too big for them to accomplish much in the remaining time, so a list of items were compiled for discussion at a later date.

1. How much is enough?
2. When do you quit?
3. When *can* you quit?
4. How can the task be partitioned?
5. How can the risks be managed?
6. V & V crosses traditional boundaries of responsibility, how can this be managed?
7. What tools are available? Must we rely on a "hot bench"?
8. How do you V & V interfaces?
9. How do you V & V complex subsystems?

Automatic theorem provers were discussed briefly. No consensus on the value of this type of tool was obtained, but its proponents pointed to

1. use as an alternative to testing
2. use for complementing testing

as benefits compared to traditional V & V.

Opponents referred to the

1. unknown reliability of new tools and techniques
2. developing methodology for when to use these tools
3. "unprovable" parts of the system still need traditional testing

as reason to take the "wait and see" attitude about the developing technology. Theorem provers have been successfully demonstrated on structured software, but have yet to make significant headway on hardware, except that produced by logic synthesis tools.

2.2.2.5.7 Certification basis

As critical flight control systems have become more complex they have evolved from pure analog implementations to include digital hardware implementations and now microprocessor based software implementations. The extreme complexity of a full time FBW flight control system virtually necessitates the use of some type of microprocessor based software implementation due to their great flexibility and computing power.

While microprocessor based software implementations are well suited for FBW and autoland systems, they are difficult to verify and validate for certification due to their extremely large number of possible failure modes and the indeterminate effects thereof. For a FBW system we would like to avoid, if possible, the time consuming and costly low level software verification and critical hardware FMEA verification, as well as exhaustive system verification and validation efforts.

Specifically, the working group briefly touched on the following topics:

1. FAA involvement during development. It is too late to plan certification after design and development of a new system. The FAA and other concerned parties should have plans prepared during development so that certification can proceed smoothly. It is too costly to build a system that can't fly.

2. Basis for certification should be done before production is complete. This point follows the same argument as above. Plans for flight testing and V & V for certification should be complete before the first plane exits the production line.

3. Validation of new technologies. The FAA should continue to accept the responsibility for looking at and approving new technologies. Standards are needed so that future systems can be designed effectively.

4. Validation procedure development. A methodology with specific guidelines is necessary to get a product through certification validation. This methodology should follow the validation of a new technology or new application of existing technologies.

2.2.2.5.8 Maintenance

It is suspected that unrecognized (non-permanent) faults are, to a good extent, responsible for unscheduled removals of aircraft digital equipment. This suspicion is generally supported by maintenance figures which indicate that for unscheduled aircraft digital equipment removals, less than 15% can be traced to the reported failure and in roughly 50% no hard-wired failure can be found. This trend accounts for the substantially lower Mean Time Between Removals (MTBRs) of high Mean Time Between Failures (MTBF) digital equipment. The direct result of these low MTBRs is higher maintenance costs.

A goal for commercial airlines, as proposed by the working group, is to have zero *unscheduled* maintenance of aircraft. A box is placed in the system, tested periodically, but is never removed. A procedure for revalidation of a system once it has been disassembled, fixed, and reassembled does not exist and it is the general consensus that maintenance itself introduces defects into the system. System developers are depending heavily on BIT and self-test for system checkout, fault detection and system go/no-go testing on the flight line, yet these are inexact technologies themselves.

On-card redundancy was offered as one way to accomplish some fault tolerance that could allow the zero unscheduled maintenance goal.

A concrete goal: make boxes as good as the cables in current systems.

How can we establish EM interference (EMI) failure protection so that this is a realizable goal?

2.2.3 Research Needs

2.2.3.1 Questions of interest

1. Identification and probability of common mode failure: how do you design to avoid common mode failure? Can you? Only in commercial? How far can the system degrade? At all? Should anybody (e.g., pilot, crew) know?

2. How do we tolerate intermittent faults? Soft fault - spontaneous recovery? No decision. What do we call it when a sensor starts to drift but has not left tolerance?

3. Definitions of fault types, e.g., soft/hard/intermittent/generic/common mode.

4. How do we recover the system from common mode failures?

   (a) How do we fail?

   (b) Leave power level same?

   (c) Go to back-up? De-emphasis?

   (d) Reconfiguring is a way to recover

   (e) Common mode failures

5. How do we model environment?

6. How do we test environment? What are the needed environmental specifications? How much? (Some work has to be done here.)

7. How do we achieve high dispatch reliability? How high a MTBR should be required?

   (a) How high with need for any maintenance?

   (b) As good as cables (inspect periodically)?

   (c) Never touch "Black Box"?

   (d) Environment changing too fast?

8. How do you maintain a critical digital system?

9. How fast does the system have to recover?

10. How long do you need to endure an intermediate transient fault?

11. How often do the various types of faults occur? (Basis for our estimates.)

12. How do we integrate user perspectives in systems design (i.e., pilots, flight crew)?

13. How do we capture the coupling effects of an integrated control system in a function specification?

14. How do you write high-level requirements for integrated systems (i.e., flight/engine) that can be translated into realizable functional specifications?

15. How do we test for reliability (acc. life test) and survivability. Unexpected situations - What are they? What do you do about them?

16. Loss of function is driven by common mode faults/design errors. How do we ferret out common mode faults early in the design phase?

17. How do you stress the system to bring out common mode/generic faults?

18. On future complex highly integrated systems, how do we do the extensive verification and validation needed? How can we decouple task verification and validation and then recouple?

19. Use automated theorem proving in addition to traditional testing methodology. This emerging technology may prove very beneficial to the V & V procedures currently in place.

20. How do we maintain a flight-crucial system over the aircraft lifetime?

21. How do you revalidate a repaired system?

2.2.3.2 Recommended research activities

The following is a table that represents the importance placed by working group members on each recommendation for a research activity. The work associated with accomplishing each of the recommendations should begin within the next two years to meet the needs and requirements identified by the working group.

Recommendations - how important are they?

H:	High priority
M^+:	Slightly less than high priority
M:	Moderate priority
L:	Low priority

Order Discussed	Recommendation	Priority
1	NASA-LaRC should do an in-house compilation and analysis of in-service reliability data for critical digital systems and present a sanitized version to the public. Here, "sanitized" means that data will not be attributed to a particular company, event, or accident, so that reluctance on the part of contributors can be minimized.	H
2	Obtain analysis and development tools. These tools will provide models and help predict fault insurance/fault tolerance/fault detection coverage.	M^+
3	Refine structured requirements methodology tools. Some rudimentary tools have been developed by industry, but are not available and are not reviewed outside the developers working environment.	L
4	Obtain structured methodology tools. These tools will ease the documentation struggle that takes place during every development cycle. These tools need to provide traceability of requirements and "correctness verification" features.	L
5	Increase knowledge base for system stress testing (random inputs, model noise environment, etc.). The biggest issue facing members of this committee is how the environment affects the systems. With more information about environmental effects, systems can be better designed to handle the effects.	H

Order Discussed	Recommendation	Priority
6	Develop cost and time effective validation and verification philosophy for complex integrated systems. The designers and implementors of today's flight-critical systems do not have a good idea about how extensive the V&V process needs to be to be adequate. This process is currently *very* costly. Responsible maintenance organizations are also extremely concerned with revalidation after reassembly and maintenance.	H
7	Validation of new technologies and background testing for certification basis (increased confidence level). The working group members are concerned with the confidence they can place in new technologies and new applications of existing technologies.	M
8	Cost trade-offs for designing complex fault tolerant systems. How much time and energy should go into each phase of the development cycle so that the end product is safe, but the producer can stay in business.	M
9	Validation and maintenance procedures of flight-critical systems over life of aircraft. Much is heard in the media about maintenance when there is an air disaster, yet there are few established procedures for safe maintenance. Developers of new systems would like to achieve a goal of no *unscheduled* maintenance over the lifetime of the aircraft.	M^+
10	Electromagnetic environment (EME) propagation analysis and testing and modeling for validation. Again, how can we get models of the environment validated so that developers can place high confidence in system modeling?	H

2.2.4 References

Davis, Alan M., "A Comparison of Techniques for the Specification of External System Behaviors", *Communications of the ACM*, September 1988, pp. 1098-1115.

2.2.5 Abbreviations

BIT:	built-in test
EM:	electromagnetic
EMC:	electromagnetic compatibility
EMI:	electromagnetic interference
EME:	electromagnetic environment
FADEC:	full-authority digital engine control
FBW:	fly-by-wire
FBL:	fly-by-light
FMEA:	failure modes and effects analysis
HERF:	high-energy radio frequency
IC:	integrated circuit
MTBR:	mean time between removals
MTBF:	mean time between failures
RF:	radio frequency

2.3 Working Group Report on Requirements For Flight-Critical Digital Systems – Space

Chair: Robert Gates, Martin-Marietta
Co-Chair: Howard Stone, NASA-LaRC
Coordinators: Robert Baker and Anita Shagnea, RTI

2.3 Requirements For Flight-Critical Digital Systems – Space

2.3.1 Introduction and Overview

High reliability has been required for the digital avionics systems used in life- or mission-critical space applications. Traditionally, reliability requirements have been met using extensively tested single string systems built with S-level parts and with backup components only for crucial single point failures. Emphasis was placed on fault avoidance as opposed to fault tolerance. Several factors contributed to this approach. The extraordinary premium placed on system weight and power consumption was the primary factor dictating a minimum of redundancy. Further, techniques to provide redundancy management were, in the past, crude and the associated hardware would have represented a significant proportion of the avionics hardware. It was also recognized that single string systems were much less complex and that adequate single string systems could be designed and tested. Consequently, there has existed a bias against the use of redundant fault-tolerant systems in the space application community. The introduction of fault-tolerant systems into these applications must not only be justified by performance and cost, but also must overcome the reluctance to depart from established practice.

Tradition notwithstanding, it was the consensus of the working group that technology advances in hardware, software, and fault-tolerant system architecture; the increases in lift capability; and the more demanding application requirements dictate the need to reassess requirements for fault-tolerant avionics. There is good reason to expect that fault-tolerant avionics systems will play an increasing role in space applications.

Applications whose requirements were considered in the working group included the aerospace plane, the shuttle, launch vehicles, earth orbiting satellites, space station, and planetary craft. However, the discussions were predominantly directed toward the requirements of the joint Air Force/NASA Advanced Launch System.

2.3.2 Critical Issues

2.3.2.1 Figures-of-Merit Appropriate Figures-of-Merit (FOM) for avionics used in space applications was the first topic considered in this working group. No single FOM was identified for all applications. Rather, several

FOM's must be considered and the specific FOM's that are appropriate depend upon the characteristics of the specific application. Reliability requirements for a number of applications result in system failure rates of 10^{-7} to 10^{-10} failures per hour, putting these systems into the very- to ultra-reliable range.

It was noted that for launch vehicles such failure rates for the avionics would be inconsistent with the overall failure rate of the vehicle and that often political and emotional consideration drive reliability requirements rather than cost and technical considerations. Fuel systems, engines, and other mechanical systems all fail at rates several orders of magnitude higher than these numbers. The goal for the ALS avionics was in the range of 10^{-5} failures per hour, which in turn was substantially lower than that for the rest of the vehicle.

It was further noted that the cost to validate such a system (10^{-10}), if indeed such a system could be validated, would be staggering by present approaches. It was stated that reliability requirements established should result in minimum life cycle costs. Presently, the cost of the avionics on a launch vehicle is a small percentage of the cost of the vehicle and the opinion was offered that the current avionics equipment is overqualified. The equipment has survived vehicle explosions and has continued operation. Since cost of this equipment is relatively small, the tendency has been to make certain that an expensive vehicle would not be lost due to such an inexpensive component.

The sensitivity of system reliability to the system recovery parameters such as coverage on short missions was discussed. It was noted that even long missions are punctuated by brief periods or mission phases requiring high reliability. During these phases, coverage would become an issue. Calculations indicate that changing the coverage parameter from unity probability of recovery to a recovery probability of .9 can change system reliability by several orders of magnitude for short missions and very high reliability. Consequently, particular attention must be given to the design of self tests and built-in test and evaluation (BITE) for these applications. It was noted that achieving coverage approaching 100% is extremely difficult in practice.

Industry representatives described an approach or philosophy used by Jet Propulsion Laboratories on certain space applications which de-emphasized evaluating reliability. Instead, systems were designed to have no single point failures unless the director approved each instance where a single point failure could occur. Systems were two-failure tolerant excluding these exceptions. It was pointed out that this philosophy could lead to costly systems and in some cases had the potential to reduce overall system reliability.

To evaluate reliability, the classes of faults or fault types against which system fault recovery and fault masking is effective must be specified in system requirements. With the environmental stresses that accompany space applications (thermal, mechanical, radiation, etc.), simple permanent "stuck-at" fault models are not adequate. More complex fault behavior such as transients must be considered. The potential value of the Byzantine resilient systems which are capable of handling arbitrary fault behavior for a set number of simultaneous faults was discussed.

Industry representatives pointed out that for launch vehicles the range safety requirements dictate more demanding design constraints than do the reliability requirements for mission success.

2.3.2.2 System Costs and Testing Currently, a substantial part of the costs for a space mission is devoted to testing. In some instances systems are subjected to excessive testing to the extent that system life could be reduced by the tests. Each unit undergoes complete burn-in tests, shock tests, and in some instances tests only appropriate for the development models. Testing continues up to launch time.

It was noted that multipath redundant avionic systems could potentially reduce testing requirements. If their use could eliminate full testing of every unit, recurring system costs would be reduced.

The computer performance requirements for flight control of launch vehicles are not very demanding. Excess capacity in the digital computer could be used to incorporate features which would lead to a reduction in overall launch systems costs as opposed to the cost of the avionics system. The cost of added complexity in the avionics system could possibly be offset by reduced costs in prelaunch testing and mission planning. Adaptive guidance and control, embedded health monitoring, system history logging and automated test and checkout are among the candidate features for the avionics system which could reduce overall launch costs but would increase the avionics system costs.

Due to the previous bias against multipath systems in the space community, development tests should include "piggy-back" tests on flight vehicles using conventional avionics. In these tests, the multipath system would perform all the functions and results would be compared to the conventional system. The conventional system would control the vehicle. This would allow many of the concerns regarding multipath systems to be examined and resolved without making what would be considered a large, risky change.

2.3.2.3 System Engineering and Integration Many of the testing and validation problems that impact current systems are related to the integration and testing of computer systems. It was felt that while individual subsystems could be demonstrated to meet their requirements, the problem of establishing that a system met overall requirements was much more difficult. The complexity of system designs and the interdisciplinary nature of modern systems (digital systems, software, advanced sensors, RF systems, etc.) contributes to this problem. A clear need exists for methods, tools, and facilities to support system engineering and integration. It was felt that some of these issues were of the scale that they could not be addressed by individual companies.

In addition to systems integration, concern was expressed over the need for validated design tools which can manage the complexity of modern systems while satisfying the need for systems reliability. It was recognized that the existing high level commercial design tools such as silicon compilers, software language compilers, control system design tools, etc. all have been developed with performance and functionality as requirements. Requirements for reliability, fault tolerance, validation, and testability were not considered. It is not known to what extent these tools could affect systems design in these areas, but the potential exists to introduce design faults which would reduce system reliability. With each high leverage design feature provided by these tools, a sequence of lower level design decisions are automatically made. These lower level design decisions were made to satisfy performance and functionality requirements, not to satisfy any of the many other requirements characteristic of mission-critical applications.

In this regard, some concern was voiced about the mandated use of Ada for future systems, even though Ada is recognized as having substantial advantages for system development.

A need for methods, tools, and facilities for rapid prototyping of systems was also identified in the working group. Again this was viewed as an area beyond the resources of a single company.

2.3.2.4 Future Systems Requirements for more advanced systems with more autonomy is the trend for space applications. As a result, systems to meet these requirements are more complex, have greater throughput requirements and have more demanding reliability requirements. Intelligent systems will be necessary to meet mission objectives. All phases of the system development process will be affected. However, a good deal of concern was expressed regarding the testing and validation of intelligent systems.

2.3.3 Research Needs

The following tables summarize, in order of priority, the urgent and longer term research needs identified by the working group.

URGENT ISSUES (NEXT 2 YEARS)*

- What is appropriate figure-of-merit for system design? Factors include cost, reliability, time, coverage, and availability
- Define approach to specifying parts levels (Class S vs Class B)
- Limit scope of production testing for multipath production acceptance testing
- Adequacy of fault coverage via bit/self test (example VLSI @ less than 90%) for advanced fault-tolerant systems
- Integration of new environments into design process - e.g., S.E.U. and E.M.E.
- How suitable are dissimilar designs?
- Use of Ada in multi-path systems
  - Rendezvous
  - Real time

*Prioritized

LONG TERM ISSUES (NEXT 5 YEARS)

- Increased emphasis on integration research
  - Health monitoring interface
  - Validation of adaptive GN&C/intelligent systems

- Plan to educate regulatory organizations and key management people on multi-path systems (requires identification of approval wickets)

- Establish criteria/architecture for lift-off with known failures to increase availability

- How to insert modern technology in a long term space program

- Specify more comprehensive fault models including transient and hardware/software design faults

- Concern that modern, high leverage design methods/tools can contribute unreliability to multi-path systems (e.g., unintended redundancy from compilers and graphic circuit design programs)

- Define recovery approaches during prelaunch phase and for reusable vehicles

2.4 Working Group Report on System Design for Validation

Chair: Gerald C. Cohen, Boeing Advanced Systems
Co-Chair: Dan Palumbo, NASA-LaRC
Coordinator: Joanne Bechta Dugan, RTI

2.4 Design for Validation/Verification Working Group

2.4.1 Introduction and Overview

Validation refers to the process by which the system requirements (from which the specifications are derived) are shown to be correct. Verification is the process by which a system is shown to meet the given specifications. Demonstrating that a system meets the given requirements is not the same as showing that the system will perform the desired function, as it does not guarantee that the requirements themselves are correct. The working group on Design for Validation and Verification (V/V) was charged with discussing how to design a flight-critical system so that it can be validated and verified. Several classes of issues were discussed, commencing with an assessment of current design methodology, and the need for better systems engineering. Many participants voiced a desire for a set of design guidelines that would facilitate V/V. Two of the many chronic V/V problems were discussed, the first being the need for a suitable language for requirements definition, and the second being the need for reliable failure data on new technologies. The working group spent the largest portion of time discussing the need for an integrated set of tools to aid in the design of verifiable and validatable flight-critical systems. Many of the other issues were then couched in terms of the integrated tool set. Each participant was asked to list the research issues thought to be the most important; the lists were then consolidated and prioritized (by majority vote). Each of these classes of issues will be summarized in subsequent sections, followed by the prioritized listing of research goals, and a discussion of the issues relating to the integrated tool set.

2.4.2 Critical Issues

2.4.2.1 An assessment of the current situation The working group meeting began with several participants discussing problems associated with recently developed systems, in that systems have been plagued with cost overruns, late delivery, etc. A major cause of schedule delay was felt to be the digital electronics systems. One participant stated that late design changes to a particular system were causing 1500 wiring changes to be performed per day, while the actual embedded computer was accounting for between 5 and 15 percent of the entire system errors. The conclusion of this discussion was that a move must be made toward a total system engineering design environment. This approach would reduce "side effects" from design changes, and would enforce traceability of design changes throughout the hierarchy of design. (This same feeling was expressed during the opening session of the workshop).

Many of the participants from the commercial side of aviation were interested in hearing from those who have been designing military systems, as they have been using FBW technology for some time. The desire for a compendium of experience gained and lessons learned from the military systems was expressed. It was thought to be extremely useful to gather and analyze data especially with respect to failure rates and modes. The participants were skeptical as to whether such a project could be realized, however.

2.4.2.2 Design guidelines Another major topic of discussion concerned the development and utilization of guidelines for designing systems that are inherently verifiable and validatable. As an example, several participants spoke of designing *deterministic* systems by avoiding the use of preemptive scheduling, interrupts and floating point numbers. The rationale behind the use of such restrictive guidelines would be the elimination, possibly, of a large number of required simulation runs. An extremely large number of simulation runs is required to obtain anomalies through the statistical runs (known in the avionics arena as "rare event data."). The large number of required runs can be easily demonstrated. If the requirement state that the system failure rate for a flight-critical system be less than 10^{-11} (failure rate of the structure) for a 6 hour mission, then 10^{-13} simulations are required to obtain the rare event data. Obviously this number is impractical and techniques must be found to reduce it, possibly by variance reduction techniques such as importance sampling.

One participant suggested the following example set of guidelines for designing a verifiable and validatable system:

- Layered hierarchy of computing functions.
  From the application layer down to the silicon layer, each layer should be strongly verified and should have tight, consistent interfaces to adjacent layers. There should exist fault-tolerant capabilities at each layer.

- Isolation of ultra-critical functions.
  For special protection, ultra-critical functions should be kernalized.

- Traceable Dependencies.
  To support validation of design modifications, dependencies among function modules should be logically traceable. (The issue of traceability was raised many times in subsequent discussions.)

- Smoothly degradable behavior.
  Efficient trade-off between multiprocessing for high performance and fault tolerance.

- Run-time testability and observability.
  The assumed attributes of each component should be testable at run time. That is, if a certain behavior was assumed to be impossible or certain, this behavior should be observable at run time, so as to verify the correctness of the assumptions and to assure the adequacy of fault tolerance.

There was a large degree of interest in developing a standard library of building blocks, each of which is formally and completely verified and validated. The functions implemented should be general enough to allow widespread usage. Each module should have formal specifications and generic designs, with provisions for modification and re-verification. Custom systems would be built by connecting the blocks and verifying the connections.

Participants also desired a set of guidelines for using new technologies and methodologies, such as N-version implementations, design diversity by dissimilarity, designing integrated vehicle management systems, etc. The development of such guidelines were recognized as research issues, and will be discussed in the section where the prioritized list of research topics is presented.

2.4.2.3 Specific V/V problems Two recurrent problems in designing for V/V were discussed; the first has a simple but economically impractical solution, while the second appears to have no satisfactory solution. The first problem arises when trying to assess the correctness, performance or reliability of a system using new technologies, in that it is difficult to predict component failure rates, failure modes and possible erratic behaviors. The classic solution to this problem is to dramatically overdesign the system (for example, in the Byzantine failure problem); this approach is frequently infeasible in systems that are constrained by cost, size, weight, etc. The efficacy for designing to withstand Byzantine failures was discussed. One participant claimed to have never seen a Byzantine failure, while another claimed that all redundant computers that have been built and monitored sufficiently have exhibited Byzantine behavior.

One participant asked whether it would be feasible and useful to build a fly-by-wire flight control system (perhaps with a mechanical backup) that is extensively over-monitored, so as to obtain the needed data. There was near-unanimous agreement that such an approach would produce badly needed data. However, there was also near-unanimous agreement that such a system would not be built by short term profit-minded management.

The second problem is the desire for a clear, precise language for requirements and specifications. From the software engineering perspective, studies have indicated that errors in specifications or requirements are more costly than any other kind of error. Specification languages have been developed that are easy to use, but only for narrow applications. (See Rich and Waters, "The Programmer's Apprentice: A Research Overview," IEEE Computer, November 1988.) It is not clear whether the design of parts (or all) of a flight-critical system is a narrow enough application for the development of a useful specification language. Even if a specification language could be developed such that the specifications could be shown to meet the requirements, only half of the problem would be solved, that being the verification part of V/V. The problem of proving that the requirements themselves are correct, consistent, and complete (the validation part of V/V) would still remain. The consideration of this problem was considered to be of paramount importance to virtually all participants in the workshop. It was suggested that perhaps rapid-prototyping could be at least a partial answer.

2.4.2.4 An Integrated Tool Set Several participants suggested the development of an integrated tool set for system design as a valuable research goal. Much of the session was spent discussing the desired attributes of such a tool; as such the tool served as a stimulus for the discussion and prioritization of many research goals. Even though the development of such an extensive tool set appears to be a rather lofty goal, it may serve as an agenda for future research.

The most comprehensive presentation of the integrated tool set is discussed here. Most participants accepted this proposal as a valid starting point, but did not necessarily agree with each proposed point. A multiphase program for the development of the tool set was proposed, commencing with the development of a formal language, in which the requirements for the flight control system or vehicle management system would be expressed. This language would be developed mindful of the goal of validation, and would provide the capability for upgrading the requirements. The Integrated Tool Set (ITS) would then be used to verify the correctness of the system against the requirements and specifications at each design phase.

- Phase I – Development of requirements and specifications for ITS. The requirements should consider (among other characteristics) the size and types of systems to be evaluated and verified, and the system characteristics and parameters to be included.

- Phase II – Critical issues (such as those listed in the prioritized list of research topics) are identified and resolved.

- Phase III – Development of an integrated set of tools for performance and reliability evaluation of systems defined by the specifications. The outputs of these tools should be in a form that is comparable to the requirement specifications.

- Phase IV – Evaluation of the tool set. The set of tools, as well as its integration should be rigorously validated and verified.

It was suggested that the development of silicon compilers for integrated circuit design might serve as a model for the development of an integrated tool set.

The working group participants saw the development of a tool set as a framework for enforcing guidelines for verifiable designs. The development of such guidelines was assumed to be the major item on the agenda for future research.

Part of the third working group session was spent compiling a list of attributes that were desired in such an integrated tool set. The optimism of the participants was evident during this phase of the discussion, as nearly every conceivable attribute was suggested and embraced. It was envisioned that the optimal tool set would nominally have the system requirements as the input, and would produce design specifications (via an interface to a CAD tool) and a verification test set as outputs. The tool would support some common language which would enforce traceability, support verifiable hierarchical development and building block designs, and interface to reliability and performance tools.

It was clearly understood that such an immense undertaking would require a long term monetary commitment from NASA and industry. It was envisioned that NASA would serve as a central control point for tool development, and would integrate tools and techniques researched and developed by others into the tool set, and would support and maintain the tools after integration. Such an undertaking also requires close ties with industry; industry should use the tools in good faith and provide feedback to NASA for future improvements.

2.4.3 Research Needs

Each participant in the working group was asked to list the research items deemed important for their work in design for validation and verification. The separate lists were compiled and then were separated into two lists, based on the desired research goal. Each of the items on the first list was characterized by the desire for a set of guidelines for using new design techniques. These items were designated (by majority vote) as high, medium or low priority. The items on the second list were other research items to support design for validation and verification.

2.4.3.1 Guidelines for New Design Techniques

2.4.3.1.1 High priority research items

- Failure containment, coverage, FMEA, redundancy management
  Although much has been done in the area of assessing coverage and assuring failure containment, this area continues to be a high priority concern, especially in the face of increased integration of flight-critical functions.

- Environmental effects
  The topic of EMF and HERF research was mentioned by several different working groups. The major interest in the V/V group was an

assessment of the effects on the failure rates and failure modes of components in flight-critical systems.

- Reusable building blocks
  The concept of designing systems by utilizing a library of pre-designed and verified building blocks has been generally accepted as a good idea. The research needed in this area is in the methodology for developing and combining building blocks and in the design, verification, and standardization of modules.

- Concurrent processing
  Because of the inherent difficulties in verifying multiprocessor systems (for enhanced performance rather than fault tolerance), concurrent systems are currently not approved for flight-critical functions. Techniques for validation of concurrent systems is thus considered a high priority item.

- N-version hardware and software
  The concept of N-version and dissimilar designs needs to be addressed more fully; for example, what are the relative advantages and disadvantages, is there an optimal N, and how can one quantify the merits.

- Guaranteed determinism
  If systems can be guaranteed to be deterministic, the task of validation is simplified considerably. Research into methods for guaranteeing determinism, even if some subsystems are nondeterministic, is needed.

- Complexity metrics and complexity reduction
  A methodology should be researched and developed for partitioning complex systems into more manageable units, perhaps hierarchically. Issues concerning interfaces and failure containment must be investigated.

2.4.3.1.2 Medium priority research items

- Integration concepts
  As systems become more integrated, guidelines must be developed that pertain to partitioning of the hardware, partitioning of the software, and partitioning of functions between the hardware and the software.

If multiple functions operate on the same computer or use data from the same sensors for example, it is difficult to guarantee independence, which makes verification and validation more difficult.

- Validation of Modeling Assumptions
The participants of the design for V/V working group were concerned with the concept of testing for the validity of assumptions made in analysis.

- Highly reliable communications
Reliable standard two-way real-time data communications networks and development of applicable verification techniques is crucial to the highly-integrated flight control application.

- Performance/Reliability Tradeoffs
Guidelines for trading performance for reliability, and a tool which would allow such trade-off case studies are needed.

2.4.3.1.3 Low priority research item Only one item was determined to be a low-priority research item, that of determining guidelines for designing for smoothly degradable behavior. The problems to be addressed in this area concern the restoration of the system following a transient error, and the derivation of bit-synchronous protocols that will allow graceful degradation following an error.

2.4.3.2 Other research needs The research items in this section were not considered to relate to the development of guidelines for design, but rather were considered to be research items otherwise related to design for V/V. This list was not prioritized.

- Technology transfer
There should be increased levels of communication between industry, military and NASA to define the lessons learned on past flight-critical systems, covering such areas as redundancy management, performance, backups, documentation, testing, failure modes. It was suggested that a fuller interaction between NASA, industry and FAA personnel should be promoted, so that research could be better focused on real problems. Perhaps researchers could be placed in industry for a time.

- Variance Reduction techniques
  Several participants wanted to see research continue in simulation methodology as applicable to flight-critical systems. Some example topics include the investigation of variance reduction techniques such as importance sampling, and investigation of the use of parallel processors to speed simulation run times.

- Software Waterfall technique as applied to multiprocessing systems
  It is not clear whether current waterfall techniques for software specification are appropriate for multiprocessing and knowledge-based machines.

- Deterministic bounds for non-deterministic behavior
  For what kinds of non-deterministic behavior can deterministic bounds be developed?

- Formal verification
  Work on formal verification should undoubtedly continue; its potentials and limitations should be realistically evaluated.

- Undocumented functionality
  Some systems may provide more functionality than is documented. This can pose a problem when errors on input pins cause the system to go into internal test mode, for example.

2.5 Working Group Report
on
Failure Modes

Chair: Douglas Frank, Douglas Aircraft
Co-Chair: Harry Benz, NASA-LaRC
Coordinator: James Watterson, RTI

2.5 Failure Modes Working Group

2.5.1 Introduction and Overview

The goal of the failure modes working group was to identify industry needs that relate to failure modes in flight-critical digital systems and to suggest research programs to NASA-LaRC that will satisfy these needs. In other words, what research help does industry need?

Motivation for this effort stems from the more extensive use of electronics in flight-critical systems (flight control, engine control, cockpit displays, etc.) in place of mechanical/hydraulic systems. This shift to electronics has undoubtedly created new failure modes that have not yet been discovered. For example, it is evident that the failure modes in a fly-by-wire flight control system are not the same as those in a mechanical/hydraulic flight control system. All failure modes must be identified and understood in order that industry be in a position to preserve safety. Such information will also allow industry to improve aircraft reliability and maintainability, which will in turn increase system availability and decrease system life-cycle cost.

The working group proceeded by (1) formulating a set of definitions that relate to failure modes, (2) identifying failure mode issues for flight-critical digital systems, (3) developing a comprehensive list of research problems by brainstorming, and finally (4) molding the list of problems into research programs that will contribute to solutions for industry problems/needs in this area. The following writeup represents the consensus of the working group participants listed in Appendix C of this report.

The working group realized early in the working session that definitions (failure, hard fault, soft fault, failure mode, etc.) were needed so professionals with different backgrounds could proceed effectively with the stated task. It was decided that the following definitions, some of which have been proposed as IEEE standards, would be used by the working group.

1. **VEHICLE:** Highest level component.

2. **SYSTEM:** Second highest level component.

3. **FAILURE MECHANISM:** A mechanism that could produce a failure (metal migration,voltage overstress, lack of grease, etc.)

4. **FAULT:** Phenomenological reason for a failure (open wire, stuck-at-1 fault, stuck-at-0 fault, design fault, mechanical friction, etc.)

5. **FAILURE:** Deviation of behavior from specification (arithmetic function failure, storage failure, flight control function failure, etc.)

6. **HARD FAILURE:** The same as a permanent failure. Repeated use of the same input and same initial conditions results in the same incorrect response

7. **SOFT FAILURE:** The same as a temporary failure. Repeated use of the same input and same initial conditions does not result in the same response

8. **LATENT FAILURE:** Fault has occurred, error has not occurred.

9. **FAILURE MODE:** A failure and the associated symptoms [pilot taking a nap while plane nose-dives (human failure mode), unwanted movement of ailerons (system failure mode), unwanted flight control command (subsystem failure mode), etc.]

10. **ERROR:** Deviation of device's state from correct state [pilot fails to note that plane is in a nose-dive (human error), erroneous position of ailerons (system error), erroneous signal from flight controller (subsystem error), etc.]

2.5.2 Critical Issues

Throughout the working session, the working group identified various technology issues that relate to the use of electronics in flight-critical systems. The list of issues include environmental threats, the existence of new failure modes, test techniques, validation/certification, and modeling. These issues appear immediately below in outline form.

- Environmental threats
  - High Energy Electromagnetic Environment (EME)
    - Lightning (direct strike)
    - High frequency RF (HERF)

        - Electromagnetic pulse (EMP)
            - Nuclear EMP
            - Lightning EMP
        - High energy nuclear particles that cause single event upset (SEU)
        - Temperature & humidity
        - Vibration

- Failure modes of new technologies
    - CMOS/SOS
    - Gallium arsenide
    - Room temperature superconductors
    - VHSIC/VLSIC
- Failure modes at various levels of system hierarchy
    - transistor level
    - gate level
    - board/module level
    - subsystem level
    - system level
- Relationship between failure modes and functional demands
- Energy required to upset or damage a component
- Testing procedures
    - Research & development
    - Manufacturing
    - Field
- Test techniques for fault detection/isolation
    - nonconcurrent test techniques (off-line testing)

  - concurrent techniques (on-line testing)
  - functional testing
  - behavioral testing

- Troubleshooting
  - Fault detection
  - Fault isolation
  - Repair & retest

- Validation/certification/integration criteria
  - How does one validate a system?
  - How does one validate a component of a system?
  - How does one certify that a module performs as intended?
  - How does one assure that the module is compatible with other modules in the system?

- Modeling problems
  - device modeling
  - fault modeling
  - component stress (over a period of time)

- Problems encountered when handling products
  - Electrostatic discharge (ESD)
  - Shock & vibration
  - Effect of temperature and humidity

2.5.3 Research Needs

2.5.3.1 General Needs During a brainstorming session, thirteen research problems were identified by the working group. These problems, which reflect industry needs as perceived by the working group, are presented below.

1. Establish electromagnetic environment (EME) internal to aircraft that results from external electromagnetic environment.

   - Obtain experimental data (data base)
   - Develop analytical results
   - Demonstration

2. Develop certification/integration criteria for digital systems.

3. Explore component trends and EME sensitivities.

4. Study chip level testing (functional test in a known environment).

   - Static/dynamic tests
   - Error detection and correction (EDAC)
   - System test interface

5. Update MIL- STD-HDBK-217E (F).

   - Add new parts (VHSIC/VLSIC, Josephson devices, etc.)
   - Include fault data (hard and soft faults)
   - Include failure mode data (including probability of failure mode occurrence)

   MIL-STD-HDBK-217 or its equivalent is the basis for failure rate prediction. However, it does not include information on transients and intermittents which represent roughly 50-90% of failures. Thus, information contained in the existing document is the tip-of-the-iceberg. A program is thus needed to supply the missing data. If not done, reliability modeling is of limited value.

6. Establish Testability Program.

   - Manufacturing test
   - Field test
   - Functional fault models for system testing
   - Failure mode modeling

7. Define and investigate reasons why fly-by-wire systems fail.

   - Multiple independent faults (never observed)
   - Single point failures (observed some times)
   - Domino failures (most common?)

   Most research has been aimed at multiple independent faults, but it is the other two that appear to be the real problem. A program is needed to confirm this observation so that research, development, and design resources are allocated to the right problem.

8. Verify that present redundancy techniques are adequate.

9. Fiber optics (life testing).

10. What is the probability of a Byzantine failure? Is the probability of a Byzantine failure high enough to require specific architectures? Without knowing this, design decisions will have to be made by flipping a coin. A program is thus needed to determine this probability and answer additional sensitivity questions such as: Does increasing clock rates, which decrease timing margins and increase metastability rates, cause an increase in Byzantine failures?

    Note: A Byzantine failure is any failure that produces different symptoms for different observers. For example, a flip-flop that outputs a signal that lies between 0 and 1 can be interpreted as a zero by some downstream devices and as a 1 by other downstream devices. Byzantine failures are more commonly related to timing. They also tend to be single point failures, and in such case the probability of system failure cannot be lower than that of the associated Byzantine failure.

11. Methods exist for detecting many degraded conditions in non-electrical components (detect vibration, bearing noise, crack in metal, etc.). However, methods have not been developed for detecting degraded conditions in electrical systems. Thus, advanced methods are needed for detecting degraded electrical components before they fail. Two possible approaches are the use of analog techniques and/or the use of failure history.

12. Determine the percentage of failures that are hard/soft in existing systems.

    - Data to be supplied by industry
    - Create data base for use by industry

13. Investigate failure mechanisms for new technologies.

2.5.3.2 Suggested Research Programs The primary objective of the failure modes working group was to identify three groups of problems to be addressed by NASA. In the first group are urgent problems where there is not time to mount a research program and for which some interim solution is the best result that can be expected. The second group consists of longer term problems that are amenable to being addressed by deliberate research programs. The third group consists of problems that may become important at some time in the future. These three groups of problems are identified in the sections immediately below as (a) short term research effort, (b) long term research programs, and (c) future research problems.

2.5.3.2.1 Suggested Short Term Research Effort The working group concluded that all problems identified in section 3.5 are very important and should be part of a long term or future program. This being the case, no short term research efforts were identified by the working group.

2.5.3.2.2 Suggested Long Term Research Programs By analyzing the research problems (brainstorming output) presented in section 3.5, the working group identified three long term research programs. Program #1 was created by combining research problems 1, 2, and 9. Program #2 consists of research problems 3 and 5, and Program #3 is comprised of research problems 3 and 4. These three long term programs are further described immediately below.

Certification and Integration Criteria (Long Term Program #1)

EME/HERF internal environments

- Perform tests to determine internal environment caused by HERF, lightning and other external EM environments (some data does presently exist)
  - Engine nacelles
  - Fuselage
  - Cockpits
- Perform any additional vehicle/component tests to achieve the needed degree of comprehensiveness in the EME response data bases
- Formulate a data base from the above tests
- Determine the transfer functions that relate the internal and external environments in typical aircraft structures
- Develop a national resource analysis capability that includes the various transfer processes (functions/models) from the external environment to the digital circuits that provide the needed data processing/functions (environment, aircraft exterior, aircraft interior, cables, equipment enclosures, circuits, etc.)
- Develop methods to assess the impact of the internal environment on systems
  - Test methods
  - Analysis methods

Design/Verification

- Develop methods to verify adequacy of hardware and software designs to prevent system functional upset due to the EME/HERF internal environments. This will include test and analysis methods which are needed to verify protection against upset in the lightning multiple stroke and multiple burst environments (How much protection is enough?).
- Determine the degree of comprehensiveness of system representation in the test configurations needed for flight-critical systems verification/validation.

Life Testing

- Perform accelerated life testing on fiber optics to determine sensitivity to EME and low level radiation and vibration environments (also consider thermal sensitivity, embrittlement, opacity, etc.).

Note: Long term program #1 should be carried out in cooperation with other organizations in the United states and Europe (e.g., FAA, DOD, RTCA, SAE committees AE4L and AE4R, EUROCAE, IEEE, and major airframe manufacturers).

Testing (Long Term Research Program #2)

Troubleshooting and Repair

- Smarter diagnostic aids are required to reduce trouble shooting and LRU turn around times. One possibility is to store acceptance test software in the system, and download the software into a PC for field testing; this could enhance testing and at the same time reduce the amount of special test equipment required for field testing. Study is required to provide verification of effectiveness and cost savings attributable to test software LRU loading and standard test busses (e.g., IEEE 488).

Error-detecting/correcting (EDAC) codes

- Most digital systems do not presently utilize EDAC codes that carry wordlength penalty that increases with detection/correction capability of the code. A study is requested to demonstrate reduced mean time between unit removals due to the use of EDAC or other concurrent in-flight monitoring, and cost effectiveness.

Fault injection guidelines

- Guidelines are requested for injecting faults to assess performance and capabilities of fault detection techniques and diagnosis of prototypes in a laboratory environment.

Component Trends (Long Term Program #3)

Empirical Data

- An on-going test program to provide empirical data on new families of digital devices is requested. Such data needs to define Energy thresholds such as speed-power product $p_d \cdot \tau_d$ and failure modes for new devices (damage thresholds, upset thresholds, degradation thresholds, etc.). This data should be maintained in a national resource data base.

Update MIL- STD-HDBK-217E (F)

- Add new parts (VHSIC/VLSIC, Josephson devices, etc.)
- Include fault data (hard and soft faults)
- Include failure mode data (including probability of failure mode occurrence)
- Provide guidance for interpreting empirical data (relate empirical data from standardized waveforms to various waveforms produced by associated EME).

2.5.3.2.3 Suggested Future Research Programs Future research problems identified by the working group consist of those problems in section 3 (problems 7, 8, 10, 11, 12, and 13) that were not included in the long term research programs of section 4. These problems are listed immediately below (A through F) for completeness.

A. Define and investigate reasons why fly-by-wire systems fail.

   (a) Multiple independent faults (never observed)

   (b) Single point failures (observed some times)

   (c) Domino failures (most common?)

   Most research has been aimed at multiple independent faults, but it is the other two that appear to be the real problem. A program is need to confirm this observation so that research, development, and design resources are allocated to the right problem.

B. Verify that present redundancy techniques are adequate.

C. What is the probability of a Byzantine failure? Is the probability of a Byzantine failure high enough to require specific architectures? Without knowing this, design decisions will have to be made by flipping a coin. A program is thus needed to determine this probability and answer additional sensitivity questions such as: Does increasing clock rates, which decrease timing margins and increase metastability rates, cause an increase in Byzantine failures?

D. Develop advanced analog techniques for detecting degraded components before they fail.

E. Determine the percentage of faults/failures that are hard/soft in existing systems.

   (a) data to be supplied by industry

   (b) create data base for use by industry

F. Investigate failure mechanisms for new technologies.

2.5.4 References

"*Process Fault Detection Based on Modeling and Estimation Methods - A Survey,*" Rolf Isermann; Automatica, Vol. 20, No. 4, pp. 387-404, 1984.

"*Engine Fault Analysis: Part I - Statistical Methods,*" Sood, Friedlander, Fahs; IEEE Transactions on Industrial Electronics, Vol. IE-32, No. 4, November 1985.

"*Engine Fault Analysis: Part II - Parameter Estimation Approach,*" Sood, Fahs, Henein; IEEE Trans. on Industrial Electronics, Vol. IE-32, No. 4, November 1985.

"*On the Identification of Certain Non-Linear Networks of Automata,*" Gollman; Cybernetics and Systems Research, North-Holland Publishing Co., 1982.

"*Continuous Systems with Digital Behavior,*" Reusch, Szwillus, Cybernetics and Systems Research, North-Holland Publishing Co., 1982.

"*Identification Evaluation Methods,*" V. Klein; AGRAD Lecture Series No. 104, November 1979, pp. 2-1 - 2-21.

"*The Choice and Use of Different Model Sets for System Identification,*" Hajdasinski, Eykhoff, Damen, van den Boom; Proceedings of the 6th IFAC Symposium on Identification and System Parameter Estimation, 1982.

"*Parameter and State Estimation: Introduction and Interrelation,*" Sorenson; Proceedings of the 6th IFAC Symposium on Identification and System Parameter Estimation, 1982.

"*System Identification - A Survey,*" Astrom, Eykhoff; Automatica, Vol. 7, 1971, pp. 123-162.

"*Multivariable System Identification - A Survey,*" Nierderlinski, Hajdasinski; Proceedings of the 5th IFAC Symposium on Identification and System Parameter Estimation, 1979, pp. 43-76.

"*Identification Methods,*" Ljung; Proceedings of the 6th IFAC Symposium on Identification and System Parameter Estimation, 1982, pp. 57-64.

"*Experiment Design*, Goodwin; Proceedings of the 6th IFAC Symposium on Identification and System Parameter Estimation, 1982, pp. 65-71.

"*Model Validation,*" Ljung, Proceedings of the 6th IFAC Symposium on Identification and System Parameter Estimation, 1982, pp. 73-75.

"*Performance Analysis of Generalized Upset Detection,*" Blough, Masson; Digest of the 17th International Symposium on Fault-Tolerant Computing, 1987, pp. 218-223.

"*Implications Associated with the Operation of Digital Data Processing in the Relatively Harsh EMP Environments Produced by Lightning,*" Hess; International Aerospace and Ground Conference on Lightning and Static Electricity, June 1985.

"*Transient Fault Behavior in a Microprocessor: A Case Study,*" Duba, Iyer; Proceedings of the ICCD, 1988.

"*Digital System Upset - The Effects of Simulated Lightning-Induced Transients on a General Purpose Microprocessor,*" Belcastro; NASA TM84652, 1983.

"*Data and Results of a Laboratory Investigation of Microprocessor Upset Caused by Simulated Lightning Induced Analog Transients,*" Belcastro; NASA TM85821, 1984.

"*EMP Upset: Overview and Test Methodologies,*" Thomas, Diloreto; AFWL-TR-84-60, March, 1985.

"*Test Waveforms and Techniques for Assessing the Effects of Lightning-Induced Transients,*" SAE AE4L Committee Report: AE4L-81-2, 1981.

"*Recommended Draft Advisory Circular: Protection of Aircraft Electrical/Electronic Systems against the Indirect Effects of Lightning,*" SAE AE4L Committee Report: AE4L-87-3, 1987.

"*Subsystem EMP Strength Verification Methods: Upset Detection and Evaluation for Military Subsystems*," Hanson; Dikewood Company Final Report DC-FR-4088.330-1, 1988.

"*ESD Considerations for Electronic Manufacturing*," Frank, Donald; American Society of Manufacturing Engineers, Westec Conference, Douglas Paper 7324, 1983.

"*Please Keep Your EMC Out of my ESD!*," Frank, Donald; Annual Reliability and Maintainability Symposium, Douglas Paper 7622, 1986.

"*Implications Associated with the Operation of Digital Data Processing in the Presence of the Relatively Harsh EMP Environments Produced by Lightning*," Hess; 10th International Aerospace and Ground Conference on Lightning and Static Electricity(ICOLSE), June 1985.

"*Sharing the Protection of Aircraft Electronic Systems Against the Effects of High-Level Electromagnetic Environments Between Traditional Protection and System Architecture*," Hess, Yount, Knoller, Masson, Larsen; 8th AIAA/IEEE Digital Avionics Systems Conference, October, 1988.

2.6 Working Group Report on System Modeling

Chair: Philip S. Babcock, C. S. Draper Lab
Co-Chair: Sal Bavuso, NASA-LaRC
Coordinator: Charlotte Scheper, RTI

2.6 System Modeling

2.6.1 Introduction and Overview

2.6.1.1 Motivation and Goals The industry representatives in this working group are looking for tools and techniques that will assist them in the design of fault-tolerant systems. They are aware of the reliability tools that have been developed at NASA-LaRC, and look to LaRC for guidance on what the tools can do and how they can be applied to their problems, as well as for instruction on how to use the tools. They are also interested in the research and techniques/tools that LaRC is developing to gather the data required as input to the reliability tools. Currently, industry is having great difficulty in selecting the proper tool for a given problem and applying it correctly. The lack of a coherent and unified presentation of how the wide variety of tools relate to each other, and how to properly exploit the richness of this variety has greatly limited the effectiveness of all the tools. The tool builders must address this issue for the tools to be accepted and used.

2.6.1.2 Industry Needs The need exists in industry for modeling and tools to support fast development of responses to RFP's, system design, and trade-off studies. Modeling is required during the design of fault-tolerant systems to translate high-level requirements into system/architecture requirements, to improve productivity during the design process, to provide a means of fleshing-out preliminary designs, and to provide justification for the resulting design. Modeling is also required to conduct trade-off studies between different designs with respect to attaining system requirements within specified constraints, especially cost constraints. To perform the required modeling, techniques that are developed should be embedded in tools. The tools in turn must be easy to use, accurate, validated, and efficient. It was the consensus of this working group that no tool currently meets these needs and that industry quite clearly can recognize a tool that meets their needs.

The current reliability tools may be adequate for most of industry's problems, but they must be made more easy to use and apply to a given problem. The current tools do provide an adequate base for extension into analysis areas that industry expects to be high growth areas, such as performance and cost analysis; however, a much tighter interaction between the industry users and the NASA model builders is required for current and future investment in tool development to be justified.

2.6.2 Critical Issues

Ultimately, industry feels that an integrated tool environment is needed to aid them both in quickly deciding between competing system designs and in completing the design and assessment of a target system. However, the primary focus of the working group discussions was (1) what tools exist for reliability analysis, (2) what methods and internal models are they based on, (3) how are they used, (4) how can the user be certain that he has accurately matched a tool to his system and accurately input a system representation to the tool, and (5) how can the user be confident of the results computed by the tool. The iterative process of modeling a system for reliability analysis and modifying it based on the results of the model analysis is illustrated in Figure 1. In this process, understanding the system means to understand the types of faults the system is subject to, what effects the faults have on the system, and how the system can detect, isolate, and recover from faults. This understanding of the system is essential in selecting a tool that can properly represent the system and compute a solution. Once the tool has been selected, the appropriate input model(s) has to be created. The model can then be evaluated, and the results used to determine if modifications to the system are necessary.

The reliability tool builders see themselves involved in an iterative development - as new fault tolerant systems are designed, they modify their tools to handle the new modeling needs. The users' primary concern is that the tools demand too much knowledge of a given technique, such as Markov modeling, and provide little support for judging the validity of their results. Some of the current means used by members of this group to validate the results include using several tools and comparing the results, and computing hand solutions of simplified models as a comparison. The members of the group also feel that there should be more explicit guidelines as to which modeling techniques are appropriate for various systems. There was general consensus that the tools perform the numerical computations correctly, but uncertainity as to correct interpretation and application of the tool models by the users. The users are also concerned about validating the system models they create and the difficulty of attaining the data required for input parameters to the tools. In general, it seems that the users and the tool

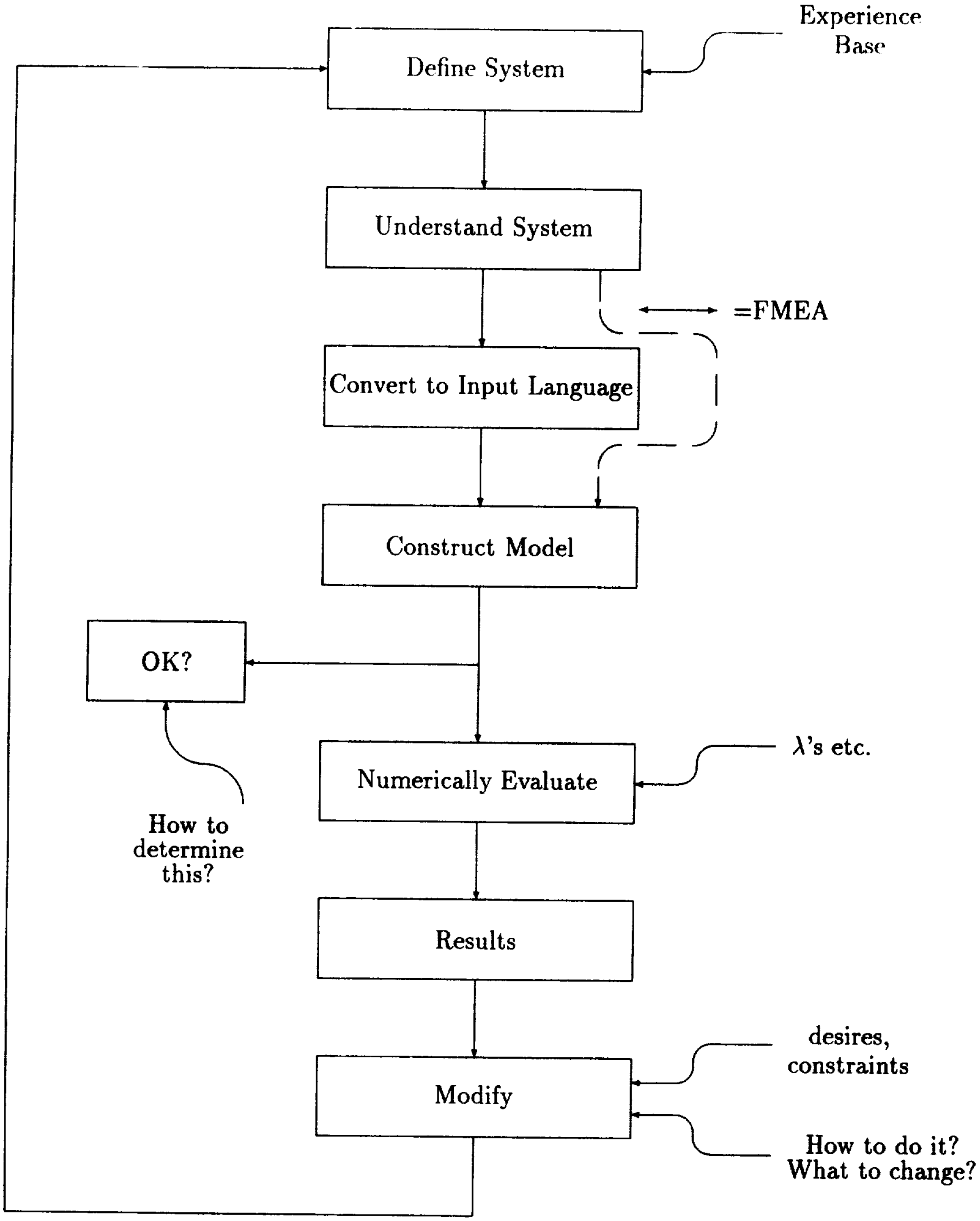


Figure 1: The Modeling Process

builders differ in their expectations of user expertise in the various modeling techniques and applications of the various tools to particular problems. Also, the disagreements among various tool developers about the utility and applications of certain techniques and the real uses to which their tools will be applied add to the doubts of the user community as to whether or not the current tools meet their needs.

The topics identified for discussion by the working group were

- What tools are available
- How to model complex systems
- How to model coverage
- How to compile, compute, and/or estimate data needed for model parameters
- What are the issues in tool development
  - Development of graphical inputs
  - Designation of beta test sites
  - Determination of what is needed
  - Determination of who develops what
  - Identification of industry needs
- What are the mathematical issues
- How can tools be used for quick justification of design decisions
- Performance modeling
- Cost modeling
- How can design and evaluation be integrated
- How to verify system models that are created as input to tools
- How to verify results from tools

2.6.3 Research Needs

As a result of the working group discussions, issues were identified that the industry representatives felt should be addressed, possibly by LaRC. The issues were categorized as urgent issues that should be resolved within a year, research issues that should be started now and resolved within 5 years, and long term research issues with no timetable for resolution.

2.6.3.1 Urgent Issues The focus of the urgent issues is to exploit the full power of the reliability tools currently available from NASA by enhancing the information that is available about the tools, by creating guidelines for their use, and by continuing research on data collection for model inputs. It was felt that an industry/NASA workshop should be held so that a more precise and detailed identification could be made of the type and scope of the problems and applications that industry wants to model than currently exists. The successful completion of the following actions is essential to the full utilization of the current tools:

- the development of guidelines for matching application to tool for all NASA tools

- the development of guidelines for selecting coverage representation and parameters

- the creation of an example-based user's guide for each tool that would show how to use that tool through an evolutionary presentation of each of its features

- the development of explicit guidelines for a user to confirm that a model or a tool's output conforms with his input and intentions

- the provision of tutorials directed at the application of the tools to the user's specific problem area

- the continuation and expansion of experiments and data collection to determine model inputs, particularly focused on the internal processes relating to coverage

- the active and continuing confirmation by NASA that the above actions focus on and apply directly to the industry users' needs.

2.6.3.2 Research Issues The research issues to be started now and completed within 5 years were selected with the goal of improving the power and productivity of NASA's current tools. First, to increase the use and value of the tools, they should be made more portable; easier to use through the addition of prompts, library functions, sophisticated on-line help facilities, graphical and textual input, user-specified defaults, input consistency checks, and facilities for flexible output manipulation. To increase the user's confidence in the tool results and to improve the user's ability to validate models and outputs, the tools should be modified to automatically bound all internal modeling and numerical approximations, to tell the user why the answer is what it is, and to include more graphic output capabilities. Information on why a certain answer resulted can provide design insight as well as permit confirmation of tool usage. Finally, to make a start toward an integrated tool environment, NASA-LaRC should select an appropriate input format or vocabulary for reliability modeling tools and develop and/or acquire performance modeling and evaluation tools.

2.6.3.3 Long-Term Research Issues For the long term, NASA should initiate the research that will be required to extend and integrate individual "ility" tools into an environment for supporting all phases of system design. Capabilities need to be added to individual tools to assist in making informed decisions with respect to a particular "ility" in the design optimization process. However, these capabilities have to be selected and developed to complement and interact with those of other tools. In particular, the interactions and tradeoffs between reliability and performance need to be identified for integration into general performability tools.

For reliability estimation, research is needed to develop a tool that can selectively create fault tree, markov, or simulation models from a unified, high-level input language, solve that model and produce output in a unified and descriptive format.

2.6.4 References

"*On the Next Generation of Reliability Analysis Tools*," Babcock, P. S., Leong, F., Gai, E.; Contractor Report 178380, NASA, October 1987.

"*CARE III Model Overview and User's Guide: First Edition*," Bavuso, S.L. and Peterson, P. L.; Technical Memorandum 86404, NASA, April 1985.

"*Evaluation of Reliability Modeling Tools for Advanced Fault Tolerant Systems*," Baker, R. and Scheper, C.; Contractor Report 178067, NASA, October 1979.

"*SURE Reliability Analysis: Program and Mathematics*," Butler, R. W. and White, A. L.; technical Paper 2764, NASA, March 1988.

"*ASSIST User's Manual*," Johnson, S. C.; Technical Memorandum 87735, NASA, August 1986.

"*The Fault-Tree Compiler*," Martensen, A. L. and Butler, R. W.; Technical Memorandum 98098, NASA, January 1987.

"*Unified Reliability Model for Fault-Tolerant Computers*," Ng, Ying W. and Avizienis; IEEE Transactions on Computers, Vol. C-29, No. 11, November 1980.

"*HARP: The Hybrid Automated Reliability Predictor: User' Guide*," Rothman, E. M., Mittal, N., Dugan, J. B., Trivedi, K. S. and Bavuso, S. J.; Technical Report, Duke University, Durham, NC, December, 1988.

2.7 Working Group Report on Reliable Software

Chair: Martin Shooman, Polytechnic University
Co-Chair: George Finelli, NASA-LaRC
Coordinator: Linda Lauterbach, RTI

2.7 Reliable Software

2.7.1 Introduction & Overview

The Software Reliability Working Group was composed of 21 members who met on December 12 and 13, 1988 to discuss and evaluate the important research issues in this area.

Most of our efforts were spent on defining and listing the various issues associated with this area. The group discussed the importance of software issues within a system context, rather than as a separate entity divorced from the hardware issues. As an example, we considered hardware and software coupled issues as falling within our area. If a system contained three computers (hardware) with a hardware voter to form a triple modular redundancy (TMR) scheme and identical versions of the software on each computer, this was considered hardware fault tolerance and was not addressed in our discussion. However, if the voting rule was an algorithm programmed on a microprocessor, then we classified the system as a software implementation of hardware redundancy and included this as a topic in our area. Similarly, if there were system issues which related to both hardware and software, we considered these as well.

The group discussed definitions of terms when necessary, to reach a common understanding. The term software fault tolerance is used to refer to software algorithms that implement hardware fault tolerance, whereas the term fault-tolerant software is used to refer to schemes to mask software faults, such as n-version programming and recovery blocks. Also, as is evidenced in the research literature, a lot of members had different ideas about the meaning of Verification, Validation and Test (VV&T). To allow for the broadest meaning of this term, the group used it to encompass all activities various people associated with the term.

It was noted that VV&T of software has traditionally been a difficult task in that it absorbs a large portion of development resources, it is difficult to formulate a well-defined methodology for effective testing, and it is difficult to identify which system failures are due to software faults. VV&T is especially difficut in the case of fault-tolerant systems where one wishes failure rates of 10^{-9} per hour. These very low failure rates make the problem of software, hardware, and system VV&T very difficult.

We spent less of our time on our attempts to classify and rank the 42 research and advanced development issues which we defined. These issues are grouped under the 6 major categories given in section 2.7.2; however, there is some overlap between categories. We voted on the importance and time needed for investigation of each issue. Importance was ranked as high (H), medium (M), or low (L). The time needed was categorized as up to two years (2), five years (5), or 10 years (10). These rankings appear to the left of each issue. In most cases, a consensus was reached on the rankings. In a few cases there was a substantial split, and in such cases both opinions are listed. For example, a ranking of (H-5,2) means that almost all agreed that the issue was of high priority and most thought it to be a five-year issue; however, a substantial minority thought it could be accomplished in two years.

Time did not permit a second round of rethinking of the issues and a grouping of them into a number of coherent interrelated research programs. If this had been done, some of the rankings might have changed a bit. For example, if a two-year issue of medium priority was found to be necessary to collect data to be used in a five-year issue of high priority, then the two-year issue would become high priority. Twenty four (57%) of the issues were rated of high priority, twelve (29%) of medium priority, and the remaining six (14%) were of low priority.

Subsequent to the workshop, all participants were provided with a rough draft of the categorized issues, and were asked to vote for the ten top issues among the 24 high priority items. The participants (NASA employees excluded) were then polled by telephone for their votes. Ten of the 11 responded. These were averaged according to the following ranking scheme: the most important issue was rated as 10, the next most as 9, down to the 10th issue, which was rated as 1 (one respondent only ranked the top five issues). Issues which were not ranked in the top ten received a score of zero. The top nine issues (highest average scores) are listed in Section 2.7.3.2. All the 24 high priority issues received at least one vote; however, only those with an average score of 2.7 or higher appear in Section 2.7.3.2 and each of these issues received votes from either 4, 5, 6, or 7 out of the nine respondents. (Assuming a uniform distribution due to random selection, the scores would have all been 2.3). A smaller number of issues emerge if we read carefully the descriptions given in Section 2.7.3.2. In a number of cases, the same issue is being raised from a different viewpoint. As examples, compare the similarities of: issues 25 and 30, issues 7 and 8, and issues 28 and 33.

On December 14 when each group made their report, a few emphasized the importance of creating a national data repository. It seems that most of the members of the group would support the concept as evidence by some of the topics discussed: "Performance, reliability, and availability analysis of real-world N-version systems", "Definition, specification, and collection of reliability data", "Correlating the measured reliability with [various factors]", "Data collection of fielded systems and lab experiments", as well as others.

2.7.2 Critical Issues

During the working group meetings, a first draft of an outline of categories was created. These categories were meant to span all detailed issues discussed. The working group had time to place most but not all issues within the categories. Following the working group meeting, the remaining issues were categorized. To adequately incorporate these remaining issues, the categories were slightly modified; the outline below shows the resulting categorization.

SOFTWARE RELIABILITY WORKING GROUP RESEARCH ISSUES

- SOFTWARE ISSUES IN FAULT-TOLERANT SYSTEMS
  - Fault-Tolerant Software Techniques
  - Hardware and Software Integration Issues
- RELIABILITY/AVAILABILITY/SAFETY ANALYSIS OF SOFTWARE
  - Reliability Growth Models
  - Common Mode (Coincident Error) Models
  - Metrics
  - Development; VV&T
  - Safety and Risk
- DATA COLLECTION
  - Fielded Systems
  - Lab Experiments

- SOFTWARE TESTING AND EFFECTIVENESS
  - Evaluation
  - Coverage Criteria
- SOFTWARE DEVELOPMENT METHODOLOGIES
  - Evaluation
  - Paradigms
  - Language Issues
  - Tools
- CERTIFICATION AND STANDARDS

2.7.3 Research Needs

2.7.3.1 Research Issues Identified The following annotated list sets forth the forty-two issues identified by the working group. The issues are presented within the category framework delineated in the previous section. Immediately preceding each issue, a priority and time rating is given, in the format <Priority>-<Time>, according to the following scheme. Priority was assigned based on the participants' assessment of the urgency in the need for research results to guide government and commercial endeavors to predict, measure, and ensure the reliability of flight-critical software.

PRIORITY	TIME
H: High	**2: up to 2 years to complete**
M: Medium	**5: from 2 to 5 years to complete**
L: Low	**10: from 5 to 10 years to complete**

- **SOFTWARE ISSUES IN FAULT-TOLERANT SYSTEMS**
  - FAULT-TOLERANT SOFTWARE TECHNIQUES

1. **(H-2) Definition, properties (robustness, convergence, etc.), and analysis of various voting strategies (as used in N-version, recovery blocks, adaptive, on-line & spares, repair, etc.).** For the various voting strategies proposed for fault-tolerant software, it would be beneficial to have a common basis for their definition and delineation of their various properties. Then, the strategies could be analyzed (e.g., performance and reliability) and compared for potential applications.

2. **(H-2) Examine the effects of reduced levels of verification, validation, and test (VV&T) on the reliability of fault-tolerant software systems.** Fault-tolerant software has been proposed in order to protect against software faults. It has also been suggested that this development technique could reduce the need for certain VV&T activities on the individual versions. The relationships between the VV&T of the versions, the fault-tolerant software strategy, and the overall systems resultant reliability characteristics must be investigated.

3. **(H-5) Cost-benefit analysis and selection criteria for various fault-tolerant software techniques.** The bottom line for the acceptance of any new technology is cost. Given various voting strategies, VV&T techniques, and reliability characteristics, analysis capabilities are needed that will assess the costs and benefits of various N-version techniques and allow for quantitative comparisons and selection criteria.

4. **(H-5) Performance, reliability, and availability analysis of real-world fault-tolerant software systems.** Data from fielded systems should be used to analyze the effectiveness of implemented N-version systems in achieving their required performance, reliability, and availability levels.

5. **(H-10) Guidelines for the development of N-version programs to minimize common mode (coincident) failures.** N-version software assumes that the individual versions fail independently. To achieve independent failure, the individual versions are usually programmed by separate programming teams. Studies have shown that this approach cannot guarantee independence. Investigations need to

be conducted which can provide the basis for guidelines for the development of N-version software.

6. (M-2) **Definition and properties of reconfiguration/ recovery techniques.** Just as there are various voting techniques to consider for N-version software, once a vote has been made on non-unanimous results, it may be necessary to reconfigure or recover the hardware, software, or system. Work to define reconfiguration and recovery techniques, and to compare the properties of these techniques, is needed.

   – HARDWARE AND SOFTWARE INTEGRATION ISSUES

7. (H-5) **Interaction and impact of fault-tolerant software on hardware redundancy management.** From a systems viewpoint, how do fault-tolerant software and redundant hardware systems interact? Specifically, how is the operation of hardware redundancy management affected when the hardware is overlaid with fault-tolerant software? Is there danger of negative synergisms?

8. (H-5) **Categorize, validate, and analyze the cost-benefits of software to manage hardware redundancy.** In all current redundant hardware systems, there is some portion of the redundancy management that is done in software. This type of software has its own attributes and properties which must be understood and analyzed.

9. (M-2) **On-line discrimination between hardware and software faults.** In a fault-tolerant system with recovery, it is imperative that faults be isolated and identified. When a redundant hardware system is running fault-tolerant software to protect against software faults, the discrimination between hardware and software faults is necessary so that proper recovery actions can be taken. If software faults are thought to be hardware faults, good hardware units will be discarded and the problem will still remain.

10. (L-2) **Redundant hardware (non-lock step) with single software version.** There are two approaches for allowing tasks (single version, non-fault-tolerant software) to execute on a redundant hardware system: synchronously (lock step) and asynchronously (non-lock

step). When task execution is asynchronous, then the job of fault recovery becomes more complicated than when tasks run synchronously.

11. **(L-5) How to design flight-critical software to be independent of underlying hardware.** In order to make flight software usable across airframes and across different vendors hardware, it would be desirable to make software independent of the underlying hardware. This would also reduce the overhead for certification and recertification.

12. **(L-10) Validation of non-deterministic scheduling of tasks.** There are two approaches to the scheduling of tasks: deterministic (according to predefined task schedule tables) and non-deterministic. When scheduling is non-deterministic, the order of task execution will be affected by various factors and thus result in tasks being executed in a random order. These factors must be determined along with their effects on the order of tasks and the execution of the system.

- **RELIABILITY/AVAILABILITY/SAFETY ANALYSIS OF SOFTWARE**

  - RELIABILITY GROWTH MODELS

13. **(M-2) Investigate the correlations among wall clock time, CPU time, input space, and test vectors for software reliability modeling.** Software reliability growth model describe the reliability of software as a function of time (i.e., clock or cpu time) or the number of executions (i.e., number of inputs or test vectors). There has been some discussion in the research literature concerning the appropriate unit of "time" for reliability estimation. Some models appear to work better for one unit over another. Controlled investigation needs to be conducted to resolve this issue.

14. **(M-2) Unified hardware, software, and systems reliability models.** From an overall systems point of view, very little, if any, work has been done to enable the estimation of the reliability of a total system, including hardware and software. The pursuit of software reliability models without regard to the total system context could limit the results of this research.

– COMMON MODE (COINCIDENT ERROR) MODELS

15. **(H-2) Coincident Error model analysis.** A theoretical basis for reliability modeling of redundant software and coincident failure has been developed. Estimation techniques based on this approach need to be pursued. Also, this modeling approach should be used to investigate the benefits and limitations of fault-tolerant software.

– METRICS

16. **(H-2) Correlating the measured reliability of the software with software metrics.** Analyses to determine which metric or combination of metrics (such as design and code complexity measures, lines of code) best predict the observed operational reliability of the software.

– DEVELOPMENT; VV&T

17. **(H-5) Correlating the measured reliability of the software with development strategies and associated VV&T.** Data from fielded systems should be used to establish empirical relationships between various software development strategies, along with their associated VV&T techniques, and the resultant measured reliability.

– SAFETY AND RISK

18. **(H-2) Hazard Analysis and Failure Modes and Effects Analysis.** The feasibility and efficacy of these techniques should be investigated for their application to software. Failure modes and effects analysis concentrates on identified failure modes, and the effects the failures have on the software or system. Hazard analysis is a much broader based activity; it entails identifying conditions and events that may result in an accident or catastrophe. Hazard analysis may include failure modes and effects analysis.

19. **(M-2) Feasibility of using Software Fault Trees.** Investigate using fault tree analysis to show dependence of software faults, and from this information develop improved test cases.

20. **(M-2) Reliability (Safety) Block Diagrams.** A traditional method for analyzing hardware reliability is to make a model of the system based on the probability of success paths existing in the system. Such models are generally referred to as reliability block diagrams or graph models. They can be used in much the same way as fault trees, to model the success of system software.

- **DATA COLLECTION**

  - FIELDED SYSTEMS

21. **(H-5,2) Definition, specification, and collection of reliability data.** Lack of voluminous, complete data on fielded systems hinders our knowledge of failure rates, and ability to validate reliability models and to estimate failure rate parameters for reliability prediction. The contents of a complete data base of error and reliability data needs to be defined and specified. Then, a mechanism to facilitate collection, storage, and access of these data needs to be set up. Note that due to the low failure rates of fielded flight-crucial systems, large quantities of data collected over time are needed to further the progress in this area.

22. **(H-2) Collection of metric data on systems for correlation with measured reliability.** Fault-tolerant systems have been fielded, such as the A320, the space shuttle, and the X29A. Fault-tolerant software design is being used to increase reliability. Metric and reliability data from these systems would provide invaluable feedback to the research and user communities. A framework for the collection of these data is needed. Such data would enable evaluation of development and VV&T strategies, modeling approaches, and cost-benefit analyses.

  - LAB EXPERIMENTS

23. **(H-5,2) Definition and execution of lab experiments.** Experiments can be used to investigate specific issues of concern. In the past, experiments have provided data on the error rates of software due to different bugs, the rates of coincident failures in N-version software, and the strengths of various testing techniques. These studies have also provided insight on how to collect meaningful data for investigating software reliability.

- **SOFTWARE TESTING TECHNIQUES AND EFFECTIVENESS**

  - EVALUATION

24. **(H-2) Analysis of error classes and their associated functional mapping, and appropriate techniques for detections.** What classes of errors are found in flight-critical software? Are more catastrophic errors observed in some error classes than in others? Does the functionality of a software unit map to the classes of errors likely to be found in that unit? Which testing techniques are better at exposing each class of errors?

25. **(H-5) Cost-benefit comparison of various testing strategies.** How do different criteria compare for selecting input spaces (e.g., error crystals, fault trees, data partitions) to emphasize in testing? How do techniques such as dynamic branch testing compare with techniques such as static structure analysis? Is it worthwhile to place more emphasis on formal testing at earlier phases (e.g., unit testing)?

26. **(M-5) Stopping rules for VV&T, and their associated metrics.** Testing often stops because of schedule and cost deadlines, rather than because a technique has been exercised to satisfaction of a technique-specific stopping rule. Studies to determine reasonable, measurable stopping rules are needed. In conjunction with these stopping rules, guidance on the time and cost to budget to accomplish them are needed.

  - COVERAGE CRITERIA

27. **(H-2) Establish integration test coverage criteria.** More quantitative guidance is needed for aiding in determining when sufficient coverage has been obtained during integration testing. For example, what are adequate coverage criteria for stack depth analysis, data interconnectivity, and timing tests?

- **SOFTWARE DEVELOPMENT METHODOLOGIES**

  – EVALUATION

28. **(H-5) Cost-benefit analysis of various software development strategies (fault avoidance).** How do fault avoidance (as opposed to fault tolerance) development strategies such as Clean room, structured analysis, Jackson methodology, compare?

29. **(H-10) Techniques and tools for requirements/specification validation.** Evaluation of existing tools and techniques (e.g., rapid prototyping, requirements languages) is needed. Areas for improving these existing tools and techniques must be identified, as well as identifying new ones. Then the definition, specification, development, and assessment of improvements must be undertaken.

30. **(H-10) Ways to make single version software more reliable.** Tools to develop test cases; can fault tree analysis be used to show dependence of faults and be used to generate test cases? A basis for determining the reliability of software is needed. Work on defining and developing testing strategies is needed.

31. **(L-2) What techniques ensure high levels of programming quality in light of the fact that VV&T has been separated from the coder.** This addresses the psychological issue of complacency among some programmers working on self-correcting software (e.g., n-version, recovery blocks). How can we impress on programmers that it is extremely important they make their code as reliable as possible, rather than rely on error-correcting facilities?

  – PARADIGMS

32. **(H-5) Design for software testability.** Methods are needed for designing software so that it can be tested more effectively and efficiently. Note that in hardware, design faults are not counted as 'errors'; only wearout is counted as an error. With software, design flaws are indeed errors. Also, complexity permitted in hardware is fairly limited. Software complexity, and the shared resource environment in software are big contributors to error. Explore designing for low complexity, etc.

33. (H-5) **Techniques for establishing error containment in software.** Techniques which lead to preventing errors in noncritical portions of code from corrupting critical portions are needed. Also, containment of errors within critical portions of code is needed, to ensure minimum damage/degradation from a given error.

34. (M-10) **Use of formal proof techniques in establishing software integrity.** There are statistical limitations on the estimation of reliability of life-critical systems. To quantify ultra-high reliability using a life testing approach, a prohibitively large number of test cases and/or test specimens must be used. Therefore, there is a growing group in the research community that is advocating the use of alternate approaches to the validation of ultra-reliable systems. One of the most powerful such approaches is formal mathematical proof of correctness. Using this approach, a system is specified in a formal specification language and this specification is refined through a series of increasingly detailed design levels all the way down to actual implementation. At each step of the process, the current level of the design is mathematically proven to be consistent with the previous level.

35. (M-10) **What is the role of software reuse in flight-critical software?.** Can this software be easily reused? What about interfacing reusable modules with software under development? How can 'robustness' of reusable software be measured and conveyed to potential reusers?

36. (M-10) **Collection of a library of reliable modules for reuse.** Research issues include determining if and how software module reuse can benefit flight-critical software development. Specific areas to explore include: 1) identifying functionality (modules) which likely can be reused at less cost than developing new ones, 2) assessing the resulting robustness of the software, and 3) pros and cons of interfacing reused modules with new code under development.

37. (L-10) **Establishing software engineering and assurance criteria and methods for artificial intelligence.** With the increased usage of AI methods, the functional domain of software will expand into more life-critical areas. Adaptations to and expansion of existing

software engineering and assurance criteria will be required by these driving technology changes.

– LANGUAGE ISSUES

38. **([*1])Effect of various languages (Ada, C) on software reliability.** The issues that should be addressed include the structure, capabilities, size, and philosophy (e.g., strong typing versus no type checking) of the language, and the status of the compilers/interpreters available.

– TOOLS

39. **(L-2) Tools and techniques for maintaining, enhancing, and retargeting flight-critical software.** What about old undocumented or poorly documented software? What about software that doesn't have source code anymore? What about the need to retarget software when the underlying hardware it was developed for has become obsolete?

- **CERTIFICATION AND STANDARDS**

40. **(H-2) Determine requirements for testing of tools.**

    (a) Development tools: compilers, linkers, code generators, etc.

    (b) Monitoring tools: coverage analysis and traceability analysis

    (c) Analysis tools: simulators and test case generators

    There is a great emphasis on automating the design, development, and VV&T processes. Automation usually means the use of a tool to do a function. Thus, software is producing and analyzing software. Therefore, there must be requirements and guidelines for the testing of these tools. This area poses some unique challenges when it is considered that the output of a piece of software is another piece of software or data about a piece of software.

[1] As an oversight, this issue was not rated by the Working Group.

41. **(H-10) Institutionalize results of research on software reliability, standards, and guidelines.** It has been said that the user community can lag as much as ten years behind the research community. This gap must be closed in the aerospace community, specifically with respect to software. Ways must be found to get the most important research results into the hands of those that need them most. Government agencies, like NASA and FAA, should play a crucial part in this technology transfer.

42. **(M-10) Certification procedures for products of tools associated with emerging software development techniques.** Given that there are many tools available for the design, development and VV&T of software, it should be possible to formulate certification procedures which take into account the use of these tools.

2.7.3.2 Nine Most Urgent Issues The following list details the results of a post-meeting poll of working group attendees, to determine the few most important issues of the forty two identified.

	VOTING	CONSENSUS	ISSUE	
RANK	SCORE	PRIORITY	NUMBER	ISSUE
1	5.4	H-2	40	CERTIFICATION AND STANDARDS. **Institutionalize the Results of Research on Software Reliability Standards and Guidelines**
2	4.4	H-10	30	SOFTWARE DEVELOPMENT METHODOLOGIES: EVALUATION **Ways to Make Single Version Software More Reliable**
3	3.5	H-5	17	RELIABILITY/AVAILABILITY/ SAFETY ANALYSIS OF SOFTWARE: DEVELOPMENT; VV&T. **Correlating the measured reliability of the software with development strategies and associated VV&T**

RANK	VOTING SCORE	CONSENSUS PRIORITY	ISSUE NUMBER	ISSUE
4	3.3	H-5	3	SOFTWARE ISSUES IN FAULT-TOLERANT SYSTEMS: FAULT-TOLERANT SOFTWARE TECHNIQUES. **Cost-benefit analysis and selection criteria for various fault-tolerant software techniques.**
5	3.4	H-5	8	SOFTWARE ISSUES IN FAULT-TOLERANT SYSTEMS: HARDWARE AND SOFTWARE INTEGRATION ISSUES. **Categorize, validate, and analyze the cost/ benefits of software to manage hardware redundancy**
6	3.3	H-5	25	SOFTWARE TESTING TECHNIQUES AND EFFECTIVENESS: EVALUATION. **Cost-benefit comparison of various testing strategies**
7	3.3	H-5	33	SOFTWARE DEVELOPMENT METHODOLOGIES: PARADIGMS. **Techniques for establishing error containment in software.**
8	2.8	H-5	28	SOFTWARE DEVELOPMENT METHODOLOGIES: EVALUATION. **Cost benefit analysis of various software development strategies (fault-avoidance).**
9	2.7	H-5	7	SOFTWARE ISSUES IN FAULT-TOLERANT SYSTEMS: HARDWARE AND SOFTWARE INTEGRATION ISSUES. **Interaction and impact of fault-tolerant software on hardware redundancy management.**

2.7.3.3 High-Priority Issues The following three lists are subsets of the full list of issues presented in the previous section. These lists highlight the high-priority issues; the first list consists of all H-2 issues; the second, all H-5 issues; and the third, all H-10 issues.

ISSUES REQUIRING 2 YEARS TO COMPLETE

ISSUE NUMBER	ISSUE
1	SOFTWARE ISSUES IN FAULT-TOLERANT SYSTEMS: FAULT-TOLERANT SOFTWARE TECHNIQUES **Definition, properties (robustness, convergence, etc.) and analysis of various voting strategies (N-version, recovery blocks, adaptive, on-line & spares repair, etc.)**
2	SOFTWARE ISSUES IN FAULT-TOLERANT SYSTEMS: FAULT-TOLERANT SOFTWARE TECHNIQUES **Examine the effects of reduced levels of verification, validation, and test (VV&T) on the reliability of fault-tolerant software systems**
15	RELIABILITY/AVAILABILITY/SAFETY ANALYSIS OF SOFTWARE: COMMON MODE (COINCIDENT ERROR) MODELS **Coincident Error model analysis**

ISSUE NUMBER	ISSUE
16	RELIABILITY/AVAILABILITY/SAFETY ANALYSIS OF SOFTWARE: METRICS **Correlating the measured reliability of the software with software metrics.**
18	RELIABILITY/AVAILABILITY/SAFETY ANALYSIS OF SOFTWARE: SAFETY AND RISK **Hazard Analysis and Failure Modes and Effects Analysis**
21	DATA COLLECTION: FIELDED SYSTEMS **Definition, specification, and collection of reliability data**
22	DATA COLLECTION: FIELDED SYSTEMS **Collection of metric data on systems for correlation with measured reliability**
24	SOFTWARE TESTING TECHNIQUES AND EFFECTIVENESS: EVALUATION **Analysis of error classes and their associated functional mapping, and appropriate techniques for detections**
27	SOFTWARE TESTING TECHNIQUES AND EFFECTIVENESS: COVERAGE CRITERIA **Establish integration test coverage criteria**
40	CERTIFICATION AND STANDARDS **Determine requirements for testing of tools**

ISSUES REQUIRING 5 YEARS TO COMPLETE

ISSUE NUMBER	ISSUE
3	SOFTWARE ISSUES IN FAULT-TOLERANT SYSTEMS: FAULT-TOLERANT SOFTWARE TECHNIQUES **Cost-benefit analysis and selection criteria for various fault-tolerant software techniques**
4	SOFTWARE ISSUES IN FAULT-TOLERANT SYSTEMS: FAULT-TOLERANT SOFTWARE TECHNIQUES **Performance, reliability, and availability analysis of real-world N-version systems**
7	SOFTWARE ISSUES IN FAULT-TOLERANT SYSTEMS: HARDWARE AND SOFTWARE INTEGRATION ISSUES **Interaction and impact of fault-tolerant software on hardware redundancy management**
8	SOFTWARE ISSUES IN FAULT-TOLERANT SYSTEMS: HARDWARE AND SOFTWARE INTEGRATION ISSUES **Categorize, validate, and analyze the cost-benefits of software to manage hardware redundancy**
17	RELIABILITY/AVAILABILITY/SAFETY ANALYSIS OF SOFTWARE: METRICS **Correlating the measured reliability of the software with development strategies and associated VV&T**

ISSUE NUMBER	ISSUE
21	DATA COLLECTION: FIELDED SYSTEMS **Definition, specification, and collection of reliability data**
23	DATA COLLECTION: FIELDED SYSTEMS **Definition and execution of lab experiments**
25	SOFTWARE TESTING TECHNIQUES AND EFFECTIVENESS: EVALUATION **Cost-benefit comparison of various testing strategies**
28	SOFTWARE DEVELOPMENT METHODOLOGIES: EVALUATION **Cost-benefit analysis of various software development strategies (fault avoidance)**
32	SOFTWARE DEVELOPMENT METHODOLOGIES: PARADIGMS **Design for software testability**
33	SOFTWARE DEVELOPMENT METHODOLOGIES: PARADIGMS **Techniques for establishing error containment in software**

ISSUES REQUIRING 10 YEARS TO COMPLETE

ISSUE NUMBER	ISSUE
5	SOFTWARE ISSUES IN FAULT-TOLERANT SYSTEMS: FAULT-TOLERANT SOFTWARE TECHNIQUES **Guidelines for the development of N-version programs to minimize common mode (coincident) failures**
29	SOFTWARE DEVELOPMENT METHODOLOGIES: EVALUATION **Techniques and tools for requirements/ specification validation**
30	SOFTWARE DEVELOPMENT METHODOLOGIES: EVALUATION **Ways to make single version software more reliable**
41	CERTIFICATION AND STANDARDS **Institutionalize results of research on software reliability, standards, and guidelines**

2.7.4 References

"*Fault Tolerance: Principles and Practice*," Anderson, Thomas and Lee, P.A.; Prentice-Hall, 1984.

"*Experimentation in Software Engineering*," Basili, V. R., Selby, R. W., Hutchens, D. H.; IEEE Transactions on Software Engineering, IEEE Computer Society Press, Silver Springs, MD, Vol. SE-12, No. 7, July 1986.

"*Structured Analysis and System Specification*," De Marco, Tom; Yourdon, Inc., New York, NY, 1979.

"*A Theoretical basis for the Analysis of Multiversion Software Subject to Coincident Errors*," Eckhardt, D. E. and Lee, L. D.; IEEE Transactions on Software Engineering, Vol. SE-11, No. 12, December 1985, pp. 1511-1517.

"*Results of Software Error-Data Experiments*," Finelli, G. B., Proceedings of AIAA/AHS/ASEE Meeting on Aircraft Design and Operations, September 1988.

"*Software Metrics: Establishing a Company-Wide Program*," Grady, R. B. and Caswell, D. L.; Prentice-Hall, Englewood Cliffs, NJ, 1987, p. 288.

"*Software Reliability. Measurement, Prediction, Application*," Musa, J. D., Iannino, A., Okumoto, K.; McGraw-Hill, New York, NY, 1987.

"*Software Considerations in Airborne Systems and Equipment Certification*," Radio Technical Commission for Aeronautics Secretariat; Washington, D.C., No. DO-178a, March, 1985.

"*System Safety Engineering and Management*," Roland, H. E. and Moriarty, B.; John Wiley and Sons, Inc., 1983.

"*Cleanroom Software Development: An Empirical Evaluation*," Selby, R. W., Basili, V. R., Baker, F. T.; IEEE Transactions on Software Engineering, IEEE Computer Society Press, Silver Springs, MD, Vol. SE-13, No. 9, September 1987.

2

"*Software Reliability: A Historical Perspective,*" Shooman, M. L.; IEEE Transactions on Reliability, Vol. R-33, No. 1, April 1984, pp. 48-55.

"*Software Engineering Design/Reliability/Management,*" Shooman, M. L.; McGraw-Hill Book Co., 1983.

2.8 Working Group Report on Flight Test

Chair: Jerry Doniger, Lear Astronics
Co-Chair: David Holmes, NASA-LaRC
Coordinator: Ed Withers, RTI

C·2

2.8 Flight Test

2.8.1 Introduction and Overview

2.8.1.1 Motivation The Flight Testing working group was established to determine the areas in which flight testing can be used to verify the performance and safety aspects of avionic systems that cannot be adequately accomplished by analysis, simulation, and laboratory testing, as well as the ways that flight test results can be applied to improve these other types of testing. Flight testing is expensive compared to simulations, but simulations do not provide sufficient confidence to be accepted without confirmation by flight testing. It may be possible to use flight testing to improve the confidence level of simulations and thus reduce the cost of testing and approving new systems.

2.8.1.2 Goals The goals of the flight testing working group were to identify research areas that would help industry reduce the cost and improve the reliability of digital system design by appropriate use of flight testing. Not only were research areas identified, but possible research directions were generated that might aid in accomplishing these goals. The flight testing session also developed a list of areas in which flight testing is important. This list is separated according to system integrity issues and system functionality issues. In addition, a list of possible candidate experimental test systems was generated that would be useful in testing and improving present tools and methods. This list of candidate systems should serve as a starting point from which a research project could be developed.

2.8.1.3 Industry Needs Industry has several needs for research in support of flight test activities, including standardized flight critical design methodologies and verification tools that NASA-LaRC should be able to provide. These capabilities must allow for design innovation and the continuing improvements in the verification methods and tools themselves. Models need to be developed for many environmental conditions that are poorly understood, such as wind shear, turbulence, lightning, and atmospheric effects on radio frequency (RF) and electro-optical (E-O) guidance sensors. In turn, flight testing can be used to support research in other technological areas, as in developing methodologies for building digital systems and the tools for implementing these methodologies.

A method for using flight testing to improve design techniques and tools was discussed by the working group and is shown schematically in Figure 4 presented below. Present techniques and tools are used to design and model a system, then testing (including flight testing) is conducted to verify performance and integrity. The differences between the model predictions and the actual performance are used as a basis for improvements to methods and tools. The process is then repeated to continue improvements in methods and tools as technology improves and new problems arise. Much of the testing could be done by industry during development of new systems, with the data collected during this testing being stored and organized by NASA-LaRC for later use by all of industry.

2.8.1.4 Industry Support Industry presently conducts flight testing of all systems that are to be certified and released for use; government agencies monitor, review, and verify test data to decide whether or not to certify a system for in-service use. Much of the flight testing being conducted may be redundant and the methods of testing are often re-invented by each company conducting flight tests. If NASA-LaRC were to develop a general data base of design verification methodologies and tools that was available to all organizations conducting flight testing, it could reduce the cost of development of new systems. The initial data base may be the current industry flight test simulation / correlation methods and tools. For the proposed data base to be useful, there would have to be industry and certifying agency support to not only contribute data to the system, but to utilize the data stored by NASA-LaRC in future developments as well.

2.8.1.5 Links to Other Verification Methods An obvious link of flight testing to other design verification methods is the link to simulation. Flight testing can be used to improve the models for later simulations, and also to confirm that present models are accurate. Simulation and flight testing must be used in complementary and efficient ways to improve the techniques for modeling and testing systems. With the current rate of technology growth, testing and modeling methods and tools must be improved to reduce development costs while ensuring that the performance and integrity requirements of the systems are met.

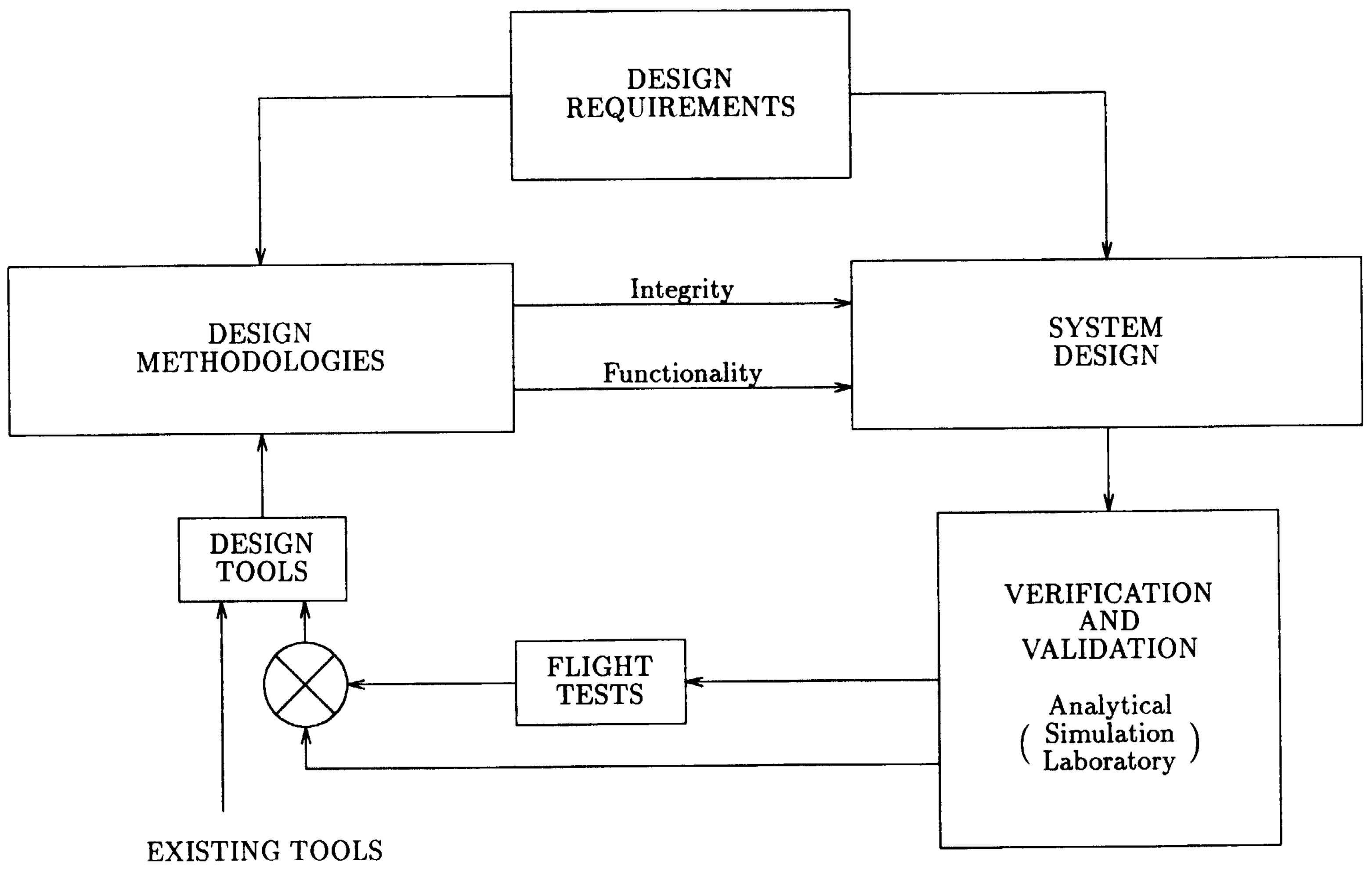


Figure 4: Proposed Cycle for Optimization of Digital System Development Methods and Tools

2.8.2 Critical Issues

Of the issues discussed during the working group, the following were of considerable interest:

- Improved design, test, evaluation, and verification processes that are used broadly by industry and government,
- Environmental information,
- The roles of testing versus simulation,
- System Functionality, and
- System Integrity.

Each of these issues is highly interrelated, and thus one or more of them appears to some extent among the research needs listed later. Research ideas were discussed that might lead to improvements in each of these areas. One of the major issues discussed was the ways to verify the tools being used during development and ways that feedback from the flight testing phase might be used to improve these tools. Some care needs to be taken to assure that development and testing are done in ways that may later be used to improve each other.

2.8.3 Research Needs

2.8.3.1 Near-Term Needs

Verification Methodologies NASA-LaRC should develop new methodologies, and improve existing ones, for verifying the performance and integrity of flight-critical digital systems that could then be used as the industry standards. These verification methodologies could utilize the advantages of flight testing in the verification process. Some possible areas to which flight testing could be applied include comparing the reliability of single version software systems versus redundant software systems, and the reliability of similar systems versus dissimilar systems in a redundant software environment. NASA-LaRC should not be involved with the development of specific flight-critical systems; rather, NASA-LaRC should develop tools for building flight-critical systems.

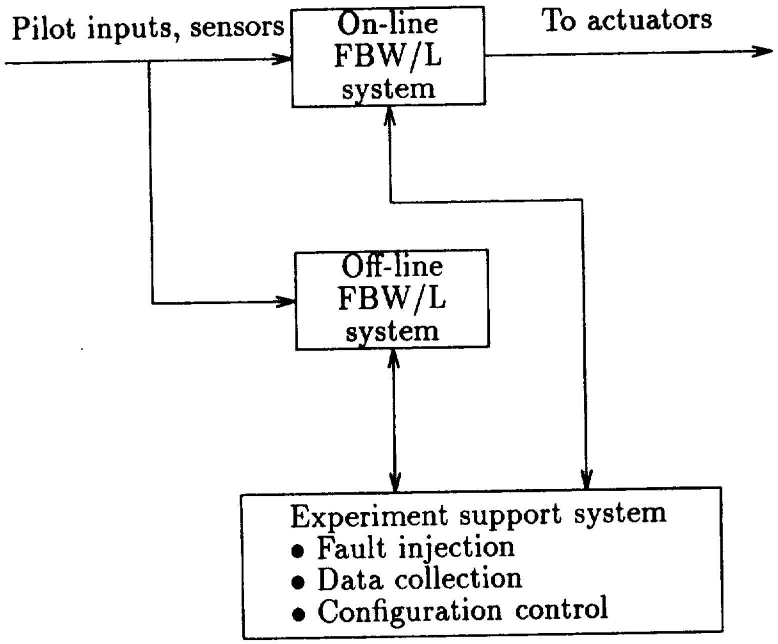


Figure 5: Experimental FBW/L System Test Bed

Verification Tools The new, or improved, verification methodologies will require new, or improved, tools to aid in their use. At a minimum, some standard type of data base needs to be defined such that the design tools will work together to aid design optimization, information transfer, and maintenance from one organization to another.

Test Bed A general-purpose flight test bed needs to be developed that would allow the verification methods and tools themselves to be verified. The test bed would also allow new systems to be operated on-line or off-line with existing on-board systems. For example, if a new flight critical system or system element is developed, it could be installed on the test bed aircraft in parallel to the present system. Testing would then reflect "live" conditions without risking the aircraft or crew on an untested system. Figure 5 shows schematically the organization of a test bed aircraft. The test bed for parallel systems also allows for testing of systems in degraded modes without risk to aircraft or crew.

Designing for Testability From the beginning of the design of a system, provisions need to be made for later testing of the system. These provisions include additional windows into systems without impacting performance and safety, but which may improve producibility and in-service maintenance.

2.8.3.2 Longer Term Needs

Environmental Models NASA-LaRC should develop, verify, and upgrade models of environmental conditions that would then be used by industry in developing new sensors and systems. NASA-LaRC would also serve as the "clearinghouse" for this information as well as any feedback provided by industry using these models.

NASA's role as a "clearinghouse" would include the functions of collecting and storing data generated by various members of the industrial community. Some of the types of information in this data base might include: lightning, High Energy Radio Frequency (HERF) effects, turbulence, wind shear, precipitation, etc. NASA-LaRC would also keep the most recent versions of design, verification, and testing tools and would also serve to insure that all newly developed tools met certain standards and thus could communicate with each other. This data base of information and these tools would be available to all members of the industrial community for use in developing new flight-critical digital systems.

How much testing Just as verification methodologies need to be developed or improved, testing methodologies need to be improved and investigated. The major issues raised concerning flight testing are concerning which systems / elements are to be tested, when sufficient testing has been accomplished to satisfy regulatory and operational requirements, and to quantify the level of confidence available in the test results.

2.8.4 List of System Functionality Issues

(Not Prioritized)

System

- Redundancy Management / Failure Detection, Isolation, and Reconfiguration (FDIR)
- Lightning/ HERF
- Design Errors / Model Accuracy
- Hardened Stability Augmentation

Software

- Ada / Real Time Issues
- Dissimilar Software
- Automated Development Tools

Computer

- High Reliability Architectures
- Timing Tolerances
- Transient Recovery Capability
- Technology (Electronic, Fluidic, etc.)
- Similar / Dissimilar

Sensors

- Solid State
- Skewed
- Analytical Redundancy

Actuators

- Smart Actuation (Electrohydrostatic (E-H), Electromechanical (E-M), Electrical or Mechanical? (EOM))
- Integrated Actuators (Local Power)
- Fault Characteristics

Communications

- Optical
- Protocol

Power (Electrical / Hydraulic)

- Centralized / Distributed
- Dissimilarity
- Uninterruptible / Redundancy? Management

Pilot / Vehicle Interface

- Head Up / Down Functional Displays
- Reconfiguration
- Controller Operation
- Caution / Warning Displays

2.8.5 List of System Integrity Issues

(Not Prioritized)

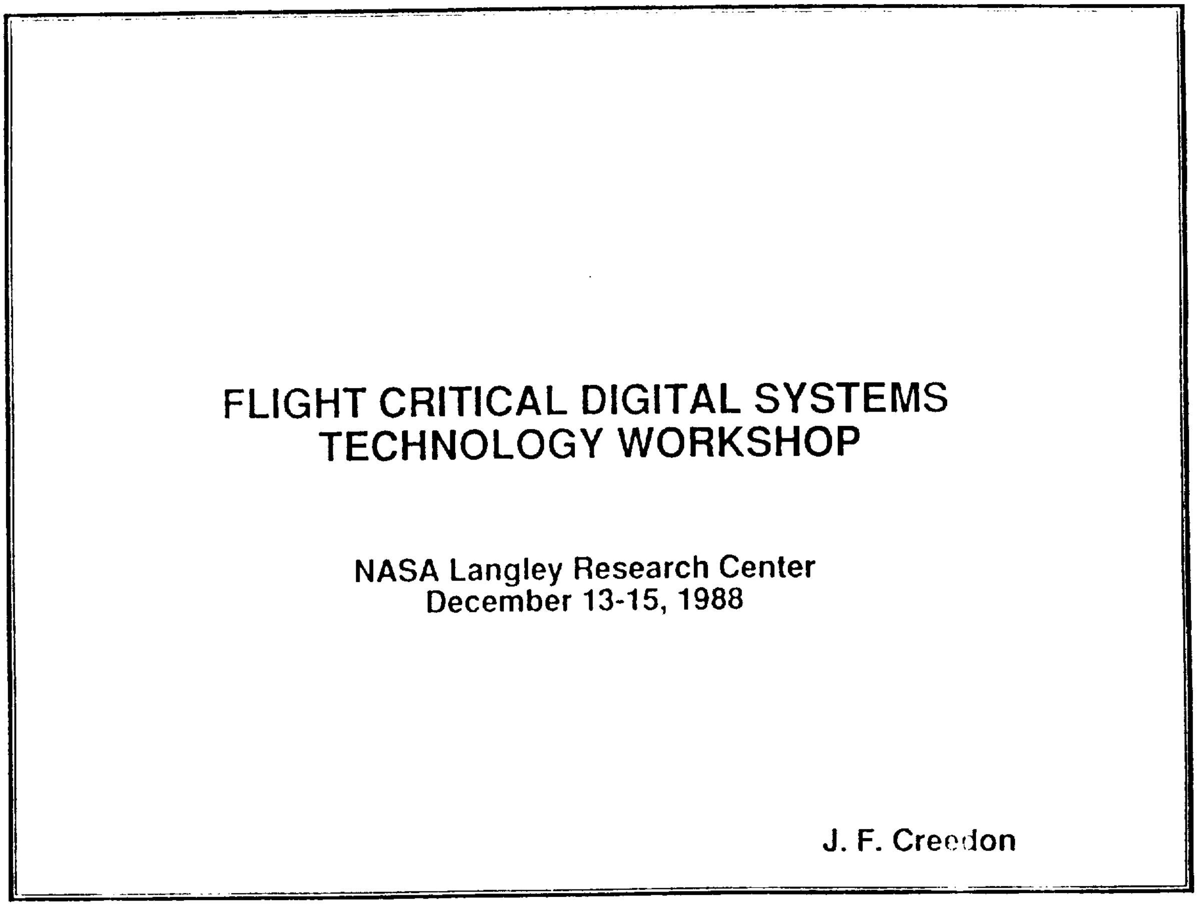
- Command / Stability Augmentation Control Law Concepts
- Envelope Limiting
- Autoflight Control Law Concepts
- Flutter Suppression
- Autonomous Landing / Obstacle Avoidance
- Autonomous Windshear Prediction
- Engine-Matching Modes
- State Estimation / Analytical Redundancy
- Pilot / Vehicle Interfaces
  - Displays (Primary, Caution / Warning)
  - Controllers, Data Entry
  - Procedures

2.8.6 List of Candidate Experimental Systems

- Fly By Wire / Light (FBW / L) Control System
- Crew Station Display System
- Autonomous Landing System
- Windshear Prediction System
- Flutter Suppression System

APPENDIX A

Overview Viewgraphs for Keynote Speakers



FLIGHT CRITICAL DIGITAL SYSTEMS TECHNOLOGY WORKSHOP

NASA Langley Research Center
December 13-15, 1988

J. F. Creedon

NASA LANGLEY RESEARCH CENTER

Founded in 1917

- First civil aeronautical research laboratory

Capabilities

- People: 2837 civil servants
- Facilities: $1.5 billion replacement value

Mission: Aeronautics Space Research

Aeronautics

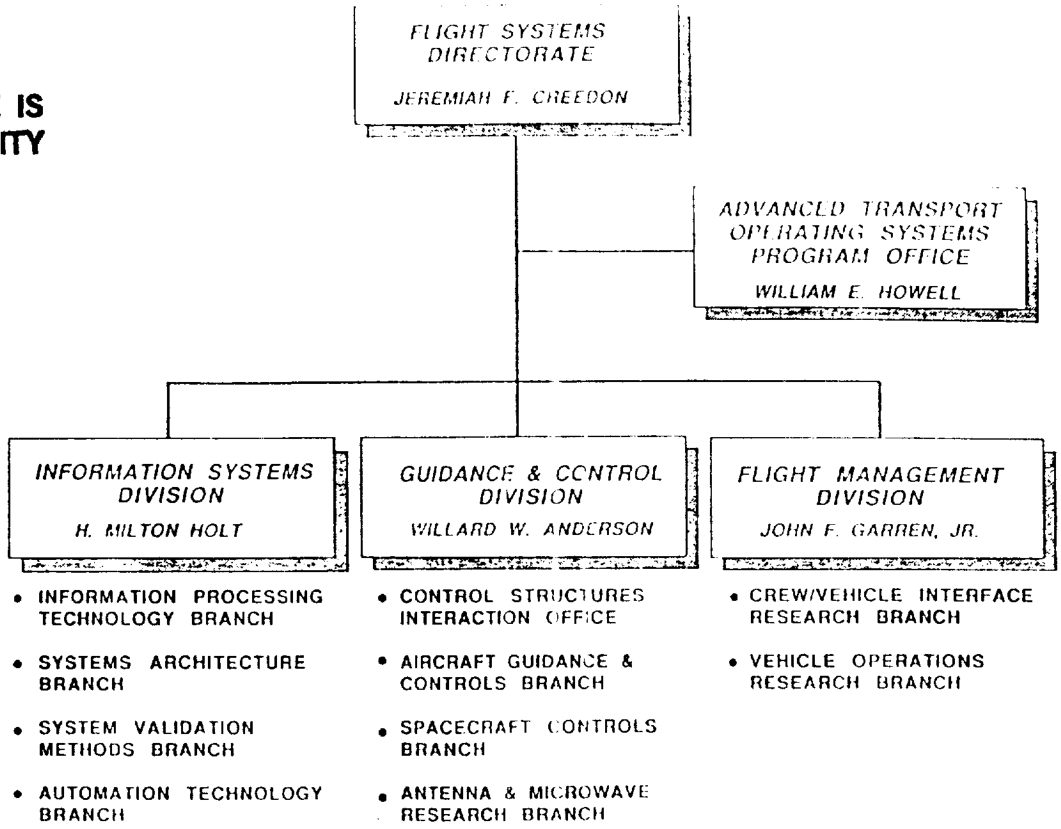
- Full range of disciplines
- 60% of resources

Space

- Selected disciplines & atmospheric science
- 40% of resources

THE PURPOSE OF NASA LANGLEY RESEARCH CENTER

- To perform innovative aerospace research that is relevant to national needs and agency goals
- To transfer the research results to the user communities in a timely manner
- To provide development support to other government agencies, industry, and other NASA centers



FLIGHT SYSTEMS DIRECTORATE
JEREMIAH F. CREEDON

ADVANCED TRANSPORT OPERATING SYSTEMS PROGRAM OFFICE
WILLIAM E. HOWELL

INFORMATION SYSTEMS DIVISION
H. MILTON HOLT

- INFORMATION PROCESSING TECHNOLOGY BRANCH
- SYSTEMS ARCHITECTURE BRANCH
- SYSTEM VALIDATION METHODS BRANCH
- AUTOMATION TECHNOLOGY BRANCH

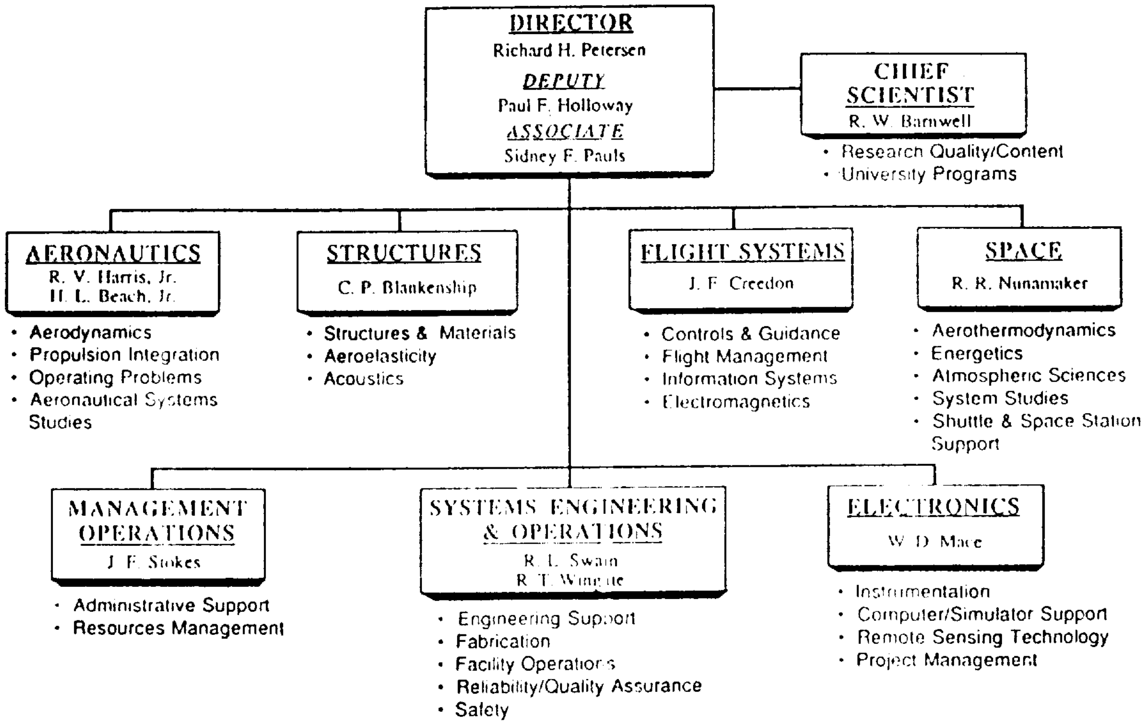
GUIDANCE & CONTROL DIVISION
WILLARD W. ANDERSON

- CONTROL STRUCTURES INTERACTION OFFICE
- AIRCRAFT GUIDANCE & CONTROLS BRANCH
- SPACECRAFT CONTROLS BRANCH
- ANTENNA & MICROWAVE RESEARCH BRANCH

FLIGHT MANAGEMENT DIVISION
JOHN F. GARREN, JR.

- CREW/VEHICLE INTERFACE RESEARCH BRANCH
- VEHICLE OPERATIONS RESEARCH BRANCH

LANGLEY RESEARCH CENTER



DIRECTOR
Richard H. Petersen
DEPUTY
Paul F. Holloway
ASSOCIATE
Sidney F. Pauls

CHIEF SCIENTIST
R. W. Barnwell

- Research Quality/Content
- University Programs

AERONAUTICS
R. V. Harris, Jr.
H. L. Beach, Jr.

- Aerodynamics
- Propulsion Integration
- Operating Problems
- Aeronautical Systems Studies

STRUCTURES
C. P. Blankenship

- Structures & Materials
- Aeroelasticity
- Acoustics

FLIGHT SYSTEMS
J. F. Creedon

- Controls & Guidance
- Flight Management
- Information Systems
- Electromagnetics

SPACE
R. R. Nunamaker

- Aerothermodynamics
- Energetics
- Atmospheric Sciences
- System Studies
- Shuttle & Space Station Support

MANAGEMENT OPERATIONS
J. F. Stokes

- Administrative Support
- Resources Management

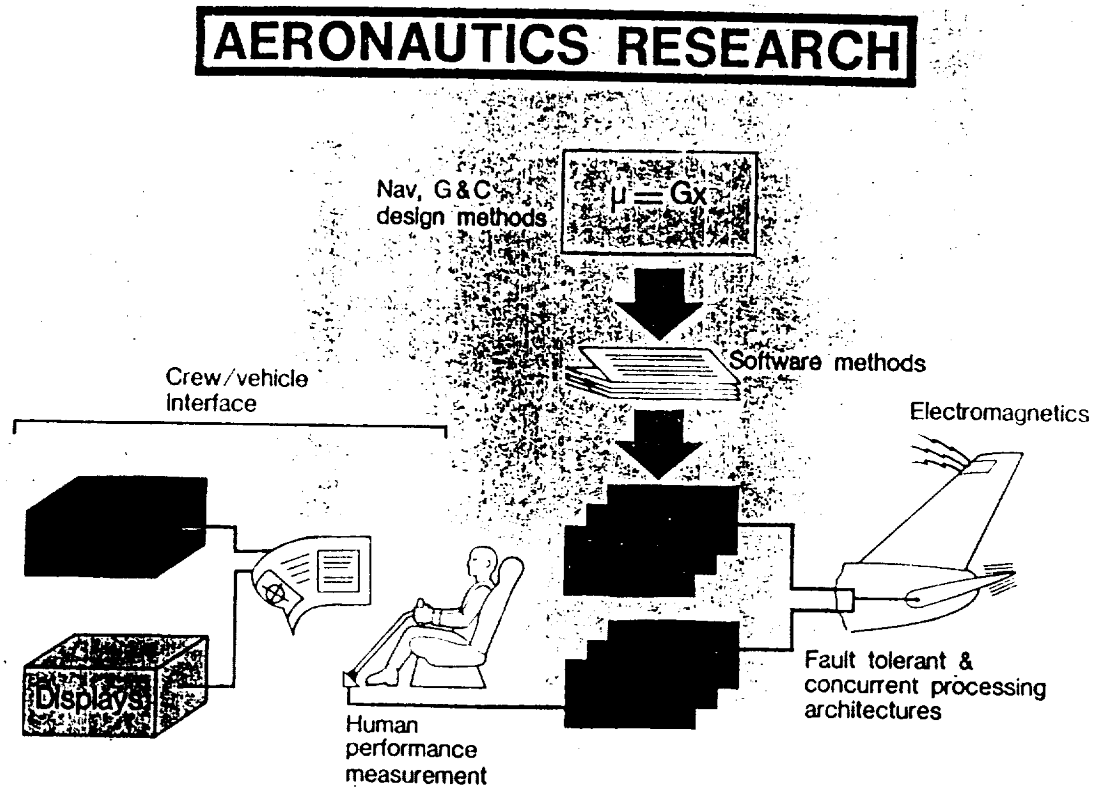
SYSTEMS ENGINEERING & OPERATIONS
R. L. Swain
R. T. Wingate

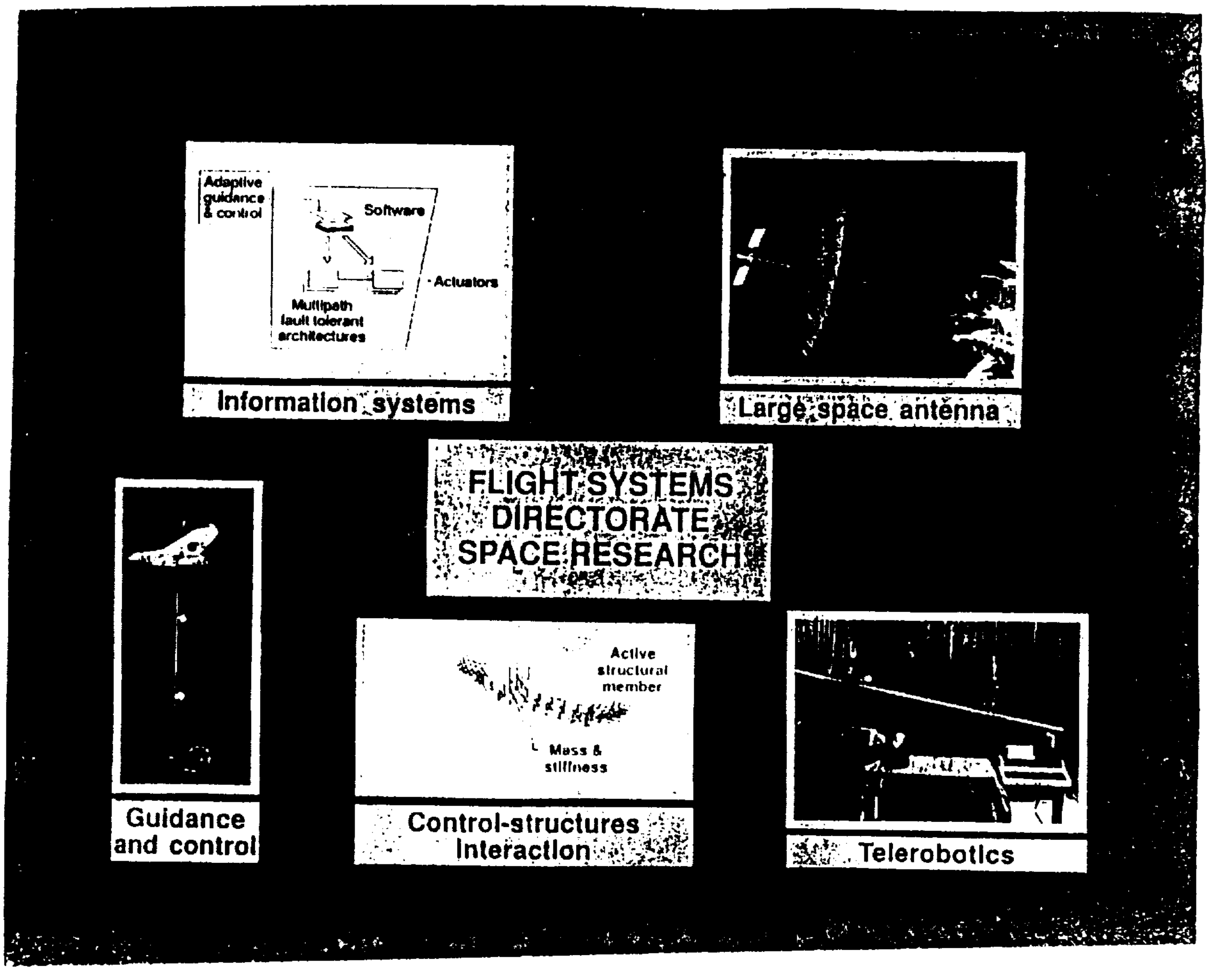
- Engineering Support
- Fabrication
- Facility Operations
- Reliability/Quality Assurance
- Safety

ELECTRONICS
W. D. Mace

- Instrumentation
- Computer/Simulator Support
- Remote Sensing Technology
- Project Management

AUGUST 1987


AERONAUTICS RESEARCH
Nav, G&C design methods
μ = Gx
Software methods
Crew/vehicle interface
Electromagnetics
Displays
Fault tolerant & concurrent processing architectures
Human performance measurement

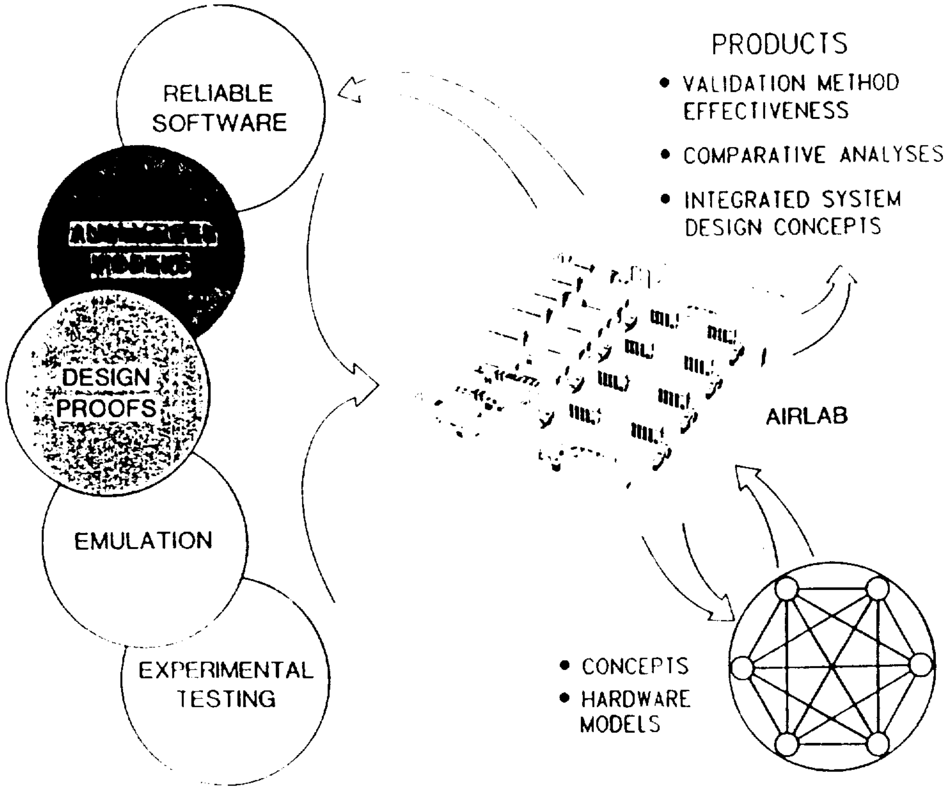

Adaptive guidance & control
Software
Actuators
Multipath fault tolerant architectures
Information systems
Large space antenna
FLIGHT SYSTEMS DIRECTORATE SPACE RESEARCH
Guidance and control
Active structural member
Mass & stiffness
Control-structures interaction
Telerobotics

INFORMATION SYSTEMS GOALS

•
•
•

- To generate architectures, concepts, design methods, and integrated design tools for information processing systems required by future aircraft and spacecraft applications.
- To develop concepts, approaches, and methods which increase the performance and reliability and decrease the cost of applications and systems software for ground and flight systems.
- To provide analytical methods, assessment techniques, experimental methodologies, and the AIRLAB facility for the evaluation and validation of fault-tolerant, concurrent processing, and distributed computer systems and software for spacecraft and aircraft applications.

FAULT-TOLERANT SYSTEMS RESEARCH



PRODUCTS

- VALIDATION METHOD EFFECTIVENESS
- COMPARATIVE ANALYSES
- INTEGRATED SYSTEM DESIGN CONCEPTS

- CONCEPTS
- HARDWARE MODELS

FLIGHT CRITICAL DIGITAL SYSTEMS TECHNOLOGY WORKSHOP

OBJECTIVES

Identify research issues which must be resolved

Define level of analysis experimentation and demonstration required for acceptance of results

OUTPUT:

Workshop document (including consensus on most important issues)

FOLLOWED BY:

Assessment of our program vs. identified needs

Feedback to industry on program

FLIGHT CRITICAL DIGITAL SYSTEMS TECHNOLOGY WORKSHOP

PROCESS

Overview presentation to provide a context for the workshop

Working Group Sessions: Aero and Space Requirements
Design for Validation
Failure Modes
System Modeling
Reliable Software
Flight Test

Some Thoughts On Flight Critical Systems

Tom Cunningham

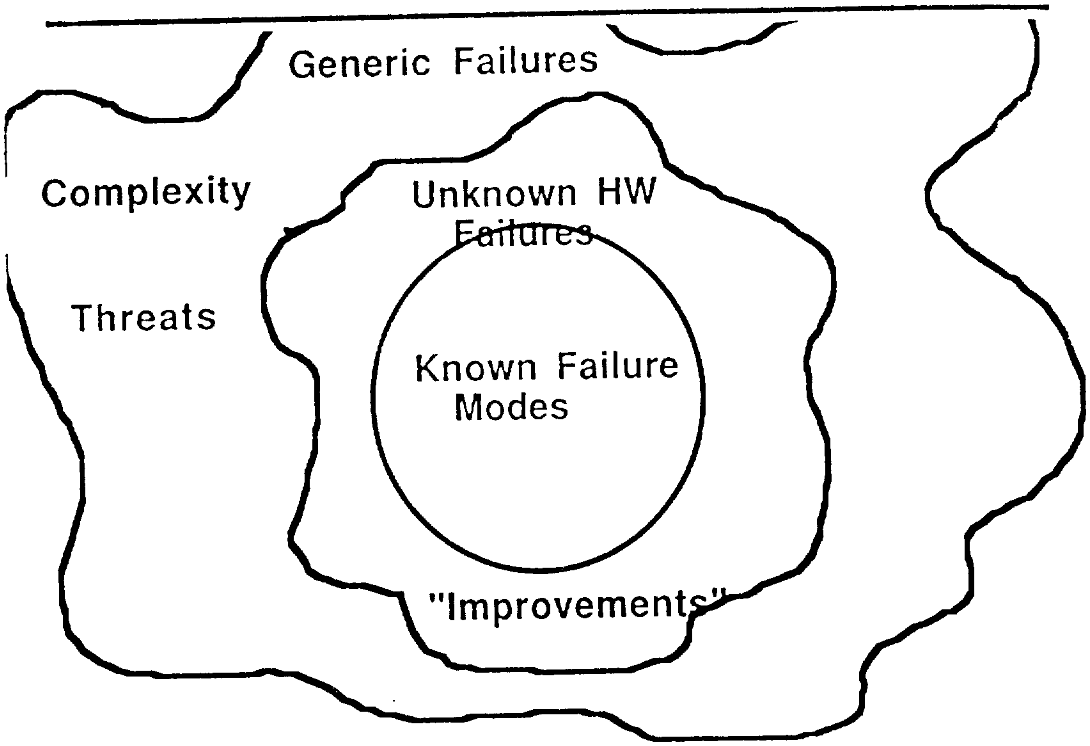
Honeywell Systems & Research Center

December 13, 1988

with help from:

Kevin Driscoll
Gautham Ramohalli
Russ Hendrick
Larry Yount
Randy Gaylor
John Weyrauch

The Coverage Umbrella Must Be Big



Complexity

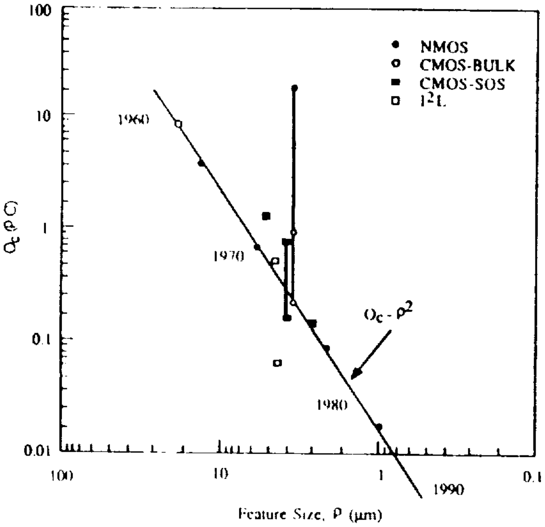
- Functional Demands
  - Reality of FBW, WLA, RSS, Envelope Limiting...
  - More functions under the fault tolerant umbrella, e.g., VMS
  - Complex feedback mechanizations (are they necessary?)
- Attempts at Safety
  - "Cover every conceived failure"
  - "If two channels are better than one, why not ten ?"
- Hardware capability
  - VLSI complexity
  - New sensors and actuators

PRECEDING PAGE BLANK NOT FILMED

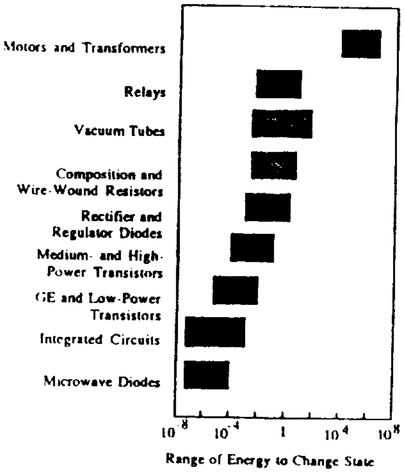
Threats

- EME
- Lightning

The digital electronic circuits used in modern FCSs are vulnerable to upset by decreasing levels of disturbance.



Critical charge for upset plotted as a function of feature size. There is surprisingly little dependence on device technology.

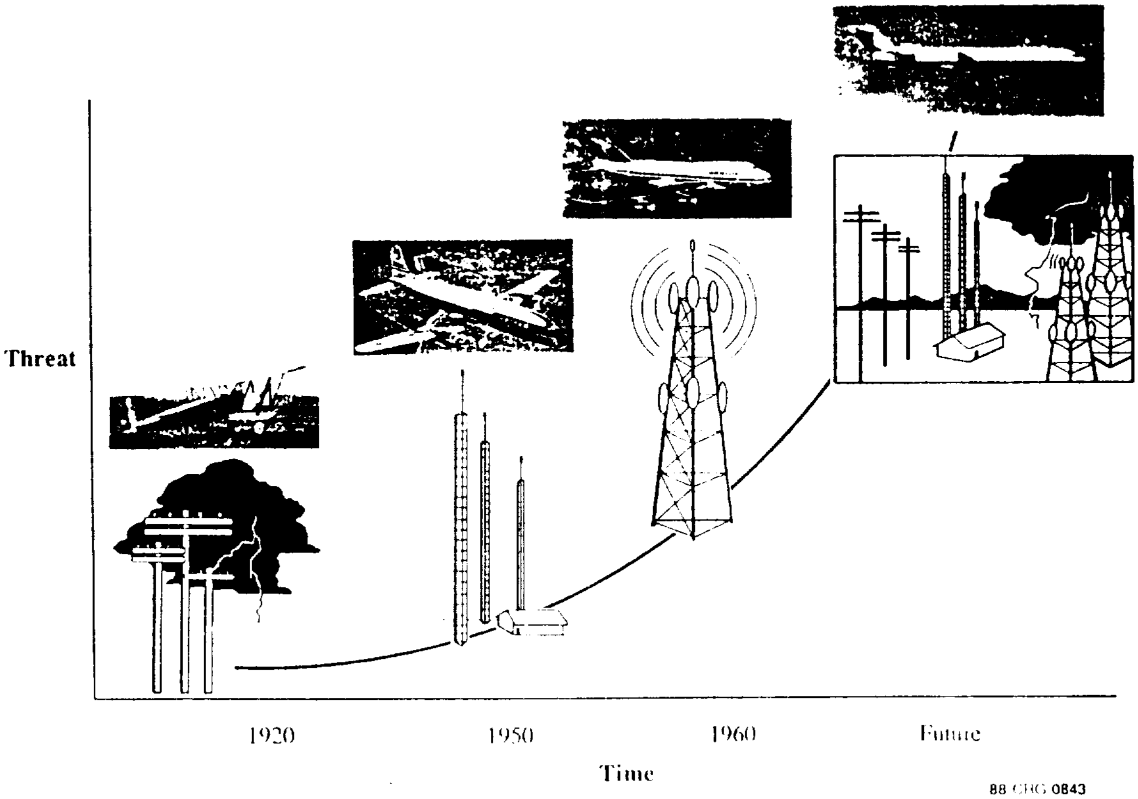


Integrated circuits are among the electronic components most sensitive to EMP. Source: *DNA EMP Awareness Course Notes*

88-CHG 0872

ORIGINAL PAGE IS
OF POOR QUALITY

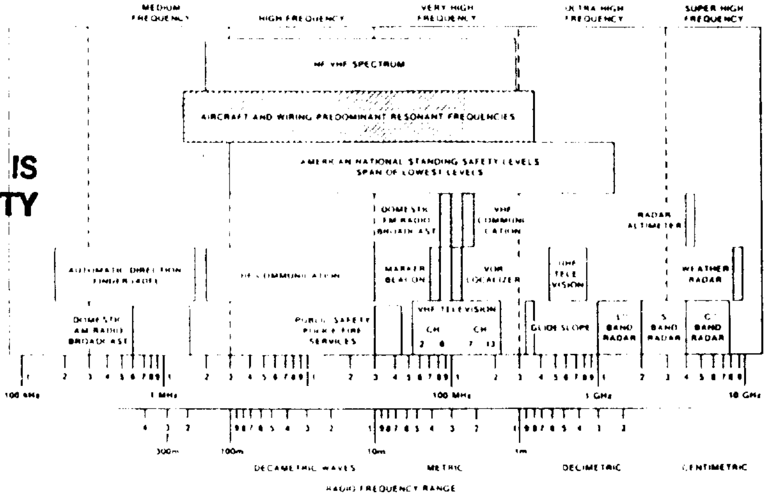
The EME threat is also increasing.



88 CHG 0843

RF Sources

ORIGINAL PAGE IS
OF POOR QUALITY



ELECTROMAGNETIC ENVIRONMENT

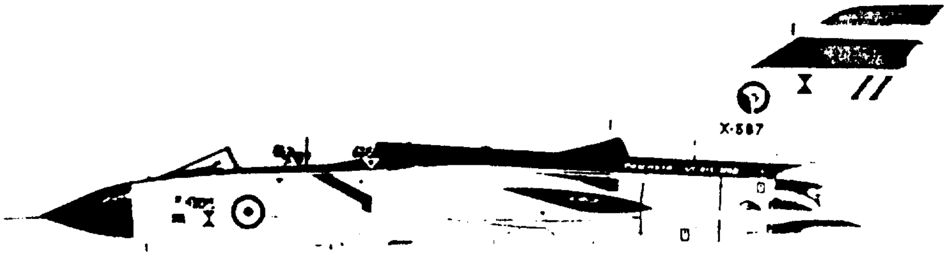
FREQUENCY		AVERAGE FIELD STRENGTH V/m	PEAK FIELD STRENGTH V/m
10 kHz	3 MHz	100	100
3 MHz	30 MHz	1000	1000
30 MHz	100 MHz	100	100
100 MHz	200 MHz	200	3000
200 MHz	1 GHz	2000	6000
1 GHz	2 GHz	2000	14000
2 GHz	8 GHz	400	14000
8 GHz	10 GHz	2000	14000
10 GHz	40 GHz	1000	6000

MANMADE RADIO FREQUENCY ENVIRONMENT

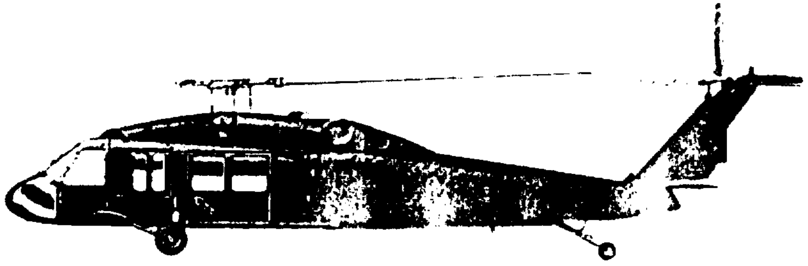
THREAT	INTRA/INTERSYSTEM ELECTROMAGNETIC INTERFERENCE	STATIC DISCHARGES	LIGHTNING	HIGH-ALTITUDE ELECTROMAGNETIC PULSE (HEMP)
THREAT EXPOSURE	Local Usually Antenna Related	local Particle Related	local Cloud Related	Regional Nuclear Burst
EFFECTS	Induced Direct	Induced	Direct Induced	Induced
CRITICALITY	Flight Safety	Mission	Flight Safety	Flight Safety
THREAT SPECTRUM	Broadband Up to 100GHz	Broadband Up to 100 MHz	Broadband Up to 100 MHz	Broadband Up to 100 MHz
PROTECTION MEASURES	Shielding Filtering Redundancy Cable/Equip. Placement Fiber Optics Channel Recovery	Discharges EMI Techniques. Channel Recovery	Diverters Surge arresters EMI Techniques Channel Recovery	EMI Techniques Channel Recovery
MIL-STDS	MIL-E-6051 MIL-STD-461/462 MIL-B-5087	MIL-E-6051 MIL-B-5087	MIL-E-6051 MIL-B-5087 MIL-STD-1757	NONE

A new hazard is now facing commercial aviation.

Military aircraft have been lost due to this problem.

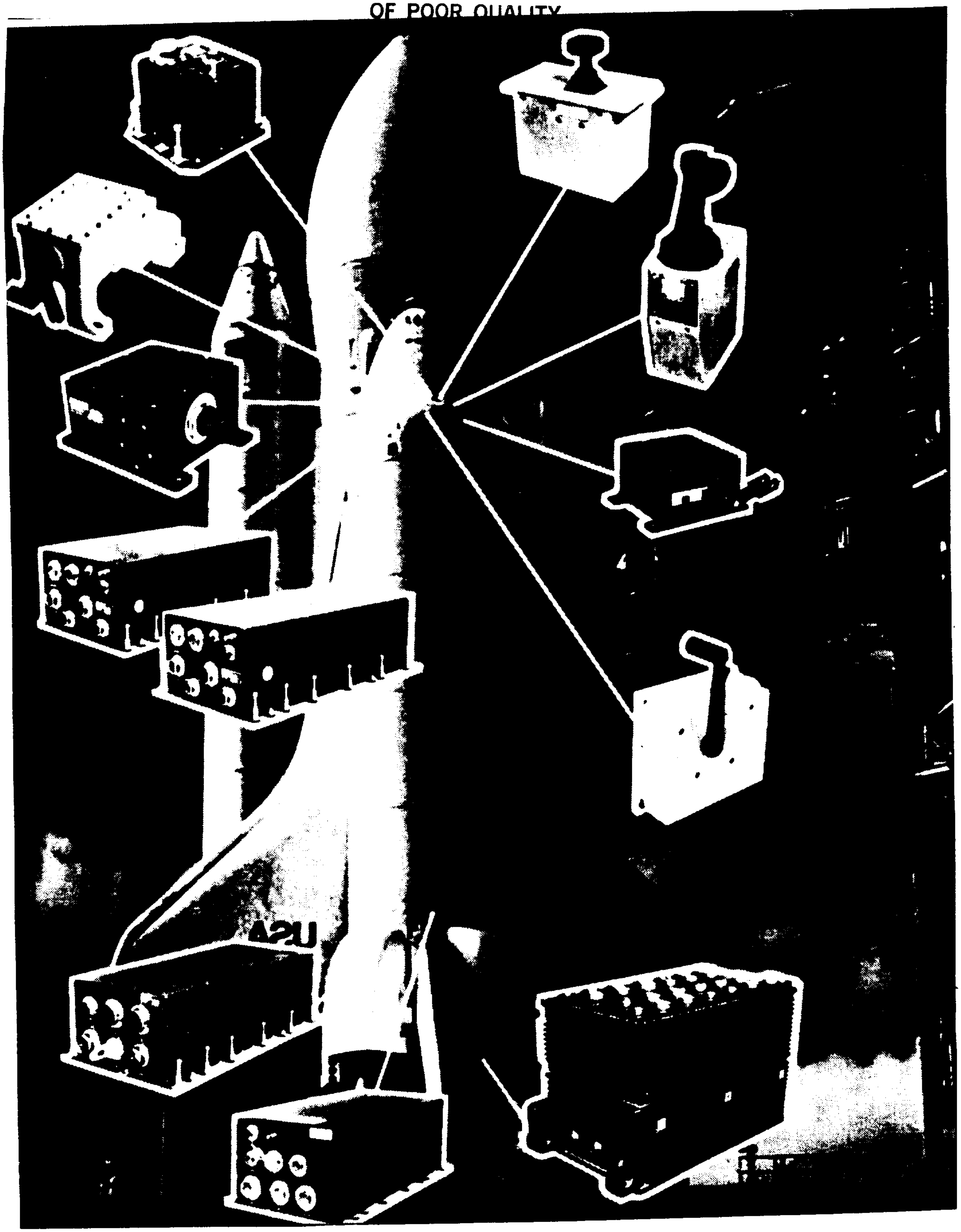


Tornado Fighter

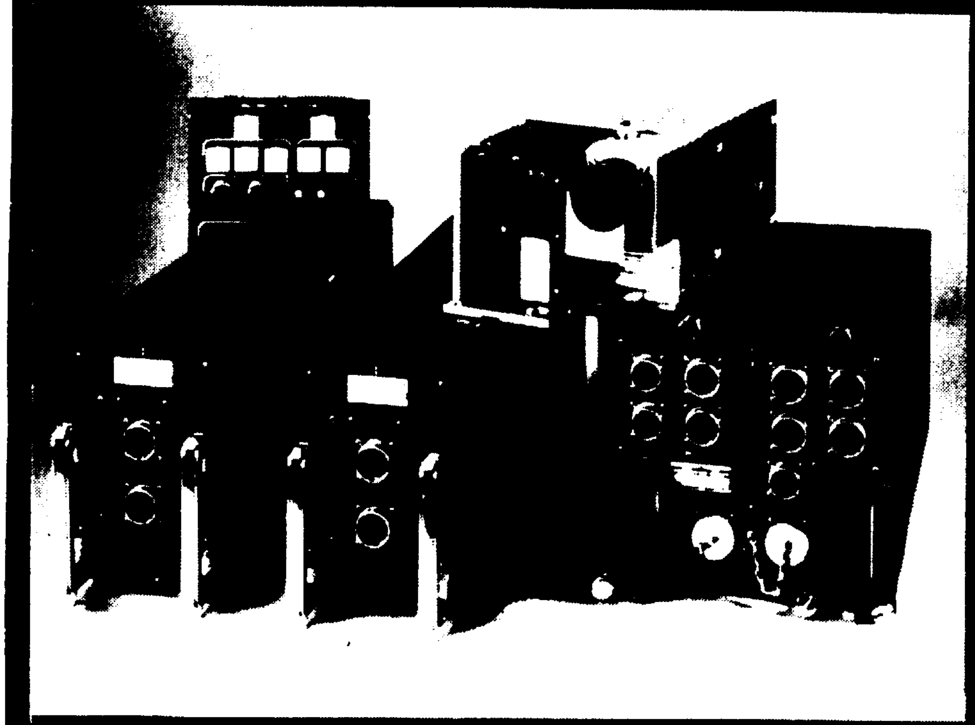


Black Hawk Helicopter

ORIGINAL PAGE IS
OF POOR QUALITY


A2U

DIGITAC



CUSTOMER: U.S. AIR FORCE

SYSTEM CHARACTERISTICS

- FULL AUTHORITY
- DIGITAL (HDP-301)
- DUAL (FAIL-SAFE)
- WIRE/FIBER OPTIC 1553
- ASSEMBLY LANGUAGE

SYSTEM STATUS

OPERATIONAL AT EDWARDS AFB
FLIGHT TEST PILOT SCHOOL

ORIGINAL PAGE IS
OF POOR QUALITY

X-29A FLIGHT CONTROL SYSTEM

CUSTOMER: GRUMMAN/DARPA/AIR FORCE/NASA

SYSTEM CHARACTERISTICS

- FLY-BY-WIRE
- DIGITAL (HDP-5301)
- TRIPLE-CHANNEL (FAIL-OP)
- DIGITAL REVERSION MODE
- ANALOG BACKUP
- ASSEMBLY LANGUAGE

STATUS

- HARDWARE DELIVERED
- FIRST FLIGHT LATE 1984

Software Concerns

- Too many design errors are blamed on "Software"
- Coding errors have not presented problems
- Ada features are of concern
- Proof of software is important:
  - Sabotage
  - "Improvements"
  - Security

Some "Software" Errors

System	Error	Design/ Algorithm	Code	Compiler	No Error	Other
JA-37 Viggen	WOW Switch not properly engaged	√				
STS-1 "Bug heard round the world"	Syncronization of computers				√	
DIGITAC	ABS/-1/ = -1					√
Apache LHX Demo	Compiler interp. of 180° to -180°			√		
X-29 FSW	Series of dangerous flight modes	√?				

ORIGINAL PAGE IS
OF POOR QUALITY

The Human Side of Design

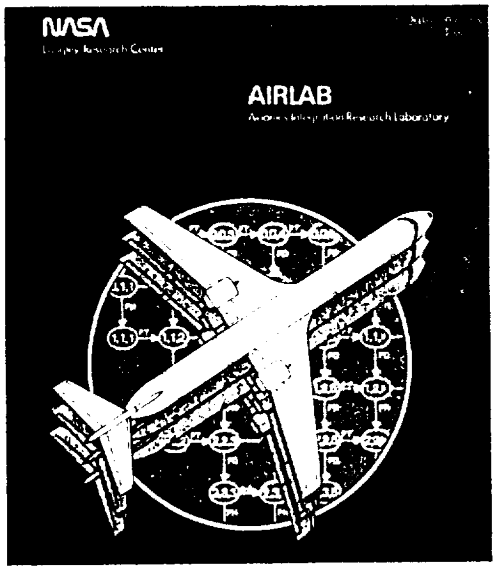
- Problems are solved by "people"
- Solutions must be understood by people who understand the problem
    - Problems are complex --> Solutions nust be simple
    - Find the one person on the team who can explain the solution
- Avoid trust in "discipline" interfaces
    - Hardware / Software
    - Aircraft / Flight Control
- Preserve the "Corporate Memory"

Some Research Needs

1. Achievable Levels of Safety Assurance
2. Relative Importance of Byzantine Problem
3. Methodologies for Correct System Timing
4. Complexity Metrics
5. Psychological Factors in Design
6. Design Diversity
7. Tradeoffs between Avionics and Related Systems
8. Level of Verification Needed for Support Tools
9. Methodologies for Designing and Evaluating FBW Systems
10. Methodologies for Developing and Verifying Correct Requirements
11. Ada Issues
12. Fiber Optics
13. An Objective List of Safe-Design Features (with relative values)

ORIGINAL PAGE IS
OF POOR QUALITY

NASA resources are invaluable for this effort.

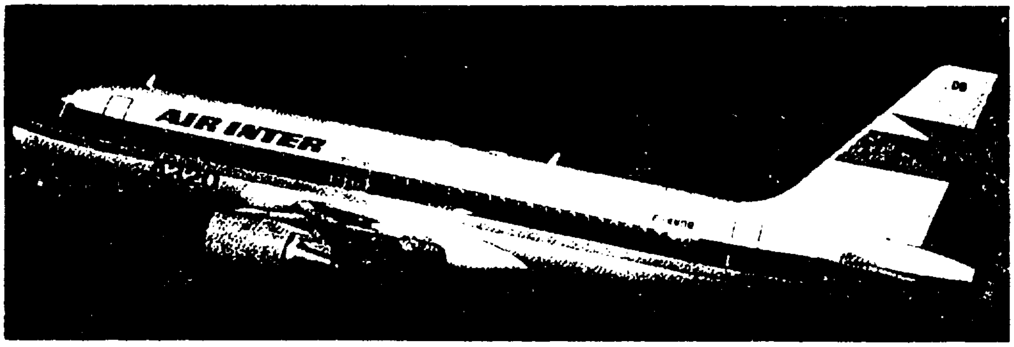


- Simulation and analysis of architectures
- Fault insertion and instrumentation
- Characterization of upsets
- ATOPS aircraft flight testing

- EME environment characterization
- EME circuit effects characterization

88 CRG 0965

Looking for a challenge ?



How about this ?

NASA FLIGHT CRITICAL SYSTEM WORKSHOP CHARTS

CARL S. DROSTE

12-13-88

GENERAL DYNAMICS FORT WORTH DIVISION

PRECEDING PAGE BLANK NOT FILMED

WHAT IS THE EXPERIENCE BASE OF THE FORT WORTH DIVISION OF GENERAL DYNAMICS (GD/FW) ON DIGITAL FLIGHT CONTROL SYSTEMS?

o GD/FW HAS BEEN FORTUNATE TO HAVE THE OPPORTUNITY TO DEVELOP AND FLY A NUMBER OF INTEGRATED DIGITAL FLIGHT CONTROL SYSTEMS.

oo AFTI/F-16 DIGITAL FLCS
oo AFTI/F-16 AUTOMATIC MANEUVERING ATTACK SYSTEM
oo F-16 QUAD DIGITAL DEMONSTRATION SYSTEM
oo F-16 PRODUCTION DIGITAL FLIGHT CONTROL SYSTEM
oo F-16 AUTOMATIC TERRAIN FOLLOWING SYSTEM

o GD/FW IS HEAVILY INVOLVED IN THE INTEGRATION OF A NUMBER OF FLIGHT CRITICAL SYSTEMS NOW BEING DEVELOPED FOR FLIGHT TEST, BUT NOT YET FLOWN (IN SOME CASES TEAMED WITH OTHER ORGANIZATIONS).

oo ATF
oo ATA
oo F-111 FLIGHT CONTROL MODERNIZATION
oo OTHER

o GD/FW IS IN THE INITIAL STAGES OF FLIGHT CRITICAL SYSTEM DEVELOPMENT FOR A NUMBER OF PROPOSED AIR VEHICLES.

oo ADVANCED VERSIONS OF THE F-16
oo NASP
oo E-7 STOVL
oo OTHERS

WHERE ARE THE SOFT SPOTS FOR FUTURE FLIGHT CRITICAL SYSTEM DEVELOPMENT?

o MANY OF THE TRADITIONAL BOUNDARIES BETWEEN TECHNOLOGIES ARE DISAPPEARING AND FLIGHT CRITICAL SYSTEM COMPLEXITY IS RAPIDLY INCREASING.

o THE CRITICAL SYSTEMS DEVELOPMENT TASK IS NOT A LINEAR FUNCTION OF SYSTEM SIZE.

o CRITICAL SYSTEMS INTEGRATION TECHNOLOGY IS DIFFICULT TO TRANSFER.

) INCREASING DIVERGENCE BETWEEN DEMONSTRATED AND THEORETICAL TECHNOLOGY -- REAL WORLD VERSUS PROMISES.

TRADITIONAL TECHNOLOGY BOUNDARIES ARE DISAPPEARING AND SYSTEM COMPLEXITY IS RAPIDLY INCREASING

o THE NUMBER AND EXTENT OF SYSTEMS THAT FALL INTO THE CATEGORY WHERE FAILURE CAN CAUSE IMMEDIATE RISK TO THE AIRCRAFT WILL BE GREATLY INCREASED.

o THERE WILL BE SIGNIFICANTLY INCREASED INTEGRATION OF FLIGHT CONTROL WITH 'OTHER SYSTEMS.

oo PROPULSION
oo AIR DATA
oo AVIONICS
oo STRUCTURES
oo STORES MANAGEMENT
oo SECONDARY CONTROL

o MANY AIRCRAFT SUBSYSTEMS THAT ARE NOT FAULT TOLERANT ON THE PRESENT GENERATION OF AIRCRAFT WILL HAVE TO BE MADE REDUNDANT OR BE PROTECTED BY ANALYTIC REDUNDANCY.

o CONTROL LAW DESIGN WILL INCLUDE MANY MORE THAN THE TRADITIONAL VARIABLES OF THE PAST.

o LIFE CYCLE COST OF THE TOTAL INTEGRATED SYSTEM IS BECOMING AN INCREASINGLY IMPORTANT FACTOR (I.E, RELIABILITY, MAINTAINABILITY, AVAILABILITY, ETC.).

THE CRITICAL SYSTEMS INTEGRATION ENGINEERING TASK IS NOT A LINEAR FUNCTION OF THE SYSTEM SIZE

o SYSTEMS INTEGRATION IS BRINGING TOGETHER MANY TRADITIONAL TECHNICAL DISCIPLINES IN AN INTERACTIVE MANNER.

o WHETHER WE LIKE IT OR NOT, THE INTEGRATED SYSTEMS OF THE FUTURE ARE FORCING US MORE AND MORE TO "DESIGN BY COMMITTEE".

o MOST OF US REALIZE HOW MUCH LONGER IT TAKES A COMMITTEE TO DO SOMETHING.

o THE CHALLENGE IS TO MAKE THIS COMMITTEE FUNCTION AS A CLOSE KNIT TEAM TO MINIMIZE OVERHEAD.

o THE FOCUS OF THE TEAM MUST BE ON INTERACTIVE COMMUNICATION.

THE CRITICAL SYSTEMS INTEGRATION ENGINEERING TASK IS NOT A LINEAR FUNCTION OF THE SYSTEM SIZE (CONTINUED)

o EXPERIENCE HAS SHOWN THAT MANPOWER REQUIREMENTS ESCALATE DISPROPORTIONATELY WITH SYSTEM COMPLEXITY.

o LOOKING AHEAD, WE MAY BE "DESIGNING BY BUREAUCRACY" IF WE ARE NOT CAREFUL, PARTICULARLY IF WE FORGET THAT COMMUNICATION ACROSS TECHNICAL BOUNDARIES IS MUCH MORE IMPORTANT THAN STRUCTURE. THE OVERHEAD MAY BECOME PROHIBITIVE UNLESS WE CAN FIND NEW WAYS OF FUNCTIONING.

o UNDETERMINISTIC SYSTEM CONCEPTS SUCH AS ARTIFICIAL INTELLIGENCE WILL REQUIRE NEW VERIFICATION AND VALIDATION METHODS.

o PHILOSOPHICAL CHANGES IN HOW WE APPROACH SYSTEM DESIGN, VERIFICATION, AND VALIDATION MAY BE REQUIRED BECAUSE EXTRAPOLATION OF PRESENT METHODS BECOMES PROHIBITIVE.

CRITICAL SYSTEMS INTEGRATION TECHNOLOGY HAS BEEN DIFFICULT TO TRANSFER

o MANY COST VERSUS SAFETY TRADES REQUIRE LARGE, EXPENSIVE PROGRAMS TO FORCE COST EFFECTIVE RESOLUTION. ON RELATIVELY SMALL PROGRAMS, FLIGHT CRITICAL SYSTEM INTEGRATION DECISIONS CAN BE GREATLY INFLUENCED BY OTHER FACTORS, SINCE PRODUCTION MANUFACTURING AND LONG TERM OPERATIONAL SUPPORT ARE NOT A PRIMARY CONSIDERATION.

o THERE IS RARELY ANY PROVABLE "BEST" WAY TO INTEGRATE SYSTEMS. DIFFERENT TECHNICAL BACKGROUNDS AND EVOLUTIONARY DEVELOPMENT MANY TIMES RESULT IN A NUMBER OF ACCEPTABLE SOLUTIONS. SELECTION OF THE "BEST" APPROACH IS OFTEN DEPENDENT ON THE GROUND RULES SELECTED TO DO THE EVALUATION.

o EFFECTIVE SYSTEMS INTEGRATIONS RELIES ON PERSONNEL AND FUNCTIONAL ORGANIZATION AS IT DOES TECHNOLOGY.

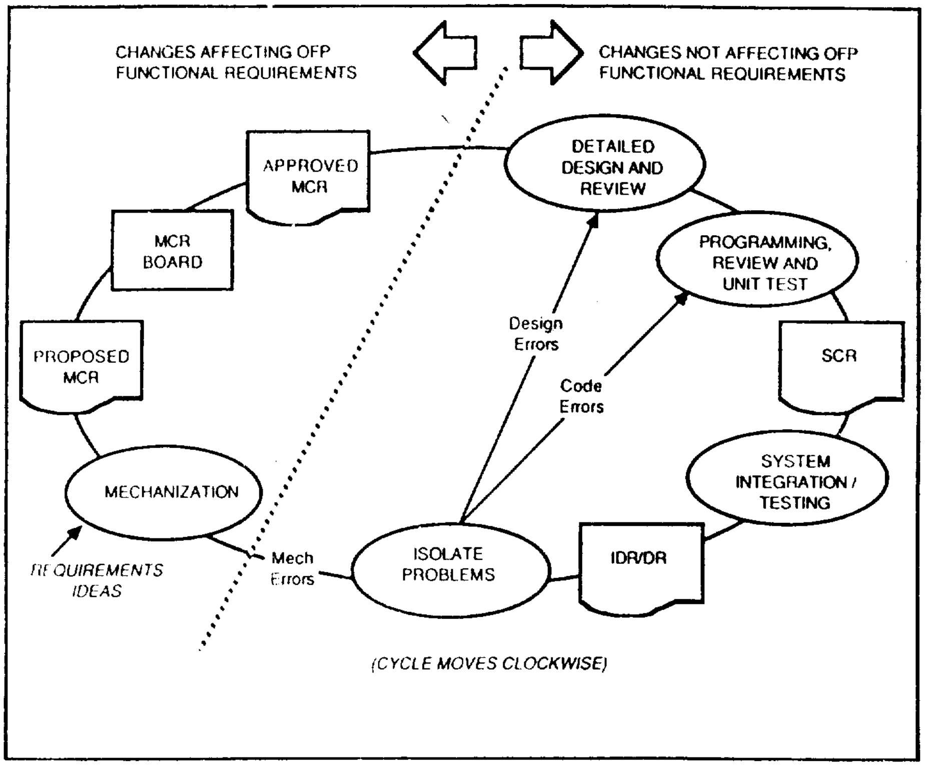
THE REAL WORLD VERSUS PROMISES

- POTENTIAL TECHNOLOGICAL ADVANCEMENTS ARE COMING AT US AT AN EVER INCREASING RATE.
- TECHNICAL POSSIBILITY IS OVER-SHADOWING TECHNOLOGY MATURATION IN THE REAL WORLD.
- IT DOES NOT TAKE A BIG ERROR IN GROUND RULE ASSUMPTIONS TO HAVE A PARTICULAR TECHNOLOGICAL OR SYSTEM ENGINEERING APPROACH COLLAPSE DOWN AROUND YOU.
- JUST BECAUSE SOMETHING HAS BEEN TALKED ABOUT FOR A NUMBER OF YEARS DOES NOT MEAN IT IS TRUE AND IS SURELY NOT SOMETHING ON WHICH YOU BUILD THE "NEXT" GENERATION.
- WE MUST BE VERY CAREFUL THAT OUR ZEAL AND INTEREST IN TECHNOLOGICAL "ADVANCEMENT" DOES NOT GET US IN SERIOUS TROUBLE.

SYSTEM WIDE INTEGRITY MANAGEMENT (SWIM) WILL BECOME A FUNDAMENTAL SYSTEM ENGINEERING CONCEPT

- THE REQUIREMENT IS NOW BECOMING APPARENT BY THE NEED TO INCLUDE NON-REDUNDANT SYSTEMS AS PART OF FLIGHT CRITICAL SYSTEMS.
  - AUTOMATIC MANEUVERING ATTACK SYSTEMS (AMAS)
  - AUTOMATIC TERRAIN FOLLOWING AND AVOIDANCE
  - AUTOMATIC GROUND COLLISION AVOIDANCE SYSTEMS FOR DISORIENTATION, INATTENTION, AND LOSS-OF-CONSCIOUSNESS.
- FAILURE PROTECTION MUST BE TREATED ON A SYSTEM WIDE BASIS AS EARLY IN THE PROGRAM AS POSSIBLE.
- DESIGN AND SUPPORT TECHNICAL DISCIPLINES MUST BE INTEGRATED INTO AN EFFICIENT SYSTEM ENGINEERING TEAM WITH EARLY EMPHASIS AND PARTICIPATION OF SAFETY, RELIABILITY, HUMAN FACTORS, PVI, AND OPERATIONS RESEARCH PERSONNEL.
- ALL SYSTEM COMPONENTS ASSUME A ROLE IN SWIM.
- PROTECTION AFFORDED BY SYSTEM MUST BE ANALYZED AND DOCUMENTED.

ORIGINAL PAGE IS
OF POOR QUALITY


CHANGES AFFECTING OFP FUNCTIONAL REQUIREMENTS
CHANGES NOT AFFECTING OFP FUNCTIONAL REQUIREMENTS
APPROVED MCR
DETAILED DESIGN AND REVIEW
MCR BOARD
PROGRAMMING, REVIEW AND UNIT TEST
PROPOSED MCR
Design Errors
SCR
Code Errors
MECHANIZATION
SYSTEM INTEGRATION / TESTING
REQUIREMENTS IDEAS
Mech Errors
ISOLATE PROBLEMS
IDR/DR
(CYCLE MOVES CLOCKWISE)

NASA PURSUIT OF INTEGRATED SYSTEM TECHNOLOGIES

o FEELING THE PRESSURE OF INCREASED SYSTEM INTEGRATION, MANY RESEARCHERS ARE PURSUING THE TECHNOLOGY AT THE INTERFACE POINTS BETWEEN THE TRADITIONAL TECHNICAL DISCIPLINES.

o NASA IS AT A STAGE WHERE DECISIONS NEED TO BE MADE RELATIVE TO HOW DEEPLY NASA OAST WANTS TO GET INTO THE SYSTEMS AREA. THERE ARE TWO WAYS TO GO.

oo CONTINUE IN THE PRESENT MANNER WITH THE VARIOUS TECHNOLOGIES EACH DEVELOPING ADVANCEMENTS IN THEIR IMMEDIATE INTERFACES AND CONDUCTING LIMITED SYSTEM INTEGRATION TASKS NECESSARY FOR EXPERIMENTS FOCUSING ON AIRCRAFT CAPABILITY.

oo DIVE COMPLETELY INTO THE FLIGHT CRITICAL SYSTEMS INTEGRATION AREA.

o EITHER ROUTE IS A PERFECTLY RATIONAL APPROACH, AND NASA WILL BE PROVIDING VALUABLE TECHNOLOGY FOR THE FUTURE AIRCRAFT IN EITHER CASE.

o THE HISTORICAL DIFFICULTY IN TRANSFERRING INTEGRATION TECHNOLOGY SHOULD BE HEAVILY WEIGHED IN ANY NASA DECISIONS.

o EFFECTIVE PURSUIT AND TRANSFER OF FLIGHT CRITICAL SYSTEMS INTEGRATION TECHNOLOGY WILL REQUIRE SIGNIFICANT LONG-TERM FUNDING COMMITMENTS AND REORGANIZATION.

The FAA Systems Perspective

Jim Treacy
Federal Aviation Administration

December 13, 1988

(There were no visual
aids used for this presentation)

COMMERCIAL AVIATION FLIGHT-CRITICAL

RESEARCH NEEDS

13 DEC 88
L.J. Yount/R.F. Hess
Honeywell/SCFSG

NEED OF AN FAA CERTIFICATION BASIS FOR ADVANCED TECHNOLOGY COMMERCIAL AIRCRAFT (e.g. FLY-BY-WIRE/LIGHT FLIGHT CONTROLS, etc.)

- Without an establish certification basis and the associated methodologies neither cost nor schedule for the development of an advanced technology commercial aircraft can be determined

- The resolution of the various substantial technical risks associated with such a certification will require a cooperative collaboration of appropriate complementary knowledgeable entities from both the private and public sectors of the U.S. Economy

- The threat posed by high energy elements of the external electromagnetic environment (Lightning, HERF) is probably the most substantial unresolved obstical to the development (for commercial aircraft of) digital computer based flight critical systems

THE ELECTROMAGNETIC ENVIRONMENT (EME) THREAT

- Since EME threats can excite waves of electrical energy throughout the entire aircraft, they represent a common mode threat to the redundant electronic elements of fly-by-wire/light control systems

- Error bounds associated with present EME technology state-of-the-art are prohibitive

EME THREAT TRENDS

↑ + ↑ + ↑

- EME THREAT (SOME ELEMENTS INCREASING ON GLOBAL SCALE)
  - -RADIO FREQUENCIES/ HIGH ENERGY RF (HERF) RADIO, TV, RADAR, MICROWAVE
  - -AIRCRAFT POWER SYSTEMS
  - -AIRCRAFT ELECTRONICS
  - -LIGHTNING
  - -MILITARY DIRECTED ENERGY WEAPONS ECM

- AIRCRAFT OPERATIONS ARE INCREASINGLY DEPENDENT ON ELECTRONIC SYSTEMS
  - -FLY-BY-WIRE/FLY-BY-LIGHT FLIGHT CONTROLS
  - -FULL AUTHORITY DIGITAL ENGINE CONTROLS
  - -INTEGRATED AIRCRAFT CONTROL SYSTEMS
  - -INTEGRATED AIRCRAFT DISPLAY SYSTEMS

- AVIONICS SUSCEPTIBILITY TO EME IS INCREASING
  - -TRADITIONAL SHIELDING IS BEING COMPROMISED COMPOSITE VS. METALIC STRUCTURE
  - -ENERGY TO INDUCE UPSET IS DECREASING

EME EFFECTS ON EXISTING FBW AIRCRAFT

o US ARMY BLACK HAWK

- Designed to withstand 1 V/M
  5 Aircraft Losses for which EMI was or is suspected as cause

o U.S. NAVY SEA HAWK

- Shielded to withstand 200 V/M
  No Losses Attributed to EMI

o PANAVIA TORNADO

- Designed to withstand 1 V/M
  1 Aircraft Loss Attributed to EMI

AIRCRAFT EME PROTECTION

- o Optimum protection of critical digital systems from the EME threat
  - Requires both survivability from hard faults and recoverability from soft faults
  - Will probably require the use of a mix of many different prevention (e.g. shieldings, optical links, etc.) and tolerance ("transparent recovery") techniques
  - Will result in minimizing weights, performance or cost penalties and maximizing the degree of immunity
- o **There is a need for a national resource for performing analytic assessments of threat penetration into electronic systems which would:**
  - Complement existing imperical techniques to achieve the needed degree of integrity to properly account for EME effects that would be associated with a total and mature EME technology for aircraft
  - **Provide the ability to access 1) EME protection effectiveness 2) the impact of changes (e.g. materials shielding, etc.) that affect airplane EM system characteristics early in the design cycle**

SOME EME PROTECTION APPROACHES

CONVENTIONAL

- o SHIELDING
- o FILTERING
- o REDUNDANCY
- o MECHANICAL CONTROLS (INCLUDING HYDRAULICS)
- o DIVERSION PATHS
- o GROUNDING
- o BALANCED CIRCUITRY
- o ELECTRICAL BONDING
- o LOCATION

EMERGING TECHNOLOGIES

- o FIBER OPTICS
- o "TRANSPARENT RECOVERY"
- o SOFTWARE TOLERANCE
- o RECONFIGURATION
- o HARDENED ICs
- o CONDUCTIVE COMPOSITES
- o OPTICAL COMPUTING AND STORAGE

RECOVERABLE FAULTS

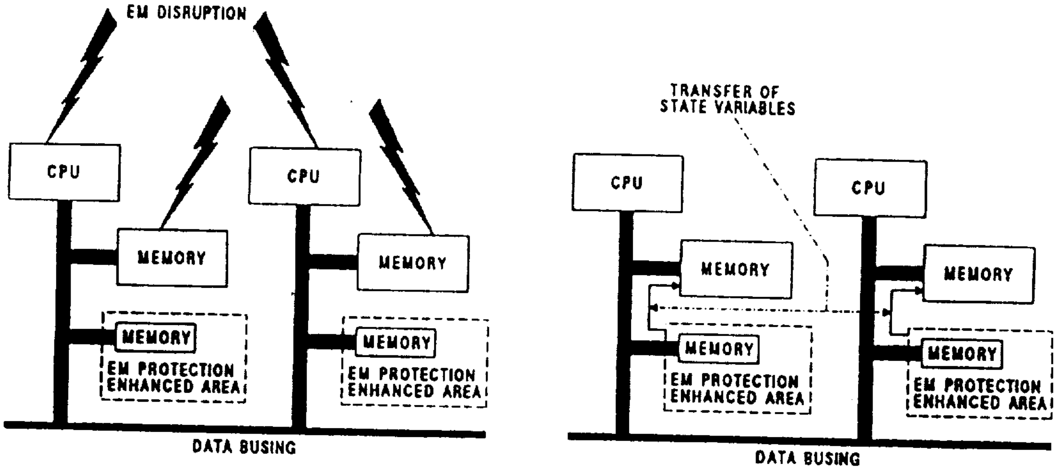
- A Characteristic associated with digital computers that knows several aliases (upset, soft fault, faults with nonstationary observability)

- Soft Faults can be induced by

  - Environmental Factors (e.g. EME, Nuclear particles,etc)
  - Hardware Factors
  - Software Factors

- In addition to being an obvious concern relative to safety, soft faults may be a major contributor to the concern associated with the costly MTBUR unconfirmed removals problem

TRANSPARENT RECOVERY


EM DISRUPTION
CPU
MEMORY
MEMORY
EM PROTECTION ENHANCED AREA
CPU
MEMORY
MEMORY
EM PROTECTION ENHANCED AREA
DATA BUSING
TRANSFER OF STATE VARIABLES
CPU
MEMORY
MEMORY
EM PROTECTION ENHANCED AREA
CPU
MEMORY
MEMORY
EM PROTECTION ENHANCED AREA
DATA BUSING

ORIGINAL PAGE IS OF POOR QUALITY

SUMMARY

- FAA CERTIFICATION BASIS FOR ELECTRONIC CONTROL SYSTEMS
  - NEEDS FURTHER DEFINITION
- EME TECHNOLOGY
  - NEEDS FURTHER DEVELOPMENT
- AIRCRAFT EME PROTECTION SCHEMES
  - NEEDS FURTHER DEVELOPMENT

KEY TECHNOLOGIES FOR THE 1990'S

RICHARD ULLMAN

FLIGHT CRITICAL WORKSHOP
LANGLEY RESEARCH CENTER
DECEMBER 13-15, 1988

KEY TECHNOLOGIES FOR THE 1990'S

AN INDUSTRY STUDY OF HIGH-LEVERAGE
ENABLING AEROSPACE TECHNOLOGIES
AND ROADMAPS TO ATTAIN THEM

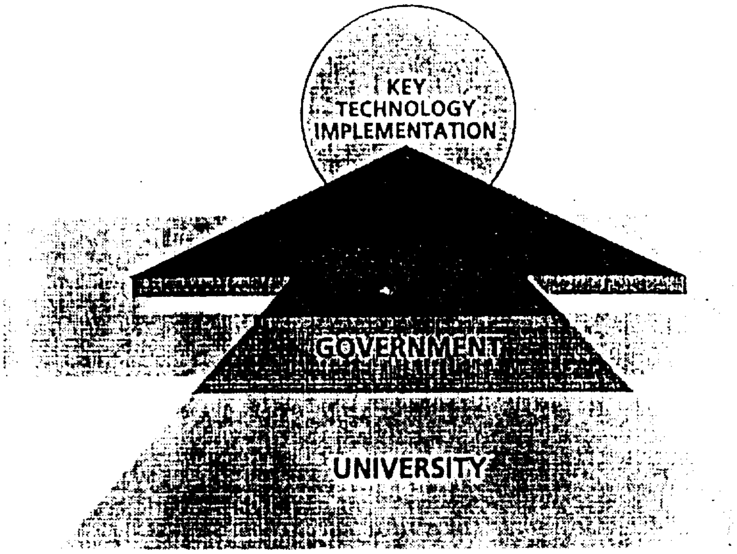
88 528-001
ITT
DEFENSE

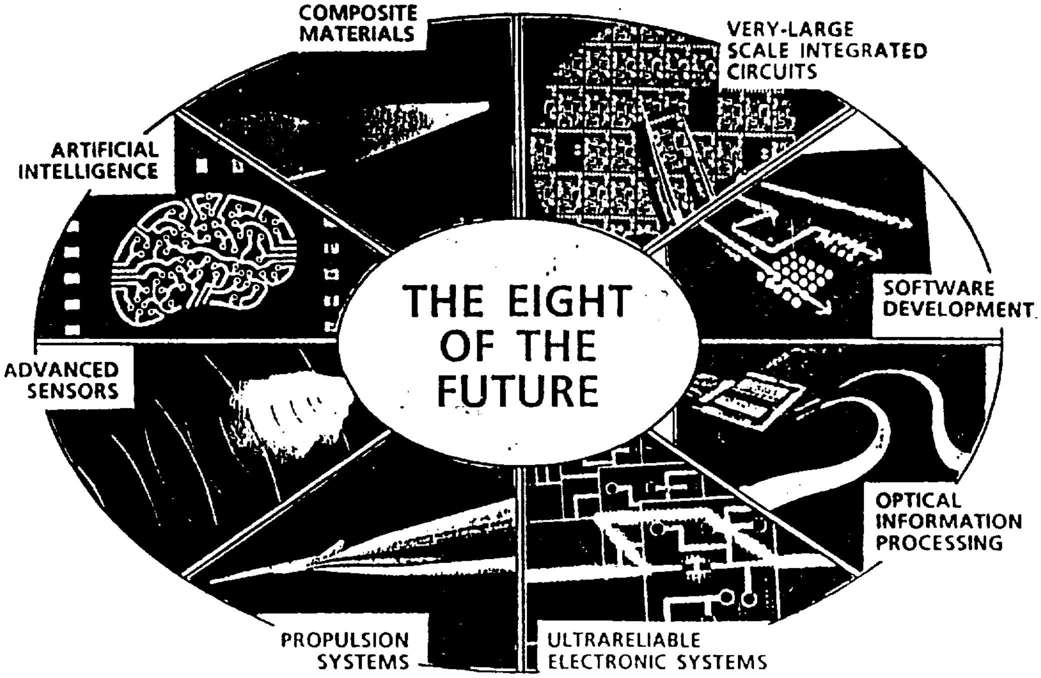
SOME HISTORICAL PRECEDENTS

1970 - THE SMOKE STACKS
1975 - APPLIANCES
1980 - AUTOMOTIVE INDUSTRY
1985 - CONSUMER ELECTRONICS
? - AEROSPACE TECHNOLOGY

88-528-008
ITT
DEFENSE

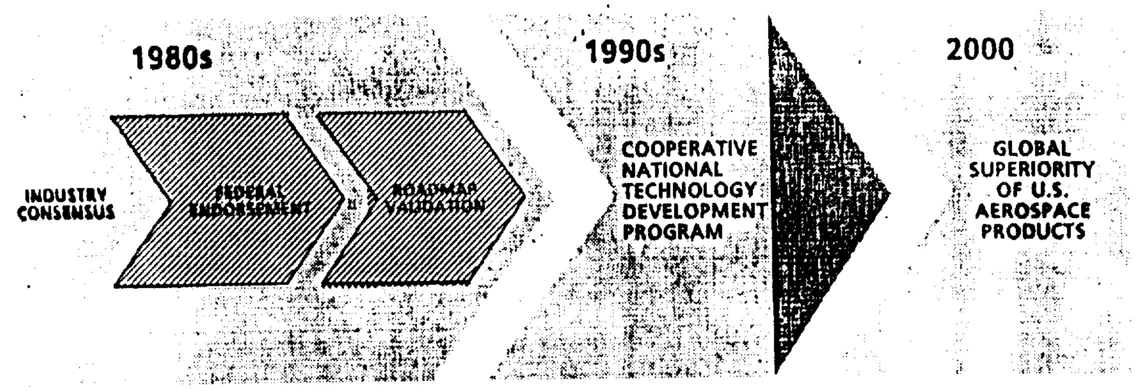
CATALYSTS - AN INDUSTRY CONSENSUS AND A NATIONAL COMMITMENT





ORIGINAL PAGE IS
OF POOR QUALITY

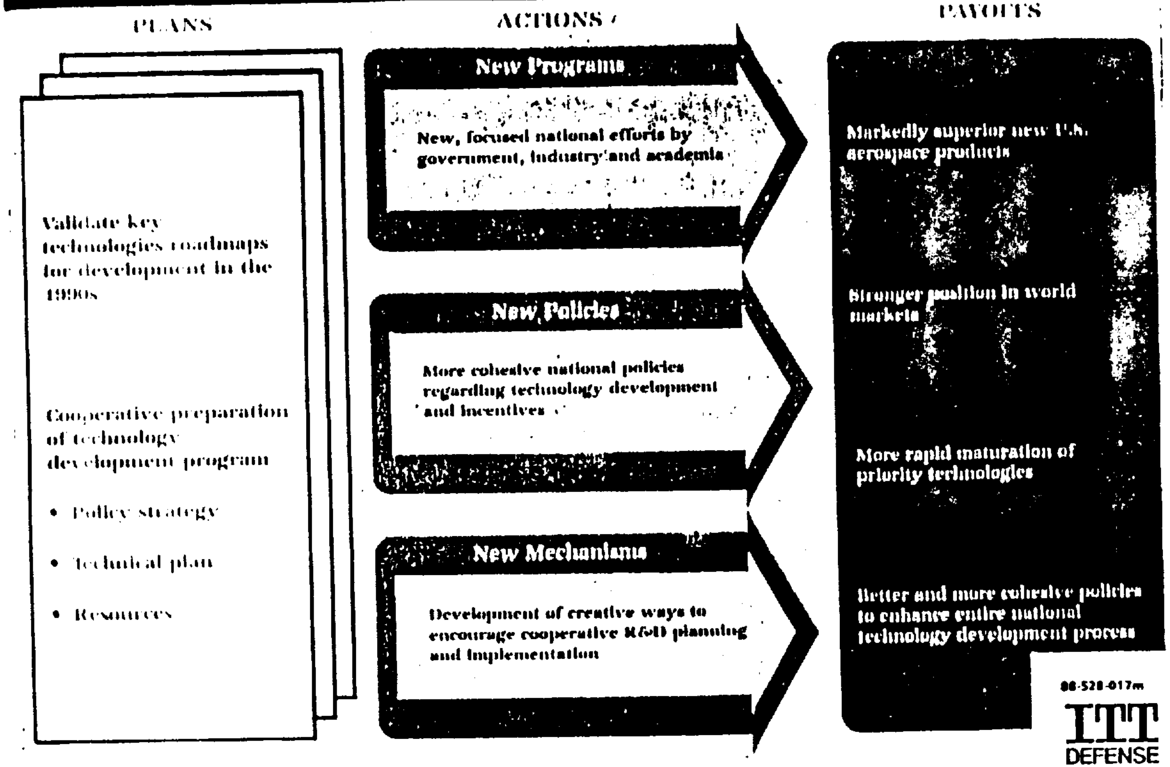
A RENEWED COMPETITIVENESS


1980s
1990s
2000
INDUSTRY CONSENSUS
FEDERAL ENDORSEMENT
ROADMAP VALIDATION
COOPERATIVE NATIONAL TECHNOLOGY DEVELOPMENT PROGRAM
GLOBAL SUPERIORITY OF U.S. AEROSPACE PRODUCTS

88-528 021

ITT DEFENSE

U.S. AREOSPACE COMPETITIVENESS INITIATIVE



PLANS

Validate key technologies roadmaps for development in the 1990s

Cooperative preparation of technology development program

- Policy strategy
- Technical plan
- Resources

ACTIONS

New Programs

New, focused national efforts by government, industry and academia

New Policies

More cohesive national policies regarding technology development and incentives

New Mechanisms

Development of creative ways to encourage cooperative R&D planning and implementation

PAYOFFS

Markedly superior new U.S. aerospace products

Stronger position in world markets

More rapid maturation of priority technologies

Better and more cohesive policies to enhance entire national technology development process

88-528-017m

ITT DEFENSE

ORIGINAL PAGE IS
OF POOR QUALITY

PROGRAM STATUS

- TECHNOLOGY ROADMAPS ARE BEING COORDINATED
- GOVERNMENT PARTICIPATION IS QUITE ACTIVE
- TECHNOLOGY TEAM MEMBERSHIP CONTINUES TO GROW
- KEY TECHNOLOGIES ASSESSED AS 40% OF TOTAL INDUSTRY TECH DEVELOPMENT
  - $1.5B IR&D
  - $1.5B CR&D
- AEROSPACE TECHNOLOGY POLICY FORUM WAS CONVENED

88-528-009

ITT
DEFENSE

URES ROADMAP

ULTRA RELIABLE ELECTRONIC SYSTEMS IN THE 21ST CENTURY

VISION

- ENHANCE THE RELIABILITY OF ELECTRONIC SYSTEMS, BY AT LEAST A MAGNITUDE WITHIN THE DECADE

- ACHIEVE THIS WHILE REDUCING
  - ACQUISITION COSTS
  - DEVELOPMENT TIME
  - CYCLE TIME THROUGH THE PLANT
  - COST OF OWNERSHIP

URES ROADMAP

APPROACH

- CULTURAL CHANGE IN HOW THE DESIGN PROCESS IS APPROACHED AND MANAGED - "CONCURRENT ENGINEERING"
- DEFECT FREE MANUFACTURING
- TECHNOLOGY INSERTION
- UNDERLYING NEED TO ENHANCE THE U.S. EDUCATIONAL SYSTEM

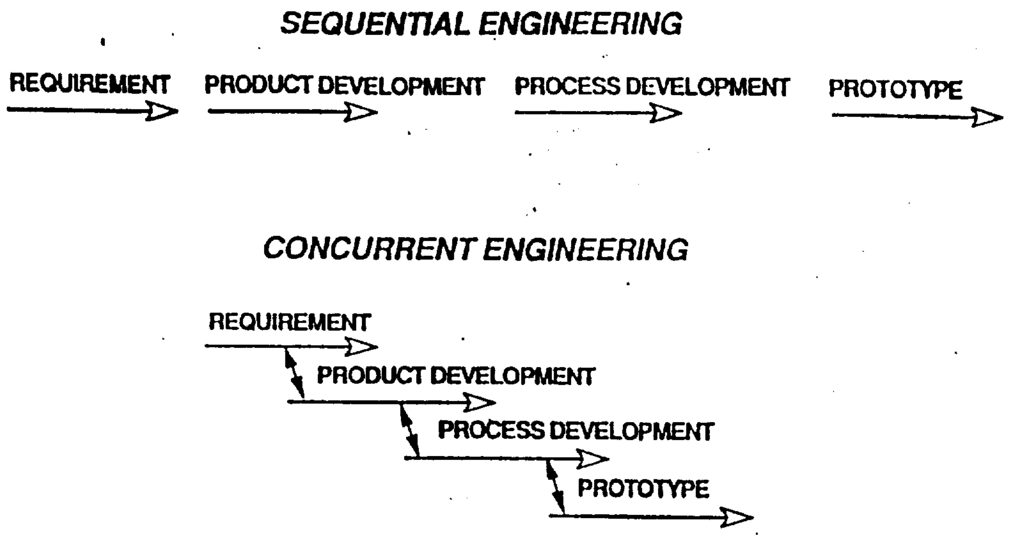
RELATIONSHIP TO TQM INITIATIVE

- CONCURRENT ENGINEERING IS THE ENGINEERING ARM OF TQM
- DEFECT FREE MANUFACTURING IS THE ULTIMATE GOAL OF CONTINUOUS IMPROVEMENT

CONCURRENT ENGINEERING

THE SYSTEMATIC APPROACH TO THE INTEGRATED, CONCURRENT DESIGN OF PRODUCTS AND THEIR RELATED PROCESSES, INCLUDING MANUFACTURE AND SUPPORT. THIS APPROACH IS INTENDED TO CAUSE THE DEVELOPERS FROM THE OUTSET, TO CONSIDER ALL ELEMENTS OF THE PRODUCT LIFE CYCLE FROM CONCEPTION THROUGH DISPOSAL, INCLUDING QUALITY, COST, SCHEDULE AND USER REQUIREMENTS.

CONCURRENT ENGINEERING



SEQUENTIAL ENGINEERING

REQUIREMENT → PRODUCT DEVELOPMENT → PROCESS DEVELOPMENT → PROTOTYPE →

CONCURRENT ENGINEERING

REQUIREMENT

PRODUCT DEVELOPMENT

PROCESS DEVELOPMENT

PROTOTYPE

AN INTEGRATED PROCESS WHICH ENGINEERS THE PRODUCT AND THE MANUFACTURING AND SUPPORT PROCESSES TOGETHER WITH EMPHASIS ON EFFICIENCY, INCREASED QUALITY AND REDUCED COST.

CONCURRENT ENGINEERING

SELECTED CASE STUDIES

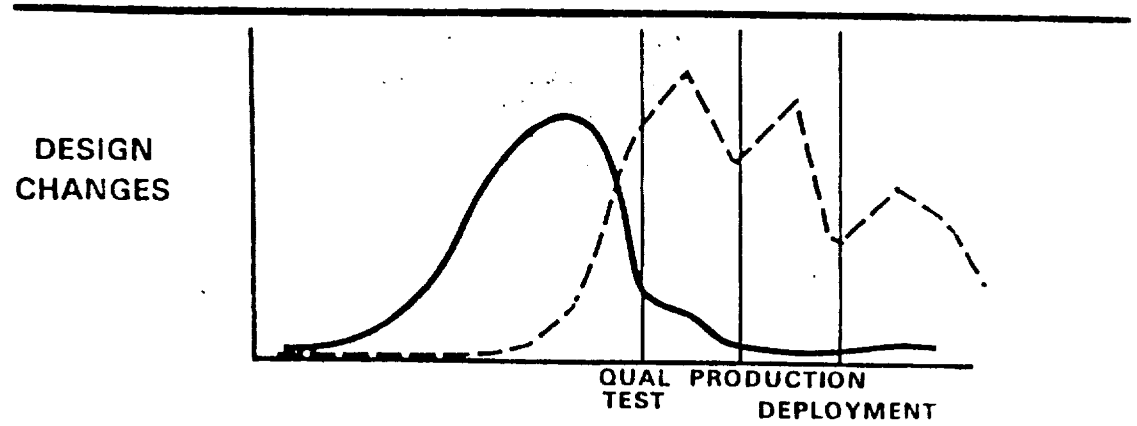
CASE STUDY	COST	SCHEDULE	QUALITY
McDONNELL DOUGLAS	60% SAVINGS ON BID FOR REACTOR AND MISSILE PROJECTS.	SIGNIFICANT SAVINGS (REDUCTION FROM 45 WEEKS TO 8 HOURS) IN ONE PHASE OF HIGH SPEED VEHICLE PRELIMINARY DESIGN; 18 MONTH SAVING ON TAV-8B DESIGN.	SCRAP REDUCED 58%, REWORK COST REDUCED 29% AND NON-CONFORMANCES REDUCED 38%; WELD DEFECTS PER UNIT DECREASED 70%; 68% FEWER CHANGES ON REACTOR; 68% FEWER DRAWING CHANGES ON TAV-8B.
BOEING BALLISTIC SYSTEMS DIVISION	REDUCED LABOR RATES BY $28/HOUR; COST SAVINGS 30% BELOW BID.	PART AND MATERIALS LEAD-TIME REDUCED BY 30%: ONE PART OF DESIGN ANALYSIS REDUCED BY OVER 90%.	FLOOR INSPECTION RATIO DECREASED BY OVER 2/3; MATERIAL SHORTAGES REDUCED FROM 12% TO 0; 99% DEFECT-FREE OPERATION.
AT&T	COST OF REPAIR FOR NEW CIRCUIT PACK PRODUCTION CUT AT LEAST 40%.	TOTAL PROCESS TIME REDUCED TO 46% OF BASELINE FOR 5ESS.**	DEFECTS REDUCED BY 30% TO 87%.
DEERE & COMPANY	30% ACTUAL SAVINGS IN DEVELOPMENT COST FOR CONSTRUCTION EQUIPMENT.	60% SAVINGS IN DEVELOPMENT TIME.	NUMBER OF INSPECTORS REDUCED BY 2/3.
HEWLETT-PACKARD CO., INSTRUMENT DIVISION	MANUFACTURING COSTS REDUCED 42%.	REDUCED DEVELOPMENT CYCLE TIME 35%.	PRODUCT FIELD FAILURE RATE REDUCED 60%. SCRAP AND REWORK REDUCED 75%.
IBM	PRODUCT DIRECT ASSEMBLY LABOR HOURS REDUCED 45%.	SIGNIFICANT REDUCTION IN LENGTH OF PMT DESIGN CYCLE. 40% REDUCTION IN ELECTRONIC DESIGN CYCLE.	FEWER ENGINEERING CHANGES. GUARANTEED PRODUCIBILITY AND TESTABILITY.

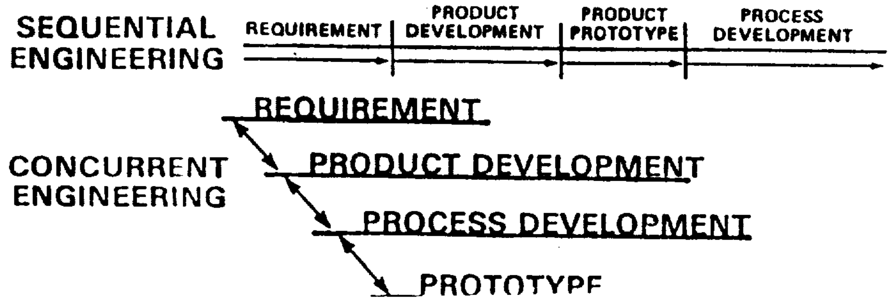
CONCURRENT ENGINEERING

SIGNIFICANT ELEMENTS

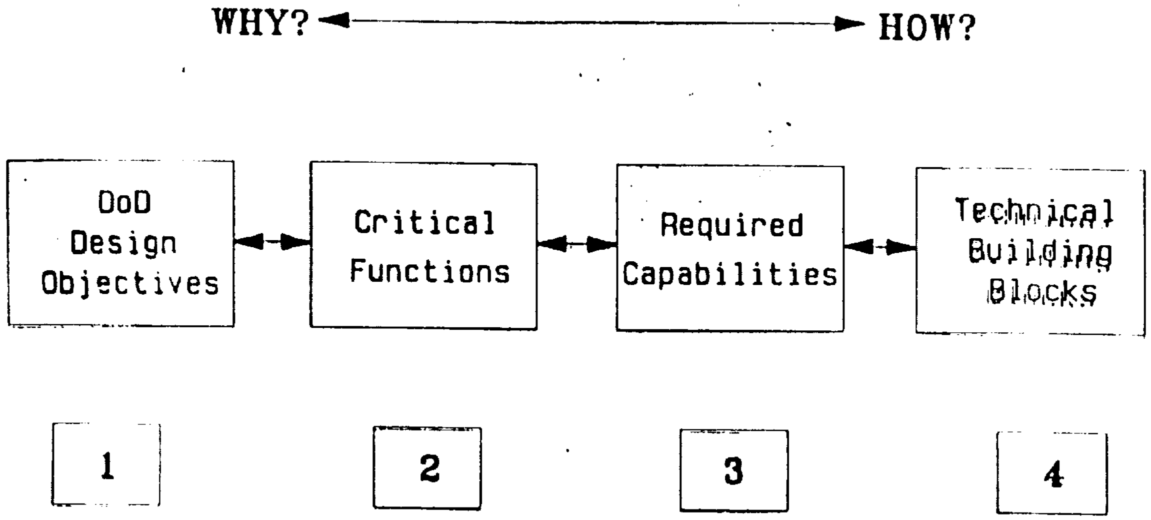
- o TOTAL SYSTEM ENGINEERING - FRAMEWORK FOR SYSTEM INTEGRATION AND OPTIMIZATION
- o MULTIDISCIPLINE TEAMS - INTEGRATED PRODUCT AND PROCESS ENGINEERING, STREAMLINED PROCESSES
- o QUALITY ENGINEERING METHODS - EFFICIENT PRODUCT AND PROCESS OPTIMIZATION
- o CAD/CAE/CAM - MANAGEMENT OF CHANGE, RAPID TRANSFER OF BENEFITS, REDUCTION OF ERRORS, EFFICIENT DATA COLLECTION AND ANALYSIS, EFFECTIVE INTEGRATION

CONCURRENT ENGINEERING




SEQUENTIAL ENGINEERING
REQUIREMENT
PRODUCT DEVELOPMENT
PRODUCT PROTOTYPE
PROCESS DEVELOPMENT

CONCURRENT ENGINEERING
REQUIREMENT
PRODUCT DEVELOPMENT
PROCESS DEVELOPMENT
PROTOTYPE


WHY?
HOW?

DoD Design Objectives

Critical Functions

Required Capabilities

Technical Building Blocks

1

2

3

4

COMPONENTS

ORIGINAL PAGE IS OF POOR QUALITY

COMPONENT 1 DoD OBJECTIVES	COMPONENT 2 CRITICAL FUNCTIONS	COMPONENT 3 REQUIRED CAPABILITIES	COMPONENT 4 TECHNICAL BUILDING BLOCKS
	Early, complete & continuing understanding of customer requirements and priorities.	Capture data on comparable products, processes & support (lessons learned).	Data Processing & Data Structures
		Define and capture data for new weapon system product, process & support. (complete and unambiguous description)	
Reduced Cost		Synthesize requirements into design of product, process & support.	Frameworks/ Architectures
		Validate design of product, process and support.	
	Translation of requirements concurrently and in an integrated fashion into optimal products and manufacturing and support processes.	Manage product, process, and support data.	
Reduced Time		Disseminate product, process, and support data.	Tools & Models
		Deliver product or data for manufacturing & supporting product.	
Increased Quality		Rapid Prototyping	Manufacturing Systems
		Process Robustness	
	Continuous review and improvement of product, process & support characteristics.	Intelligent oversight for impact assessment of changes.	
		Proactive, concurrent availability of current design.	Design Processes

Figure C.2 - Concurrent Engineering Framework

LIST EACH OF THE TOOLS TO BE DEVELOPED (Continued)

- DEFECT FREE MANUFACTURING
  - TRANSITION TEAMS
  - CONTINUOUS IMPROVEMENT
  - PROCESS OPTIMIZATION
  - SPC
  - TQM
  - RELIABILITY MODEL TO REFLECT MANUFACTURING PROCESS
- TECHNOLOGY INSERTION
  - VLSI
  - MIMIC
  - VHSIC
  - SUPPORTING TECHNOLOGIES

RLLATED ACTIVITIES UNDERWAY

- IDA STUDY ON THE ROLE OF CONCURRENT ENGINEERING IN WEAPON SYSTEMS ACQUISITION
- DARPA STUDIES ON CONCURRENT ENGINEERING
- TQM INITIATIVES
- SUPPORTING TECHNOLOGIES

RECOMMENDED APPROACHES

BASIC ROADMAP CONSIDERATIONS	CHANGE IN DESIGN PROCESS	DEFECT FREE MANUFACTURING	TECHNOLOGY INSERTION
STATE OF ONGOING RESEARCH AND STUDY			
INVESTMENT LEVELS, AND THEIR IMPLICATIONS			
THE MIX OF RESEARCH FUNDS			
ONGOING PROGRAMS			
NEW APPROACHES			
MAJOR ROAD BLOCKS, SOLUTIONS			
NEED FOR INCENTIVES			

88 520 015

ITT
DEFENSE

ORIGINAL PAGE IS
OF POOR QUALITY

ULTRA RELIABLE ELECTRONIC SYSTEMS (URES) (Continued)

NEEDED: HUMAN CAPITAL

- INDUSTRIAL EXHIBITION
  - CRYSTAL PALACE, LONDON
  - 1851
- DOMINANT WORLD POWERS
  - #1 - BRITAIN
  - #2 - U.S.
- BRITISH BUSINESSMEN AMAZED AT U.S. PRODUCTS
- LITERACY RATE
  - U.S. - 90%
  - BRITAIN - 67%

88-528-007

ITT
DEFENSE

ULTRA RELIABLE ELECTRONIC SYSTEMS (URES) (Continued)

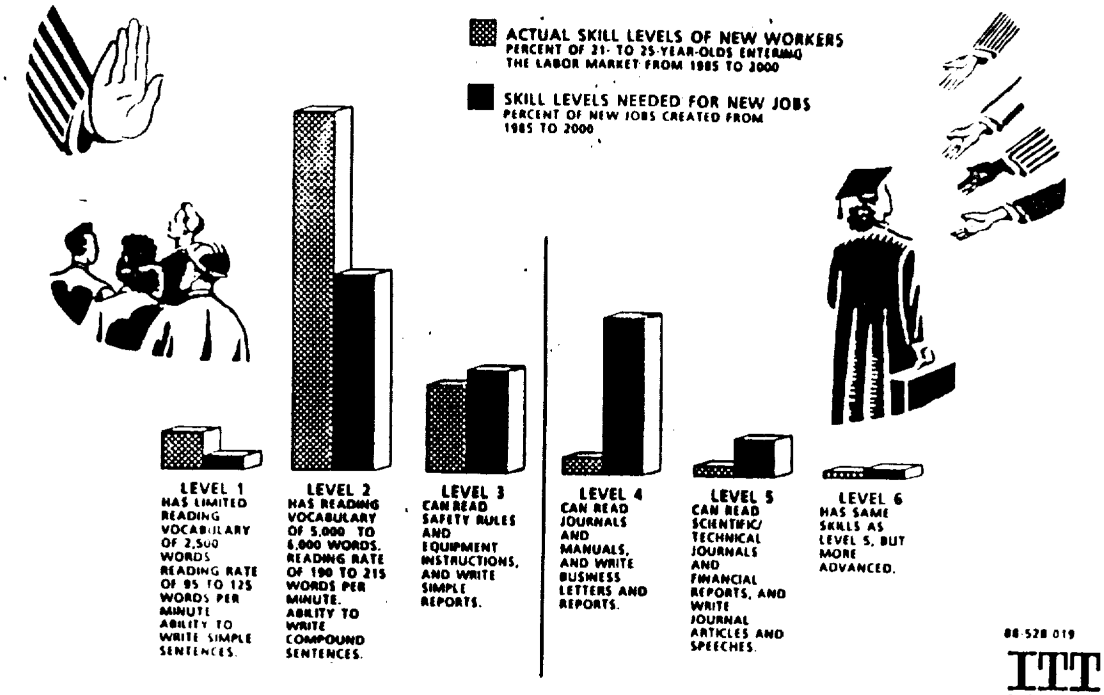
NEEDED: HUMAN CAPITAL

- 1980's
- DOMINANT WORLD POWER
  - #1 - U.S.
  - #2 - JAPAN
- AMERICAN CEO's MARVEL AT THE QUALITY OF JAPANESE PRODUCTS FLOODING THE MARKET
- LITERACY RATE
  - JAPAN - 95%
  - U.S. - 80%

88-528-006

ITT
DEFENSE

THE LOOMING MISMATCH BETWEEN WORKERS AND JOBS

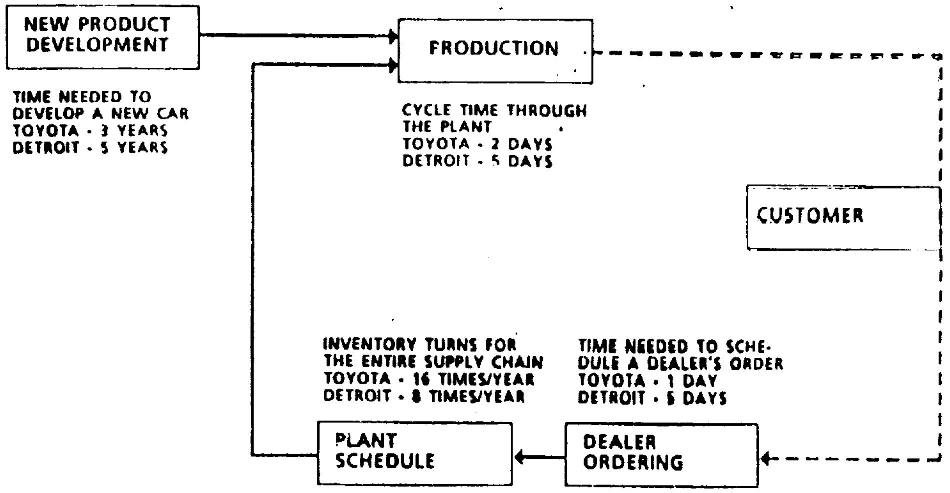


88 528 019

ITT
DEFENSE

HARVARD BUSINESS REVIEW

NOVEMBER-DECEMBER 1988



NEW PRODUCT DEVELOPMENT

TIME NEEDED TO DEVELOP A NEW CAR
TOYOTA - 3 YEARS
DETROIT - 5 YEARS

FRODUCTION

CYCLE TIME THROUGH THE PLANT
TOYOTA - 2 DAYS
DETROIT - 5 DAYS

CUSTOMER

INVENTORY TURNS FOR THE ENTIRE SUPPLY CHAIN
TOYOTA - 16 TIMES/YEAR
DETROIT - 8 TIMES/YEAR

TIME NEEDED TO SCHEDULE A DEALER'S ORDER
TOYOTA - 1 DAY
DETROIT - 5 DAYS

PLANT SCHEDULE

DEALER ORDERING

88 526-023

ITT
DEFENSE

ORIGINAL PAGE IS
OF POOR QUALITY

URES ROADMAP (Continued)

BASIC FACTS

1. TOYOTA CAN DEVELOP A NEW CAR IN 3 YEARS VS DETROIT'S 5 YEARS

2. INVENTORY TURNS FOR ENTIRE SUPPLY CHAIN
   TOYOTA 16 TIMES/YEAR
   DETROIT 8 TIMES/YEAR

3. TIME NEEDED TO SCHEDULE A DEALERS ORDER
   TOYOTA - 1 DAY
   DETROIT - 5 DAYS

4. PRODUCTION CYCLE TIME THROUGH THE PLANT
   TOYOTA - 2 DAYS
   DETROIT - 5 DAYS

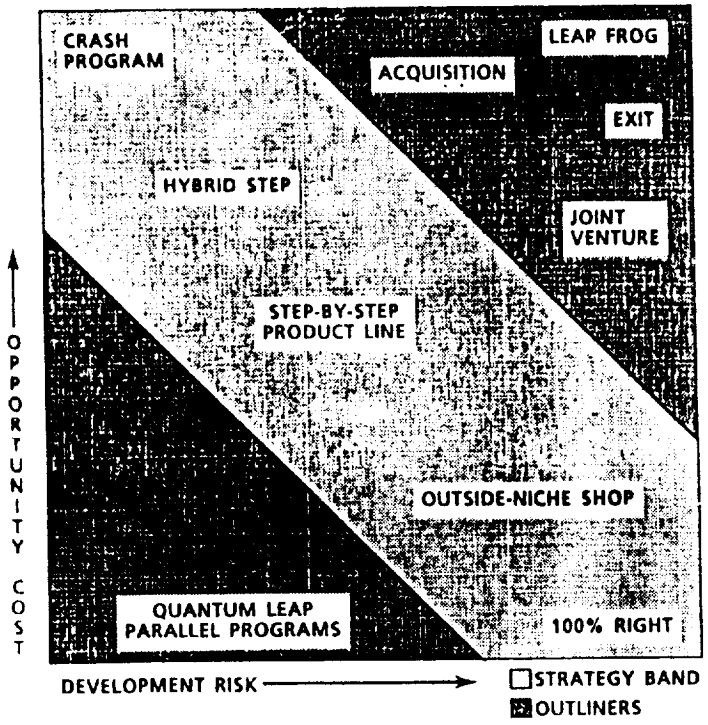
5. F-15 A/C BUILT IN JAPAN DEMONSTRATE HIGHER RELIABILITY, REDUCED MAINTENANCE

88-528-014

ITT

HARVARD BUSINESS REVIEW

NOVEMBER - DECEMBER 1988



88-528-024

ITT
DEFENSE

APPENDIX B

Summary Viewgraphs

NASA-LaRC FLIGHT-CRITICAL DIGITAL SYSTEMS TECHNOLOGY WORKSHOP

December 13-15, 1988

SUMMARY VIEWGRAPHS

OBJECTIVES

- Identify flight-critical systems research issues
- Address benefits obtainable by using digital systems in flight-critical applications
- Emphasize realization in a practical sense
- Define level of analysis, experimentation, and demonstration required for acceptance of results

SOME WORKSHOP STATISTICS

115 Participants

- 49 – Industry
- 39 – Government
- 22 – Research Institute
- 5 – University

85 Off-site participants

50 Organizations represented

31 Commercial organizations

ORGANIZATIONS REPRESENTED

Martin Marietta
Douglas Aircraft
Boeing Adv Systems
Lear Astronics
Lightning Technologies
Grumman Aircraft
MDAC Astronautics
Boeing Comm Aircraft
Pratt &Whitney
Lockheed California
Boeing Helicopter
Bell Helicopter
Bendix Aerospace
Arinc
Boeing Aerospace
Lockheed Georgia
Carnegie-Mellon U

Hamilton Standard
GE Aerospace
GE Aircraft Engines
Garrett Engine
Bendix Flight Systems
Honeywell/Sperry
Bendix/King
ITT Defense
TRW
Honeywell/SRC
General Dynamics
Lockheed Aero Sys
GD Space Systems
Boeing Electronics
Comp Sys Dev Corp
AFWAL
Michigan State U

NASA Langley
NASA Moffitt
NASA Dryden
NASA HQ
JPL
PRC Kentron
AVRADA
Argonne Nat'l Lab
Johns Hopkins U
RTI
Polytechnic U
SRI
FAA/Seattle
FAA Tech Center
Draper Lab
Duke U

AGENDA

- Opening remarks by Dr. J. F. Creedon,
  Director for Flight Systems, NASA-LaRC

- Invited overview presentations from industry to provide a workshop context
  - Dr. Thomas B. Cunningham,
    Honeywell Systems Research Center
  - Dr. Carl S. Droste, General Dynamics
  - Mr. Jim Treacy, Federal Aviation Administration
  - Mr. Larry J. Yount and Mr. Richard F. Hess,
    Honeywell Systems Research Center
  - Mr. Richard S. Ullman. ITT Defense Technology Corporation

AGENDA (Continued)

- Seven parallel working group sessions
  - Aeronautical Requirements
  - Space Requirements
  - Design for Validation
  - Failure Modes
  - System Modeling
  - Reliable Software
  - Flight Test

REQUIREMENTS FOR FLIGHT-CRITICAL DIGITAL SYSTEMS – AERONAUTICAL

Chair: John Todd, Douglas Aircraft
Co-chair: James Kelly, NASA-LaRC
Coordinator: Jill Hallenbeck, RTI

Goal

- Address what levels of dependability (e.g., performance and reliablility) must be achieved in order that flight-critical digital systems can fulfill useful roles in their respective flight regimes

Participants

- 9 – Industry
- 3 – Government
- 2 – Research Institute

REQUIREMENTS FOR FLIGHT-CRITICAL DIGITAL SYSTEMS – AERONAUTICAL

(Continued)

Key Recommendations

- Compilation and analysis of in-service reliability data and present a sanitized version to the public
- Increase knowledge base for system stress testing (e.g., random inputs, model noise environment, etc.)
- Develop cost and time-effective V & V philosophy for complex integrated systems
- Electromagnetic Environment (EME) propagation analysis and testing for validation

REQUIREMENTS FOR FLIGHT-CRITICAL DIGITAL SYSTEMS – SPACE

Chair: Robert Gates, Martin-Marietta

Co-chair: Howard Stone, NASA-LaRC

Coordinator: Robert Baker & Anita Shagnea, RTI

Goal

- Address what levels of dependability (e.g.. performance and reliability) must be achieved in order that flight-critical digital systems can fulfill useful roles in their respective flight regimes

Participants

- 6 – Industry
- 5 – Government
- 5 – Research Institute

REQUIREMENTS FOR FLIGHT-CRITICAL DIGITAL SYSTEMS – SPACE (Continued)

Key Recommendations

- Addressing what the appropriate figure-of-merit for system designs is. Factors include cost, reliability. time coverage, and availability
- Define approach to specifying parts levels (Class S vs Class B)
- Increased emphasis on integration research
  - Health monitoring interface
  - Validation of adaptive GN&C/intelligent systems

SYSTEM DESIGN FOR VALIDATION

Chair: Gerald C. Cohen, Boeing Advanced Systems
Co-chair: Dan Palumbo, NASA-LaRC
Coordinator: Joanne Dugan, RTI

Goal

- Address how flight-critical digital system technology can be made a part of the initial vehicle design, thus escaping the traditional "add-on" role of validation of electronic systems

Participants

- 8 – Industry
- 6 – Government
- 3 – Research Institute
- 2 – University

SYSTEM DESIGN FOR VALIDATION (Contd.)

Key Recommendations

- *Integrated Tool Set*:

  – Current design and manufacturing techniques are limited in the scope of problems that can be reasonably handled; thus. the development of an integrated methodology (tool set) is required. Future integrated systems will increase the size and complexity of systems. thus increasing the need for an integrated set of design and analysis tools

- *Design Guidelines*:

  It is easy, and quite common. for designers to design systems that cannot be validated. Guidelines are needed for designing validatable systems. including guidelines for using new technologies and techniques

- *Data Aquisition*:

  - Unlike structures, digital systems retain little evidence of logical or timing failure causes. Not only must past data be acquired from earlier systems, but future systems must contain sufficient monitoring systems so as to gather data on failure modes and effects of new technologies

FAILURE MODES

Chair: Don Frank, Douglas Aircraft
Co-chair: Harry Benz, NASA-LaRC
Coordinator: James Watterson, RTI

Goal

- Address how the various failure modes impact the design of digital systems used in flight-critical applications

Participants

- 5 – Industry
- 2 – Government
- 3 – Research Institute
- 1 – University

Key Recommendations

- Establish certification and integration criteria for EME/HERF internal environments
- Develop improved trouble shooting and repair procedures for flight-critical systems
- Investigate impact of component trends on failure modes of flight-critical systems

SYSTEM MODELING

Chair: Philip S. Babcock, C. S. Draper Labs
Co-chair: Sal Bavuso, NASA-LaRC
Coordinator: Charlotte Scheper, RTI

Goal

- Address what modeling techniques and support tools are required to permit designers to adequately judge the merits of different system designs

Participants

- 9 – Industry
- 9 – Government
- 9 – Research Institute
- 1 – University

SYSTEM MODELING
(Continued)

Key Recommendations

- Exploit the full power of current reliability tools by enhancing the information that is available for the tools. by creating guidelines for their use, and by continuing research on data collection for model inputs
- Improve the power and productivity of the current reliability tools by making them easier to use and supportive of model and results validation
- Integrate individual "ility" tools into an environment for supporting all phases of system design from quick justification of design decisions to completion of the design and assessment of a target system

RELIABLE SOFTWARE

Chair: Martin Shooman, Polytechnic University
Co-chair: George Finelli, NASA-LaRC
Coordinator: Linda Lauterbach, RTI

Goal

- Address how software should be treated as a component of flight-critical digital systems

Participants

- 8 – Industry
- 12 – Government
- 3 – Research Institute
- 1 – University

Key Recommendations

- Institutionalize the results of research on software reliability standards and guidelines
- Research ways to make single version software more reliable
- Correlate the measured reliability of the software with development strategies and associated V,V&T

ORIGINAL PAGE IS
OF POOR QUALITY

FLIGHT TEST

Chair: Jerry Doniger, Lear Astronics
Co-chair: David Holmes, NASA-LaRC
Coordinator: Ed Withers, RTI

Goal

- Address the role of flight test in demonstrating the acceptability of flight-critical digital systems

Participants

- 4 – Industry
- 4 – Government
- 2 – Research Institute

Key Recommendations

- Improve design. test. evaluation. and verification processes that are used broadly by industry and government
- Increase the amount of environmental information available to be used during simulation and testing
- Define the roles of testing and simulation. particularly where they may be used in complementary ways

SUMMARY OF ISSUES COMMON TO MANY WORKING GROUPS

- Lack of fully effective design and validation methods with support tools to enable engineering of highly-integrated, flight-critical digital systems

- Lack of high quality laboratory and field data on system failures

SUMMARY OF RECOMMENDATIONS COMMON TO MANY WORKING GROUPS

- Collect and analyze data for both operational and experimental systems
- Evaluate the cost-effectiveness of design and validation technologies
- Provide an easy-to-use, integrated, and validated environment of tools, guidelines, and results
- Establish criteria for EME validation

APPENDIX C

List of Participants

NAME OF ATTENDEE	AFFILIATION
Ailinger, Deborah F.	C. S. Draper Laboratory, Inc.
Alger, Linda	C. S. Draper Laboratory, Inc.
Anderson, R. E.	General Electric Company
Babcock, Philip S.	C. S. Draper Laboratory, Inc.
Baker, Robert	Research Triangle Institute
Banks, Dave	Boeing Electronics
Bansal, Indar	GE Aircraft Engines Group
Bavuso, Salvatore J.	NASA Langley Research Center
Beatty, Bob	McDonnell Douglas/Huntington Beach
Becher, Bernice	NASA Langley Research Center
Belcastro, Celeste	NASA Langley Research Center
Benz, Harry F.	NASA Langley Research Center
Bleeg, Bob	Boeing Commercial
Bond, David G.	Boeing Aerospace Co.
Bott, Charles F.	Bell Helicopter Textron, Inc.
Boyd, Mark	Duke University
Briggs, Donald R.	General Dynamics
Bryant, Wayne H.	NASA Langley Research Center
Bunting, J. O.	Martin Marietta Astronautics Group
Butler, Rick W.	NASA Langley Research Center
Caldwell, James C.	NASA Langley Research Center
Calloway, Raymond S.	NASA Headquarters
Carreno, Victor A.	NASA Langley Research Center
Chacon, Vince	NASA Dryden Flight Res. Facility
Clary, James B.	Research Triangle Institute
Cohen, Gerald C.	Boeing Advanced Systems
Creedon, Jerry F.	NASA Langley Research Center
Cunningham, Tom B.	Honeywell SRC
DeWalt, Michael P.	FAA/Seattle Aircraft Cert. Office
Deyst, John	C. S. Draper Laboratory, Inc.
Dodge, John L.	Garrett Engine Division
Doniger, Jerry	Lear Astronics Corp.
Driscoll, Kevin R.	Honeywell Systems and Research Ctr.
Droste, Carl S.	General Dynamics
Dubbury, Michael	Bendix/King
Dugan, Joanne	Research Triangle Institute

NAME OF ATTENDEE	AFFILIATION
Dunham, Janet R.	Research Triangle Institute
Eckhardt, Dave E.	NASA Langley Research Center
Elks, Carl R.	NASA Langley Research Center
Finelli, George B.	NASA Langley Research Center
Frank, Donald E.	Douglas Aircraft Company
Gai. Eliezer G.	C. S. Draper Laboratory, Inc.
Gangsass, Dagfinn	Boeing Advanced Systems
Gates, Robert L.	Martin Marietta Corp.
Goldberg, Jack	SRI International
Hallenbeck. Jill	Research Triangle Institute
Harper, Rick	C. S. Draper Lab
Hayes, Paul	NASA Langley Research Center
Hess. Richard F.	Honeywell, Inc.
Holmes, Dave	NASA Langley Research Center
Holt, Milt	NASA Langley Research Center
Howell, Sandra	NASA Langley Research Center
Jambor, Bruno J.	Martin Marietta Denver Aerospace
Johnson, Sally C.	NASA Langley Research Center
Kelly, Jim R.	NASA Langley Research Center
Kinlaw, Jeff	JPL/California Institute of Tech.
Lala, Jay	C. S. Draper Laboratory, Inc.
Lauterbach. Linda	Research Triangle Institute
Legere, Bob	Pratt & Whitney
Leonard, Bruce	McDonnell Douglas/Huntington Beach
Maras, Matt	McDonnell Douglas Astro.
Martinec, Dan	ARINC
Masson, Gerald M.	Johns Hopkins University
May. Philip J.	Computer Systems Development Corp.
McElvany. Michelle C.	Allied-Signal Aerospace Co.
McGough. John	Bendix Flight Systems
McManus. Bruce L.	Boeing Helicopters
Meissner, Charles W.	NASA Langley Research Center
Mulcare, Dennis B.	Lockheed Aeronautical Systems Co.
Neal. Brian	General Electric
Newman, Michael	Douglas Aircraft Co.
Nordstrom. John	GDSS

NAME OF ATTENDEE	AFFILIATION
Padilla, Peter A.	NASA Langley Research Center
Palumbo, Dan L.	NASA Langley Research Center
Pitts, Felix L.	NASA Langley Research Center
Plumer, Andy	Lightning Technologies, Inc.
Polley, John A.	GE Aircraft Engines
Reeck, Francis	Hamilton Standard UTC
Reed, John E.	FAA Technical Center
Rooney, Robert H.	Lockheed Aeronautical Sys. Co.
Rosch, Gene	C. S. Draper Laboratory, Inc.
Rottman, Michael S.	AFWAL/FDCL
Saraceni, Peter	FAA Tech Center
Scheper, Charlotte	Research Triangle Institute
Schmid, Herman	GE Aerospace
Schor, Andrei L.	C. S. Draper Laboratory
Shagnea, Anita	Research Triangle Institute
Shaw, Jack	Boeing Commercial
Shih, K. C.	NASA Ames/Moffitt
Shooman, Martin	Polytechnic University
Shull, Tom	NASA Langley Research Center
Siewiorek, Daniel	Carnegie-Mellon University
Sjogren, Jon	NASA Langley Research Center
Snyder, Marcia	Hamilton Standard
Spitzer, Cary	NASA Langley Research Center
Stech, George	NASA Langley Research Center
Stokes, John	NASA Langley Research Center
Stone, Howard	NASA Langley Research Center
Stump, Charlie	NASA Langley Research Center
Swain, Robert	NASA Langley Research Center
Swann, Jerry	GE Aircraft Engines
Szkody, Ron	Wright R & D Center
Thambidurai, Philip	Bendix Aerospace Tech Center
Thomas, Mitch	NASA Langley Research Center
Thompson, Daniel	AFWAL FDCL
Tice, Howard	McDonnell Douglas/Huntington Beach
Todd, John R.	Douglas Aircraft Co.

ORIGINAL PAGE IS
OF POOR QUALITY

NAME OF ATTENDEE	AFFILIATION
Toolan, Bill	Grumman Aircraft
Treacy, Jim	FAA/Seattle Certification Office
Ullman, Richard	ITT Defense Tech Corp
VanAlen, Derek	Boeing Aerospace Co.
Vitale, Anthony	McDonnell Douglas Corporation
Voigt, Sue	NASA Langley Research Center
Walker, Carrie	NASA Langley Research Center
Walter, Chris	Bendix Aero Tech Center Adv Sys
Watterson, Jim	Research Triangle Institute
Weixeman, Kent	Boeing Military
White, Allan	NASA Langley Research Center
Withers, Ed	Research Triangle Institute
Wojcik, Tony	Michigan State University
Wright, Bill	General Electric
Young, Steve	NASA Langley Research Center
Yount, Larry J.	Honeywell, Inc.
Zaepfel, Pete	NASA Langley Research Center

NASA
National Aeronautics and Space Administration

Report Documentation Page

1. Report No.
NASA CP-10028

2. Government Accession No.

3. Recipient's Catalog No.

4. Title and Subtitle
NASA-LaRC Flight-Critical Digital Systems Technology Workshop

5. Report Date
April 1989

6. Performing Organization Code

7. Author(s)
C. W. Meissner, Jr.
J. R. Dunham
G. Crim (Editors)

8. Performing Organization Report No.
412U-3181-29

9. Performing Organization Name and Address
NASA Langley Research Center
Hampton, VA 23665-5225

10. Work Unit No.
505-66-21-03

11. Contract or Grant No.

12. Sponsoring Agency Name and Address
National Aeronautics and Space Administration
Washington, DC 20546-0001

13 Type of Report and Period Covered
Conference Publication

14. Sponsoring Agency Code

15. Supplementary Notes The workshop was organized and chaired by Charles W. Meissner, Jr. and Felix L. Pitts of NASA Langley Research Center. The Center for Digital Systems Research, Research Triangle Institute, Research Triangle Park, NC, provided support in preparing this document. C. W. Meissner, Jr.: NASA Langley Research Center, Hampton, Virginia, J. R. Dunham and G. Crim: Research Triangle Institute, Research Triangle Park, North Carolina.

16. Abstract

This publication documents the outcome of a Flight-Critical Digital Systems Technology Workshop held at NASA-Langley Research Center on December 13-15, 1988. The purpose of the workshop was to elicit the aerospace industry's view of the issues which must be addressed for the practical realization of flight-critical digital systems. The workshop was divided into three parts: an overview session; three half-day meetings of seven working groups addressing aeronautical and space requirements, system design for validation, failure modes, system modeling, reliable software, and flight test; and a half-day summary of the research issues presented by the working group chairmen. Issues that generated the most consensus across the workshop were (1) the lack of effective design and validation methods with support tools to enable engineering of highly-integrated, flight-critical digital systems, and (2) the lack of high quality laboratory and field data on system failures especially due to electromagnetic environment (EME).

ORIGINAL PAGE IS
OF POOR QUALITY

17. Key Words (Suggested by Author(s))
Aerospace
Highly Integrated
Flight-Critical
Digital Systems
Systems Design and Validation

18. Distribution Statement
[redacted]

19. Security Classif. (of this report)
Unclassified

20. Security Classif. (of this page)
Unclassified

21. No. of pages
190

22. Price