DTIC FILE COPY

RADC-TR-90-239
Final Technical Report
September 1990

AD-A230 067

TESTABILITY/DIAGNOSTICS DESIGN ENCYCLOPEDIA

Giordano Associates

George Neumann, George Barthlenghi, et al.

DTIC
ELECTE
DEC 20 1990
S
D

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

Rome Air Development Center
Air Force Systems Command
Griffiss Air Force Base, NY 13441-5700

90 12 19 167

This report has been reviewed by the RADC Public Affairs Division (PA) and is releasable to the National Technical Information Services (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

Although this report references limited documents listed below, no limited information has been extracted:

RADC-TR-84-57 dated Apr 84 - Distribution authorized to USGO agencies and their contractors.

RADC-TR-85-148 dated Aug 85 - Distribution authorized to USGO agencies and their contractors.

Built-in-Test Design Guide Joint AMC/CNO/AFLC/AFSC Commander dated Mar 81 - Subject to Export Control

RADC-TR-90-239 has been reviewed and is approved for publication.

APPROVED: Frank H. Born

FRANK H. BORN
Project Engineer

APPROVED: John J. Bart

JOHN J. BART
Technical Director
Directorate of Reliability & Compatibility

FOR THE COMMANDER: James W. Hyde III

JAMES W. HYDE III
Directorate of Plans & Programs

If your address has changed or if you wish to be removed from the RADC mailing list, or if the addressee is no longer employed by your organization, please notify RADC (RBET) Griffiss AFB NY 13441-5700. This will assist us in maintaining a current mailing list.

Do not return copies of this report unless contractual obligations or notices on a specific document require that it be returned.

REPRODUCED FROM
BEST AVAILABLE COPY

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY (Leave Blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	September 1990	Final Jun 87 - Mar 89

4. TITLE AND SUBTITLE

TESTABILITY/DIAGNOSTICS DESIGN ENCYCLOPEDIA

5. FUNDING NUMBERS

C - F30602-87-C-0099
PE - 62702F
PR - 2338
TA - 02
WU - 3B

6. AUTHOR(S)

George Neumann, George Barthlonghi, et al.

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Giordano Associates
21 White Deer Plaza
Sparta NJ 07871

8. PERFORMING ORGANIZATION REPORT NUMBER

N/A

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Rome Air Development Center (RBET)
Griffiss AFB NY 13441-5700

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

RADC-TR-90-239

11. SUPPLEMENTARY NOTES

RADC Project Engineer: Frank H. Born/RBET/(315) 330-4726

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

The military services have been experiencing problems in the adequacy of their weapon systems' diagnostic capabilities. There are available a variety of techniques, procedures, standards, and devices which can be applied to the acquisition of a system's diagnostic capability. However, this type of information appears in a variety of reports, military standards, specifications, and other documents. The objective of this document is to collect such individual and diverse pieces of information as they relate to the function of weapon system design. This guide is one of the three guides aimed at the government program manager, the contractor program manager, and the system designer. This guide is specifically designed to help the design engineer to meet or exceed the diagnostic requirements imposed on the system he is designing.

General diagnostics, diagnostic equipment, Test methods/equipment; Integrated systems; Standards specifications; handbooks; systems approach/engineering; Data bases; Troubleshooting/repair; Maintainability test evaluation, Computer aided diagnosis; Logistics support; Fault isolation; Malfunction detection (FD)

14. SUBJECT TERMS	15. NUMBER OF PAGES
Diagnostics, Integrated Diagnostics, Testability, Diagnostics Capability, Design, Guidance	544
	16. PRICE CODE

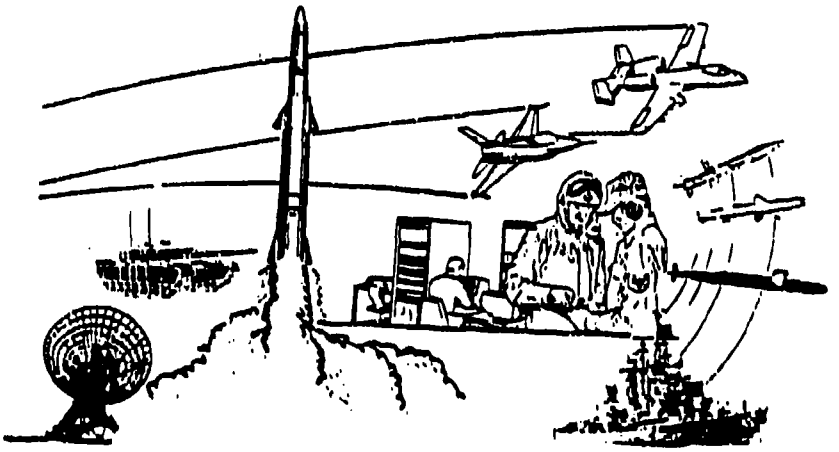
17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

AN ENCYCLOPEDIA FOR THE WEAPON SYSTEM DESIGNER



HOW YOU CAN DESIGN AND DEPLOY AN IMPROVED WEAPON SYSTEM DIAGNOSTIC CAPABILITY

– A DIAGNOSTIC DESIGN ENCYCLOPEDIA



ROME AIR DEVELOPMENT CENTER

PREFACE

One of the major missions of the Rome Air Development Center (RADC) is the development of procedures and techniques for improving the readiness and supportability of weapon systems. In support of this mission, RADC has sponsored a myriad of studies, analyses, and developments that have resulted in techniques, standards, and procedures aimed at reaching this goal.

In the 1980s all the military services have recognized the importance of improving the diagnostic capability of weapon systems as a means for rapid troubleshooting and repair of these systems. The research and development efforts conducted by RADC are reflected in this guide by synthesizing the results of these many efforts and filling gaps to provide both government and industry with a compendium of procedures and techniques which may be used to improve the fielded weapon systems' diagnostic capability.

Many other programs have made the contributions that are included in these guides. Information has been freely included from various military service and industry work. Among these is the Air Force's Generic Integrated Maintenance Diagnostics Program (GIMADS). The GIMADS Program has made available much of its Air Force-oriented material, which is included in this guide. In this manner, material from all of the other service organizations is now available for Joint Service use.

Three (3) guides have been written which are aimed at the following users:

- o Government Program Manager
- o Contractor Program manager
- o System Designer

Thus, the guidance material required by a specific user will be included in one of these three (3) guides.

It is believed that this guidance material represents a comprehensive look at the problems in fielding a satisfactory diagnostic capability and a structured system engineering approach to solving these problems. RADC solicits comments on this guidance material, as a means for improvements in the coming years.

DIAGNOSTIC DESIGN ENCYCLOPEDIA

These guides have been prepared under contract by Giordano Associates, Inc., with subcontractor assistance from Grumman Aerospace Corporation and Rockwell International.



Accession For	
NTIS CRA&I	[X]
DTIC TAB	[ ]
Unannounced	[ ]
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or special
A-1	

DIAGNOSTIC DESIGN ENCYCLOPEDIA

Table of Contents

	Page
Preface	I
Table of Contents	III
Introduction	XI
Why All The Worry About Diagnostics?	XI
How Can This Guide Help You?	XIII
Definitions	XV
What Do We Mean By Integrated Diagnostics?	XV
How to Use This Guide	XVII

Requirement #1 — Programmatic

Establishing and Justifying a Program for Acquiring a Diagnostic Capability	1-1
1.1 Reviewing a Statement of Need	1-3
1.2 Responding to an RFP,SOW,Specification	1-7
1.3 Diagnostic Capability Program Planning	1-27
1.4 Preparation of SCPs/DCPs	1-31

Requirement #2 — Requirements

Establishing and Allocating Diagnostic Requirements	2-1
2.1 Translating Mission Requirements	2-3
2.2 Allocation of Diagnostic Requirements	2-17
2.3 Optimization of the Diagnostic Element Mix	2-25
2.4 Risk Assessment	2-33

Table of Contents...continued

Requirement #3 – Design

Designing the Diagnostic Capability 3-1

3.1 Providing a Cohesive Diagnostic Design Process 3-3
3.2 Criteria for Designing Diagnostic Capability 3-15

Requirement #4 – Assessment

Analysis and Assessment of the Performance of the Diagnostic Capability 4-1

4.1 In-Process Testability/Diagnostic Analysis 4-3
4.2 Maintainability Demonstrations 4-13

Requirement #5 – Reviews

Conducting Design Reviews 5-1

5.1 Conducting Technical Reviews and Audits 5-3

Requirement #6 – Evaluation

Conducting Test and Evaluation 6-1

6.1 Preparation of the TEMP 6-3
6.2 Development Test and Evaluation 6-9
6.3 Operational Test and Evaluation 6-15

Requirement #7 – Maturation

Maturation of the Diagnostic Capability 7-1

7.1 Maturation Planning 7-3
7.2 Data Collection and Feedback 7-9

Appendix A – Lessons Learned A-1
Appendix B – Acronyms B-1
Appendix C – Design Automation Tools C-1
Appendix D – Vertical Test Methods D-1
Appendix E – Testability/Diagnostic Design Techniques E-1

LIST OF TABLES

Table	Title	Page
1	Established Requirements	XVIII
2	Categories of Tool Types	XIX
3	Application Matrix	1-28
4	Sample Allocation of 95% BIT FD Requirement	2-21
5	MIL-STD Application to Design	3-6

LIST OF TABLES - APPENDIX C

Table	Title	Page
1	Fact Sheet Utilized to Characterize Each Software Tool Surveyed	C-6
2	Types of Testability / Diagnostic Tools vs. Areas of Application	C-9

LIST OF TABLES - APPENDIX E

Table	Title	Page
1	Diagnsotic Support Activities	E-7
2	State Sequence for Figure 16	E-32
3	TM Bus Design Parameters and Characteristics	E-87

LIST OF ILLUSTRATIONS

FIGURE	Title	Page
1	Roadmap	XXI
2	Diagnostic Growth Concept	1-16
3	Notional Diagnostic Allocation Specification	2-20
4	Design Integration of Diagnostic Capability	3-7
5	Cone of Tolerance	3-9
6	Design Integration of Diagnostic Capability	3-10
7	Common Architecture	3-27
8	Diagnostic Strategy	3-28
9	Diagnostic System Implementation	3-29
10	B-1B Maintenance Concept	3-32

LIST OF ILLUSTRATIONS - APPENDIX D

FIGURE	Title	Page
1	Cone of Tolerance	D-7
2	Ideal Distribution for a 100 = 10 OHM Resistor	D-10
3	Test Limits Must be Set to Compensate for Measurement Uncertainties	D-13
4	Effect of Skewed Nominal Test Limits at the Factory and Measurement Uncertainty of Depot ATE -- RTOK Condition at Factory	D-15
5	ATE Emulation -- The Classical Approach -- Hardware Reconfiguration	D-20
6	ATE Emulation -- PE Approach -- Software Reconfiguration	D-21
7	Vertical Testability Design Procedures and Documentation	D-24

LIST OF ILLUSTRATIONS – APPENDIX E

Figure	Title	Page
1	Partitioning of Circuits for Scan Design	E-13
2	General Structure of Circuit for Scan-Path Discussion	E-13
3	Latches and Flip-Flops: Symbols and Waveforms	E-15
4	Flip-Flop With Multiplexer (2) Multiplexer Circuit Diagram, (b) and Symbol, (c) For Multiplexed Data (MD) Flip-Flop	E-15
5	Stanford Scan-Path Architecture	E-16
6	Two-Port D Flip-Flop Circuit	E-18
7	General Structure of Circuit Using Two-Port Flip-Flops to Provide Scan Path	E-19
8	General Structure of Circuit Using Two-Port Latches to Provide Scan Path - LSSD Double-Latch Design	E-20
9	General Structure of Circuit Using Scan-Set Techniques	E-20
10	General Structure of Circuits Using Multiplexer to Scan-Out Latch Contents	E-22
11	Addressable Latch Circuit (a) and Symbol (b)	E-22
12	General Structure of Circuit Using Random-Access Scan	E-24
13	Addressable Latch With Coincident Selection	E-24
14	General Structure of Circuit Using Scan Latches on Input Output Pins	E-26

LIST OF ILLUSTRATIONS – APPENDIX E

Figure	Title	Page
15	Test ROM Stimulus	E-28
16	Standard Form of Autonomous Linear - Feedback Shift Register	E-31
17	General Modular Realization of An LFSR	E-31
18	Four-Stage Linear-Feedback Shift Register	E-33
19	A Four-Stage Multiple-Input LFSR	E-34
20	Example Use of Signature Analysis With TM Bus	E-36
21	Typical Microprocessor-Based System	E-39
22	Test Data Wraparound	E-41
23	Residue Codes	E-45
24	Watchdog Timer	E-45
25	Pin Electronics Channel	E-48
26	Example Architecture for Fault-Tolerant Circuit	E-51
27	Example Architecture for Fault-Tolerant Circuit Using Voting Arrangement	E-51
28	Example of Two Fault-Tolerant Circuit Architectures	E-52
29	Example Use of TM Bus With a Fault - Tolerant Microprocessor System	E-54
30	Impact of Test Points Limited to Certain Outputs	E-56

LIST OF ILLUSTRATIONS – APPENDIX E

Figure	Title	Page
31	Examples of Approaches to Achieve Testability In Multi-Package Shift Register, Comparator, Parity Generator/Checker Chains	E-56
32	ROM/RAM Test Points	E-57
33	Logic Test Points for Multi-Package Sequential Circuits	E-58
34	Optimum Test Point Selection for Lamp or Led Circuits	E-58
35	Test Points for Triple Modular Redundant Circuits	E-60
36	Isolating the Circuit From A Test Point with a Resistor	E-60
37	Isolating the Circuit From A Test Point with A Buffer	E-61
38	Circuit Design Approach to Isolate Free-Running Oscillators	E-63
39	Techniques for Isolating Free-Running Oscillator	E-63
40	Delay Line Technique to Stop Oscillator And to Substitute External Clock	E-64
41	Techniques to Synchronize Two-Rail On-Board Clock to Test Equipment	E-64
42	Examples of Breaking Up Counter Strings	E-65
43	Example of Insertion of Additional Logic Component to Control Feed-Back Path	E-65

LIST OF ILLUSTRATIONS – APPENDIX E

Figure	Title	Page
44	Examples of Design Approaches to Control Feed-Back Paths	E-66
45	Additional Example of Using Logic Components to Provide Feed-Back Control	E-66
46	Additional Example of Using Added Logic to Control Feed-Back Path	E-67
47	Example of Breaking Feed-Back Loops With a Multiplexer	E-67
48	Improving Observability With Limited I/O Pins (Shift Register)	E-71
49	Improving Observability With Limited I/O Pins (MUX)	E-71
50	Examples of Probe Contact Pad Layout	E-74
51	Examples of Probe Contact Pad Layout	E-75
52	PCB Area	E-76
53	Surface Mounted Leadless Chip Carrier	E-79
54	Surface Mounted Leaded Chip Carrier	E-79
55	Chip Carrier Characteristics	E-81
56	Small Outline (SO) Package	E-82
57	Pin Grid Array (PGA)	E-82
58	TM Bus Signals	E-85

WHY ALL THIS WORRY ABOUT DIAGNOSTICS?

Let's put the diagnostic problem in its proper perspective. You've got a problem with your automobile and you turn to a mechanic for help. Historically, you realize that the problem may be fixed or indeed, for some reason, you may have to go back one or more times before the problem is corrected, or you give up. We are talking about automobiles. Automobiles, which a manufacturer has produced tens of thousands of times, have a historical record of their reliability and maintainability and have been redesigned and reengineered many times.

When comparing your automobile to an extremely complex weapon system that is pushing the state of the art and produced in limited numbers, with questionable historical data on their operation, one can easily understand the magnitude of the problem.

It is not the purpose of this guide to provide a comprehensive discussion of the diagnostic problems, but rather to guide government and industry people in circumventing known problem areas. However, to understand the magnitude of the problem a few examples follow.

In one six-month period, at one F-16 tactical fighter wing, over 13,600 maintenance manhours were reported for the processing of unnecessary removals. This equals about 20 people just working on troubleshooting these "good" items.

A DoD Task Force on Productivity in Support Operations (1986) found that 20 to 50 percent of avionics maintenance actions resulted in removal of items with no evidence of failures.

The deployment of an avionics intermediate shop for fighter aircraft to a remote location can require anywhere from three to eleven C-141B equivalent loads. In wartime, there just will not be enough cargo aircraft to respond to this need. In peacetime, it's just plain costly.

The diagnostic problem is not unique to any one service, nor to any one type of weapon system. It manifests itself throughout the military services. The problem can be an engineering or a field problem. It can be a man or a machine problem. It can be a wartime or a peacetime problem. It can be a prime system or a supportability problem. The problem manifests itself in different ways for different types of weapon systems, but the consequences are all the same--long times to troubleshoot, removal of items which have not failed, long

logistic tails, and an overall lack of confidence in the entire diagnostic capability. Obviously, the result is lack of readiness and a waste of dollars and manpower.

There are a multitude of reports which adequately describe the problem. Two of these reports give a comprehensive picture of the problem and possible solutions. These are:

"Isolation of Faults in Air Force Weapon and Support Systems," Committee on Isolation of Faults in Air Force Weapon and Support Systems, Air Force Studies Board, Commission on Engineering and Technical Systems, National Research Council.

"Report for the Department of Defense on the Implementation of Integrated Diagnostics," prepared by the National Security Industrial Association's Integrated Diagnostics Working Group, September 1984.

HISTORICALLY THE FIELDED DIAGNOSTIC CAPABILITY HAS NOT LIVED UP TO THE PROMISES

Government and industry must share the responsibility for what has happened in the past. On the government's side, there tends to be a lack of knowledge on how to specify what is needed and how to make sure the government gets what it needs. On the contractor's side is a lack of understanding of the importance of fielding a satisfactory diagnostic capability and still maintain schedule and cost limitations. Hopefully, the series of guides produced under this program will help to alleviate this situation.

The military services, as well as the Office of the Secretary of Defense, understand the urgency of this problem and have established multimillion dollar programs to help alleviate this situation, both from a technology and a management perspective. For the most part, these programs are generic--applicable to a variety of weapon systems.

DIAGNOSTIC IMPROVEMENT PROGRAMS ARE UNDERWAY

HOW CAN THIS GUIDE HELP YOU?

The outputs from many of the government-sponsored programs are a variety of techniques, procedures, standards, and devices which can be applied to the acquisition of a system's diagnostic capability. However, this type of information appears in a variety of reports, military standards, specifications, and other documents. The major focus of this guide is to bring together this knowledge in a usable form and tie this to the various diagnostic activities which occur during the acquisition and deployment of a weapon system. In addition, where holes exist in this acquisition process, the guide attempts to fill them. Following this procedure will help you in doing a better job of designing a weapon system diagnostic capability.

This guide is for use by the weapon system **DESIGNER**-- to help him understand the relationship of the design of the diagnostic capability to the prime system itself and to provide detailed methods, procedures, tools, and trade-off information which can be applied to the design and demonstration of the weapon system's diagnostic capability.

This guide provides the user with a thorough compilation of available testability/diagnostic design information.

This guide is the last of three documents. The first two are for use by the Government and Contractor Program Managers, respectively.

THREE GUIDES - ONE FOR EACH TYPE OF USER. THIS ONE IS FOR THE DESIGNER

DEFINITIONS

WHAT DO WE MEAN BY INTEGRATED DIAGNOSTICS?

Before using this guide it is imperative that you understand the definition of a few words. The first term is "**testability**", which is defined as "a design characteristic which allows the status of the unit to be confidently determined in a timely manner." It includes the capability to detect, isolate failures and minimize false alarms. Therefore, testability may be regarded as inherent to the item's design.

"**Diagnostics**" is defined as "the hardware, software and/or other documented means used to determine a malfunction has occurred and to isolate the cause of the malfunction." It also refers to "the action of detecting and isolating failures."

"**Integrated diagnostics**" is defined as a "structured design and management process to achieve the maximum effectiveness of a weapon system's diagnostic capability by considering and integrating all related pertinent diagnostic elements." The process includes interfaces between design, engineering, testability, reliability, maintainability, human engineering, and logistic support analysis. The goal is a cost-effective capability to detect and unambiguously isolate all faults known or expected to occur in weapon systems and equipment in order to satisfy weapon system mission requirements.

"**Diagnostic capability**" refers to the capability of the system to detect and isolate faults, utilizing automatic and manual testing, maintenance aids, technical information and the effects of personnel and training.

"**Diagnostic element**" is defined as one part of the diagnostic capability (e. g., ATE).

"**Diagnostic Subsystem**" is defined as all the diagnostic elements which constitute a weapon system's diagnostic capability.

"**Embedded diagnostics**" is defined as any portion of the weapon system's diagnostic capability which is an integral part of the prime system or support system. "Integral" implies that the embedded portion is physically enclosed in the prime system and/or permanently attached--physically or electrically.

DEFINITIONS

"External diagnostics" is defined as any portion of the weapon system's diagnostic capability which is not embedded.

It is important to understand that integrated diagnostics is a structured **process** for acquiring a diagnostic capability.

HOW TO USE THIS GUIDE

For a better understanding of the various diagnostic activities that take place during the acquisition and deployment of a weapon system, a Roadmap has been prepared for your use. The Roadmap depicts all of the diagnostic activities that take place during each phase of weapon system acquisition and deployment. The Roadmap is shown in Figure 1 (located at the end of this section), with inputs and outputs for each activity. This Roadmap gives the reader the entire picture of diagnostic activities from beginning to end. It is recognized that there is no single Roadmap that can apply to all situations. Thus the Roadmap is designed with multiple entry points to provide flexibility.

THE ROADMAP GIVES YOU THE BIG PICTURE

The structure of the guide is built around this Roadmap. The following seven requirements were established which apply to the Roadmap activities. These requirements are listed in Table 1.

Reference to a specific requirement is shown on the Roadmap, so the reader can quickly relate a diagnostic activity on the Roadmap to specific guidance information contained in this guide.

HOW TO USE THIS GUIDE

TABLE 1. ESTABLISHED REQUIREMENTS

REQUIREMENT #	REQUIREMENT
1	ESTABLISHING AND JUSTIFYING A PROGRAM FOR ACQUIRING A DIAGNOSTIC CAPABILITY
2	ESTABLISHING AND ALLOCATING DIAGNOSTIC REQUIREMENTS
3	DESIGNING THE DIAGNOSTIC CAPABILITY
4	ANALYSIS AND ASSESSMENT OF THE PERFORMANCE OF THE DIAGNOSTIC CAPABILITY
5	CONDUCTING DESIGN REVIEWS
6	CONDUCTING TEST AND EVALUATION
7	MATURATION OF THE DIAGNOSTIC CAPABILITY

Each one of these basic requirements is followed by detailed requirements (e. g., Requirement 3.1, Providing a Cohesive Diagnostic Design Process). Each of these detailed requirements is tied to a weapon system activity and a weapon system acquisition phase.

The three guides are structured in essentially the same way--the difference being the guidance material supplied is tailored for the USER of each specific guide. Each requirement is highlighted for easy access to the information the user requires.

Each of the guides contains a Lessons Learned Appendix (Appendix A), which will help the user to understand how this guidance information can be applied to real-world acquisitions. Appendix B lists the Acronyms used.

WHAT'S DIFFERENT ABOUT THIS GUIDE AS OPPOSED TO THE OTHER TWO?

HOW TO USE THIS GUIDE

It is recognized that the designer of the diagnostic capability has a vital interest in several of the seven requirements listed in Table 1 and very little interest in the others. The information contained relative to Requirement #'s 3 and 4 is supplemented by Appendices C, D, and E. It is important for the user of this guide to refer to these appendices, because they contain information on the techniques and tools which can assist the user. Appendix C not only describes tools (both automatic and manual) which can assist in performing the design activities under Requirement #'s 3 and 4, but also addresses tools which are applicable to other requirements. As shown in Table 2, these tools are categorized by the requirement to which they apply. However, some tools apply to more than one requirement (e.g., a design tool can also be used to assess the design). In those cases, the guide identifies its primary and secondary use.

TABLE 2. CATEGORIES OF TOOL TYPES

REQUIREMENT (FROM TABLE 1)			
# 2 ESTABLISHING REQUIREMENTS	# 3 DESIGN	# 4 ASSESSMENT	# 7 MATURATION
O SETTING REQUIREMENTS O ALLOCATE REQUIREMENTS O OPTIMIZATION OF MIX O RISK ANALYSIS	O SYSTEM ARCHITECTURE (eg., PARTITIONING, FAULT TOLERANCE) O DESIGN RULES & PRACTICES O DIAGNOSTIC AUTHORING (eg., DIAGNOSTIC STRATEGY, TEST/DIAGNOSTIC GENERATION, TRAINING INFORMATION AUTHORING, FAILURE MODES & EFFECTS ANALYSIS)	O INHERENT TESTABILITY O DIAGNOSTIC EFFECTIVENESS (eg., FAULT SIMULATION, BIT EFFECTIVENESS) O MAINTAINABILITY DEMONSTRATION (eg., MIL-STD-471)	O FEEDBACK ANALYSIS

THE GUIDE IS STRUCTURED TO GIVE THE USER EASY ACCESS TO THE INFORMATION REQUIRED

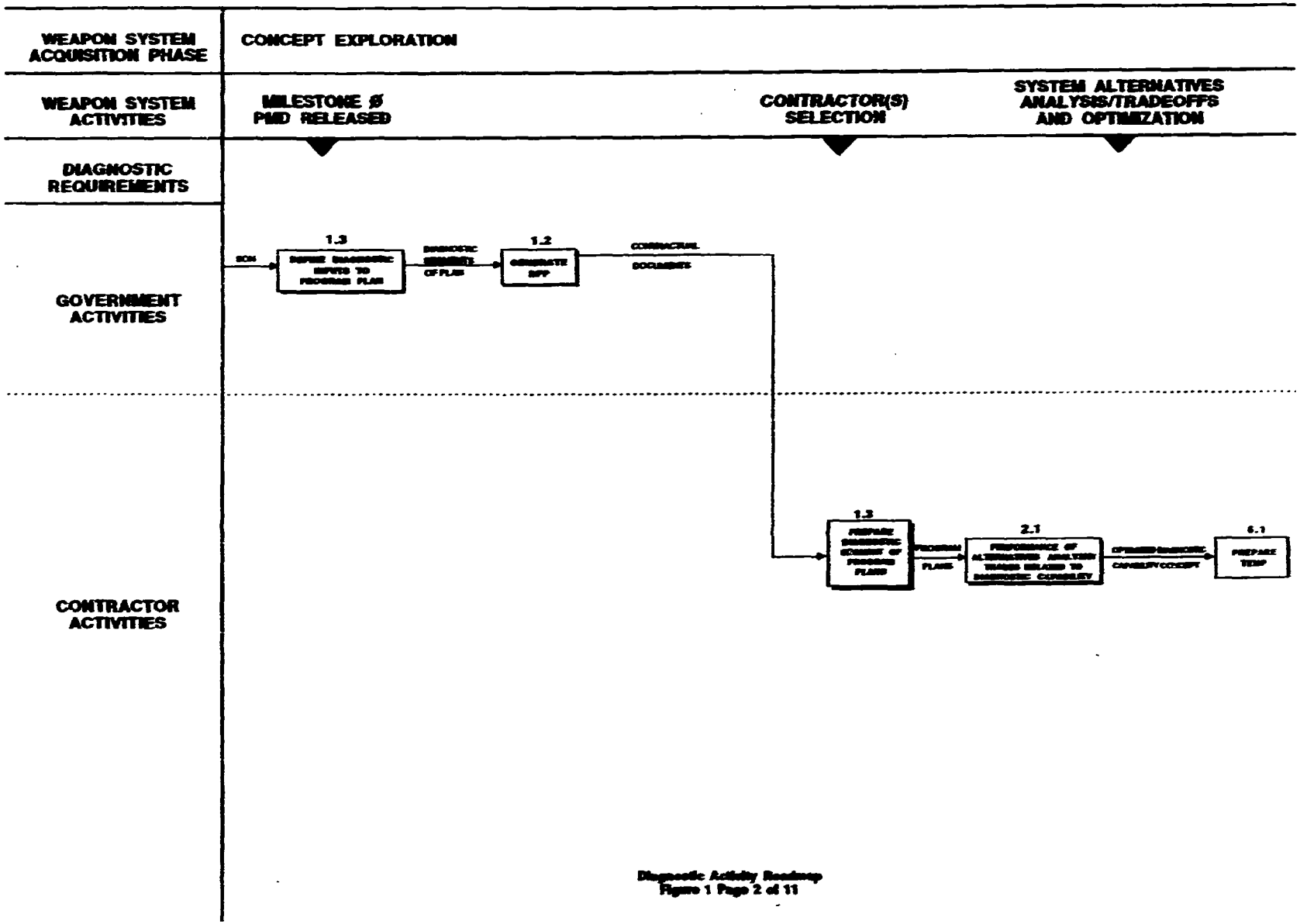
It is recognized that a guide of this type cannot contain all the necessary information that the user requires. In these cases, an attempt has been made to cite reference documents, such as military standards, handbooks, and reports.

AN AUTOMATED VERSION OF THIS GUIDANCE, PLUS SOME COMPUTERIZED TOOLS ARE AVAILABLE

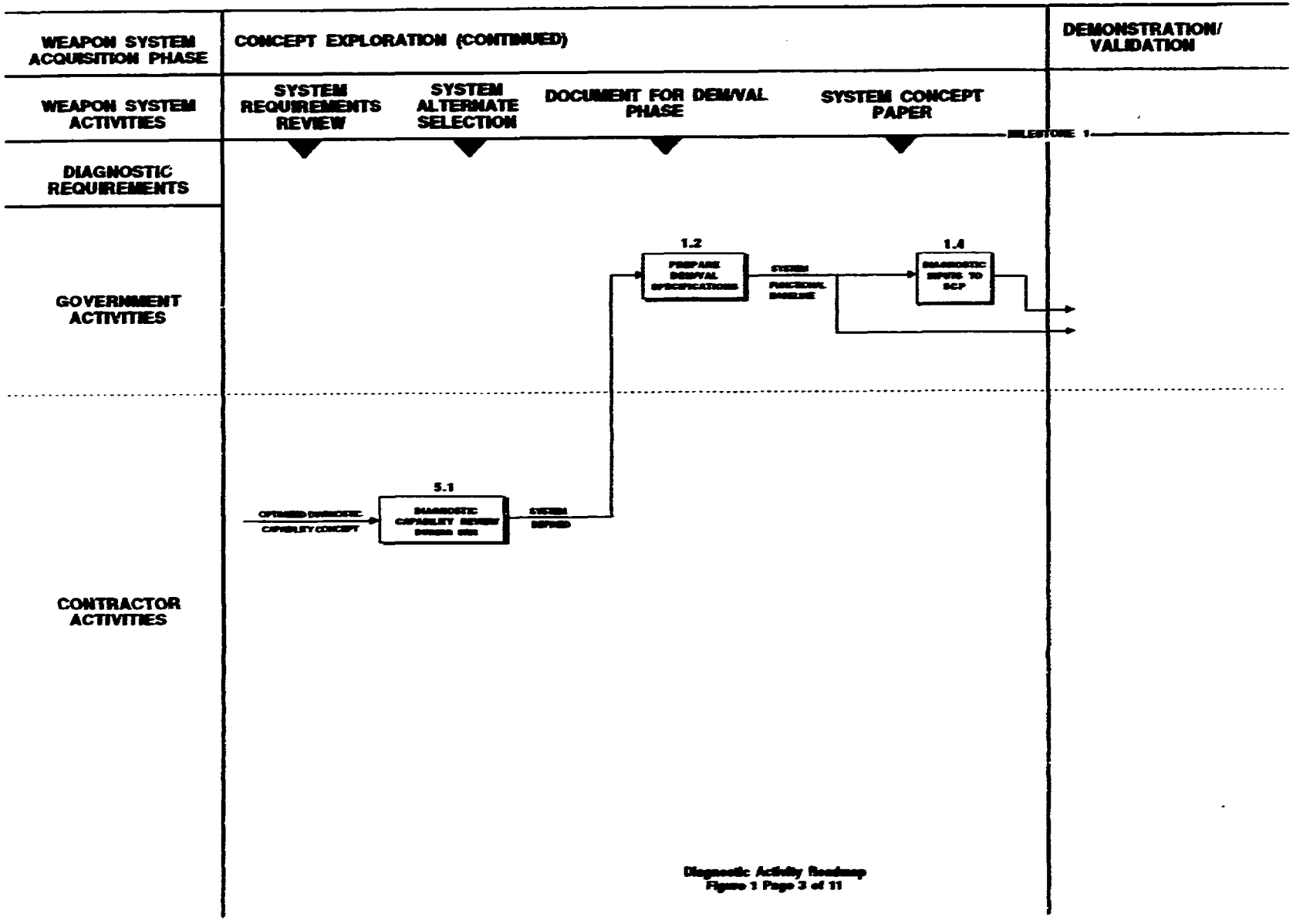
To aid in the use of these guides, a computerized, interactive version of all three guides has been developed. If you are interested in obtaining these software programs, you may contact Rome Air Development Center, RADC/RBET, Griffiss Air Force Base, New York, 13441-5700, (telephone number: 315/330-4726, AV 587-4726).

WEAPON SYSTEM ACQUISITION PHASE	OPERATIONAL REQUIREMENTS	CONCEPT EXPLORATION
WEAPON SYSTEM ACTIVITIES	SON /MNS/PPBS PROCESS	
DIAGNOSTIC REQUIREMENTS		
GOVERNMENT ACTIVITIES	1.1 [illegible]	
CONTRACTOR ACTIVITIES		

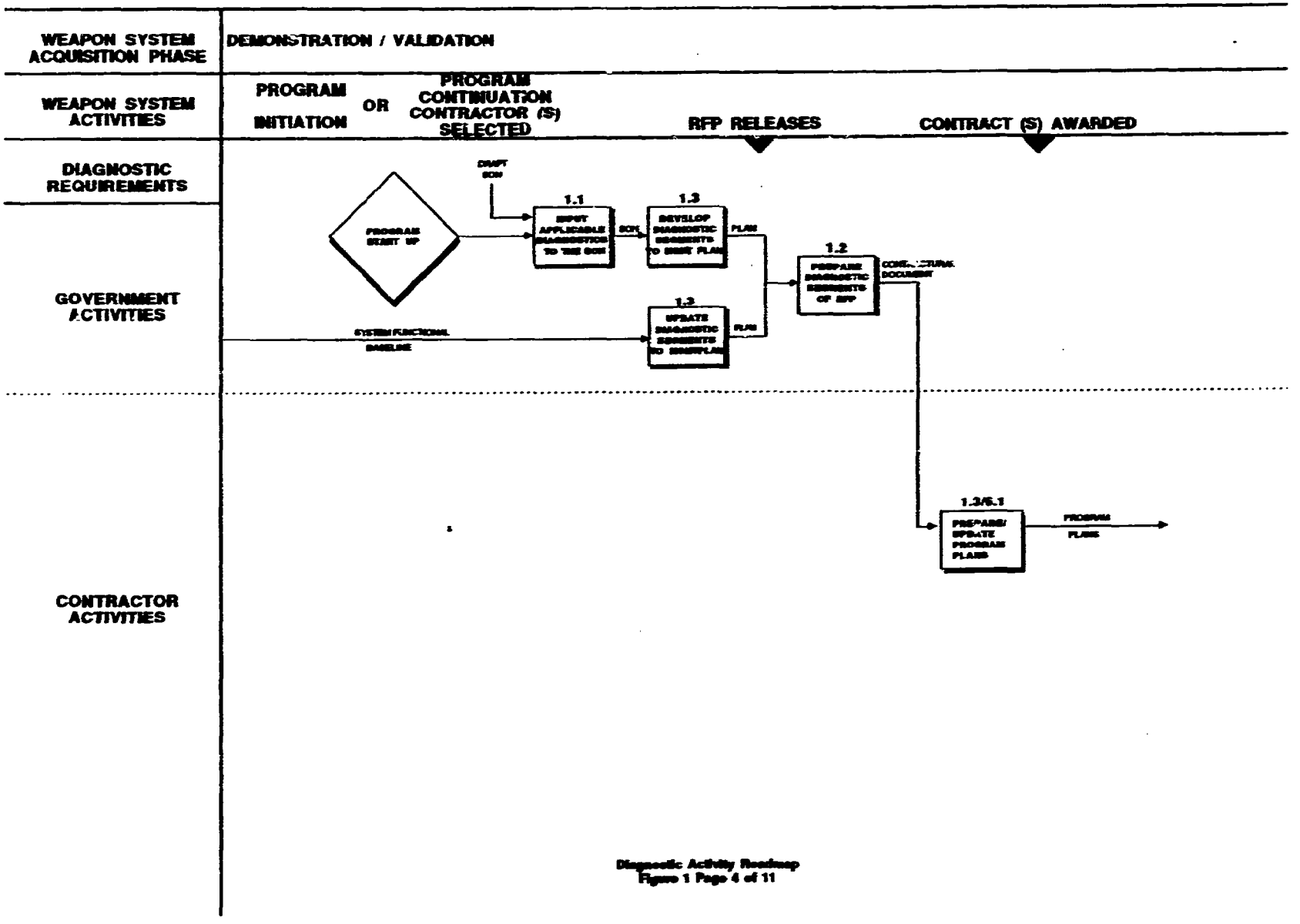
Diagnostic Activity Roadmap
Figure 1 Page 1 of 11



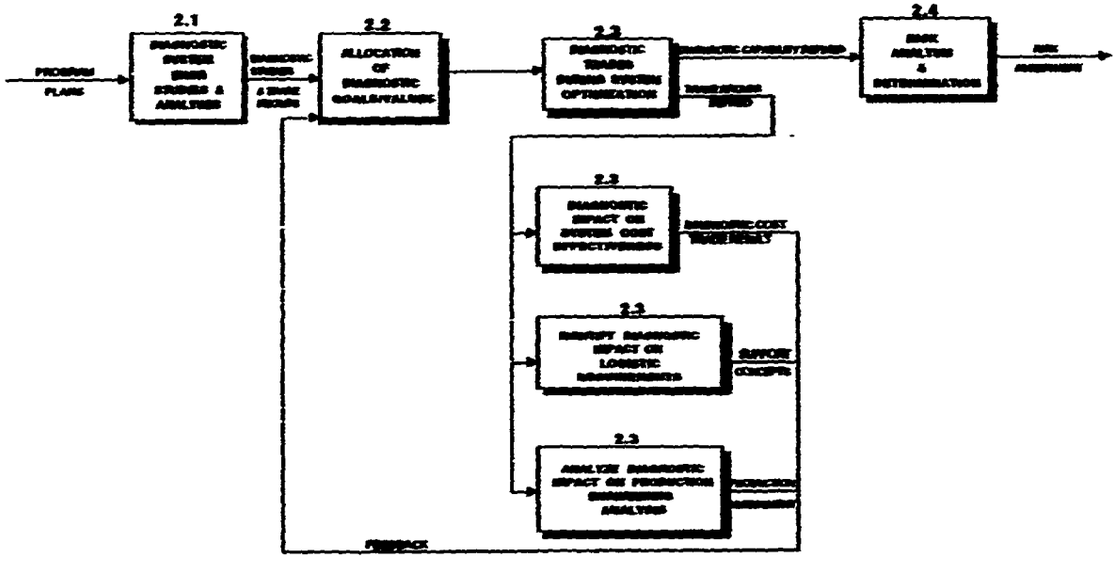
Diagnostic Activity Roadmap
Figure 1 Page 2 of 11



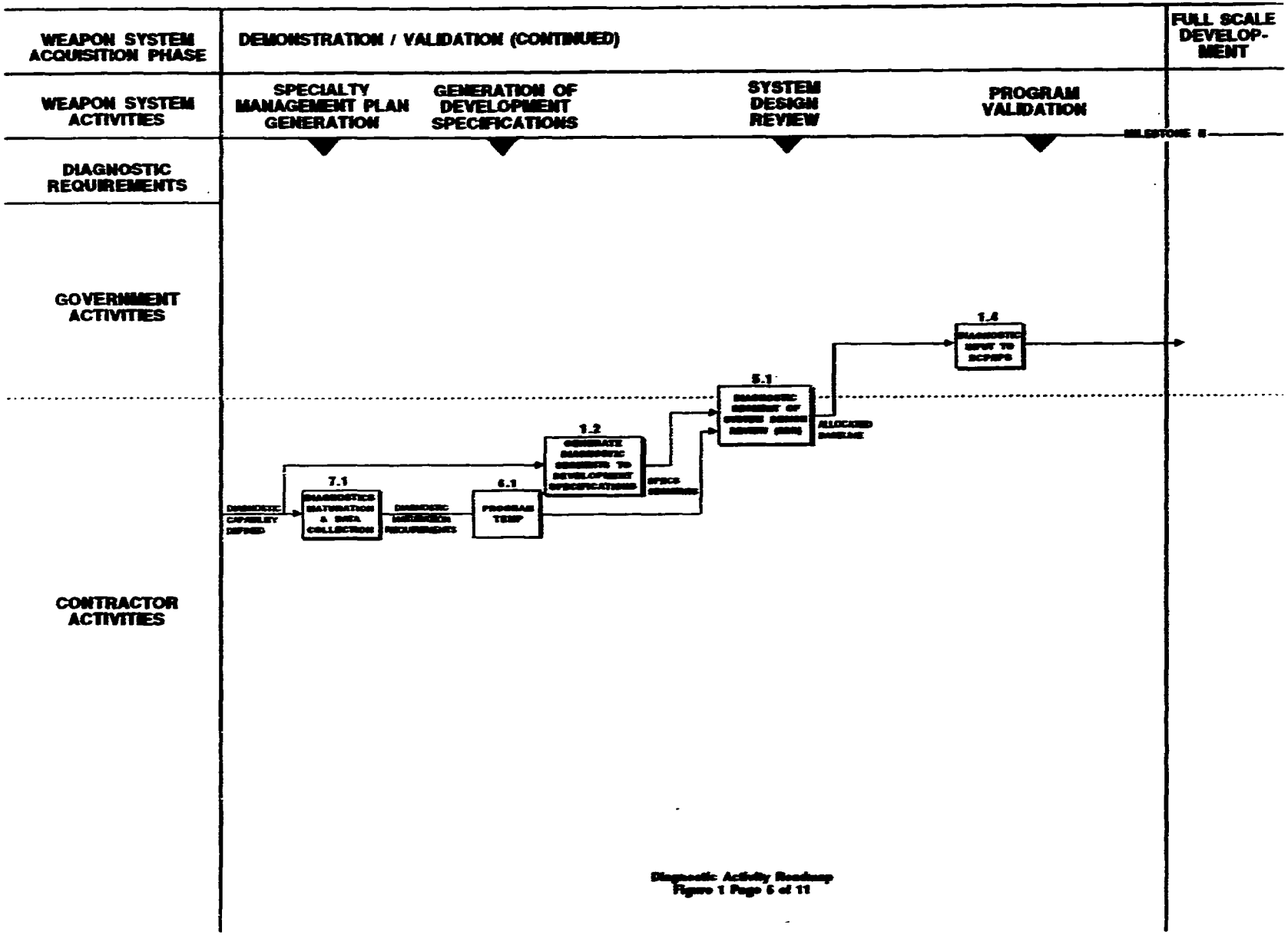
Diagnostic Activity Roadmap
Figure 1 Page 3 of 11



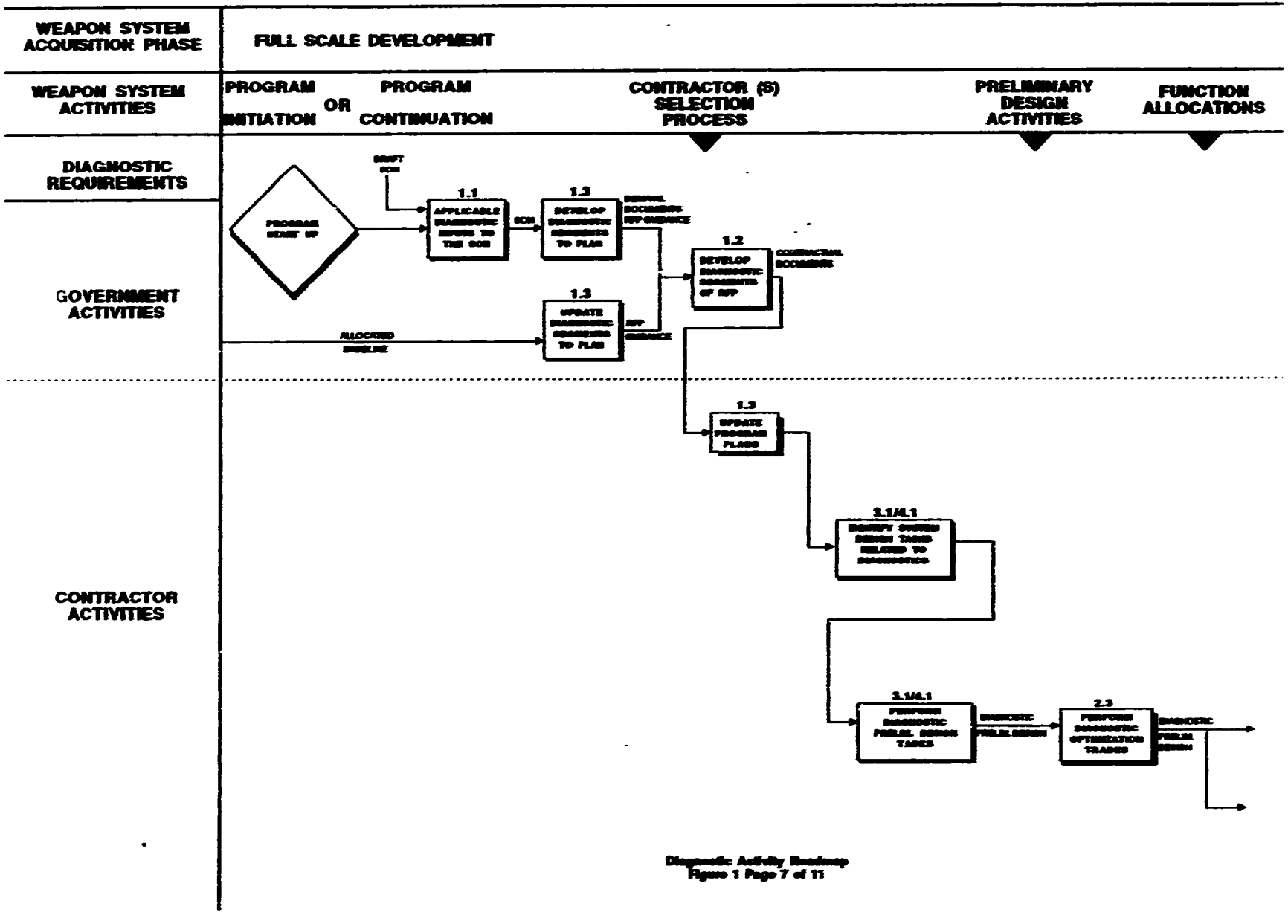
Diagnostic Activity Roadmap
Figure 1 Page 4 of 11

WEAPON SYSTEM ACQUISITION PHASE	DEMONSTRATION / VALIDATION (CONTINUED)	
WEAPON SYSTEM ACTIVITIES	SYSTEM ENGINEERING DESIGN PROOFING/PROTOTYPING LOGISTIC SUPPORT ANALYSIS REPAIR LEVEL ANALYSIS SUPPORT EQUIP. PLAN	
DIAGNOSTIC REQUIREMENTS		
GOVERNMENT ACTIVITIES		
CONTRACTOR ACTIVITIES		

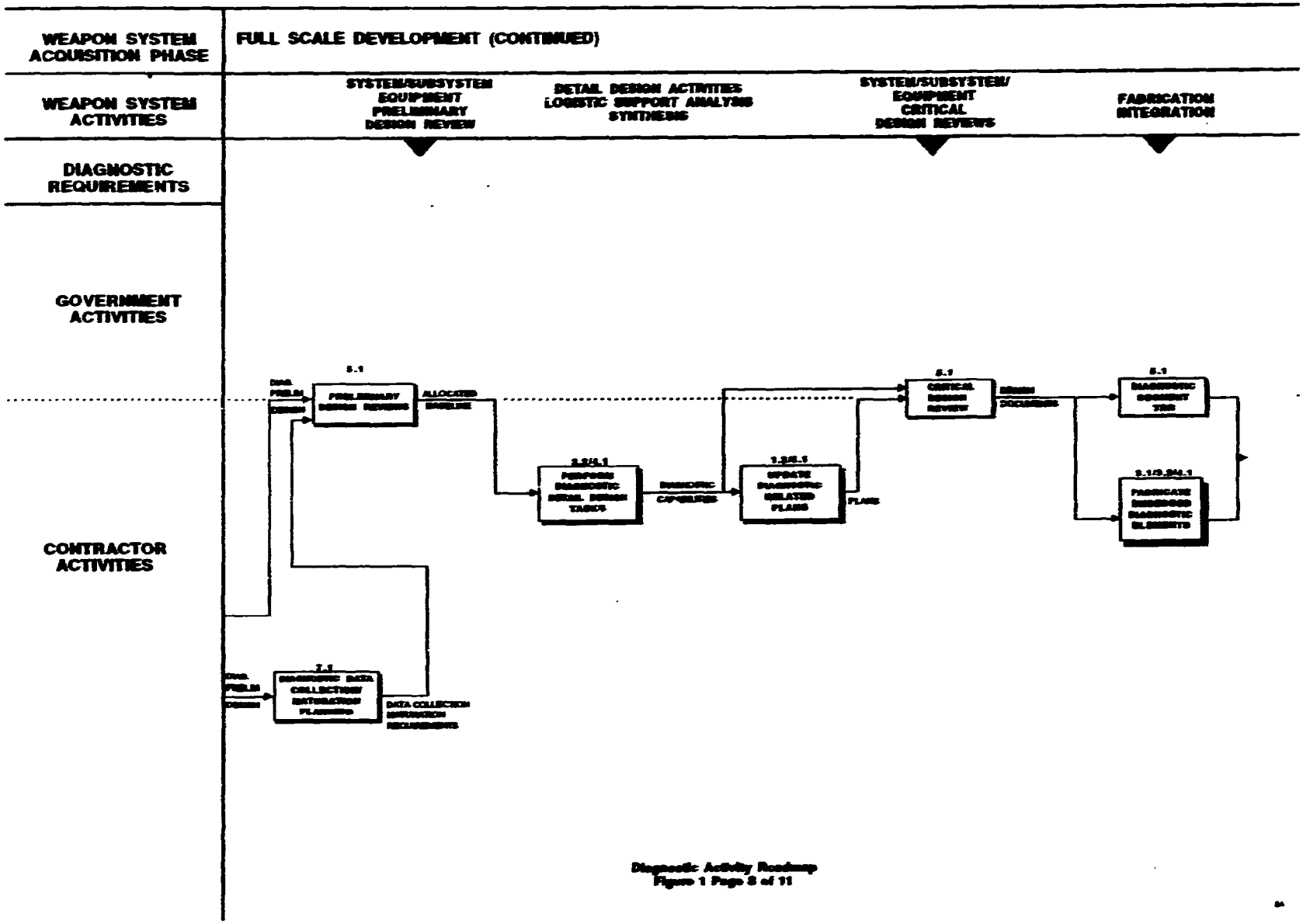
Diagnostic Activity Roadmap
Figure 1 Page 5 of 11



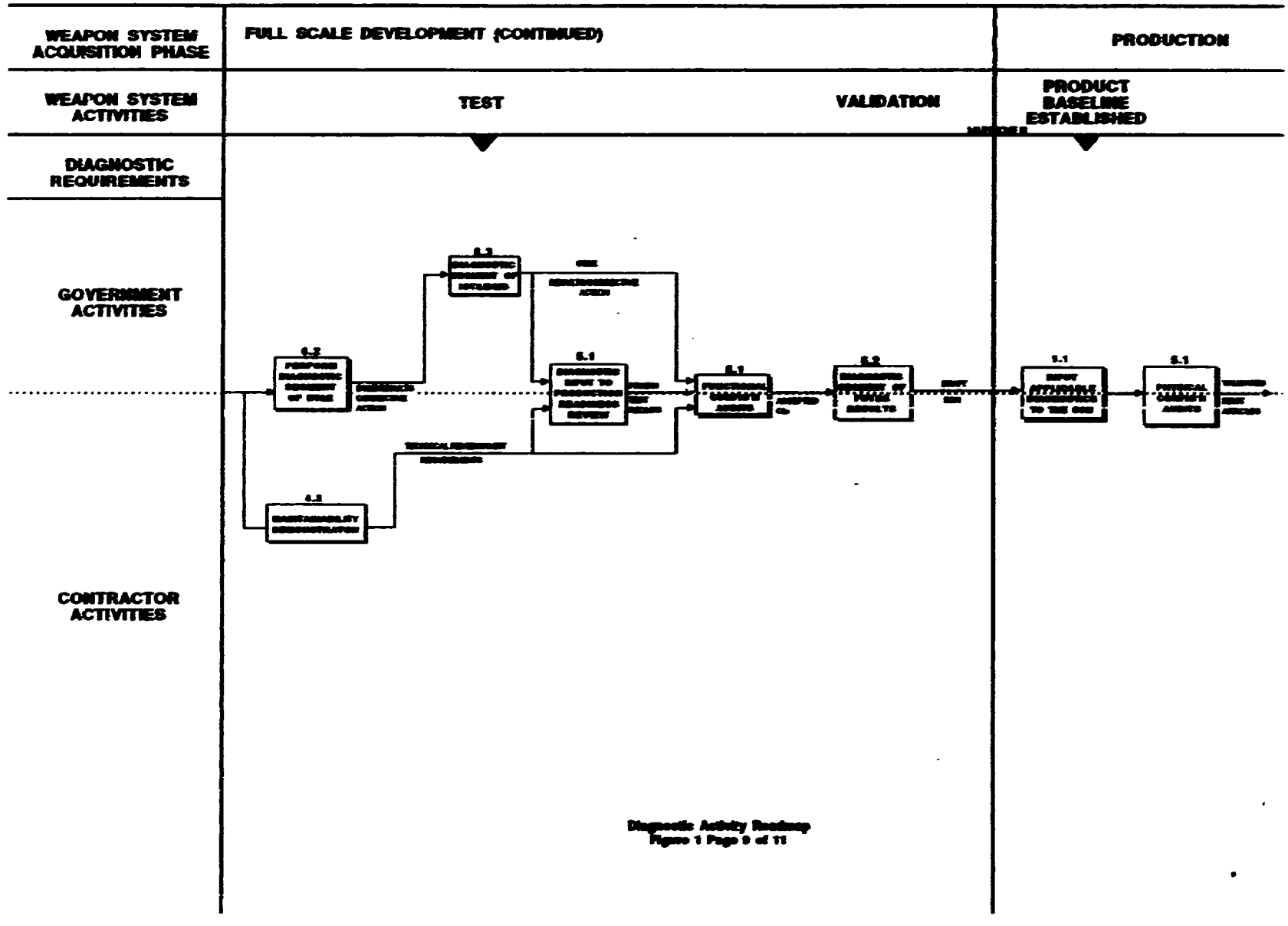
Diagnostic Activity Roadmap
Figure 1 Page 6 of 11



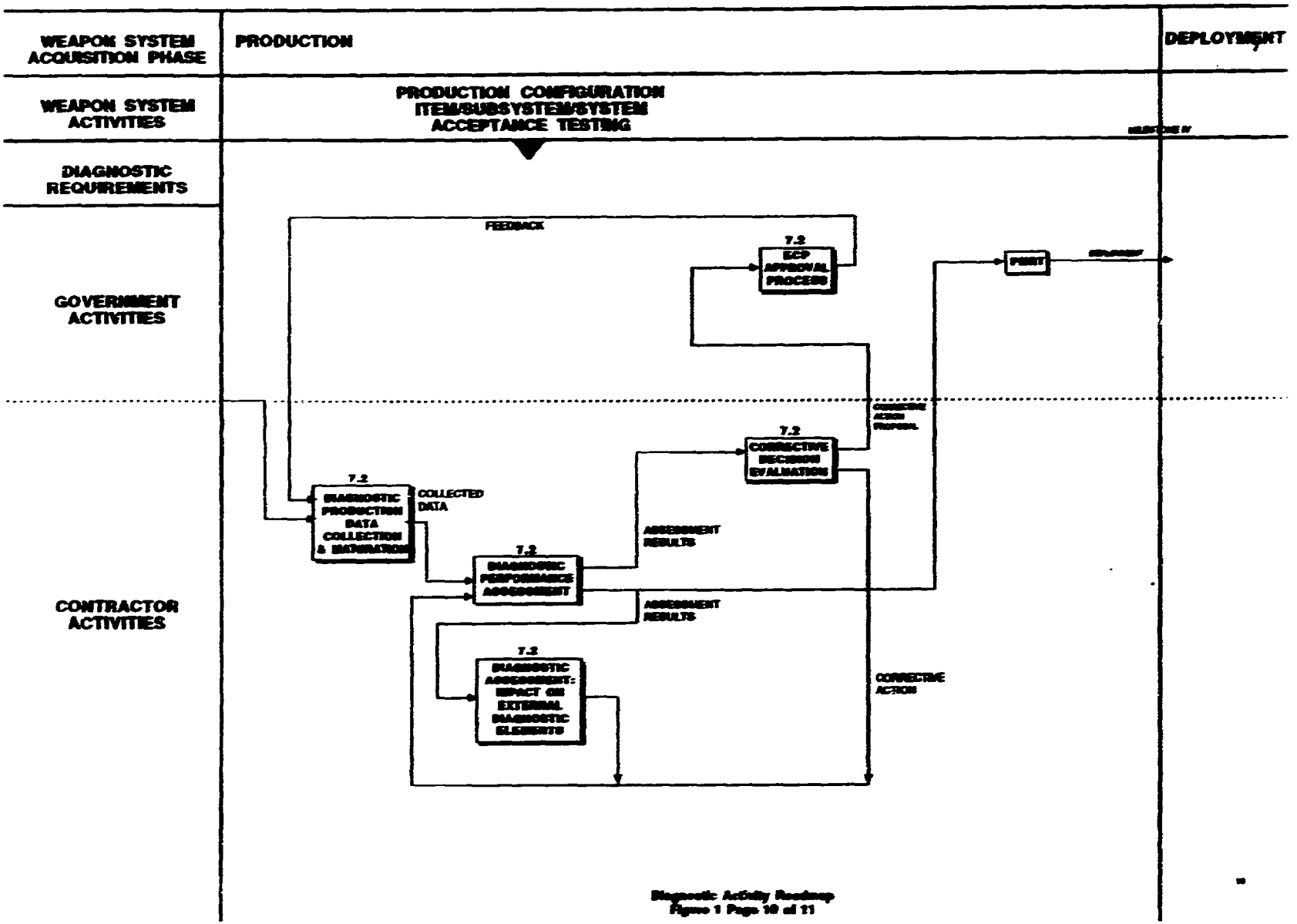
Diagnostic Activity Roadmap
Figure 1 Page 7 of 11



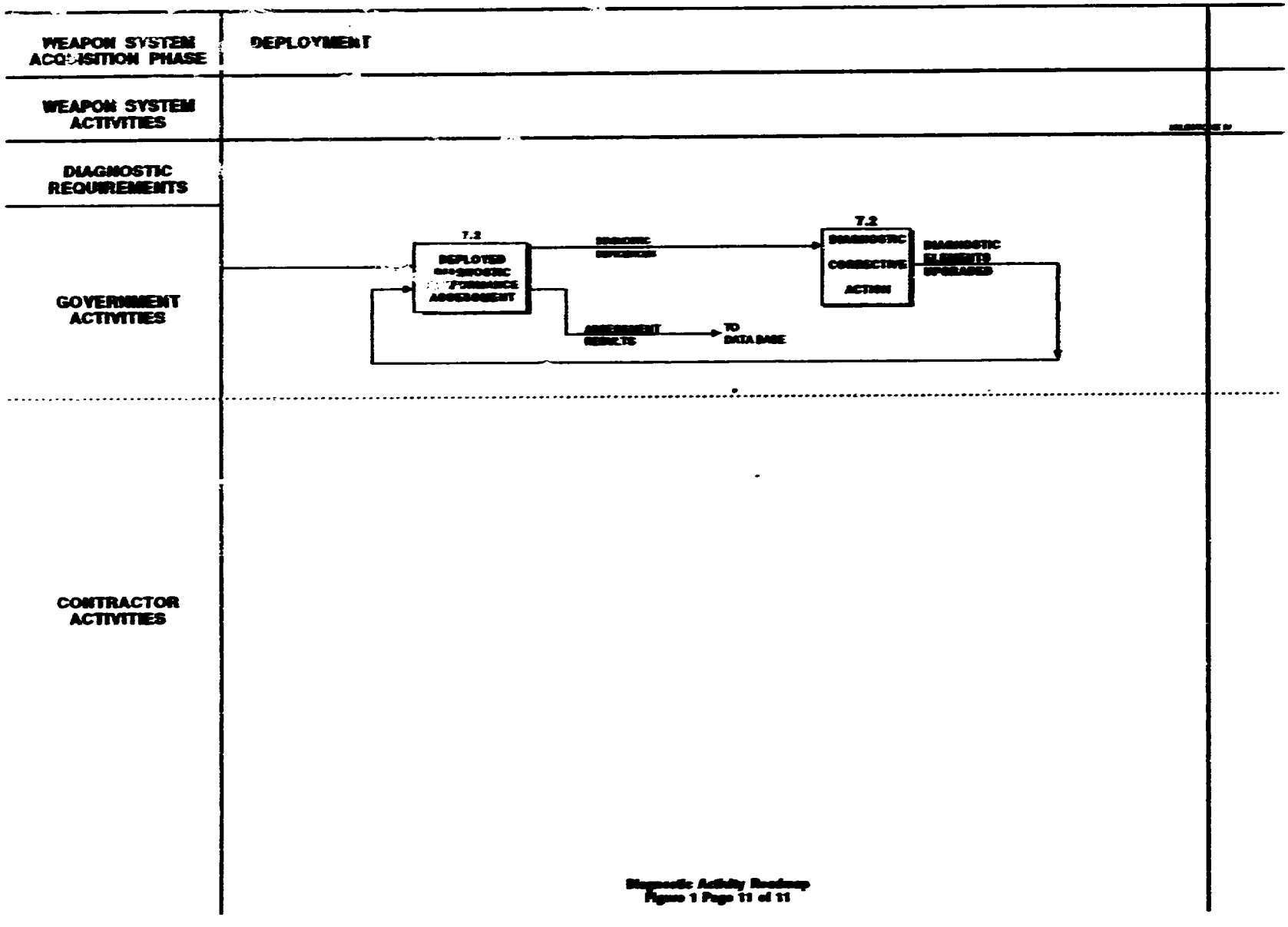
Diagnostic Activity Roadmap
Figure 1 Page 3 of 11



Diagnostic Activity Roadmap
Figure 1 Page 9 of 11



Diagnostic Activity Roadmap
Figure 1 Page 10 of 11

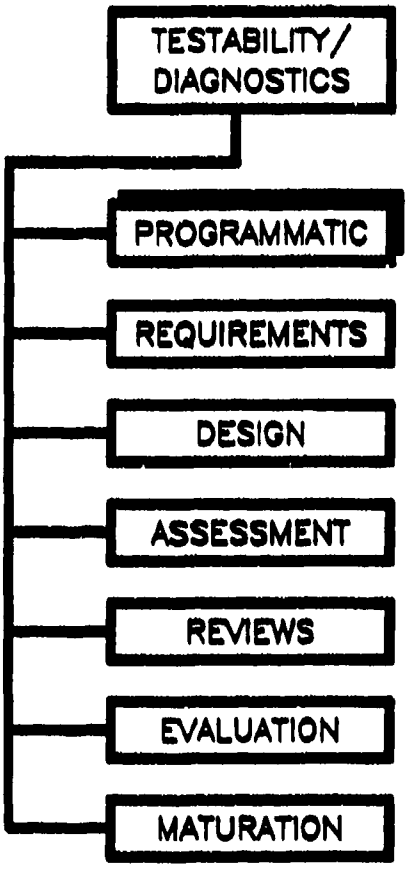


Diagnostic Activity Roadmap
Figure 1 Page 11 of 11

ESTABLISHING AND JUSTIFYING A PROGRAM FOR ACQUIRING A DIAGNOSTIC CAPABILITY

OVERVIEW

DoD organizations are often disappointed at the performance of a weapon system's diagnostic capability once it is deployed. This disappointment often results in frustration by the user, an adversarial relationship between the acquirer and the producer, and costly engineering changes. The fact is that the quality of the acquired diagnostic capability is a two-way street. In the case of the government, careful consideration must be given to what you want the contractor to deliver. In the case of the contractor, he must be dedicated to producing a quality product. Specifying fault detection and isolation requirements is a difficult, complex job for the Government Program Manager. Justifying his program to a higher authority in clear, concise terms is essential. Establishing realistic and feasible plans for satisfying these requirements is a prime responsibility of the Contractor Program Manager. Implementing these plans is the responsibility of the weapon system designer. Without a clear understanding and close cooperation among these people, production of a less-than-satisfactory diagnostic capability is inevitable.



IMPORTANT CONSIDERATIONS TO BE ADDRESSED

Reqmt.

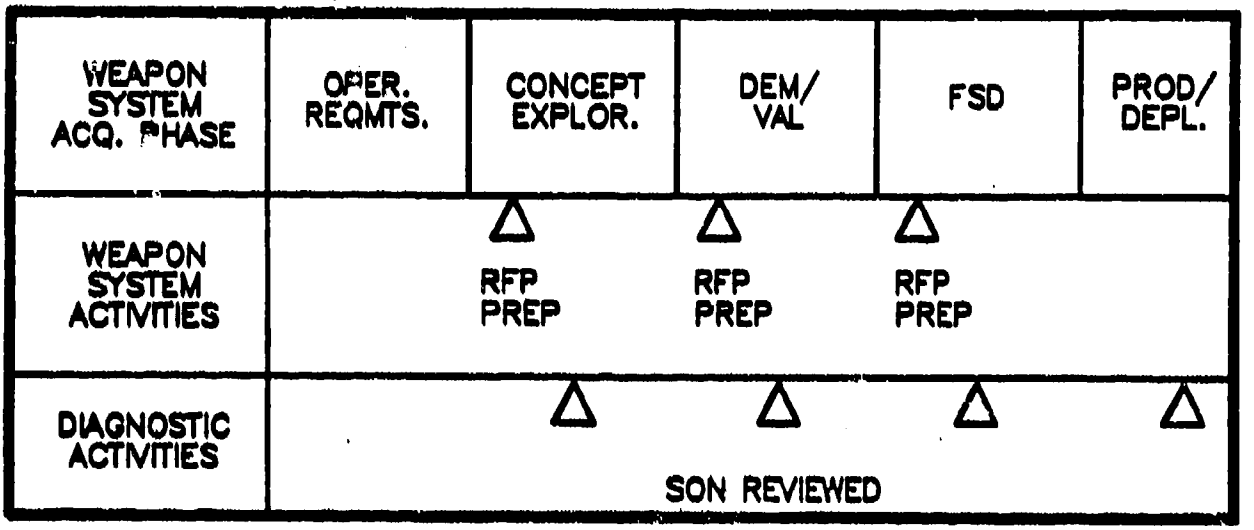
1.1 Review the Statement of Need (SON) to assure a clear understanding of the basis for the development program.

1.2 Diagnostic considerations are a very important part of your proposal.

1.3 Tie diagnostic capability plans to the system engineering plans.

1.4 Make sure that specific information on diagnostic and capability issues are available for inclusion in SCPs and DCPs.

REVIEWING A STATEMENT OF NEED	REQUIREMENT #1.1



DIAGNOSTIC ACTIVITY

The major consideration in the initiation of a weapon system acquisition program is the preparation of a Statement of Need. Each of the services has its own designation for this document. For a major weapon system, DoD Instruction 5000.2 designates this document as a "Mission-Need Statement (MNS)." For less-than-major systems, the services use other terms, such as "Operational Requirements Document" and "Required Operational Capability." It is important that these documents reflect these operational requirements in terms which can be properly interpreted to produce diagnostic requirements.The contractor is not involved in the generation of this document but must be aware of it's content since it forms the baseline upon which the diagnostic requirements are derived.

PROCEDURE

Each of the military services has issued policy directives and guidance relating to the preparation of a Statement of Need (SON). DoD Instruction 5000.2 delineates the format for an MNS. This format does not differ appreciably from the formats used for less-than-major new starts, thus the following guidance will be discussed in relation to the MNS.

The SON is issued prior to Concept Exploration. When the Concept Exploration Phase is not conducted, the SON should be issued prior to initiation of work. In addition, the validity of the SON can be reevaluated prior to the initiation of Dem/Val, FSD, and Production.

GUIDANCE

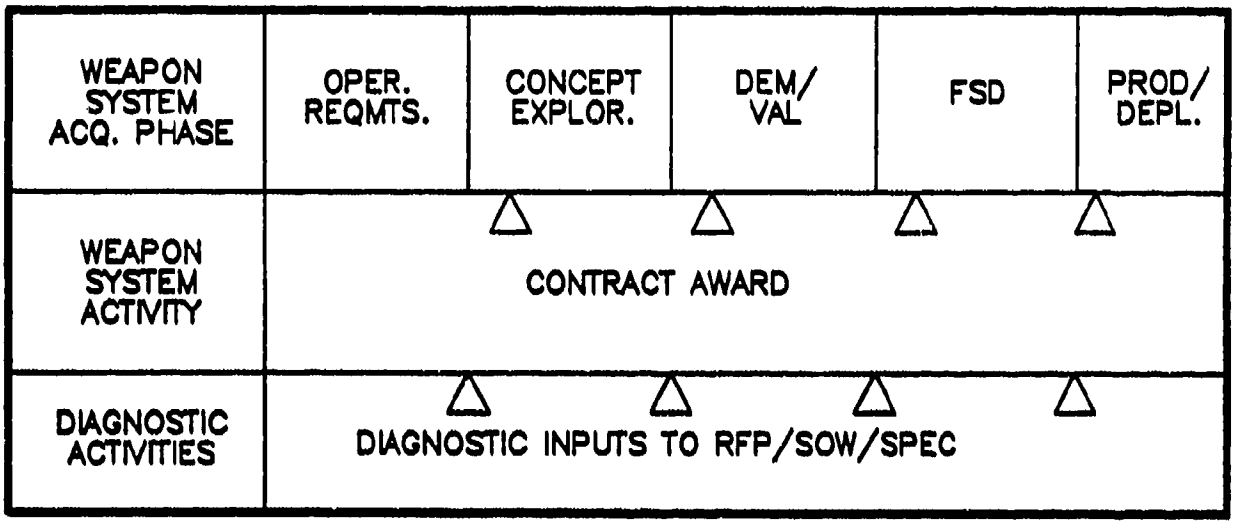
It is almost as important to ensure what should not be put into an SON as what should be put into an SON. From a diagnostic point of view, initially there are no diagnostic requirements per se, only requirements which reflect a threat and mission and operational needs, plus certain constraints put on the weapon system, such as resource limitations. At the initiation of a weapon system development, it is important that the Government Program Manager and his contractor not be limited by establishing premature diagnostic requirements, such as a certain percent fault detection/fault isolation to a given unit, or an MTTR. Rather, the contractor should be given the flexibility to derive the diagnostic requirements from mission needs, such as sortie rates, mobility requirements, and the mission scenario.

The designer's interest is mainly centered on the content of the weapon system's specification and not the SON. However, it may prove useful for the designer to review the SON to facilitate an understanding of the basis for the specification.

CHECKLIST

☑ Am I satisfied that the system specification adequately reflects the SON?

RESPONDING TO AN RFP, SOW, SPECIFICATION	REQUIREMENT #1.2



DIAGNOSTIC ACTIVITY

Clear, concise, and feasible provisions must be inserted into the Request for Proposal, the Statement of Work, and the System Specification, as a means for assuring that the contractor and his subcontractors have a clear understanding of what is required of the diagnostic capability.

PROCEDURE

One of the initial tasks which must be undertaken by the Government Program Manager, at the beginning of each acquisition phase, is the development of the Request for Proposal, which will subsequently lead to a contractual document. For the Concept Exploration Phase, normally the RFP contains a Statement of Work without an associated weapon system specification. The specification is normally invoked no sooner than the Demonstration and Validation Phase. It is normally written by the contractor, with final review by the Government Program Manager. The requirements for this diagnostic capability must appear in a variety of places throughout these documents to assure the acquisition of a satisfactory diagnostic capability. For the Concept Exploration Phase, these requirements are general in nature and allow the maximum flexibility for the contractor to do his job. As the weapon system design proceeds, these requirements become more and more specific. The thrust and content of the provisions contained in these documents vary, depending on the acquisition strategy developed by the Government Program Manager, the phase in which these documents are invoked, and the size and complexity of the weapon system.

Although the bulk of the guidance on the preparation of these documents, which follows, is for the Government and Contractor Program Managers, it is included in this guide for the following reasons:

1. To promote understanding with the designer on how to address diagnostic requirements.

2. To provide information to contractors in their review of draft RFPs/SOWs and Specifications.

3. To provide insight to the contractor in the preparation of the diagnostic portions of his proposals.

GUIDANCE

No guidance on the content and form of what should be included in an RFP and System Specification can be made, which is applicable to every system or equipment. Thus the information which follows must be perceived as examples of how to best specify a weapon system's diagnostic capability. Tailoring is a must.

RESPONDING TO AN RFP:

Diagnostics impacts a number of sections within an RFP, as shown in the following paragraphs.

Special Contract Requirements (Section H) - Contractor incentives and warranties are contained in this section of the RFP. The type and content of these incentives and warranties are almost limitless, depending on the innovation of the RFP writer.

Instructions to Offerors (Section L) - Emphasis must be placed on introducing the concept of integrated diagnostics. Although no standard format exists for this section of the RFP, this section must address the need for managerial and technical information relative to integrated diagnostics and the meeting of the diagnostic requirements. Explanation is required so that the contractor understands that integrated diagnostics interfaces with logistics, reliability, maintainability, testability, human engineering, and safety requirements.

Automation of the diagnostic design process should be addressed, because it can provide for a more efficient and effective design process. It is not the government's job to dictate the use of these design tools, but rather to encourage their use. This can be accomplished by adding provisions to the Instructions to Offerors relating to:

- o A discussion of design aids which will facilitate the design and integration of the diagnostic capability into the system engineering process
- o The development and use of a diagnostic data base which supports the application of these tools
- o Identification of how automation will reduce risk in the design of the diagnostic capability
- o Means for providing the government with appropriate documentation for understanding and validating the output of the automation process
- o Additional information on motivating the contractor to utilize automation in the diagnostic design process is included in the Computer-Aided Acquisition and Logistic Support (CALS) Implementation Guide, MIL-HDBK-59. Copies can be obtained from the CALS Policy Office, Office of the Secretary of Defense. Included in this guide are means for tailoring CDRLs and DIDs to encourage the use of design automation.

Evaluation Factors for Award (Section M) - This section will likely be written to assure that the proposal writer understands that integrated diagnostics and diagnostic requirements will have a significant impact on the selection of a contractor. The evaluation factors reflect the diagnostic content of the Instructions to Offerors (Section L) from both technical and management points of view. Thus the recognition that integrated diagnostics is part of the system engineering process must be described in this section, along with the ability to utilize advanced technology in solving this problem. From a management viewpoint, the evaluation normally will emphasize the need for a single person being responsible for the entire diagnostic capability, often requiring a person full-time.

In addition to having the evaluation factors reflect the content of the Instructions to Offerors, several other evaluation factors are of importance. These include:

- o The amount and type of specialized education and training given to both contractor program managers and designers which relate to testability and integrated diagnostics
- o The independent research and development conducted by the contractor which relates to testability and diagnostic design tool development and integrated diagnostics demonstrations

RESPONDING TO A STATEMENT OF WORK

The Statement of Work varies, depending on which weapon system acquisition phase is being addressed. Four sample Statements of Work - one each for Concept Exploration, Demonstration and Validation, Full-Scale Development, and Production are contained in the Contractor Program Manager's Guide. The principal attributes of these Statements of Work are:

1. An engineering analysis (including gathering of field data) from a previously fielded weapon system(s) to determine diagnostic capability performance deficiencies experienced

2. Identification of specific risk areas which require design attention

3. A requirement for preparation and implementation of a Diagnostic Capability Maturation Plan, including assets required, activities required, and data collection

4. Thorough analysis of the design of the embedded diagnostics to be completed by CDR

5. Design analysis and specification of the external diagnostic capability, including overlap, by CDR

6. A requirement for demonstration of the diagnostic capability, including a thorough, statistically valid sample in selected areas of the system.

PREPARING A SPECIFICATION:

Preparation of the diagnostic portion of a weapon system specification is a job which necessitates a full understanding of the design and fielding of the diagnostic capability. There is a lack of agreement on a standard methodology for this specification. The contractor must recognize the intricacies of this job to ensure that the specifications utilized to acquire a weapon system clearly define the diagnostic requirements.

What is a Failure?

An initial requirement in the specification is to establish the definition of a failure at system, subsystem, and unit levels. This requirement is essential in demonstrating graceful degradation through the use of fault-tolerant design, reconfigurability, redundancy, and performance monitoring. A failure may be defined in a number of ways, depending on the mission to be performed. It may be defined as causing the mission and performance requirements of the prime system to be compromised.

What Means are Available to Perform Diagnostics?

Too often the word "diagnostic" is used interchangeably with "test." The specification must recognize the different types of diagnostics, which include:

- o Visual observations (e.g., the display isn't working)
- o Symptomatic (e.g., the case is too hot)
- o Test (e.g., a voltage is out of spec.).

Means through which systems' diagnostics can be addressed include:

- o Automatic testing (i.e., embedded or external)
- o Manual troubleshooting, utilizing technical manuals, troubleshooting procedures, manual test equipment
- o Operator and maintenance technicians' observations and various forms of performance monitoring
- o A combination of the above.

What Terms Can be Used in Specifying Diagnostic Requirements?

As indicated previously, various terms may be used to specify diagnostic requirements. A preferred set is contained in an RADC report, "A Rationale and Approach for Defining and Structuring Testability Requirements," (RADC-TR-85-150), August 1985. The set includes the following terms. Additional terms, along with techniques for verification are contained in an RADC report, "BIT/External Test Figures of Merit and Demonstration Techniques", (RADC-TR-79-309).

Fraction of Faults Detected (FFD)

FFD can be defined as that fraction of failures which occur over operating time which can be correctly identified through direct observation or other specified means by an operator and/or other specified personnel under a given set of conditions. The quantitative definition of FFD is:

$$FFD = \frac{F_D}{F_A}$$

Where:

F_A = Number of actual failures (faults) which (will) occur over operating time, T.

F_D = Number of actual failures correctly identified through direct observation and other specified means by an operator and/or other specified personnel under a given set(s) of conditions.

In specifying FFD, all the various means which can be used to detect faults must be taken into consideration. The requirement for FFD should be stringent enough to exclude the application of the types of detection means which are unsatisfactory/unacceptable for the system needs/objectives/philosophies, but flexible enough to allow the contractor to tailor his design cost effectively. In general, the specific nature and mix of the means to be employed to achieve a given minimum FFD should be dependent on results of an analysis of each such alternative and its cost and performance effectiveness, in conjunction with other system/equipment design factors and requirements. The contractor should be tasked to perform such analyses and provide results/recommendations to the procuring activity based on these factors.

The FFD specification parameter must be specifically defined to take into account frequency of failure (failure rates) of the components making up the system. It is only in this way that FFD will be representative of what occurs during operational life.

In specifying FFD, care must be taken to define that set of detection conditions which are acceptable: for example, who can perform the detection function; what are the acceptable means through which detection can be performed; during which equipment status modes can detection be performed (operation, pre- or post-mission checks, etc.); and whether or not a failure must be detected within a certain period of time?

Fraction of Faults Isolated (FFI)

FFI can be defined as that fraction of failures which occur over operating time which can be correctly isolated to x units, or fewer, at a given maintenance echelon through use of specified means, by a maintenance technician or other specified personnel. The quantitative definition for FFI is:

$$FFI = \frac{F_I}{\overline{F_A}}$$

Where:

F_A = Number of actual failures (faults) which (will) occur over operating time T.

F_I = Number of actual failures (faults) which (will) occur over operating time T that can be correctly isolated to x units, or fewer, at a given maintenance echelon through use of specified diagnostic scheme(s)/ procedure(s) (or a defined set of such), by a maintenance technician or other specified personnel.

In specifying FFI, all the various generic means acceptable in general for the mission/operational/maintenance environment which can be used to isolate faults must be taken into consideration. The requirement for FFI should be stringent enough to exclude the application of isolation means which are known in general to be unsatisfactory/unacceptable to the system needs/maintenance philosophy/objectives but are flexible enough to allow the contractor to tailor his design cost effectively. The specific nature and mix of the means to be employed should be dependent on the results of an analysis task (levied on, and performed by, the contractor) of each fault isolation alternative, in conjunction with system/equipment design factors, maintainability requirements, and support system needs. Generally speaking, unless there is clear evidence that unacceptable weight, volume, or cost penalties would accrue, primary diagnostic means based on: (1) signal tracing and analyses through the use of schematics and test equipment, and (2) repetitive item remove/replacement/performance check actions should be avoided.

In specifying FFI, care must be taken to indicate the conditions under which isolation must take place:

- Where it takes place (i.e., Organizational Level, Shop Level)
- What are the acceptable means of isolation (i.e., built-in test, external testers, general-purpose testers, peculiar testers, manual means, degree of manual means)
- Who will perform the isolation (i.e., operator or maintenance technician)
- Its constraints (i.e., prohibition of wholesale removal of units, time allowable)

- Its second isolation tier requirements (what happens after isolation to proper ambiguity level)

- The time constraints levied by the maintainability requirement.

The FFI parameter must be specifically defined to take into account frequency of failure (failure rates) of the components making up the system. It is only in this way that FFI will be representative of what occurs during operational life.

False Alarm Rate

A false alarm is defined as an apparent indication of failure when, in fact, no failure exists. The false alarm rate is the number of false alarms per unit of time.

Intermittent faults can be difficult to distinguish from false alarms during operational test and in use. A properly structured qualification test, however, can exclude the influence of intermittent faults.

False alarm rates are controllable through the use of such design techniques and features as:

- Scope and magnitude of performance monitoring

- Definition of test tolerances

- Transient monitoring and control

- Multiple-run decision logic

- Environmental effects filtering and identification.

Fraction of Erroneous Fault Isolation Results (FEFI)

FEFI is the fraction of BIT or external tester isolations that identify the wrong removable unit (subunit) or group of units (subunits) as failed. FEFI is primarily a design problem resulting either from test system design error or low sensitivity thresholds and tolerance levels of system/equipment components and/or signals. It can have serious consequences by creating confusion during fault isolation and by eroding maintenance technician confidence in the test system. The quantitative definition is:

$$FEFI = \frac{F_E}{\overline{F_A}}$$

Where:

F_A = Number of actual failures (faults) which (will) occur over operating time T.

F_E = Number of actual failures (faults) which (will) occur over time T that are isolated to a nonfailed unit or group of units.

What Does 100% Fault Detection/Fault Isolation Mean?

In defining FFD as a contractual requirement for most programs, it is sometimes simpler to exclude those types of direct detection means (for example, detection through the use of technicians) which would, in general, be unsatisfactory to a given mission environment than to define those that are acceptable. The fact that an FFD requirement is imposed should not imply that 100% of all expected failures should not be detectable. The contractor should be tasked with the development of cost-effective, defined procedures to detect all expected failures. All of these, however, need not be direct means or belong to the type of direct means which are defined as satisfactory for general mission operational use, provided maintainability and other requirements can still be met. Detection can include direct or indirect indications to an operator, the use of maintenance technicians or other personnel performing in accordance with a series of defined routines, or some combination of these.

For FFI 100% coverage is required, which simply states that using a combination of all diagnostic resources, all faults can be isolated, given an adequate amount of time. Applying restrictions in time means that 100% of all expected faults will be isolatable, but a certain fraction (1-FFI) may have ambiguity levels greater than the value stated or be isolatable through means which are definable, but which do not belong to the class of diagnostic means cited as being acceptable for general use in the given mission or use environment. Consideration must be given as to how and where isolation to the faulty unit(s) must take place.

In summary, specifications should indicate a 100% fault detection/fault isolation coverage at each maintenance level (e.g., combinations of automatic and manual troubleshooting means should equal 100%). This does not mean that 100% of faults can be isolated to a given unit within a given time using specific diagnostic resources.

What is Diagnostic Growth?

Another aspect which is recommended to be introduced is that of diagnostic growth--similar in concept to the already established reliability growth. This growth requirement is especially important in the maturation of the system. Figure 2 is a conceptual version of this growth process. Demonstrations that these goals, or requirements, have been achieved at various phases of weapon system development must be tailored to the specific weapon system acquisition strategy. For instance, if the performance of an aircraft is to be evaluated at the conclusion of Dem/Val, then the entire diagnostic capability for the aircraft should reach the specified requirement at that point in time. On the other hand, if only specific units (usually high risk) of a weapon system are developed during Dem/Val, then the diagnostic capability for that specific unit may be demonstrated. The maturation of a diagnostic capability for the entire weapon system, in most cases, will extend into the Deployment Phase.

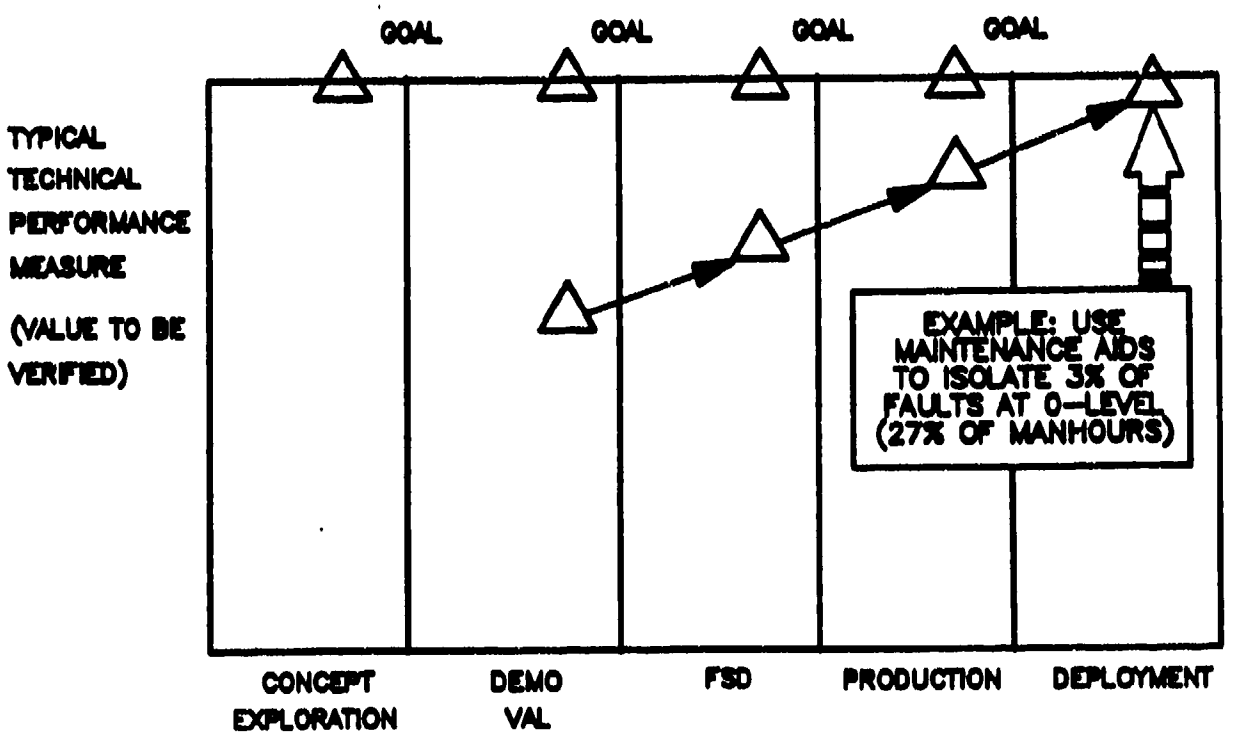


FIGURE 2. DIAGNOSTIC GROWTH CONCEPT

What Does a "Diagnostic Specification" Contain?

The diagnostic portions of the weapon system specification differ, depending on the stage of development. Normally, these specifications take the form of a Preliminary System Specification, resulting from Concept Exploration; a System Specification, derived from the Demonstration and Validation Phase; and a Configuration Item Development Specification, which allocates requirements down to subsystem and item levels. (A more complete definition of the various types of specifications is contained in MIL-STD-490, Specification Practices.) The following examples of diagnostic portions of specifications follow this form. The content must be tailored to fit a specific system requirement.

Preliminary System Specification

The Preliminary System Specification is a result of Concept Exploration Phase studies, prior to conducting the detailed diagnostic/testability requirements analysis during the Demonstration and Validation Phase.

Quantitative diagnostic/testability parameters are not specified in the Preliminary System Specification. Rather, qualitative system-level diagnostic/testability goals are included.

The model paragraphs below may be included in the Preliminary System Specification primarily to alert the performing activity that diagnostics/testability is considered to be an important aspect of the design and that quantitative requirements will be imposed in the final System Specification.

> **3.X.X Design for Testability**
>
> **3.X.X.1 Partitioning. The system shall be partitioned based, in part, upon the ability to confidently isolate faults.**
>
> **3.X.X.2 Test Points. Each unit within the system shall have sufficient test points for the measurement or stimulus of internal circuit nodes so as to achieve an inherently high level of fault detection and isolation.**
>
> **3.X.X.3 Built-In Test. Mission critical functions shall be monitored by Built-In-Test. BIT tolerances shall be set to optimize fault detection/false alarm characteristics. BIT indicators shall be designed for maximum utilization by intended personnel (operator/maintainer).**

System Specification

Quantitative diagnostics/testability requirements are derived from the trade-off analysis during the Demonstration and Validation Phase and are incorporated in the System Specification. Requirements may be expressed in terms of goals and thresholds rather than as a single number. Requirements for diagnostics/testability in a System Specification are provided in the following model.

3.2.4.X Diagnostics/Testability-The_____ system (insert name) shall be designed for testability and constructed to permit the status of the system and the unambiguous location of faults to be confidently determined and reported in a timely fashion.

3.2.4.X.1 Partitioning (Functional Modularity) - System/Subsystems will be partitioned into Line Replaceable Units (LRU) based on the function, minimum or optimum number of interconnections, the ability to fault isolate to the correct unit. LRUs will be subdivided into next level replaceable items (e.g., Shop Replaceable Units, or SRU) based on function, minimum or optimum number of interconnections, and the ability of personnel with the aid of support equipment, training, and technical manuals to fault isolate.

3.2.4.X.2 Test Points and Contacts - Test points and contacts shall be conveniently located and have safe access to signal nodes and shall be provided for the measure or injection to significant parameters for the purpose of evaluating or troubleshooting the circuit mechanisms. The number and choice of accessible nodes shall be sufficient to obtain the equipment fault detection/isolation requirements listed herein.

3.2.4.X.3 Diagnostic Capability - For each level of maintenance: all diagnostic resources shall be integrated to provide a consistent and complete diagnostic capability. A complete diagnostic capability must identify the diagnostic resources that will be used to have full FD/FI coverage. The degree of diagnostic automation shall be consistent with the proposed personnel skill levels and maintenance repair times.

3.2.4.X.4 Built-in Test - Built-in Test (BIT) provisions shall be designed into the _____ system to test system/equipment and to inform the operator of the ability of the equipment to perform a particular mission.

3.2.4.X.4.1 On-Line BIT Performance Monitoring - The on-line BIT performance monitoring features shall be operative and shall provide valid performance indications prior to and during operation. The performance monitoring operation shall be automatic and continuous and shall:

1. *Ensure subsystems are operational and are capable of satisfying their designated mission functions.*

2. *Detect any system failure or degradation which would adversely affect the system's ability to satisfy its mission objectives.*

All BIT implementations of this requirement shall be contained within the system or subsystem hardware and will not degrade mission performance at any time.

3.2.4.X.4.1.1 No-Go Condition Detection - The system on-line BIT performance monitoring features shall detect at least _____ percent of all no-go condition occurrences over the mission time. (As applied at the weapon system level or as applied independently at the subsystem level.)

Continued on the following page...

3.2.4.X.4.1.2 False Alarms - The number of false alarms shall not exceed _____ percent of all indicated no-go condition occurrences or alternately no more than _____ indicated no-go condition occurrences in any integrated 24-hour period of system operating time.

3.2.4.X.4.1.3 Performance Monitor and Self-Test Data - Performance monitor and self-test data shall be transmitted in a manner such that the transmitted data shall follow the actual condition of the system, that is, a malfunction which corrects itself shall change the fault data line accordingly.

3.2.4.X.4.2 Off-Line BIT - The _____ system BIT provision shall furnish the means for an operator to initiate BIT tests for purposes of determining and displaying the functional status of the systems/subsystems including a fault detection and isolation capability. The intended use of the off-line BIT tests is two-fold: as a system readiness test to permit operating crew to accumulate status and fault information on an opportunity basis prior to and during operations; and then verify a fault indicated during operation and to isolate the fault at the Organizational level of maintenance.

3.2.4.X.4.2.1 No-Go Condition Detection - The system off-line BIT features shall detect at least _____ percent of all no-go condition occurrences. (As applied at the weapon system level or as applied independently at the subsystem level.)

3.2.4.X.4.2.2 False Alarms - The number of false alarms shall not exceed _____ percent of all indicated no-go condition occurrences or alternately no more than _____ indicated no-go condition occurrences in any integrated 24-hour period of system operating time.

3.2.4.X.4.2.3 Off-Line BIT Fault Detection - The off-line BIT fault detection capability shall be designed to monitor, detect, and evaluate faults on all system or subsystem functions available at the system or subsystem interface. When a fault or system degradation is detected, the off-line BIT provision may determine the amount of degradation and automatically branch into the appropriate diagnostic fault isolation routine.

3.2.4.X.4.2.4 Off-Line BIT Fault Isolation - The off-line BIT fault isolation routines shall be provided at each fault detection point and shall be automatically entered when a no-go is detected. The off-line BIT shall provide fault isolation to one Line Replaceable Unit _____ percent of the time, fault isolation to _____ or fewer Line Replaceable Units _____ percent of the time. In no case shall the ambiguity group be greater than _____ LRU(s).

Continued on the following page...

3.2.4.X.2.5 Off-Line BIT Fault Isolation Time - The off-line BIT fault isolation time shall be consistent with the requirements of the Organizational level Mean Time To Repair (MTTR) requirements.

3.2.4.X.4.3 BIT Self Test - BIT self test provisions shall be incorporated into the _____ ___ system. The time for the BIT self test shall be less than _____ (and/or the duty cycle of the BIT self test shall be __ ____). The BIT failure rate shall be less than _____ percent of the prime system BIT failure indication rate.

3.2.4.X.4.4 Fail Safe Provisions - The circuits and devices which provide BIT and fault isolation functions shall be designated in such a manner that failure of these circuits and/or devices will not cause a critical failure or unsafe action of the system.

3.2.4.X.4.5 Skill Levels - A personnel skill level of _____ is required to permit the accomplishment of all actions associated with the fault isolation and removal/replacement of LRUs at the Operational/Organizational level. BIT provisions and, where required, Organizational level test equipment and maintenance procedures will be used to provide fault isolation within the MTTR specification.

3.2.4.X.5 Test Equipment Interface - Signals shall be included at the module interface which maximizes the similarity of built-in testing by the equipment and external testing by manual test equipment and/or on ATE systems. The system shall be designed for compatibility for test with the selected or targeted ATE (or _____ (insert test equipment name/designator)). Maximum use shall be made of operational pins to provide test control and access to satisfy the fault detection/fault isolation requirements of external test.

3.2.4.X.6 Test Tolerances - Appropriate tolerances and signal limits shall be established in diagnostic routines at each level which the system/equipments are subject to testing such that false alarms and Retest Okay rates are minimized.

3.2.4.X.7 Technical Information Access Time - Average time required for the maintenance technician to access maintenance technical information shall be less than _____ minutes at the Organizational level.

Configuration Item Development Specification

A model testability specification suitable for inclusion in the CI development specification is provide as follows.

3.2.4.X Diagnostics/Testability - The _____ subsystem/item (insert name) shall be designed for testability and constructed to permit the status of the subsystem/item and the unambiguous location of faults to be confidently determined and reported in a timely fashion.

3.2.4.X.1 Partitioning (Functional Modularity) - Subsystems/items will be partitioned into Shop Replaceable Units (SRU) based on the function, minimum or optimum number of interconnections, the ability to fault isolate the correct unit and the ability of personnel with the aid of support equipment, training, and technical manuals to fault isolate to the correct unit.

3.2.4.X.2 Test Points and Contacts - Test points and contacts shall be conveniently located and have safe access to signal nodes on the unit under test, and shall be provided for the measure or injection of significant parameters for the purpose of evaluating or troubleshooting the circuit mechanisms. The number and choice of accessible nodes shall be sufficient to obtain the equipment fault detection/isolation requirements listed herein.

3.2.4.X.3 Diagnostic Capability - For each level of maintenance: all diagnostic resources shall be integrated to provide a consistent and complete diagnostic capability. A complete diagnostic capability must identify the diagnostic resources that will be used to have full FD/FI coverage. The degree of diagnostic automation shall be consistent with the proposed personnel skill levels and maintenance repair items.

3.2.4.X.4 Built-in Test - Built-in Test (BIT) provisions shall be added to the _____ subsystem/item to satisfy system-level performance monitoring and off-line BIT requirements.

3.2.4.X.4.1 On-Line BIT Performance Monitoring - The on-line BIT performance monitoring features shall be operative and shall provide valid performance indications prior to and during operation. The performance monitoring operation shall be automatic and continuous shall be automatic and continuous and will monitor self-contained signal generating circuitry. All BIT implementations of this requirement shall be contained within the system or subsystem hardware and will not degrade mission performance at any time.

3.2.4.X.4.1.1 No-Go Condition Detection - The system on-line BIT performance monitoring features shall detect at least _____ percent of all no-go condition occurrences. (As applied independently at the subsystem level.)

3.2.4.X.4.1.2 False Alarms - The number of false alarms shall not exceed _____ percent of all indicated no-go condition occurrences or alternately no more than _____ indicated no-go condition occurrences in any integrated 24-hour period of system operating time.

3.2.4.X.4.1.3 Performance Monitor and Self-Test Data - Performance monitor and self-test data shall be transmitted in a manner such that the transmitted data flow shall be transmitted in a manner such that the transmitted data shall follow the actual condition of the system, that is, a malfunction which corrects itself shall change the fault data line accordingly.

Continued on the following page...

3.2.4.x.4.2 Off-Line BIT - The _____ subsystem BIT provision shall furnish the means for an operator to initiate BIT tests at the system level for purpose of determining and displaying the functional status of the system/subsystems including a fault detection and isolation capability. The intended use of the off-line BIT test is two-fold: as a system readiness test to permit operating crew to accumulate status and fault information on opportunity basis prior to and during operations; and to verify a fault indicated during operation and to isolate the fault at the Organizational level of maintenance.

3.2.4.X.4.2.1 No-Go Condition Detection - The system off-line BIT features shall detect at least _____percent of all no-go condition occurrences. (As applied independently at the subsystem level.)

3.2.4.X.4.2.2 False Alarms - The number of false alarms shall not exceed _____ percent of all indicated no-go condition occurrences or alternately no more than _____ indicated no-go condition occurrences in any integrated 24-hour period of system operating time.

3.2.4.X.4.2.3 Off-Line BIT Fault Detection - The off-line BIT fault detection capability shall be designed to monitor, detect, and evaluate faults on all system or subsystem functions available at the system or subsystem interface. When a fault or system function degradation is detected, the off-line BIT provisions shall determine the amount of degradation and automatically branch into the appropriate diagnostic fault isolation routine.

3.2.4.X.4.2.4 Off-Line BIT Fault Isolation - The off-line BIT fault isolation routines shall be provided at each fault detection decision point and shall be automatically entered when a no-go is detected. The off-line BIT shall provide fault isolation to one Shop Replaceable Unit _____ % of the time, fault isolation to ___ or fewer Shop Replaceable Units _____ % of the time.

3.2.4.X.4.2.5 Off-Line BIT Fault Isolation Time - The off-line BIT fault isolation time shall be consistent with the requirements of Mean Time To Repair (MTTR) requirements.

3.2.4.X.4.3 BIT Self Test - BIT self test provisions shall be incorporated into the _____ subsystem/item. The time for the BIT self test shall be less than _____ (and/or the duty cycle of the BIT self test shall be _____). The BIT failure rate shall be less than _____ % of the prime system BIT failure indication rate.

3.2.4.X.4.4 Fail Safe Provisions - The circuits and devices which provide BIT and fault isolation functions shall be designed in such a manner that failure of these circuits and/or devices will not cause a critical failure or unsafe action of the subsystem/item.

Continued on the following page...

3.2.4.X.6 Skill Levels - A personnel skill level of _____ is required to permit the accomplishment of all actions associated with the fault isolation and removal/replacement of SRUs at the intermediate maintenance level. BIT provisions, test equipment and maintenance procedures will be used to provide fault isolation within the MTTR specification.

3.2.4.X.7 Test Equipment Interface - Signals shall be included at the module interfaces which maximize the similarity of built-in testing by the equipment and off-board testing by manual test equipment and/or on ATE systems. The system shall be designed for compatibility for test with target off-line automatic test equipment. Maximum use shall be made of operational pins to provide test control and access to satisfy the fault detection/fault isolation requirements of off-board test.

3.2.4.X.8 LRU Fault Detection/Isolation Requirements - The following requirements apply to fault detection/isolation capability at the intermediate level of maintenance using automatic test resources (ATE/TPS and BIT).

- *Fault isolation shall be _____ percent of all organizational detected failures.*
- *Average (or maximum) test time for GO/NO-GO end-to-end tests shall be less than _____ (minutes/hours).*
- *Maximum rate of false NO-GO indications resulting in Cannot Duplicates and Retest Okays shall be _____ percent of all Organizational level detected failures.*
- *Fault isolation capability shall provide fault isolation to one SRU _____ percent of the time, fault isolation to _____ or fewer SRUs _____percent of the time. In no case shall the ambiguity group size be greater than _____ SRU(s).*
- *Average (maximum) diagnostic fault isolation time shall be less than _____ (minutes/hours).*

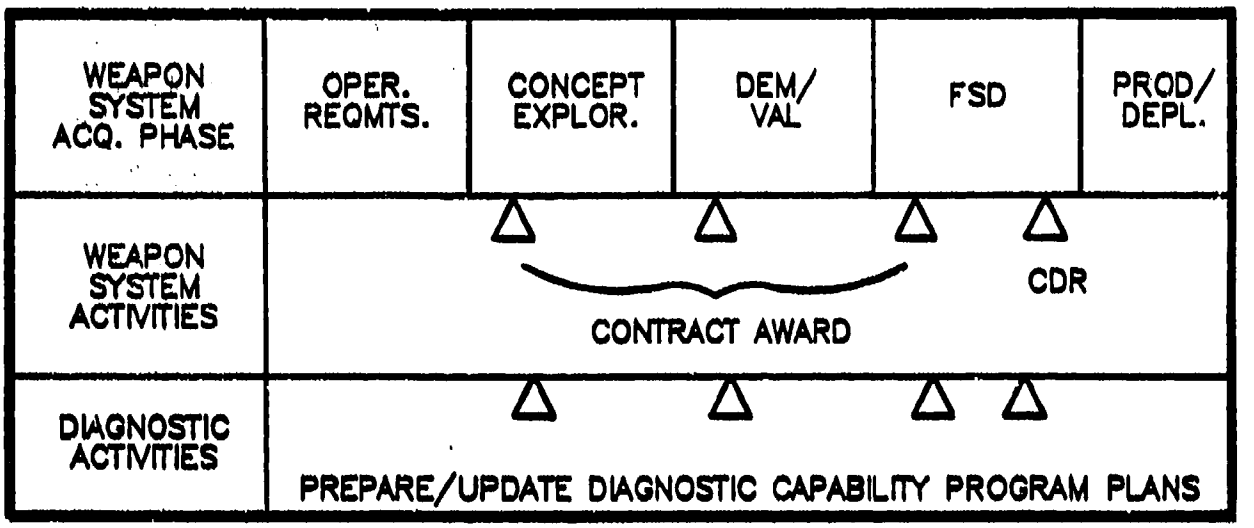
3.2.4.X.9 SRU Fault Detection/Isolation Requirements - The following requirements apply to fault detection/isolation capability at the Depot level of maintenance using automatic test resources (ATE/TPS and BIT).

- *Fault isolation shall be _____ percent of all detected failures.*
- *Average (or maximum) test time for GO/NO-GO end-to-end tests shall be less than _____ (minutes/hours).*
- *Maximum rate of false NO-GO indications resulting in Retest Okays shall be _____ percent of all detected failures.*
- *Fault isolation capability shall provide fault isolation to one component _____ percent of the time, fault isolation to _____ or fewer components _____ percent of the time. In no case shall the ambiguity group size be greater than _____ components.*
- *Average (maximum) diagnostic fault isolation test time shall be less than _____ (minutes/hours).*

CHECKLIST

- ☑ Does the specification cover 100% of the FD/FI requirements for each level of maintenance?
- ☑ Has the concept of "diagnostic growth" been invoked in the diagnostic specifications?
- ☑ Have all the diagnostic element requirements been quantitatively specified? Both fault detection and fault isolation?
- ☑ Has inherent testability been addressed adequately?

DIAGNOSTIC CAPABILITY PROGRAM PLANNING	REQUIREMENT #1.3



DIAGNOSTIC ACTIVITY

Program planning is required to ensure that the development and support of the diagnostic capability is properly managed throughout the acquisition of a weapon system. This planning must address how this development will be conducted to achieve this goal.

PROCEDURE

Program planning for the development of the diagnostic capability is required throughout the acquisition of the weapon system. It begins soon after the award of the first developmental contract and is expanded and updated as the development proceeds.

The program planning can take the form of a single Diagnostic Capability Program Plan or can be incorporated in a series of program plans which are described in a number of programmatic-type military standards. The requirements of these planning documents will be defined in the contract's Statement of Work. To avoid unnecessary duplication of programs plans, the inclusion of this planning information in existing documents is preferred.

GUIDANCE

Each of the management-type plans is required during specific phases of the weapon system acquisition. The following (Table 3) is a listing of the plans and phases where these plans are generally required. The designer's input to the System

Engineering Management Plan is particularly important because of the need to incorporate diagnostic design as an integral part of the system engineering process.

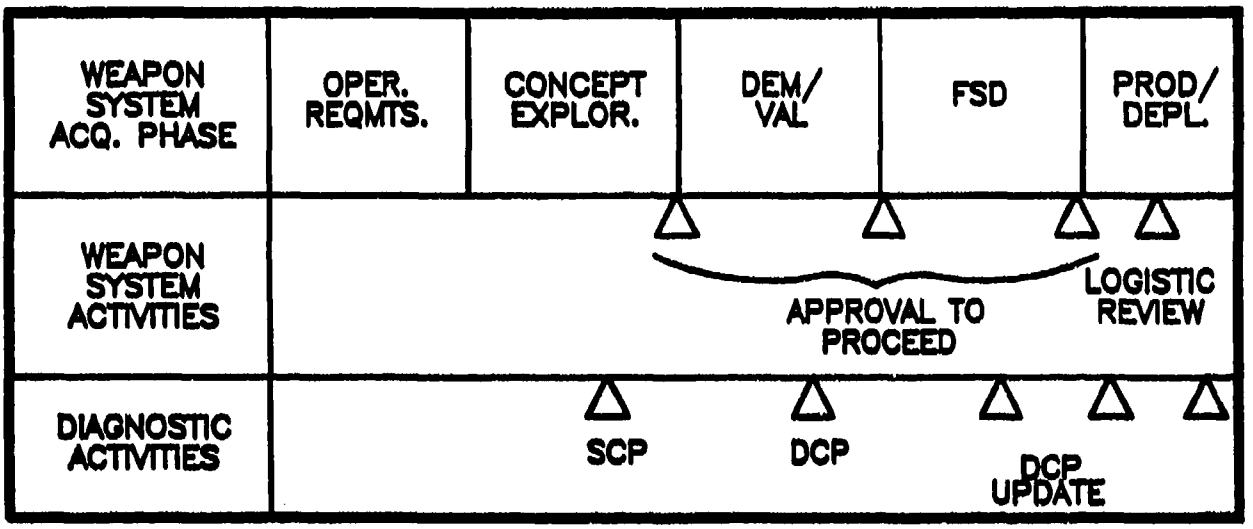
TABLE 3 – APPLICATION MATRIX

PLAN TITLE	GUIDANCE DOCUMENT	PROGRAM PHASE			
		CE	DEM/VAL	FSD	PROD
SYSTEM ENGINEERING MANAGEMENT PLAN (SEMP)	MIL-STD-499	X	X	X	X
LOGISTIC SUPPORT ANALYSIS PLAN (LSAP)	MIL-STD-1388-1	X	X	X	
TESTABILITY PROGRAM PLAN	MIL-STD-2165		X	X	
RELIABILITY PROGRAM PLAN	MIL-STD-785		X	X	
MAINTAINABILITY PROGRAM PLAN	MIL-STD-470		X	X	
INTEGRATED SUPPORT PLAN (ISP)	EITHER DOD-D-5000.29 OR MIL-STD-1388	X	X	X	X
SYSTEM SAFETY PLAN	MIL-STD-882		X	X	
HUMAN ENGINEERING PROGRAM PLAN	MIL-H-46855		X	X	
TEST AND EVALUATION MASTER PLAN (TEMP)	DOD-D-5000.3		X	X	X

CHECKLIST

☑ Has the designer read the SEMP? Contributed to its preparation?

PREPARATION OF SCPs/DCPs	REQUIREMENT #1.4



DIAGNOSTIC ACTIVITY

Prior to Milestones I through V (DoD Instruction 5000.2), the preparation of a paper is required to summarize the results of the acquisition and deployment of a major weapon system. Prior to Dem/Val, a System Concept Paper (SCP) is required. Prior to FSD, preparation of a Decision Coordinating Paper (DCP) is required. An update of the DCP is required prior to Milestones III, IV, and V. The SCP and the DCPs for Milestones II and III are required to secure approval to proceed to the next acquisition phase. Milestone IV is a logistic readiness and support review, which is conducted one or two years after deployment to assure that operational readiness and support objectives are achieved. Milestone V is a major upgrade or system replacement decision, which also requires an updating of the DCP. Diagnostic issues should be addressed in these documents.

PROCEDURE

DoD Instruction 5000.2 delineates the need and the format for both an SCP and a DCP. It is likely that this documentation will address diagnostic issues. Although this type of documentation is required only for major weapon systems, similar documentation may be required by the individual services at significant milestones.

PREPARATION OF SCPs	REQUIREMENT #1.4

GUIDANCE

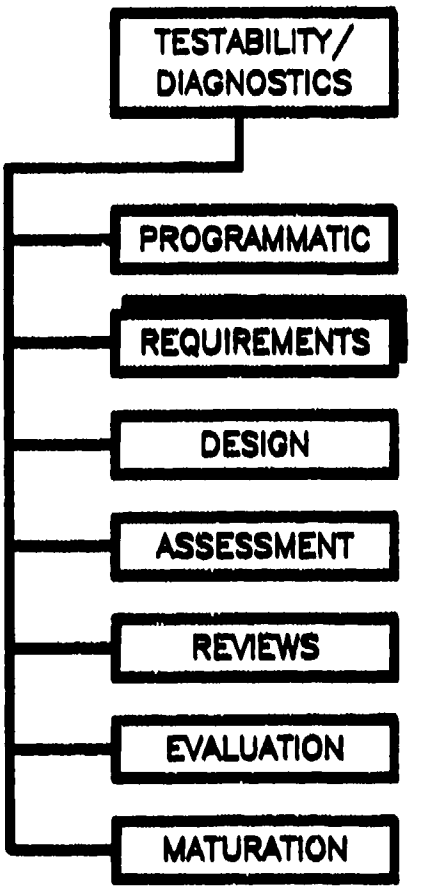
These documents are always prepared by the Government Program Manager. The designer is aware that his inputs to the preparation of these documents may have an inflect on the future of his design.

CHECKLIST

☑ Am I furnishing the Government Program Manager with the proper information for him to adequately justify his program?

ESTABLISHING AND ALLOCATING DIAGNOSTIC REQUIREMENTS

OVERVIEW



Good diagnostics and testability are based on the ability to properly establish diagnostic requirements, which are in turn based on weapon system mission,sustainability, operational, and support requirements and the ability to allocate these requirements at system, subsystem, and unit levels. Lack of appropriate attention to this process results in diagnostic designs with questionable basis and justification. Unfortunately, this process has not been transformed from an art to a rigorous methodology. An integrated series of proven tools does not exist and thus the quality of the analysis depends on the expertise of the persons performing this analysis. The system is further complicated by the advanced weapon system architecture which is now being applied. This architecture involves complex redundancy, reconfigurable elements, and configurations which require graceful degradation. A proper analysis is an integral part of logistic support, reliability, and maintainability analyses and is based on the weapon system's mission scenario and performance requirements. The analyses are an iterative process, which extend over the acquisition phases and often into deployment of the weapon system. Implementation of these analyses is normally the responsibility of the Contractor Program Manager, with the results reviewed by the Government Program Manager.

IMPORTANT CONSIDERATIONS TO BE ADDRESSED

Reqmt.

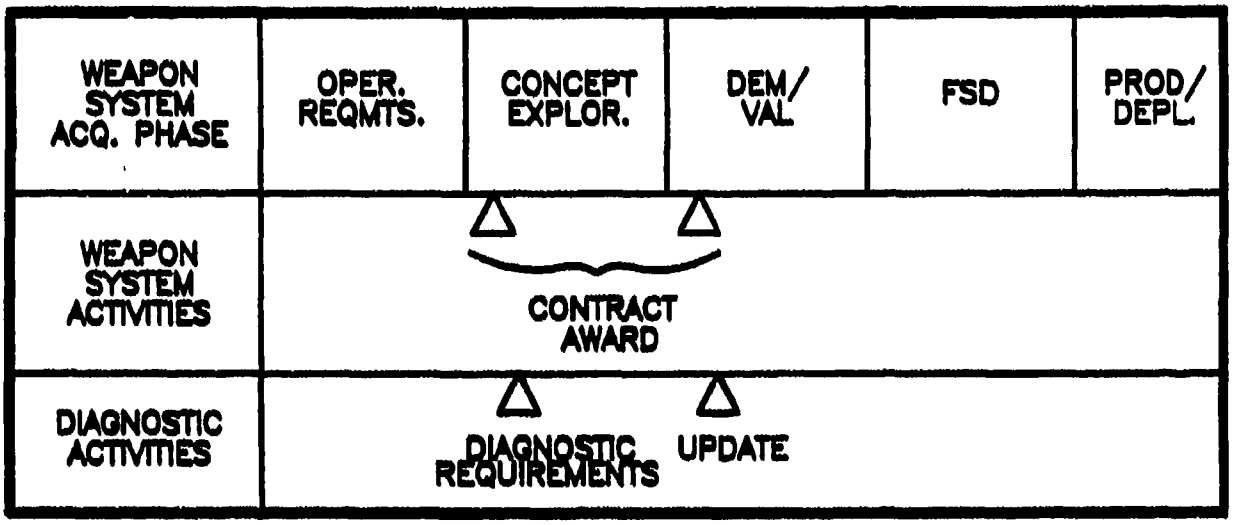
2.1 Translate mission and performance requirements into diagnostic requirements.

2.2 Allocate diagnostic requirements to system, subsystem and unit elements.

2.3 Optimize the mix of diagnostic elements.

2.4 Assess the risk of each diagnostic alternative.

TRANSLATING MISSION REQUIREMENTS	REQUIREMENT #2.1



DIAGNOSTIC ACTIVITY

Diagnostic requirements are identified in the Concept Exploration Phase from an analysis of the prime system mission and operational requirements.

PROCEDURE

The generation of weapon system operational requirements is usually performed by the government from mission studies and analyses based on the Statement of Need for a weapon system. The translation of those requirements and weapon system performance characteristics into diagnostic requirements is performed during Concept Exploration. The tasks involved in translating these requirements may be performed by the contractors or the government depending upon the acquisition process selected during Concept Exploration. For "in-house" programs, this task is performed by government engineers. Frequently, however, the translation of mission and performance characteristics into diagnostic requirements, the selection and integration of the diagnostic elements to meet these requirements, the allocation of these requirements to subsystem and unit level, and the assessment of risk is performed by the weapon system contractors in a competitive environment.

The proper implementation of this task is that it be performed in conjunction with the system engineering and logistic support analysis process and include synthesis and analysis of the various mixes of resources which make up a total diagnostic subsystem. The diagnostic requirements analysis process involves the development of a strategy for a comprehensive diagnostic capability including a mix of resources to be defined for providing FD/FI capability at each level which the system is subject to maintenance.

Descriptions of tools that are available to assist in the entire process of establishing and allocating diagnostic requirements are contained in Appendix C.

In order to translate mission and operational requirements to diagnostic capability, it is important to postulate a "diagnostic subsystem." Characteristics defining the capability of the "diagnostic subsystem" represents the results of the translation. In other words, one must change mission requirements into diagnostic capability definitions in order to successfully complete this task.

The diagnostic elements constituting the diagnostic subsystem include embedded support, support equipment at all levels of maintenance, technical data in all its forms, and personnel numbers and required skill levels.

In order to be responsive to weapon system mission and performance requirements, it is essential that the translations start by reviewing all the requirements documentation and studies. The key document is the Statement of Need which contains the weapon system mission and operational requirements. Also important is the prime system architecture driven by technology infusion into the prime system. This is an essential element in the translation since many prime system architectural concepts contain an inherent diagnostic capability which must be identified and addressed early in the analysis process.

There are two key factors which will influence the translation of weapon system mission and operational requirements into diagnostic requirements. They are:

- o Specific requirements as spelled out in the Statement of Need
- o Available technology.

Analysis of these specific requirements will translate to both requirements for the diagnostic capability as well as constraints on the diagnostic subsystem dictated by the operational parameters. The technology will impact the inherent diagnostic capability of the prime system architecture as well as impact the assessment of risk of the final diagnostic subsystem implementation.

Based upon the above analyses, translation of mission and operational requirements to a diagnostic capability will result in a preliminary set of diagnostic requirements for the entire diagnostic subsystem. The optimum mix of diagnostic elements which constitute the diagnostic capability will follow the requirements allocation to the weapon system, subsystem and unit levels.

During the Demonstration/Validation Phase and Full-Scale Development Phase the detailed trade studies will formally optimize the diagnostic element mix and provide implementation specifications for the diagnostic subsystem to be produced. This process

is obviously iterative but most dependent upon a thorough job of mission and performance requirements analysis and initial translation into diagnostic requirements.

For example, the Dem/Val Phase may result in a System Specification only, with the allocation of the system requirements to be performed and redefined in the FSD Phase. For some less-than-major systems, Dem/Val Phase may be bypassed altogether. In this case, both the system-level specifications may be developed during FSD Phase. The analyses described within this section should be performed at appropriate points based upon, and commensurate with, the level of detail achieved in the definition of the system and the definition of the support and maintenance concepts for the system.

GUIDANCE

Tools are available to assist in establishing weapon system diagnostic requirements. Appendix C has a compilation of these tools. Mainly, they address the logistic support analysis process along with readiness and cost models. However, there is no formal DoD model for translating mission and operational requirements into a diagnostic capability. Using system engineering approaches defined in MIL-STD-499, along with available models, the contractor can, indeed, develop an initial set of diagnostic subsystem requirements which are traceable to weapon system requirements, weapon system priorities, and available technology.

Success in translating mission and operational requirements into diagnostic requirements is embodied in the ability to develop higher order measures for defining weapon system characteristics that relate to fault detection and fault isolation parameters.

Typical weapon system characteristics which must be evaluated include the following:

- Probability of Mission Success
- Availability
- Utilization Rate
- Population
- Turnaround Time
- Threat
- Mobility
- Safety
- Alert
- Deployment
- Basing
- Weight
- Repair Concept
- Personnel
- Training
- Cost
- Etc.

Typical weapon system priorities are as follow:

- War fighting capability
- Survivability
- Mobility
- Manpower
- Life Cycle Cost.

TRANSLATING MISSION REQUIREMENTS	REQUIREMENT #2.1

During the Concept Exploration Phase, mission-oriented measures are overriding for diagnostic requirements generation. Resource criteria (manpower, cost, facilities, etc.) become significant during synthesis of specific diagnostic element mixes. The mission data to be collected and considered for generating the diagnostic requirements is as follows:

- o Mission scenarios definition (prioritized in order of criticality)
- o Mission rate/length
- o Mission operation (continuous vs. intermittent)
- o Mission phases
- o Time demands and operational constraints per mission phase
- o Subsystem/function utilization per mission phase (survivability or safety critical)
- o Functions/failures impacting personnel safety
- o Functions/failures impacting system/equipment safety (sustainability or mission critical)
- o Functions/failures impacting mission success (per mission phase).

A key diagnostic parameter to be determined through the analysis of mission requirements is the maximum failure latency per operating function for each mission phase. This parameter will drive the fault detection requirements which, in turn, serve as the basis for BIT design. Failure latency is the elapsed time between fault occurrence and failure indication. Maximum failure latency is the maximum allowable time between the occurrence of a fault and the reporting or "handling" of the failure. As a simplistic example, if a fire control system fault occurs, and the fire control system function is highly critical to mission success, then the maximum failure latency will be very small -- perhaps expressed in microseconds or nanoseconds. The fault detection (FD) time requirement will reflect the failure latency factor -- thereby driving the BIT technique to provide concurrent performance monitoring. Fault tolerance through redundancy may be required or considered. This simplistic example is made more complex by factoring in the time demands per mission phase of the fire control system. It is made still more complex by factoring in operating anomalies and intermittents into the FD requirements.

In the definition of diagnostic requirements, it is important to note that the diagnostic capability is made up of the inherent diagnostic capability of the prime system (active arrays are fault tolerant), as well as added diagnostic elements. It is therefore important that diagnostic analysis be integral to the prime weapon system engineering

process, since performance and support parameters can no longer be isolated from design.

The prime configuration represents a performance capability. The mission requirements can be related directly to the configuration by analysis of the behavior of the utilized configuration items over the time demands imposed by mission. A representation of performance over time integrates the "ility" measures and can be easily presented to management for setting requirements. This measure is referred to as P_t (Performance time dependency). P_t can be calculated and plotted using equations for mission reliability in MIL-STD-756B.

Operational constraints also must be addressed. The checklist below presents the operational data to be collected and considered in diagnostic requirements analysis.

- Environmental conditions (temperature, rain, dirt, salt spray, etc.)
- Operating locations (dispersed vs. centralized) (remote/accessible/inaccessible)
- Space limitations (for personnel and/or test equipment)
- Mobile or fixed maintenance facilities
- Independent operation or part of a battle group
- Manpower constraints (number and skill levels).

The constraints under which a weapon system must operate must be identified and evaluated in terms of the impact on testability requirements. System design and supportability factors must take into account these constraints. Operating constraints will often drive the diagnostic strategy to use of embedded versus external test resources.

Prime System Architecture/Configuration

Data must be collected on the architecture and configuration alternatives of the prime system to be developed with respect to partitioning, interconnections and flow as input to the testability requirements analysis. The architectures under consideration will have inherent characteristics which may support or impede diagnostics. The performance capability of alternative prime system architectures must be evaluated against the mission requirements, time phases and equipment utilization/demands.

It is useful for this evaluation to plot curves of capability vs. time demands imposed by the mission. The resulting P_t (Performance over Time) curve can include resource constraints (spares, personnel) and operational constraints (maximum allowable repair time).

The following prime system configuration data should be collected for input to this step:

- o Work Breakdown Structure (MIL-STD-881)
- o List of government furnished equipment/ off-the-shelf equipment/ non-developmental items (for above, item or candidate item)
- o Prime system architecture alternatives
- o Initial failure rate projections and characterization
- o Fault-tolerant or redundant functions
- o Technologies to be used (if known)
- o Level of integration vs. autonomy.

Based upon analysis of architectures under consideration, high-level diagnostic opportunities should be identified. This includes incorporation of a test and maintenance bus, fault-tolerant design coordination, system-level diagnostic resources - such as data acquisition/ collection subsystems or embedded adaptive diagnostic subsystem and use of standard diagnostic connections and interfaces.

Diagnostic inputs must be made within the system engineering process prior to the final selection of the prime system architecture.

Evaluate Technology Opportunities

Advanced diagnostic technology opportunity or implications must be identified based on the following areas:

- o Baseline comparison system major drivers, supportability problem areas, targets of improvement
- o Incorporation of LSI, VLSI, VHSIC, expert system or other advanced design technology in system
- o Need to improve requisite operational capability having no prior design solution.

Examples of advanced diagnostic technology opportunities which may be exploitable on the new system include:

- o Expert system based maintenance aids
- o Test and Maintenance bus concepts
- o "Smart" BIT techniques

- o Adaptive diagnostic subsystems
- o Prognostics concepts
- o Automated technical information authoring
- o Advanced packaging techniques
- o Advanced instrumentation (stimulus and measurement) technologies
- o Automatic capture of CAD data for diagnostics generation.

Upon determination of advanced technology applications, inputs must be made to the design engineering effort regarding design constraints related to the above concepts.

Diagnostic Element Constraints

In order to define specific diagnostic characteristics and requirements of the system or to further "close in" the envelope within which tradeoffs are conducted, diagnostic-related constraints are established. This includes constraints placed on built-in test design attributes and functions, testability constraints and test equipment constraints. This may also include broader diagnostic-related constraints, such as page count of technical information or maintenance technician skill-levels. These constraints are driven by mission requirements, design, operation and support characteristics, or standardization policies imposed.

Sample diagnostic-related constraints are provided below.

Driving System Requirement	**Resulting Diagnostic Constraint/Requirement**
<u>Mission Requirement</u>	
Mobility	Test Equipment Size/Weight
Continuous Operation	BIT Interface Planned Maintenance Duty Cycle
Sustainability	Redundancy
Reconfigurability	Fault-Tolerant Design
<u>Standardization Imposed</u>	
Standard Test Equipment	Standard Diagnostic Connectors Controllability, Observability Interface to UUT
Standard Bus	Interface Design/Protocol

GFE	Bit Design/Capabilities
Design Characteristics	
Power Availability	BIT Power Consumption
System Weight	BIT and Test Connector weight
System Size	Volume of BIT Circuitry and Test Connectors (Real estate available for BIT circuitry) Volume required for increased modularity
Memory Limitation	Memory allocatable to BIT functions
Operating System Char.	Software BIT function constraints
Cost	Cost of additional hardware required for BIT and testability

Establish Diagnostic Objectives

Analysis of weapon system data ascertained must be performed to identify diagnostic objectives based on system requirements. Diagnostic objectives to be considered include:

- o BIT FD/FI requirements to support preliminary maintenance concept
  - Repair Times
  - Reconfigurability
  - Deferred Maintenance
  - Fault Tolerance
- o BIT requirements to support system confidence checks
- o Requirement to deal with intermittent faults or operational anomalies
- o Prime system architecture testability opportunities
- o GFE testability factors/constraints
- o Requirements for vertical testability.

Examples of typical objectives to be established at this point are provided below.

DRIVING SYSTEM FACTOR	SAMPLE DIAGNOSTIC OBJECTIVE
Maximum Acceptable Failure Latency----⇒	Fault Detection Time
Mission/Safety Critical Function------⇒	Performance Monitoring
MTTR, Spares----------⇒	Fault-Isolation Level
Manpower and Skill Levels----------⇒	BIT Fault-Isolation Level
GFE Constraints----------⇒	System-Level BIT Requirements
Fault-Tolerant Design Coordination----⇒	Performance Monitoring
2-Level Maintenance----------⇒	Ambiguity Group Size
Life Cycle Cost Priorities----------⇒	Reliance on Embedded Diagnostics
Minimize RTOK Rate Between----------⇒ Maintenance Levels	Utilize Compatible Test Equipment, Techniques, Tolerances

Initial diagnostic requirements result from analysis of weapon system characteristics, prioritized as needed, on diagnostic elements. It is convenient to partition the diagnostic elements as embedded and external.

Some of the tradeoffs to be made for generating embedded and external diagnostic requirements include:

- o Functions and level of built-in test vs. external diagnostics
- o Functional vs. parametric testing
- o Built-in test fault detection
  - Concurrent performance monitoring
  - Periodic BIT routines
  - Operator initiated BIT routines
- o Level of diagnostic capability at each level of maintenance (e.g., detect 95% of faults; isolate to 3 LRUs; within 30 minutes, utilizing a specific diagnostic resource).
- o Diagnostic elements to be used at each level of maintenance (e.g., test equipment, technical information and maintenance aids, training and skill levels).

Once the level of built-in test is established, a maintenance workload generated by operational and failure rate data can be projected. At this point, detailed tradeoffs can be performed regarding the optimization of testability, including level of diagnostic capability at each level of maintenance, and the effectiveness and efficiency of the mix of diagnostic elements to be used at each level of maintenance. A baseline comparison system is used to project failure data. The requirements that need to be established are outlined below:

Embedded Diagnostic Requirements

- o BIT requirement for monitoring of mission-critical functions and functions affecting personnel safety (derived from maximum allowable failure latency)
- o BIT/SIT requirements to support operational constraints
- o Requirement to deal with/handle intermittents/anomalies
- o BIT/SIT requirements to support system confidence checks
- o Prime system architecture, testability opportunities, and GFE testability factors/constraints

- o BIT requirements to support preliminary maintenance concept, based on:

  - Level-of-repair analysis
  - Manpower available
  - Skill levels available/required
  - Deferred maintenance goals
  - Repair times (driving fault isolation time)
  - Sparing concepts (driving fault isolation levels)
  - Standardization requirements/goals (test equipment, personnel qualifications)

- o Requirement to provide handshake to external diagnostic resources (vertical testability, vertical diagnostics).

External Diagnostic Requirements (Support Equipment, Technical Data, and Personnel)

OPERATIONAL/ORGANIZATIONAL MAINTENANCE LEVEL:

- o Requirement for elements to optimize interface/utilization of embedded diagnostic elements

- o Define FD/FI functions to satisfy O-Level maintenance operations (driven by inputs from operational constraints and preliminary maintenance concept), based on:

  - Minimization of unnecessary removals
  - Mobility requirements/space available
  - Level-of-repair analysis
  - Sustainability (spares replenishment)
  - Manpower available
  - Skill levels available/required
  - Repair times
  - Sparing concepts
  - Standardization requirements/goals

- o O-Level technical information (including maintenance aids)

- o O-Level test equipment

  - Manual test equipment
  - Automatic test equipment and test programs
  - Portable maintenance aids

- o O-Level training requirements to support skills required

- On-the-job training
- Formal school training

o O-Level data acquisition/collection system (and data management)

o Requirements to provide O-Level handshake to I-Level diagnostic elements (vertical testability, vertical diagnostics)

INTERMEDIATE MAINTENANCE LEVEL:

o Define FD/FI functions to satisfy I-Level maintenance operations based on:

- Minimization of unnecessary removals
- Mobility requirements/space available
- Level-of-repair analysis
- Sustainability of spares pipeline
- Manpower available
- Skill levels available/required
- Repair times
- Sparing concepts
- Standardization requirements/goals

o I-Level technical information requirements (including maintenance aids)

o I-Level test equipment requirements

- Manual test equipment
- Automatic test equipment and test program sets

o I-Level training requirements to support skills required

- On-the-job training
- Formal school training

o I-Level data acquisition, collection, management, analysis, processing system requirements

o Requirement to provide I-Level handshake to Depot-Level diagnostic elements (vertical testability)

DEPOT MAINTENANCE LEVEL:

o Define FD/FI functions to satisfy Depot-Level maintenance operations, based on:

- Level-of-repair analysis
- Sustainability of spares pipeline
- Manpower availability
- Skill levels available/required
- Repair times
- Sparing concepts
- Standardization requirements/goals

o D-Level technical information requirements (including maintenance aids)

o D-Level test equipment requirements

- Manual test equipment
- Automatic test equipment and test program sets

o D-Level training requirements to support skills required

- On-the-job training
- Formal school training

o D-Level maintenance data acquisition, collection, analysis, processing

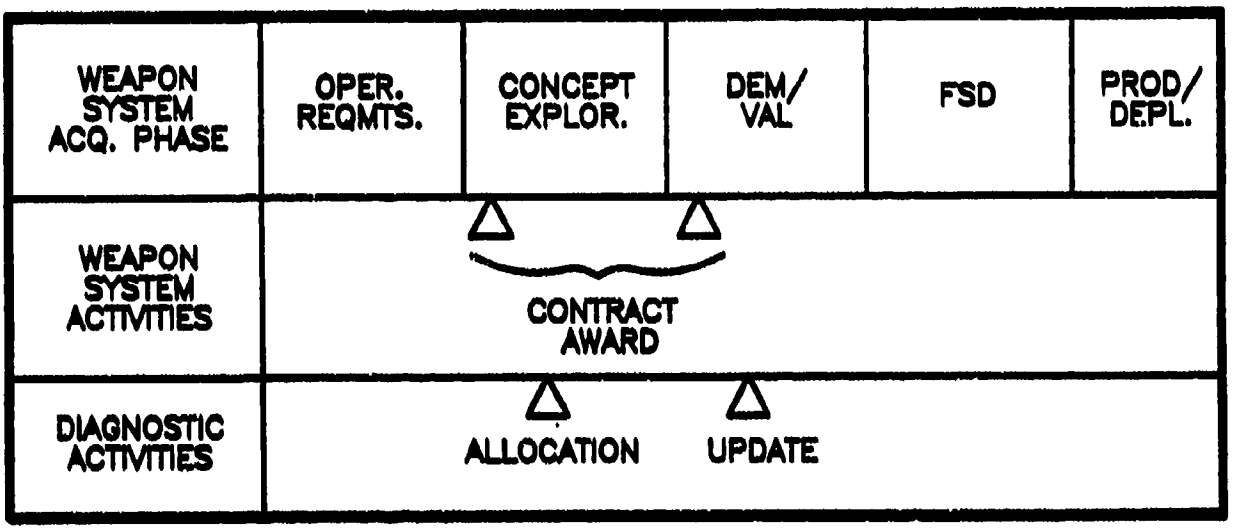
o Requirement to capture and utilize factory test resources and results and/or data for Depot use (vertical testability, vertical diagnostics).

Since the overall diagnostic capability must be defined, quantified, designed, evaluated, etc., it is best defined as a "diagnostic subsystem." This subsystem can be broken down into its component parts and defined in a type of format. This format will facilitate the hierarchical allocation and diagnostic mix optimization process because function and cost parameters can be quantitatively assigned to each element. Alternative diagnostic subsystems may then be easily synthesized and evaluated.

CHECKLIST

- ☑ Has the inherent diagnostic capability of the prime system architecture been included in the analysis?
- ☑ Have the requirements been generated for both embedded and external diagnostics? Are they feasible and implementable?
- ☑ Has the mission data been defined and utilized in the diagnostic requirements generation?

ALLOCATION OF DIAGNOSTIC REQUIREMENTS	REQUIREMENT #2.2



DIAGNOSTIC ACTIVITY

Allocation of diagnostic requirements from the system level to the subsystem and unit level is required in order to assign specification values to each configuration item which forms part of the weapon system. The allocation process, which is normally done by the contractor, shall assure that the weapon system diagnostic requirements and the constraints on the diagnostic subsystem are not violated during the "flow down" process.

PROCEDURE

Initial allocation of diagnostic requirements to lower system levels must be based upon the time demands placed upon the system configuration by the mission requirements.

After the initial set of diagnostic requirements has been defined, a diagnostic mix is postulated from the synthesized diagnostic subsystem alternatives in order to implement the initial set of diagnostic requirements.

Whereas the initial diagnostic requirements are driven by mission time demands, the optimization of the diagnostic element mix is driven by resource constraints. Simply stated, the requirements generation process indicates what is needed and the diagnostic mix generation process indicates the most affordable solutions. A risk analysis performed during the subsequent phases of system development confirm the solutions as feasible. It is, therefore, important to note that the allocation procedure is a partial step in the development of a diagnostic system. During the diagnostic element optimization and design process, it may be cost effective to reallocate the diagnostic requirements in order to achieve better implementations with respect to resource constraints. Many of these

tradeoffs are driven by both technology and the acquisition business decisions that are made for each weapon system program. For example, allocation of a testability strategy to each subsystem may not be feasible due to the existence of many government-furnished equipments within a particular weapon system. In those cases, a centralized system-level test approach may be more desirable. A shift in allocation from subsystem to system level will prove effective in the implementation of that particular weapon system diagnostics.

To achieve this flexibility, the allocation process must be tied to the system-level reliability model. This model will contain the allocated parameters with traceability back to system-level parameters. In this way, as the program proceeds from Concept through Dem/Val into Full- Scale Development, each of the values can be traded off as the diagnostic subsystem is configured and optimized as a result of knowledge gained from trade studies.

GUIDANCE

A preliminary diagnostic allocation should be prepared. The allocation should include all diagnostic elements and consideration of all maintenance levels. The allocation of diagnostic goals/values should be accomplished through the application of structured processes, based on task description and guidance provided within applicable military standards. The tasks and guidance paragraphs that define the allocation process to be employed are:

MIL-STD-499	Task 10.2.3	Allocation
MIL-STD-785	Task 202	Reliability Allocation
MIL-STD-470	Task 202	Maintainability Allocation
MIL-STD-2165	Task 201	Testability Requirements.

MIL-STD-499 addresses the entire allocation process for all performance and design requirements. Time requirements, which are prerequisites for a function, or set of functions, affecting mission success, safety, and availability are derived. It is essential that the diagnostic requirements be derived in conjunction with the entire weapon system allocation process. Reliability and maintainability allocations are derived as part of the overall weapon system allocations. Thus, they have a direct affect on the diagnostic allocations. Failure rates and repair rates are the drivers in establishing diagnostic allocations. However, other considerations dealing with safety monitoring, readiness monitoring, and logistic functions all play a part in this process. The allocation of diagnostic requirements is usually performed as part of the overall LSA process. Closely tied to the LSA process is the establishment of testability requirements, including performance monitoring, BIT, test equipment, diagnostic test points, etc.

It is important that this allocation process includes:

- o FD/FI coverage for all (100%) faults known or expected to occur at each maintenance level, and
- o Quantification of all diagnostic elements.

Figure 3 is a Notional Diagnostic Allocation Specification, which exemplifies these concepts. This allocation process is also closely tied to the optimization process (Requirements #2.3). It is important that this allocation process includes quantification of all diagnostic elements. For instance, the time to access technical information can determine whether paper or electronic delivery of technical information is required. Formal training time may influence the need for on-the-job training aids.

This system-level allocation forms the basis for the System Specification discussed under Requirement #1.2. It also is followed by allocation down to subsystem and item levels.

Allocate Requirements to Item Development Specification

System-level diagnostic requirements are allocated down to subsystem and item levels for the purpose of the development of those items. Diagnostic requirements for Configuration Items (CI) support two distinct requirements: system test (primarily BIT) and shop test (ATE and GPETE).

Quantitative testability requirements for each Configuration Item are allocated from system diagnostic requirements based upon FMEA data, relative failure rates of CIs, mission criticality of the CIs, what is achievable for each CI or other specified criteria. The failure detection level of the CI is weighted by the items' failure rate to ensure that system-level fault detection capability is achieved. Table 4 is an example of an allocation of a system- level BIT fault detection requirement which is allocated to five configuration items. The table shows three alternative FD allocations which meet the system- level BIT FD requirement of 95%.

LEVEL OF MAINTENANCE[1]	DIAGNOSTIC CAPABILITY[1]	FAULT DETECTION COVERAGE[2]	FAULT ISOLATION COVERAGE[2]	MEAN TIME TO DIAGNOSE	TECH INFO ACCESS TIME	OTHER REQUIREMENTS[1]
ORGANIZATIONAL	STATUS MONITOR/ BIT	____ %		____ %		____ % OF FAULT COVERAGE BY STATUS MONITOR FOR MISSION-CRITICAL FUNCTION
	BIT MANUAL DIAGNOSIS	____ %	____ %	____ %	____ %	
	VISUAL	____ %	____ %	____ %		BIT MEMORY ALLOCATION NOT TO EXCEED X WORDS
	TOTAL:	100 %	100 %			
INTERMEDIATE	EXTERNAL ATE/ EXPERT SYSTEM	____ %	____ %	____ %	____ %	
	MANUAL TEST	____ %	____ %	____ %	____ %	ATE LIMITED TO X LB. Y CUBIC FT.
	TOTAL:	100 %	100 %			
DEPOT	EXTERNAL ATE	____ %	____ %	____ %	____ %	
	MANUAL TEST	____ %	____ %	____ %	____ %	
	TOTAL:	100 %	100 %			

1 – LISTED BY WAY OF EXAMPLE

2 – UNAMBIGUOUS PERCENTAGE OF FAULT COVERAGE FOR EACH CAPABILITY SHOWN. TOTAL AT EACH LEVEL OF MAINTENANCE SHOULD ADD TO 100% OF THE IDENTIFIED REPLACEABLE ITEMS FOR THAT LEVEL.

FIGURE 3. NOTIONAL DIAGNOSTIC ALLOCATION SPECIFICATION

TABLE 4 SAMPLE ALLOCATION OF 95% BIT FD REQUIREMENT

CONFIGURATION ITEM	λ X 10^{-3}	FD ALLOCATION #1	FD ALLOCATION #2	FD ALLOCATION #3
A	100	.95	.98	.95
B	10	.95	.80	1.00
C	50	.95	.70	.98
D	200	.95	.99	.90
E	100	.95	.98	.99
SYSTEM	460	.95	.95	.95

The BIT performance capability and testability characteristics of GFE portions of the system should be considered in the allocation. For example, assume a GFE item has only 70% BIT fault-detection capability. In order to accomplish the 95% system-level capability required in the above example, the allocation distribution must take into account the capability of each of the items which make up, or contribute to, the system level. The capability of the GFE then serves as a constraint in the allocation. In the above example, given that item C is GFE with 70% BIT fault-detection capability, the FD allocation scheme #2 is a real world alternative and the others, #1 and #3 are not.

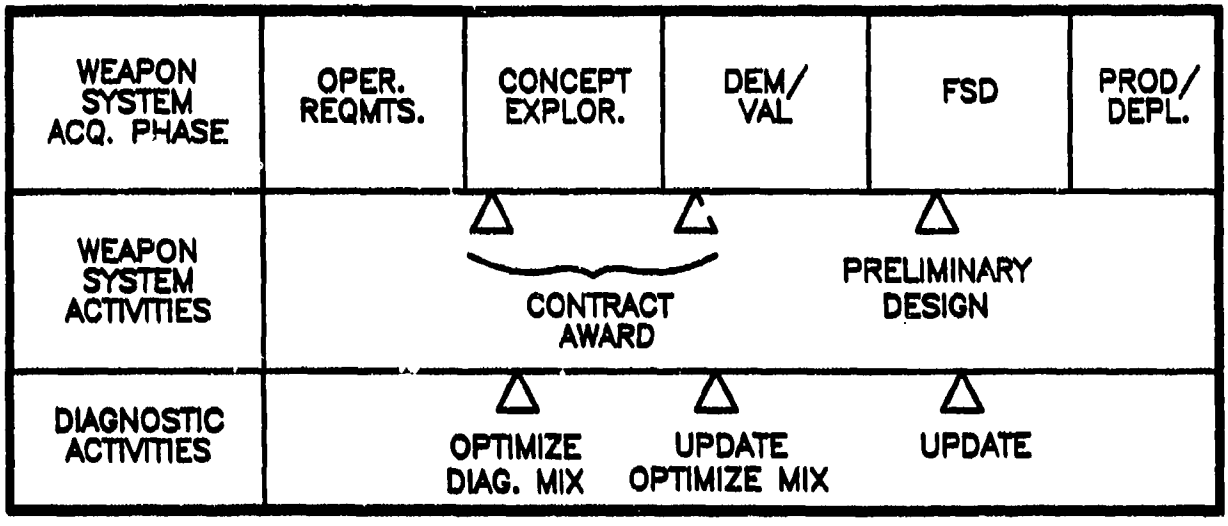
Shop test requirements are determined by how the CI is further partitioned, if at all, into Units Under Test (UUT). Diagnostic requirements for each UUT should be included in the appropriate CI Development Specification. These parameters are not allocated from the system-level requirements, but rather are driven by the diagnostic concept of off-line test requirements of the Configuration Items.

In many digital systems, built-in test is implemented in whole or in part through software. Here, diagnostic requirements will appear in a Computer Program Configuration Item (CPCI) development specification. The CPCI may be dedicated to the built-in test function (i.e., a maintenance program) or may be a mission program which contains test functions.

CHECKLIST

- ☑ Are all diagnostic elements quantitatively defined?
- ☑ Were constraints allocated to all diagnostic elements?
- ☑ Were constraints assigned to all maintenance echelons?

OPTIMIZATION OF THE DIAGNOSTIC ELEMENT MIX	REQUIREMENT #2.3



DIAGNOSTIC ACTIVITY

Given the allocation of diagnostic requirements to the subsystem and unit level, the "diagnostic subsystem" must be defined as part of the overall weapon system specification. The resulting diagnostic subsystem includes both embedded and external support. External support must be defined at all levels of maintenance and includes technical information, support equipment, and personnel numbers and skill levels.

PROCEDURE

The starting point for developing the diagnostic subsystem is the generation of a diagnostic subsystem profile from the weapon system characteristics and priorities. Each of the diagnostic elements will have a differing impact on the weapon system characteristics. For example, a high priority constraint on logistic support would favor a high degree of embedded diagnostics. On the other hand, constraints on personnel may favor technical information systems with a high degree of artificial intelligence. Operational constraints, which are common across the military services, are:

- Environmental conditions (temperature, rain, dirt, salt spray, etc.)
- Operating locations (dispersed vs. centralized) (remote/accessible/inaccessible)
- Space limitations (for personnel and/or test equipment)

- o Mobile or fixed maintenance facilities
- o Independent operation or part of a battle group
- o Manpower constraints (number and skill levels).

Analysis of the weapon system characteristics in terms of their impact on the support elements will generate various support element diagnostic profiles.

The diagnostic profiles will portray various mixes of diagnostic elements for different weapon system characteristics and constraints. Each of the diagnostic element profiles infers a diagnostic subsystem which can be built and delivered with the weapon system. The optimization issue is the selection of a diagnostic subsystem which can be implemented at low risk and which meets the requirements allocated to system, subsystem, and unit level.

The key to optimization, therefore, is the development or synthesis of various alternative diagnostic subsystems based upon the weighted diagnostic element profiles. This is an engineering task and represents an important aspect in the overall development of a diagnostic capability for the weapon system. By generation of a diagnostic subsystem, early in Concept Exploration, the overall design integration of support and prime design elements will be achieved. During the Dem/Val and Full-Scale Development Phases, the diagnostic subsystem is refined based upon trade studies.

The key is to identify the sensitivity of the various diagnostic element function contributions to the overall life cycle costs, and to ensure that all diagnostic functional requirements are considered and included as part of the total diagnostic subsystem synthesis.

Each diagnostic subsystem alternative synthesized is evaluated with respect to:

- o Impact on Mission Performance Over Time
- o Impact on Resource Requirements
  - Acquisition Cost
  - Life Cycle Cost
  - Manpower Requirements
- o Responsiveness to operational constraints.

The evaluation is performed by assigning values related to the evaluation factors listed above to the diagnostic subsystem or to the elements of the diagnostic subsystem.

OPTIMIZATION OF THE DIAGNOSTIC ELEMENT MIX — REQUIREMENT #2.3

To evaluate the responsiveness of the diagnostic subsystem to mission performance requires defining the specific configuration utilized for a specific mission. The reliability of that configuration to deliver the specified performance is then evaluated for the time intervals demanded by the mission scenario. This is called "intervals reliability." Interval reliability is the basis for determining diagnostic requirements at these intervals and is the basis for reconfigurability, deferred maintenance, and performance monitoring.

To evaluate the responsiveness of the diagnostic subsystem to operational constraints, the operational constraints must be assigned qualitative or quantitative values. The impact of the diagnostic subsystem characteristics on those values (time demands) must then be determined. This analysis includes availability parameters as well as mission reliability calculations based upon the stated time demands and subsystem utilization. The system reliability model is a very effective and available tool for this analysis.

To evaluate the impact of the diagnostic subsystem on resources, cost factors must be assigned to each element of the diagnostic subsystem. Non-recurring (development) and recurring (production and support) costs must be considered. The manpower requirements associated with the alternative diagnostic subsystems must be evaluated. Specific program existing LCC models should be used in this analysis. Data items should be standardized wherever possible.

The cost deltas associated with alternatives must be evaluated with respect to the off-line maintenance workload costs and efficiencies generated by the alternative embedded diagnostic subsystems. A key diagnostic element driving workload is ambiguity group size and RTOK rates.

Based upon the evaluations performed, the optimum diagnostic subsystem alternative is selected and the weapon system diagnostic concept is established and documented. The diagnostic concept includes prime system architecture considerations, BIT requirements at the system and subsystem levels, test equipment, technical information and personnel and training requirements for each level of maintenance. The diagnostic function of each element must be clearly and quantitatively defined as a diagnostic requirement.

Utilizing the above procedure, the result of the optimization process is the development of a diagnostic subsystem early in Concept Exploration. This parallels the development effort for radar subsystems, fire control subsystems, etc. The diagnostic subsystem becomes a weapon system attribute early in Concept Exploration and continues to evolve during subsequent program phases.

GUIDANCE

RADC issued the report, "Tools for Integrated Diagnostics" (RADC-TR-88-195), which established criteria for developing an optimized diagnostic mix of built-in, external, and manual testing resources for an electronic system. As of the publication date of this document, there is no formal algorithm for defining an optimized diagnostic mix. A methodology for performing diagnostic optimization will be a product of the RADC Automated Testability Decision Tools Program, which will be completed and published in mid-1990.

A generic hierarchical view of a diagnostic subsystem which includes engineering and program management disciplines as well as embedded and external support elements is included below to serve as guidance for the contractors. This indentured diagnostic subsystem breakdown will allow costing by the contractor for various alternatives proposed to satisfy the diagnostic requirements which have been allocated at all system levels. As experience data is accumulated on diagnostic subsystem effectiveness and cost, it will be possible to predict many of these values early in Concept Exploration using the diagnostic profile.

DIAGNOSTIC SUBSYSTEM HIERARCHY

I. PROGRAM MANAGEMENT/ENGINEERING

A. Requirements Analysis
B. Diagnostic Design & Analysis/Assessment
C. System Integration & Test
D. Maturation Program

II. EMBEDDED DIAGNOSTIC ELEMENTS

A. System-Level Diagnostic Elements
    1. System-Level Diagnostic Hardware
        a. Test and Maintenance Bus
        b. Sensors/Monitors
        c. Diagnostic Panel/Display
        d. Embedded ATE

    2. System-Level Diagnostic Software
        a. Status Monitoring
        b. Self Test/Expert Systems
        c. Prognostics
        d. Reconfigurability

    3. Diagnostics Information System

B. Subsystem Diagnostics

1. Subsystem "A" BIT
   a. BIT Hardware
      1. On Chip
      2. On Printed Circuit Board
   b. BIT Software & Firmware
   c. Interface to T&M Bus

2. Subsystem "B" BIT (Radar), etc.

III. EXTERNAL DIAGNOSTIC ELEMENTS

A. O-Level Diagnostics

1. Technical Information
   a. Maintenance Aids
   b. Paper-Based Manuals
   c. Diagnostic Data Base

2. Test Equipment
   a. Manual Test Equipment
   b. Automatic Test Equipment
      1. ATE Hardware
      2. Diagnostic Software
         a. Expert Systems
         b. Test Program Sets (TPS)
      3. ATE/ILS

3. Trained Personnel
   a. Manpower
   b. Skills
      1. Formal Training
      2. On-The-Job Training

4. Diagnostic Data Collection/Analysis System (MIS)

B. I-Level Diagnostics

1. Technical Information
   a. Maintenance Aids
   b. Paper-Based Manuals
   c. Diagnostic Data Base

2. Test Equipment
    a. Manual Test Equipment

    b. Automatic Test Equipment
        1. ATE Hardware
        2. TPS
        3. ATE/ILS

3. Trained Personnel
    a. Manpower
    b. Skills
        1. Formal Training
        2. On-The-Job Training

4. Diagnostic Data Collection/Analysis System

C. D-Level Diagnostics

1. Technical Information
    a. Maintenance Aids
    b. Paper-Based Manuals
    c. Diagnostic Data Base

2. Test Equipment
    a. Manual Test Equipment
    b. Automatic Test Equipment
        1. ATE HW
        2. TPS
        3. ATE/ILS

3. Trained Personnel
    a. Manpower
    b. Skills
        1. Formal Training
        2. On-The-Job Training

4. Diagnostic Data Collection/Analysis System

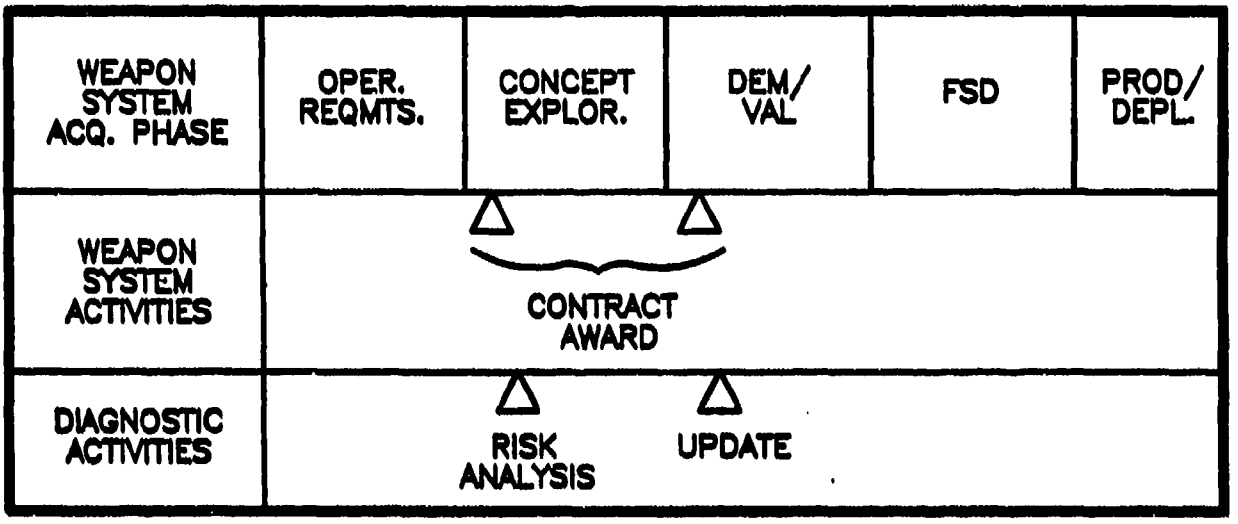
OPTIMIZATION OF THE
DIAGNOSTIC ELEMENT MIX

REQUIREMENT #2.3

CHECKLIST

- ☑ Does the "diagnostic subsystem" include all maintenance levels?
- ☑ Was the optimization of the "diagnostic subsystem" performed by doing a unit criteria analysis of various proposed "diagnostic subsystems"?
- ☑ Do you have a reasonable degree of certainty that the chosen "diagnostic subsystem" represents an optimal solution?

RISK ASSESSMENT	REQUIREMENT #2.4



DIAGNOSTIC ACTIVITY

The initial diagnostic subsystem, generated to implement the allocated diagnostic requirements, must go through a thorough risk analysis during the Dem/Val Phase. During subsequent Full-Scale Development, the diagnostic subsystem is optimized utilizing results of trade studies. The initial diagnostic element mix postulated during Concept Exploration is analyzed by the contractor for risk during that phase by technology assessment. However, risk assessment can take into account threat, technology, resources, schedule, and cost.

PROCEDURE

The procedure for performing risk analysis on the diagnostic subsystem will follow the same type of assessments conducted for risk analysis for other weapon system elements. For example, the risk assessment for a radar will include assessment of its new development components, assessment of schedule, cost risks, and assessment of the overall technologies involved in the development and integration of a total system to meet the performance requirements. Since the diagnostic subsystem will be treated as a major element of a weapon system, the same procedures should apply for it. Heretofore, diagnostic subsystems were not treated as an entity and risk analysis was limited only to the physical diagnostic hardware, such as automatic test equipment and built-in test.

Risk assessment shall include the isolation within the diagnostic subsystem of all development and non-development items. For development items, waiting factors in terms of criticality of that item shall be assigned and the items shall be categorized with

respect to risk. For items considered high development risk, workarounds shall be developed and trigger points for decisions on their implementation shall be listed. The risk analysis documentation shall be utilized to impact the Statement of Work for the Dem/Val Phase. During Dem/Val high-risk items shall be prototyped and demonstrated to the satisfaction of the Government Program Manager.

GUIDANCE

The Defense Systems Management College has generated guidance on risk management, which includes risk assessment, risk analysis, risk handling techniques, and risk control. This guidance covers risk management for the entire weapon system, but is equally relevant to the weapon system's diagnostic capability. Both risk assessment and risk analysis need to be addressed early in the development of the weapon system. Risk assessment is the process of examining a situation and identifying areas of potential risk. Risk analysis is examining the change of consequences with the modification of risk input variables. At the time this Contractor Program Managers Guide was issued, the Defense Systems Management College is publishing a risk management guide, which further defines the methodology for doing risk assessment and analysis. Appendix C describes several tools for assessing risk in relation to time, cost and producibility.

MIL-STD-1388-1 (Logistics Support Analysis) contains in Tasks 203, 205, and 303 guidance on comparative analysis, supportability related design factors, and evaluation of alternatives for trade-off analysis, all of which are directly related to the weapon system's diagnostic capability.

Lessons learned have pointed to some overriding areas of risk which must be considered during the initial risk analysis assessment. These high-risk areas are listed in the following paragraphs:

The logistic support analysis process will usually generate requirements for each of the logistic elements comprising the overall logistic system. These requirements are based upon inputs regarding the level of embedded support to be designed into the weapon system. The Logistic Element Manager, given these inputs, proceeds to develop sparing requirements, support equipment requirements, training requirements, etc. A large program risk area occurs when the promised embedded support area does not materialize. It is imperative, therefore, to close the loop between assessment during Dem/Val of the diagnostic element capability and that impact on all logistic elements.

A second major risk area occurs when a prime weapon system, which has been developed for a specific maintenance strategy and concept, is utilized in a completely different mission environment. For example, a major weapon system deployed in a three-level maintenance environment may be required to operate for extended periods of time in a dispersed operating location with less than full support. The sustainability and mobility requirements imposed upon that weapon system may not have been included with sufficient priority in the initial analysis to develop capability for that operational

environment. It is, therefore, imperative that as part of the risk analysis, the assessment of weapon system characteristics and the application of weapon system priorities be reviewed prior to commitment of system development resources.

A third high risk area worthy of special consideration is the analysis of the very large scale integrated circuits and very high speed integrated circuits (VLSI/VHSIC). Despite the intensive use of on-chip testing for these devices, it is imperative that a standard systems approach be generated by the contractor. Testability techniques including signature analysis and boundary scan designs must be evaluated at the system and subsystem level prior to commitment of development resources. Standardization by the contractor of the embedded support architecture will eliminate many high-risk problems caused by multiple vendors supplying different types of on-chip testing.

A fourth high risk area occurs when weapon system managers fail to comprehend and implement the existing fielded maintenance standards that are used to support the deployed system. For example, the military has for many years been formalizing the use of IEEE-STD 716 C/ATLAS language for Depot maintenance. The CASS, IFTE and MATE programs have institutionalized this approach. Despite this level of standardization, many programs completely ignore this fact during the Dem/Val and FSD Phases of a program. Since the targeted Depot ATE has been standardized, it is possible to develop test programs starting with Factory-level testing through integration and test of the products that are compatible and easily translatable to the fielded environment. This concept, called vertical commonality, will mature the test programs so that during deployment the logistic system will have a major capability and remove many of the risks associated with transition from interim contractor support to full government support. Utilizing expert system knowledge during these same phases will allow the test program to contain levels of artificial intelligence to extract and utilize experience data on prior failures during the Deployment Phase.

The fifth high risk area is the incorporation of government furnished equipment (GFE) in weapon systems. Care must be taken to ensure that the diagnostic requirements and capability are known and verified. The Government Program Manager must be informed if the required weapon system diagnostic capability is compromised by deficiencies in the GFE.

The sixth and final large risk area is the integration and test of the weapon system prior to delivery. Since weapon systems have become extremely software dependent and since many weapon systems are multi-mission in nature utilizing shared resources, it is imperative that the integration and test function in a program be utilized to remove as much risk as possible from the weapon system. Integration of the diagnostic elements into the weapon system will provide a major "handle" for the contractor in terms of enhancing the integration and test functions. If no attention is paid early in the game to this high potential risk, the integration and test functions will be extremely time consuming, may not come together on schedule, and may cause program hardships. If properly achieved, integration and test can be streamlined to recover much of the upfront monies

spent on enhanced testability features. It is therefore imperative that this area be given serious attention by risk assessment studies early in Concept Exploration and proceed through Dem/Val and Full-Scale Development.

RISK ASSESSMENT REQUIREMENT #2.4

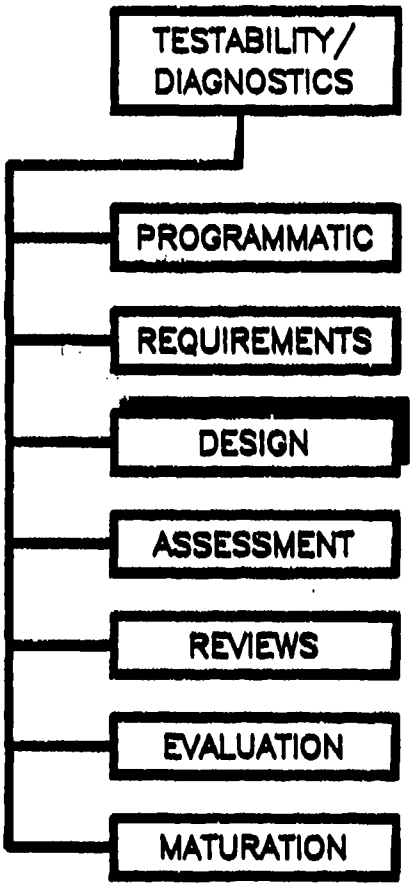
CHECKLIST

- ☑ Was risk analysis performed for the entire "diagnostic subsystem" ?
- ☑ Were adjustments planned for in those cases where one of the diagnostic elements fails to meet expectations?
- ☑ Were available standards taken into account?
- ☑ Have the integration and test risks been defined?

DESIGNING THE DIAGNOSTIC CAPABILITY

OVERVIEW

The design of the diagnostic capability is fractionated among a number of system engineering and supportability functions. Reliability, maintainability, integrated logistic support, testability, human engineering, and safety considerations all play significant roles in determining the requirements of the diagnostic capability and the design of this capability. The design process is further fractionated by the relegation of this capability to the various levels of maintenance. The diagnostic design process is controlled by a large number of military standards, which deal with the design process and criteria. All of these "pieces" of the design process must not only work together, but the diagnostic data produced must be available at specific times. A break in one of the links can compromise the design. A cohesive, integrated design process is required. It is the Program Manager's job to assure that this integration is realized and the designer's job to produce this effective diagnostic capability in an efficient manner.



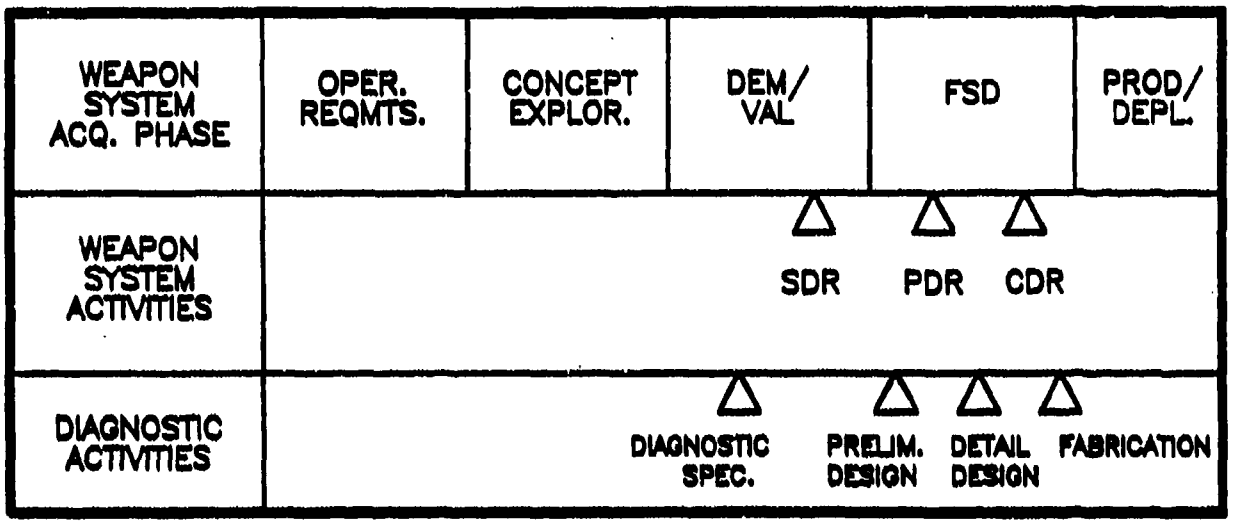
IMPORTANT CONSIDERATIONS TO BE ADDRESSED

Reqmt.

3.1 Assure cohesiveness and efficiency in the design of the diagnostic capability.

3.2 Establish diagnostic design criteria which can be effectively utilized.

PROVIDING A COHESIVE DIAGNOSTIC DESIGN PROCESS	REQUIREMENT #3.1



DIAGNOSTIC ACTIVITY

The designer is responsible for the efficient development of an effective diagnostic capability.

PROCEDURE

The cohesiveness of the diagnostic design process is dependent upon the cohesiveness of the design information flow. Many factors effect the effectiveness and efficiency of this information flow. The first is timing - What is done in what sequence? The second factor relates to the various disciplines involved in the design of the diagnostic capability. These disciplines are controlled by a sizeable number of military standards, which relate to reliability, maintainability, testability, safety, human engineering, software, and training. These standards tend to fractionalize the design of the diagnostic capability, inasmuch as each plays a significant part of the process. The third factor deals with the automation of the design process. Computer-aided tools can promote the cohesiveness and the efficiency of the process. Thus, the designer must understand the capabilities of these tools and be able to apply them effectively and efficiently.

GUIDANCE

The guidance provided in this section is designed to permit maximum visibility into the diagnostic design process. The designer must understand the design process flow, timing, and data requirements which must be satisfied. In addition, it is important to recognize that current data item procurement practices may not always be supportive of the design activity in-process data needs. At times, the CDRL and associated DID do not adequately reflect these in-process needs. The high data item generation/revision costs generally experienced are strong motivators for delaying data item preparation to a point where the design has stabilized. Such motivation is in direct conflict with the utilization of the data to make design decisions. A complete, detailed data submittal indicating that the design is flawed is of little use after the design has been completed. The guidance that follows has been designed to provide the necessary insight into the design process, which will assist the designer in doing a better job.

Design Environment

The diagnostic design environment is an essential component of the overall diagnostic design activity, which has been established by the contractor in response to the RFP requirements. This environment encompasses both the implementation methodology and the specialty coordination associated with the diagnostic design process. Evidence of these should be apparent in the interim products of the design effort, which are made available to the government program management function (at both informal in-process reviews and formal system-level design reviews).

Diagnostic design is characterized by its iterative nature and a high degree of interdependence with the supportability engineering specialties (i. e., reliability, maintainability, integrated logistic support, testability, human engineering, and safety). The allocation of diagnostic resources must be based on inputs from these disciplines. Therefore, the timing and quality of data interchanges must be in accordance with the program plans. A breakdown in data availability and exchange can be responsible for program delays and shortfalls in the fielded diagnostic capability.

The data flow required to develop the composite diagnostic capability must be responsive to the diagnostic mix established for the specific system under consideration. Embedded diagnostic features, such as built-in test (BIT), built-in test equipment (BITE), system integrated test (SIT), performance monitoring, status monitoring, embedded training, embedded maintenance aiding, adaptive AI-based diagnostic systems, etc., are an integral part of the prime equipment design. Therefore, the diagnostic data flow associated with these features must be incremental and continue until the detail prime system Configuration Item designs are complete. For the external diagnostic elements, such as automatic test equipment and the associated test program sets, manual test equipment, portable maintenance aids, technical information (hard copy or electronic delivery), training, etc., the diagnostic data flows into the LSA process up to the point where the firm requirements for these diagnostic elements can be established. Once firm

requirements exist, the diagnostic design environment must facilitate a smooth transfer of data, which is sufficient in terms of detail and format to permit fabrication of the required external diagnostic capability.

Table 5 is a listing of the major military standards which influence the design of the diagnostic capability. Some of these military standards are programmatic in nature, in that they establish a specific program and describe the tasks which can be undertaken. The remainder of the standards are process or product-oriented. As can be seen, these standards influence various aspects in the design of the diagnostic capability, starting from establishing diagnostic requirements, through the design and assessment of the diagnostic capability. There is a sequence of tasks and procedures cited in these standards which can be applied to the diagnostic capability. The interfaces and relationships between these various activities are complex and cannot be easily diagrammed to promote understanding. Establishing diagnostic requirements is described in Requirement #2, and the assessment is described under Requirement #4. Thus the following guidance will address the design of the embedded and external diagnostic capability.

Design Integration

Figure 4 is a simplified diagram of the information flow in the design of the diagnostic capability. The design process begins with a maintenance concept and design data, such as specifications, block diagrams, and schematics. Establishing the system's architecture is the next step. System's architecture has a major impact on the design of the fielded diagnostic capability. The concept of fault tolerance supports the maintenance concept by promoting graceful degradation of the system's performance, thereby allowing the maintenance to be performed at the user's convenience rather than dictated by when faults occur. Design for testability concepts play an important part at this time. Partitioning especially is closely tied to fault tolerance, because the performance monitoring capability must be able to detect failed items in order that the capability of the system is known, that necessary switching to alternate means is facilitated, and that maintenance actions can be identified.

The Failure Modes and Effects Criticality Analysis (FMECA) utilizes the system's architecture and design data to determine the modes, causes and effects of item failures. This data drives the maintenance and safety requirements which in turn help to establish the diagnostic logic, test point selection, and test requirements. From this information, the diagnostic capability is designed and fabricated, including the testing, (built-in and external), technical information, training, and personnel capability. Obviously this entire process is iterative in nature - a factor not indicated in Figure 4.

TABLE 5. MILITARY STANDARDS APPLICABLE TO THE DESIGN OF THE DIAGNOSTIC CAPABILITY

		REQUIREMENT				DESIGN							ASSESS	
		ESTABLISH	ALLOCATE	OPTIMIZE	RISK ASSESS.	FAULT TOLERANT	INHERENT TSTAB	BIT/SIT	ATE/TPS	MANUAL TEST EQ	TECH. INFO.	PERS. & TRNG.	ANALYSIS	DEMONSTRATION
PROGRAMMATIC	MIL-STD-1388-1 Logistic Support Analysis	X	X	X	X								X	X
	MIL-STD-785 Reliability	X	X	X	X	X		X						
	MIL-STD-470 Maintainability	X	X	X	X			X	X	X	X	X	X	X
	MIL-STD-2165 Testability	X	X	X	X	X	X	X	X	X			X	X
	MIL-STD-882 Safety	X				X		X						
	MIL-STD-2167 Software Development	X	X	X	X	X		X	X				X	X
	MIL-H-46855 Human Engineering	X	X	X							X	X		
	MIL-STD-1591 Analysis		X	X										
PRODUCT/PROCESS	MIL-STD-415 Test Provisions						X	X	X					
	MIL-STD-1519 Preparation of -1345 Test Rqt. Doc.								X	X				
	MIL-STD-1629 Procedures for FMECA					X		X						
	MIL-STD-2077 Requirements for TPS								X					
	MIL-STD-471 Maintainability Demonstration													X
	MIL-STD-756 Reliability Modeling & Pred.		X											
	MIL-STD-1379 Contract Training Prog.											X		X

PROVIDING A COHESIVE DIAGNOSTIC DESIGN PROCESS REQUIREMENT #3.1

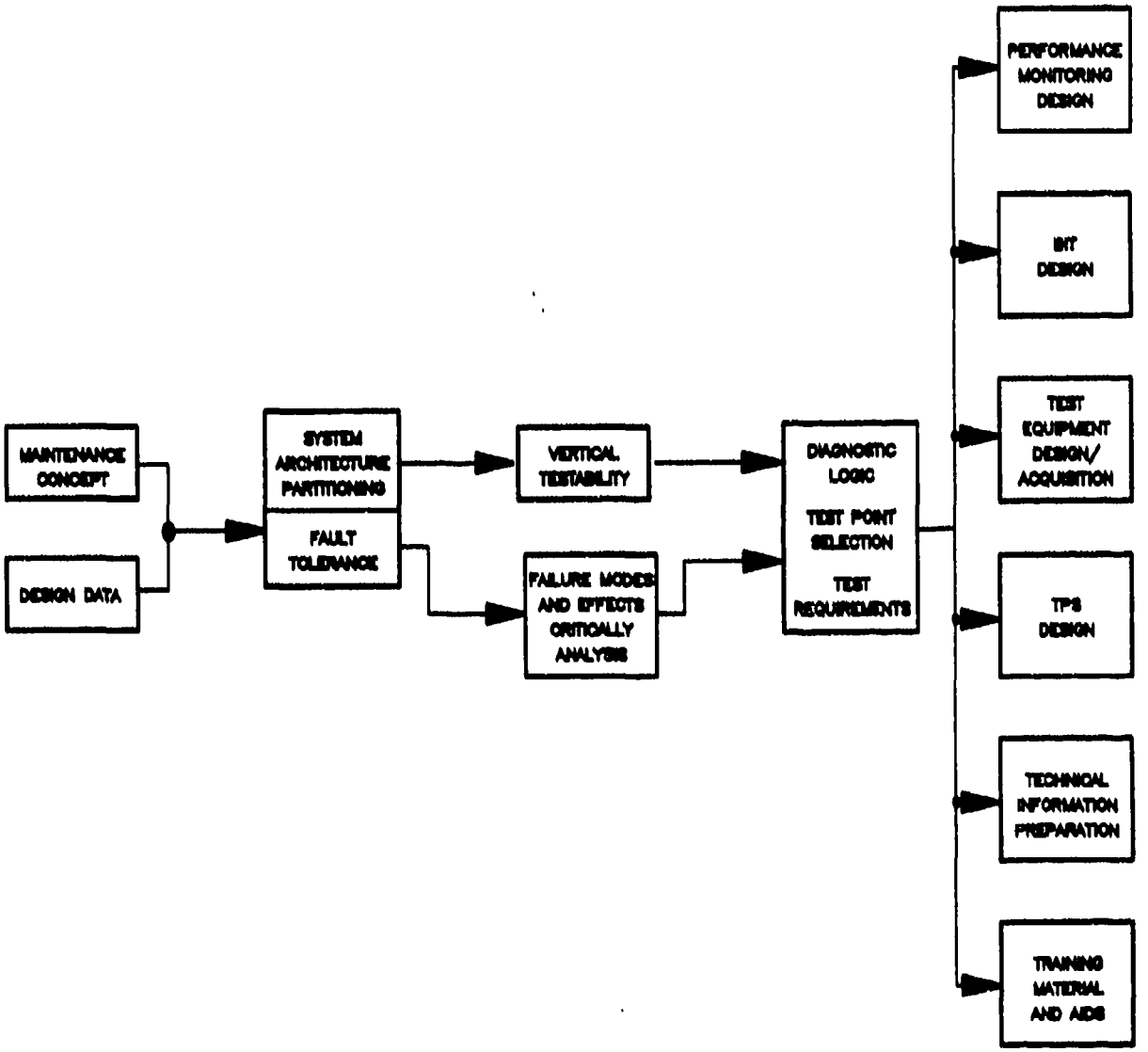


FIGURE 4. DESIGN INTEGRATION OF DIAGNOSTIC CAPABILITY

PROVIDING A COHESIVE DIAGNOSTIC DESIGN PROCESS REQUIREMENT #3.1

The concept of vertical testability was introduced years ago. In essence, this concept addressed the compatibility of testing among all levels of maintenance, including factory testing. The core of this concept is twofold. The first is the establishment of a Cone of Tolerance among these levels, and the second deals with the compatibility of environments under which these tests are performed.

The Cone of Tolerance concept is depicted in Figure 5, in which the testing tolerances are widened as the unit is tested closer to its operational environment.

The compatibility of testing environments can be implemented best through the use of common test equipment at Intermediate, Depot, and Factory Levels. This commonality of test equipment and any associated test programs is the method for implementing this compatibility.

The concept of vertical testability is key to the integration of the design of the diagnostic capability. Therefore additional guidance on vertical test methods is contained in Appendix D. This appendix also includes guidance on documenting the results of vertical testability analysis to assure this information will be integral to the diagnostic design process.

Extension of this vertical testability concept is recommended for the entire fielded diagnostic capability. Figure 6 depicts this concept, in which vertical testing is shown on the left and is joined by technical information and personnel and training compatibility requirements. Not only is this compatibility required vertically, but also horizontally. All elements that make up the diagnostic capability must be compatible at each maintenance level.

This concept of vertical and horizontal compatibility is key to the integration of diagnostic capability. The entire process is driven by the diagnostic logic which effects the requirements for all of the diagnostic elements. This diagnostic logic can be established by a variety of means including the use of maintenance dependency charts, fault trees, etc. To implement this concept, a series of matrices similar in format to Figure 6 can be prepared at various system hierarchy levels (e.g., system, subsystem, LRU, SRU). These matrices should be tailored to the unique requirements of a specific weapon system and may be used in conjunction with other required data deliverables (e.g., test requirements document).

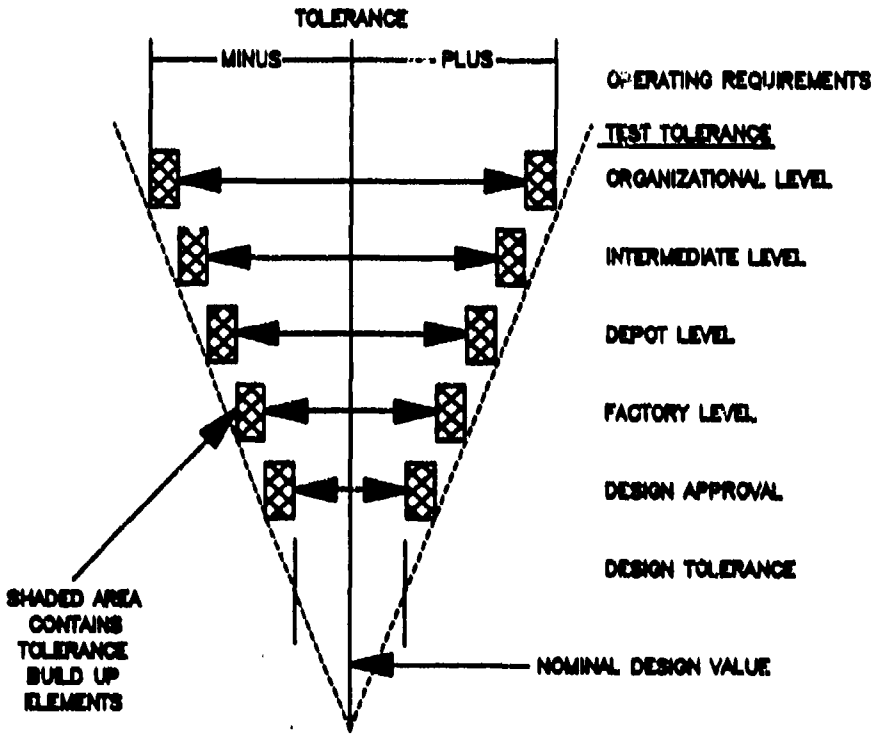


FIGURE 5. CONE OF TOLERANCE

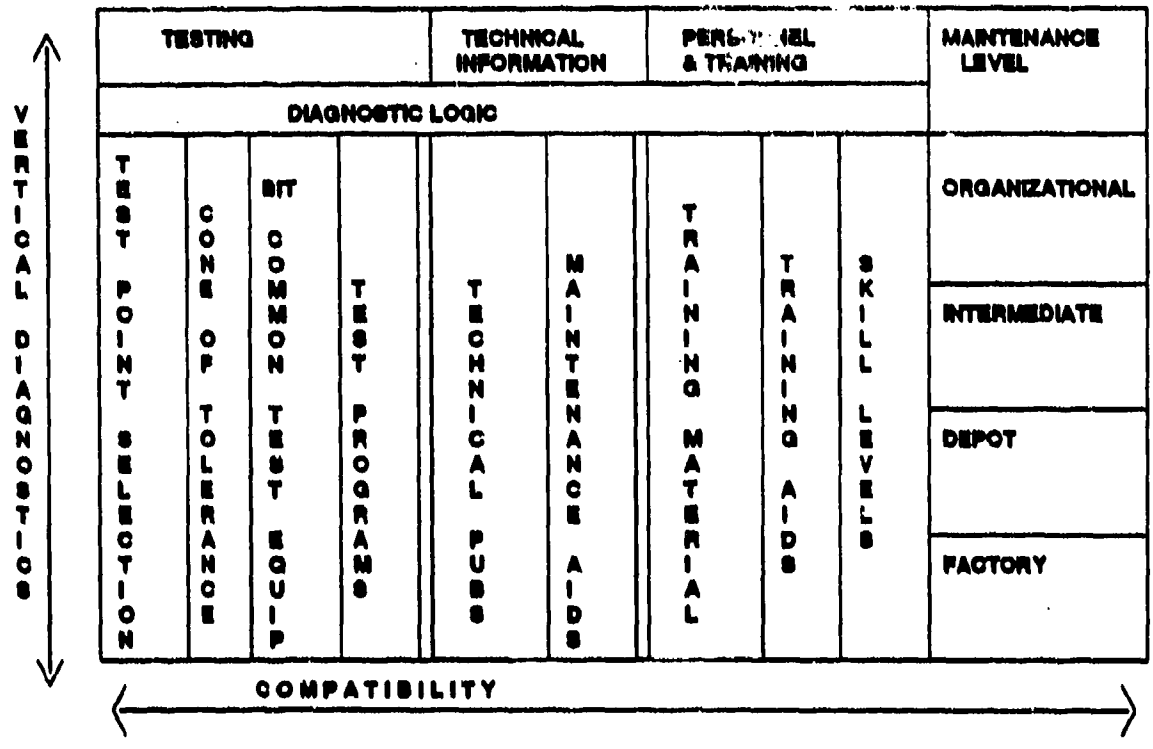


FIGURE 6. Design Integration of Diagnostic Capability

AUTOMATION OF THE DIAGNOSTIC DESIGN PROCESS

Automation of the diagnostic design process is encouraged to provide a more efficient and effective design process. The diagnostic design process should be an integral part of prime system computer-aided engineering and design.

The added efficiency and effectiveness in the use of automation is reflected in a number of ways. The effect of changes in either the diagnostic design or the prime system design can be readily ascertained as the design progresses. This iterative process then can give the designer information on whether or not the diagnostic specification requirements will be met. In addition, automation permits the concurrent development and evaluation of the entire diagnostic capability along with the remainder of the prime system.

Diagnostic Design Tools

Diagnostic design tools enhance the effectiveness and efficiency of the process. A description of available tools and processes is available in Appendix C. Appendix C identifies automated tools which can assist the designer in performing three major facets of the design process: system architecture design, implementation of design rules and practices, and diagnostic authoring.

The first element pertains to design automation tools that assist the designer in synthesizing a prime system functional capability, as well as providing an "environment" for developing a diagnostic capability concurrently with the prime weapon system development process. The architectural tools not only provide a capability to synthesize a functional capability, but also assist the designer in understanding systems methods of doing design work (i.e., operation).These tools generate documentation data bases, which are either explicitly or implicitly usable in the testing, technical information and/or personnel training disciplines.

The second element pertains to automated or manual tools which "capture" expert knowledge bases in diagnostic-related matrices for use by the designer. These knowledge bases may range from highly sophisticated and automated expert system software to unautomated, rudimentary checklists.

The final element pertains to tools and/or techniques which enable the designer to "author" (i.e., generate) diagnostic software routines and procedures utilizing prime system design data bases or heuristic information sources. These diagnostic authoring tools typically take the form of either expert system "knowledge bridges," which facilitate the extraction and/or generation of diagnostic-related procedures; or automatic test generators/fault simulators, which generate digital test vectors to fault detect/fault isolate an explicitly defined fault universe (i.e., stuck "1"/stuck "0"). In addition, time-tested analog and mixed mode simulators may be utilized not only as functional design tools, but also as diagnostic authoring tools in deriving and analyzing diagnostic test tolerances utilizing worst case or Monte Carlo analysis techniques.

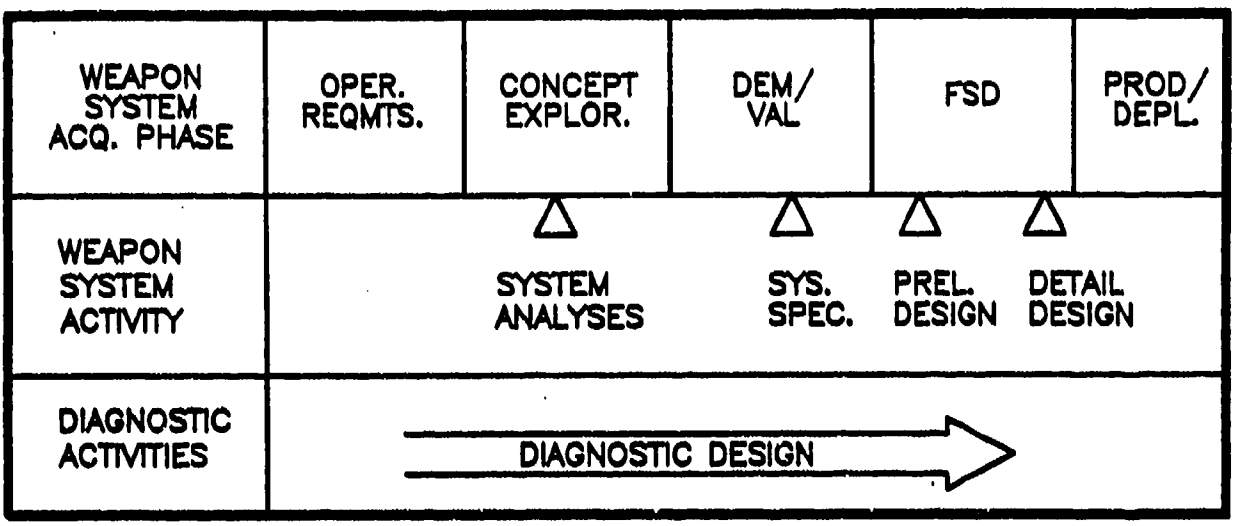
PROVIDING A COHESIVE DIAGNOSTIC DESIGN PROCESS	REQUIREMENT #3.1

CHECKLIST

- ☑ Has a concerted effort been made to assure vertical and horizontal integration and compatibility of all elements which comprise the diagnostic capability? Has this been documented for review?

- ☑ Have steps been taken to utilize automation of the diagnostic design process to enhance design efficiency and to improve the effectiveness of the fielded diagnostic capability? Have available design tools been utilized?

CRITERIA FOR DESIGNING DIAGNOSTIC CAPABILITY	REQUIREMENT #3.2



DIAGNOSTIC ACTIVITY

Design of the diagnostic capability and the elements that make up this capability are initiated early in weapon system development. It begins soon after initial analyses and allocation are completed and extends at least until Full-Scale Development has been completed. Design criteria and guidance need to be available for use as the diagnostic capability design progresses. Obviously, the bulk of this design guidance is utilized by the designer of the prime system and its support capability. He needs to be totally familiar with guidance that is available and be able to apply it appropriately.

PROCEDURE

Design criteria and guidance relating to the diagnostic capability and individual diagnostic elements are available from a number of sources, including standards, handbooks, and guides. Most often, this guidance is not a contractual requirement, except when a specific military standard is invoked. However, for the most part, the contractor should utilize this guidance material as he sees fit, as long as diagnostic requirements are met and interfaces are controlled. In addition, examples which depict the integration of the various diagnostic elements will be of value to both the manager and the designer.

Guidance to the designer consists of material contained in this section and identification of additional guidance where published material is not readily available. Tools to assist in the design process are described in Appendix C, 3.0.

GUIDANCE

The following are references to existing design criteria and guidance.

General Guidance

1. MIL-STD-454, Standard General Requirements for Electronic Equipment

   This standard covers the common requirements to be used in military specifications for electronic equipment. Reliability, maintainability, and human engineering requirements are included in this standard. However, for these types of engineering disciplines, the guidance stresses that this standard does not establish requirements and must not be referenced in contractual documents. These three requirement examples offer direction on what should be considered in preparing contractual documents.

2. MIL-STD-415, Design Criteria for Test Provisions for Electronic Systems and Associated Equipment

   This standard establishes design criteria for test provisions that permit the functional and static parameters of electronic systems and associated equipment to be monitored, evaluated, or isolated. The standard, in its present form, (Revision D) addresses older technologies and thus, if referenced in contractual documents, must be tailored to address only certain provisions in this standard.

3. The RADC Reliability Engineers Tool Kit

   The Tool Kit is intended for use by a practicing reliability and maintainability (R&M) engineer. Emphasis is placed on his role in the various R&M activities of an electronic systems development program. The Tool Kit is a compendium of useful R&M reference information to be used in everyday practice.

4. Study of the Causes of Unnecessary Removals of Avionics Equipment (RADC-TR-83-2)

   This study cites and verifies the causes of unnecessary removals of suspect items from avionics equipment and contains information on minimizing this problem.

System Architecture

Appendix E contains a compendium of testability and diagnostic design techniques, which provides designers various approaches and techniques for achieving improved testing of weapon systems. There are a number of other guides, which address the architecture of the system design, that promote improvements in the system's diagnostic capability. Included are:

1. Architecture Specification for PAVE PILLAR Avionics, January 1987, SPA90099001A

   This specification addresses the advanced avionics architecture which is specifically targeted for advanced tactical fighters and, in general, for all military aircraft applications. This architecture promotes a much-improved approach, which will foster an improved diagnostic capability. An example of this approach is contained later in this section.

2. Reliability, Testability Design Considerations for Fault Tolerant Systems (RADC-TR-84-57)

   Furnished reliability and testability evaluation and application guidance for fault-tolerant designs.

For fault tolerant systems, it is important that the design's inherent testability provisions include the ability to detect, identify, recover, and if necessary, reconfigure and report equipment malfunctions to operational personnel. In addition, because fault tolerant systems often are characterized by complex non-serial reliability block diagrams, a multitude of back-ups with non-zero switch over times, and imperfect fault detection, isolation, and recovery, it is imperative that the technical manager assure that effective testability provisions are incorporated in the system design concept. If not, the design when fielded will exhibit long troubleshooting times, high false alarm rates, and low levels of system readiness.

Fault tolerance and recovery strategies will have a significant impact on the degree to which testability is designed into the system. For example, when incorporating testability/diagnostic capability into the design, the penalties imposed by a fault tolerant system design which employs active redundancy and voting logic may be less than those imposed by a design employing standby redundancy. With active redundancy, the prime system hardware and software are more readily adaptable to perform multiple functions (including those required for testability). In active redundancy systems with voting logic, the performance/status monitoring function assures the operator that the equipment is working properly. This approach also simplifies the isolation of faults since the failure is easily isolated to the locked out branch, by the voting logic. In systems employing standby redundancy, test capability and diagnostic functions must often be designed into

each redundant or substitute functional path (both on-line and off-line) in order to determine their status.

Although the addition of redundancy is usually effective in improving system reliability, the technical manager is cautioned that the reliability improvement may be highly dependent on achievable FD/FI levels. In some cases, it is possible for imperfect FD/FI to actually cause system reliability to degrade as more redundant equipment is added. In general, the effect that varying levels of FD/FI have on system reliability can be evaluated by parametric analyses. The range of FD/FI values used in the analyses should be based on past experience with similar hardware/software systems and adjusted by evolutionary trends and expectations for state-of-the-art devices and designs.

Test Methodology for Fault Tolerant Systems - The following discusses a number of desirable design considerations for fault tolerant system testability.

- Comparison Method - An effective method for testing similar systems with similar inputs and outputs is to compare outputs and flag any gross disagreements. A means to determine which branch is faulted and an error log entry should be mandatory.

- Redundancy Verification - Each redundant path should be tested individually to prevent the mashing of faults in redundant items.

- Flexing of Spares - Periodically activate the built-in-test of the hot spares, log any errors found, and report status before these items are needed for system operation. This will prevent a faulty unit from being switched in when the system reconfigures.

- Voting Scheme Technique - A typical example of a voting scheme technique is to compare output values from three different sources. Confidence is placed in that value where at least two of the three sources agree. Errors found should be logged, and the source of the erroneous value should be recorded and corrected at an appropriate maintenance interval. Since diagnostic procedures are generally designed to locate a single fault, potential exists for the occurrence of multiple faults (e.g., a stuck-at-1 in multiple locations) than can go undetected. It may be necessary to add logic or test circuitry to ensure that each state, and each state transition, occurs correctly.

- Error Correction - Detection of degraded performance in states preceding an error correcting function is difficult since the error correcting function makes its preceding degraded state appear healthy. The error correcting functions should keep count of the number of times a correcting function had to be made and a record made in an error log. When a predetermined threshold

count is exceeded, a test signal may be injected to determine if the input stage is unacceptably degraded.

- Multiple Redundancy - In redundant systems, which are allowed to degrade gracefully through failures of redundant elements, a test should be established to verify that minimum acceptable system performance and redundancy levels are available at the start of a mission.

- Caution Indications - Fault tolerance can be applied to a variety of system types (i.e., electrical, mechanical, hydraulic, environmental, etc.). Regardless of the system type, it is customary to include a cautionary indication whenever a back-up system is called into service, especially for safety critical functions.

Fault Detection Latency Times - One of the most rigid demands imposed upon the testability design of fault tolerant systems is the quick response time necessary to reconfigure. Hence, the testability design process must take into account both spatial and temporal considerations for fault detection. The failure detection approach selection must be based upon the requirement for maximum acceptable failure latency. Continuous failure detection techniques should be used to monitor those functions that are mission critical and/or affect safety and where protection must be provided against the propagation of errors through the system. Periodic testing may be used for monitoring those functions which provide backup/standby capabilities or are not mission critical. Operator initiated testing is typically used for monitoring those functions which require operator interaction, sensor simulation, etc., or which are not easy, safe, or cost-effective to initiate automatically. The maximum permitted latency for failure detection determines the frequency at which diagnostic procedures should be run and must take into account function criticality, failure rate, possible wear out factors, and the overall maintenance concept.

Testability

There are a number of guidance documents which address testability issues. Some of these are listed below. These deal with the design techniques of controllability, observability, and partitioning. Controllability is a design attribute which describes the extent to which signals of interest may be controlled by the test process. It relates to difficulty of test generation, length of test sequence, and diagnostic resolution. Observability is another design attribute which describes the extent to which signals of interest may be observed by the test process. The emphasis is upon selection of the most appropriate test points. Partitioning deals with both the physical hardware and the functional partitioning of the circuitry. Test times and test generation costs escalate rapidly as partitioning size increases.

1. RADC Testability Notebook, Final Technical Report, June 1982

This notebook presents a consolidation of information relating to testability design techniques, procedures, cost trade-off tools, and the relationship of testability to other design disciplines and requirements. Specific examples of good testability design are contained in this document.

2. MIL-STD-2165, Testability Program for Electronic Systems and Equipments

   Appendix B of MIL-STD-2165 cites a series of factors which affect the inherent testability of a weapon system. This information can be used either as design guidance or, if weighted and scored, can actually provide a Figure of Merit for a specific system/unit.

3. Testability Analysis Handbook (Draft)

   At the time of printing the Contractor Program Managers Guide, the Testability Analysis Handbook was in draft form. Publishing is scheduled during FY89. The Preparing Activity is the Naval Sea Systems Command, CEL-DST. This handbook provides a systematic methodology for implementing testability analysis and design requirements, which are prescribed in MIL-STD-2165, Tasks 201, 202, and 203.

4. Predictions of Organizational Level Testability Attributes (RADC-TR-87-55)

   This report documents a methodology for predicting fraction of faults detected, fraction of faults isolated, and fraction of false alarms utilizing field measured data.

Built-In Test

1. Built-In Test Design Guide--Joint AMC/CNO/AFLC/AFSC Commanders, April 1987

   This Joint Service BIT Design Guide provides detailed guidelines on the implementation of BIT, including BIT design techniques at all levels within the weapon system.

2. Smart BIT (RADC-TR-85-148)

   Application of Artificial Intelligence techniques in the design of BIT, to minimize false alarms, retest OKs and non-required maintenance.

3. Sensor Handbook for Test, Monitoring, Diagnostic, and Control System Applications to Military Vehicles and Machinery, National Bureau of Standards

This handbook is intended as a guide for those who design, specify, use, and test weapon systems containing monitoring sensors. The handbook addresses measures and principles of measurement, data acquisition, sensor calibration and testing, environmental considerations, stability, durability, reliability, and error assessment for various types of sensors.

4. Analysis of Built-In Test (BIT) False Alarm Conditions (RADC-TR-81-220)

   This study analyzes the root causes of the false alarm problem and provides design guidelines for avoiding BIT false alarms.

5. Design Guidelines and Optimization Procedures for Test Subsystem Design (RADC-TR-80-111)

   This study provides guidelines and procedures to optimize the design of built-in test.

6. BIT Verification Techniques (RADC-TR-86-241)

   This report covers practical verification techniques for formal and field demonstration of BIT effectiveness.

The problem of Built-In-Test False Alarms and Cannot Duplicates have plagued design for many years. These problems must receive the full attention of system designers. Future generations of BIT must include more emphasis on interpretation of detected system anomalies and better accounting for real world system operating conditions such as fielded system performance, environmental and operational factors.

In order for the BIT to reach, and remain at, its full potential in the field, it must be designed with sufficient flexibility, including the ability to easily adjust test limits and to change BIT software without affecting tactical software.

According to the above referenced document "Analysis of Built-In-Test (BIT) False Alarm Conditions", a common cause of false alarms are sudden environmental stresses such as momentary high temperatures, or a high "G" turn. The Rome Air Development Center, at the time of this printing, is developing a Time Stress Measurement Device (TSMD) chip which will monitor and categorize (in compacted form) data relative to the temperature, axial vibration and shock, and power quality that the equipment sees over time. A larger module has already been developed and flight tested which can monitor such characteristics. In the future, BIT indications can be correlated with TSMD data to help eliminate the occurrence of false alarms and CNDs. The integration of BIT indications, TSMD data, and smart (artificial intelligence) processing may also potentially yield even greater accuracy for onboard diagnostics.

Automatic Test Equipment (ATE)

1. Modular Automatic Test Equipment (MATE) Handbook

   Although Air Force-oriented, this handbook describe procedures and techniques for acquiring automatic test equipment.

2. MIL-STD-2077, General Requirements, Test Program Sets

   This standard covers the acquisition of test program sets for use with ATE. Design criteria is included, which addresses many detail requirements for TPSs.

Human Engineering

1. MIL-STD-1472, Human Engineering Design Criteria for Military Systems, Equipment, and Facilities

   This standard covers general human engineering design criteria which can be applied to any weapon system.

Technical Information

There are a variety of standards which address the preparation of technical publications. Most of these documents are directed at a specific military service. All address the delivery of paper-type documentation. There is no firm guidance relating to other, perhaps more, innovative means for generating and delivering technical information. In the past, many technical publications have been cited to have deficiencies. These deficiencies can best be described in the DoD Audit Report No. 87-115, April 3, 1987, "Summary Report on the Defense-Wide Audit on Acquisition of Technical Manuals and Related Data From Contractors."

Means should be sought for generating and delivering this technical information in a less costly manner, without compromising its quality. There are a number of tools available, or under development, which can assist the designer of this technical information in authoring the text, when electronic delivery of technical information is contemplated. Some guidelines and standards for automatic generation of technical information and its delivery electronically can be obtained from the Human Resources Laboratory at Wright-Patterson Air Force Base. This guidance information has been developed under the Integrated Maintenance Information System (IMIS) Program.

Innovative ideas for displaying this technical information are encouraged, as stipulated in Task 303, MIL-STD-1388-1. Care should be taken to provide for quick access to the required data. For electronic delivery of this data, formats may vary substantially from paper-based technical manuals. Previously specified access times and

information modification times should influence the type of generation and delivery methods. DoD-Instruction 4151.9 requires the services to plan and schedule the acquisition of technical manuals (technical information) to ensure their availability in final form before, or concurrently with, delivery of the system to the field. During design, final plans should be developed, along with the support equipment which is furnished.

Maintenance Aids

There is a need to present technical information and troubleshooting advice to the technician on location and readily available for his use. The maintenance aid, sometimes called a job performance aid, presents information generated by experts to assist the less-experienced technician.

The maintenance aid is a device, publication, or guide used on the job to facilitate performance of maintenance. It delivers:

- o Historical information on what fault was found when similar symptoms were experienced
- o Troubleshooting logic to assist in finding the fault
- o Procedural information which assists the technician in finding and correcting a failure.

Normally, a maintenance aid is used in conjunction with a testing capability. Maintenance aids could be paper-based or employ electronic delivery systems.

Electronic delivery of this type of information opens the door to solving some of the problems associated with paper maintenance aids. Two attributes of electronic delivery systems are:

- o Information can be available to the technician in a matter of seconds by carefully constructed menus, in lieu of the technician having to page through a paper document.
- o The collection of historical data and the subsequent modification to the software programs which deliver this technical information can be updated in a matter of seconds, instead of a matter of months.

This latter attribute lends itself to the introduction of expert systems, which often employ artificial intelligence technology. The expert system has the capability of combining various pieces of information to lead the technician to a logical decision on what is faulty and how it can be repaired.

Another important aspect of the maintenance aid is its ability to train technicians on the job. Training programs must be closely associated with the design and development of a maintenance aid.

Over the past 20 years, many maintenance aids have been designed, developed, and tested. These tests, for the most part, have proven successful. However, the transition of these maintenance aids into the field has not been accomplished to any great extent. One of the reasons is that specifications, standards, and guidance information on how to invoke this requirement are lacking.

A few important facts should be remembered when applying maintenance aids and expert systems.

- The design of the maintenance aids must be done with the user in mind. Once a working model of the equipment is available, there should be a dynamic interchange of information between the maintenance technician and the design engineer to ensure an effective and efficient man-machine interface is attained.

- User acceptance and adoption of maintenance aids will be facilitated in cases where potential users are given a trial period in which to become familiar with these devices prior to their formal implementation.

- A system must be devised to assure timely updating of information to correct errors and to add newly acquired information. Without such a system, the maintenance aid will quite rapidly become obsolete.

Manpower and Training

After personnel and training requirements/allocations have been made, the training curriculum needs to be established concurrently with the system detail design. This includes the formal schooling curriculum, as well as on-the-job training. One of the alternatives available, if electronic delivery of technical information is employed, is combining training aids with the delivered technical information. These two types of information (aiding and training) are somewhat similar in nature and, at times, indistinguishable. The training curriculum should be aimed at the user(s) and accessed in a manner which can be utilized by a variety of users.

These training devices can be freestanding or embedded in the prime system. They can serve as just maintenance training devices or they can be incorporated with operational training. Separate and distinct training devices (maintenance trainers) may be required to be developed for the formal schooling.

PROGNOSTICS

Although rarely considered in electronic system design, prognostics (incipient failure detection) techniques may have a significant impact in improving the operational readiness and mission success rate of future systems. Having the ability to test an equipment to see if any of its' critical components are soon to fail could radically change the repair philosophy for a system. An RADC study entitled "Marginal Checking" indicated that often prognostics techniques must be developed on an item by item basis. This being the case, it makes sense to concentrate first on developing techniques for detecting incipient failure of high failure rate critical components. The "Marginal Checking" study has identified potential prognostics techniques which are appropriate to cables, power supply bridge rectifiers and CMOS circuitry.

Integration of Diagnostic Elements

There are a variety of ways in which diagnostic elements can be integrated to produce a more effective and efficient diagnostic capability. Expert system technology can be incorporated in either ATE or BIT to supplement the basic testing capability. Fault-tolerant design and testability design can be introduced into prime system architecture to promote ease of testing, along with graceful degradation. Maintenance aiding and maintenance training can be combined to provide on-the-job maintenance and training information, utilizing a single portable device or embedded into the prime system. Two examples of this type of integration follow.

ADVANCED AVIONICS ARCHITECTURE EXAMPLE AS EMBODIED IN THE PAVE PILLAR

Advanced Modular Avionics Diagnostic Requirements

Mission availability and probability of mission success expectations for advanced modular avionic architectures such as PAVE PILLAR are totally dependent upon the embedded diagnostic capability that is implemented as an integral part of the design. The implication of this statement represents a significant departure from the traditional relationship that has existed between the circuit design and BIT/testability disciplines. An overview of the PAVE PILLAR modular avionics architecture with its integral diagnostic features is provided in the paragraphs that follow to facilitate an understanding of the new relationship that must be developed.

PAVE PILLAR Overview:

PAVE PILLAR is a highly modular flexible avionic system architecture which employs a common module set to exploit the commonality in air-to-air and air-to-ground missions. The major functional partitions of the avionics suite are: Digital Signal Processing, Mission Processing, Vehicle Management Processing, and Avionics System Control. Figure 7 depicts the enclosing boundaries (i.e., Digital Signal Processing, Mission Processing, and Vehicle Management Processing) for resource sharing, sparing, and substitutions. The unique characteristics of the functions within each of these boundaries preclude the utilization of resources across boundaries for the purpose of recovery or reconfiguration. As a consequence of this partitioning, the diagnostic process takes place within these boundaries and the associated results are communicated across the boundaries to provide the pilot with the required system status. The system architecture which supports the diagnostic process is described in the paragraphs below.

Diagnostic Strategy:

The PAVE PILLAR advanced system architecture employs a hierarchical approach that spans system elements from the individual chips to the entire system. Essential features of this hierarchical diagnostic architecture are shown in Figure 8. The incorporation of a test control function at each level of fault detection (i.e., chip, LRM, subsystem, and system) can facilitate both fault screening and test augmentation functions. The inherent flexibility provided by this architecture provides the system designer the ability to meet weapon system specific diagnostic requirements with a suite of standard modules.

It is essential that both the system and detail designer recognize the importance of implementing such a hierarchical diagnostic structure. A suite of standard line replaceable module (LRMs) will have individual fault detection, fault isolation and false alarm rate specifications that are not necessarily adequate to meet the system-level requirements imposed without fault screening and test augmentation. Advanced BIT concepts, which have been introduced through the "Smart BIT" project, are both compatible and dependent upon the availability of a hierarchical diagnostic architecture. A representative diagnostic implementation is provided in Figure 9.

LRM Bus Structure:

The bus structure associated with the diagnostic system implementation, shown in Figure 9, is dependent upon local maintenance and data buses, as well as system level multiplex data communications. At the chip and module level, the primary diagnostic control is provided over the VHSIC Phase 2 defined TM - Bus.

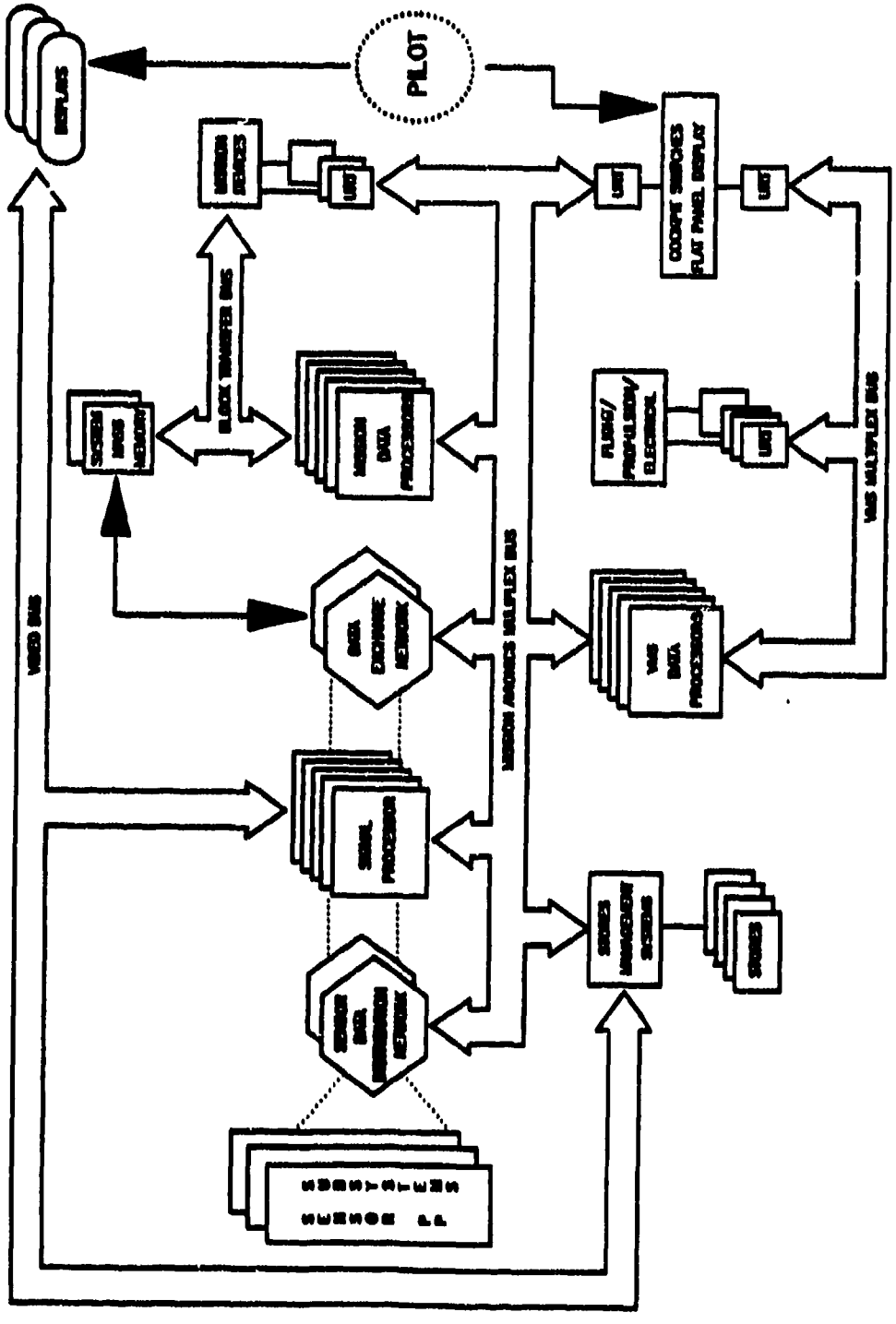


FIGURE 7. – COMMON ARCHITECTURE

HIERARCHICAL APPROACH USED FOR FAULT DETECTION, ISOLATION & SYSTEM/SUBSYSTEM STATUS REPORTING

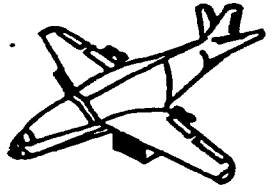
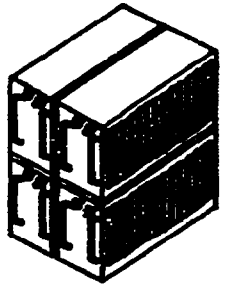
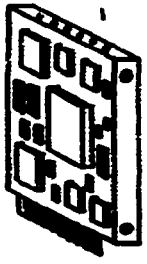
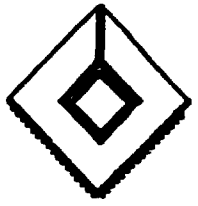
FAULT MANAGEMENT COMPUTER	SUBSYSTEM TEST CONTROLLER	MODULE TEST CONTROLLER	ON-CHIP TESTING
			
• MONITOR STATUS OF SUBSYSTEMS • ENABLES SUB-SYSTEM TESTING • PERFORMS INTER-CONNECT INTEGRITY TESTING • REPORT STATUS TO PILOT AND MAINTAINER	• MONITORS MODULE TEST STATUS • ENABLES MODULE TESTING • TESTS MAINTE-NANCE BUSSES • REPORTS SUB-SYSTEM STATUS TO FAULT MANAGEMENT COMPUTER	• MONITORS COMPONENT FAULT STATUS • INITIATES AND CONTROLS ALL MODULE TESTING • RETRYS BIT • REPORTS STATUS TO SUBSYSTEM TEST CONTROLLER	• CONTINUALLY TESTS ITSELF • FAULTS DETECTED AT LOWEST LEVEL • STATUS INDICATED TO MODULE TEST CONTROLLER

FIGURE 8. DIAGNOSTIC STRATEGY

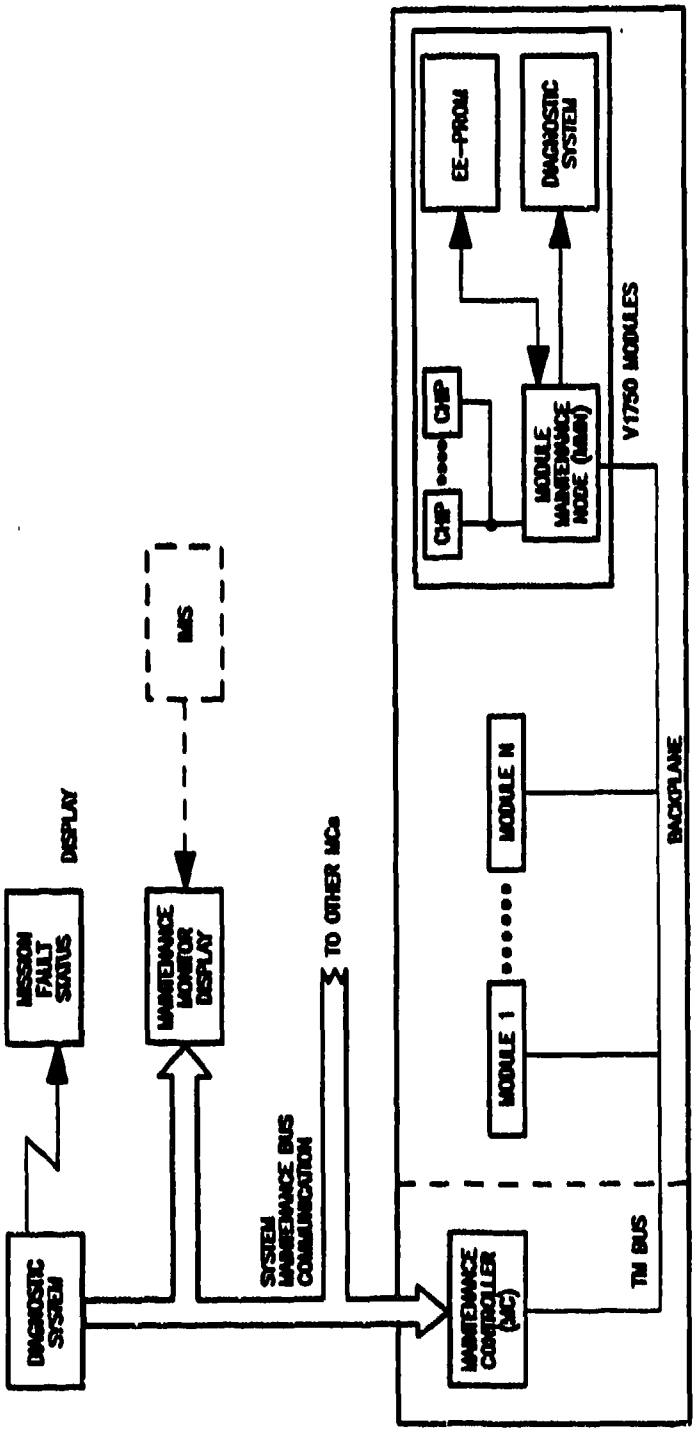


FIGURE 9. DIAGNOSTIC SYSTEM IMPLEMENTATION

Communication of diagnostic information between subsystems within the major PAVE PILLAR partitions (i.e., Signal Processing, Mission Processing, and Vehicle Management Processing) takes place over the respective partition Multiplex Bus. This configuration permits the use of a separate diagnostic/maintenance processor, or the incorporation of these functions within the mission processing function. This diagnostic system implementation is also compatible with the requirements for the System Status function, as currently defined under the Pilots Associate Program. (The Pilots Associate Program is sponsored by the Defense Advanced Research Projects Agency under Program Element Number 62301E. The performing activity is the Wright R&D Center (WRDC).

Diagnostic Applications, Summary and Conclusions:

The diagnostic architecture described above incorporates all the necessary functions to support the following operational and diagnostic/testability objectives:

Fault Detection
Fault Screening
Fault Isolation (Controllability, Observability)
Fault Recovery (Redundancy, Reconfiguration, or Graceful Degradation)
System Status Indication
Maintenance Data Recording
Repair Verification
Off-Board Maintenance Interface (IMIS)

In additional to the complete range of functions identified, the hierarchical diagnostic architecture offers sufficient flexibility for the system designer to achieve the weapon system supportability required. The layered access to the diagnostic capability within the system is essential for the application of artificial intelligence based BIT techniques being developed under programs such as: Smart BIT, Flight Control Maintenance Diagnostic System, Pilots Associate, etc. (These latter two programs are being performed by the Wright R&D Center (WRDC), WPAFB, Ohio.)

B-1B EXAMPLE OF DIAGNOSTIC INTEGRATION

The B-1B Bomber provides an example of many different facets of diagnostics deliberately combined to provide a total integrated diagnostics capability. The diagnostic elements include conventional and artificial intelligence diagnostic techniques, real-time in-flight performance monitoring and ground readiness testing, system performance monitoring for aircrew information and LRU fault isolation for maintenance personnel, detailed embedded data acquisition equipment and ground processing, standard inspection and other scheduled maintenance tasks (primarily in mechanical areas), and status information developed by the defensive and offensive avionics.

As shown in Figure 10, the core of the B-1B diagnostics is an on-board Central Integrated Test System (CITS). The general philosophy of CITS is to monitor and record activity on the aircraft equipment buses as well as performance status information generated by the defensive and offensive avionics system. Signal levels are compared to standards by the CITS computer. In the event of a failure, a CITS Maintenance Code (CMC) is generated identifying the faulty LRU. Both the CMC and measured signal levels are recorded for later analysis by a ground processor located in the intermediate shop.

Embedded equipment which makes up CITS include four data acquisition units, a CITS computer, an airborne printer, a CITS control and display panel, and the CITS maintenance recorder.

Associated ground equipment is the CITS Ground Processor. It is used for retrieving and interpreting the data recorded during each flight. The artificial intelligence portion of the diagnostics (CITS Expert Parameter System or CEPS) is also resident in a separate ground computer. The CITS Ground Processor is used to evaluate the maintenance codes recorded in flight and issue work orders directing the removal of the faulty LRU. CEPS is used when the CMC does not isolate the fault to single LRU. CEPS utilizes past history, expert diagnostic approaches, and monitored environmental data at the time of the failure to further break the CITS ambiguity groups for isolation to the single failed LRU.

Technical Orders (TOs) and crew training still play an important part in the overall diagnostics. Ground readiness tests are manually initiated following an LRU replacement. These tests are to assure proper system operation. They are performed per instructions in the TOs using the applicable tests of CITS. In addition there are limited physical inspections directed by the TOs covering the traditional but still effective monitoring of wear and fluid contamination.

The design process of integrating all of the above centered around three established disciplines. They are 1) a structured systems engineering approach, 2) a Failure Mode, Effects, and Criticality Analysis (FMECA), and 3) Logistic Support Analysis and Support Equipment Recommendation Data development.

CITS design manuals governed the systems engineering process. These manuals were created following MIL-Standard software development specifications and associated reviews. A basic document called CITS Autoflow was created for each system/subsystem which delineated the tests to be made for fault detection and isolation. The Autoflow identifies which inputs and outputs from each box are to be checked to assure that the problem is within the box and not caused externally.

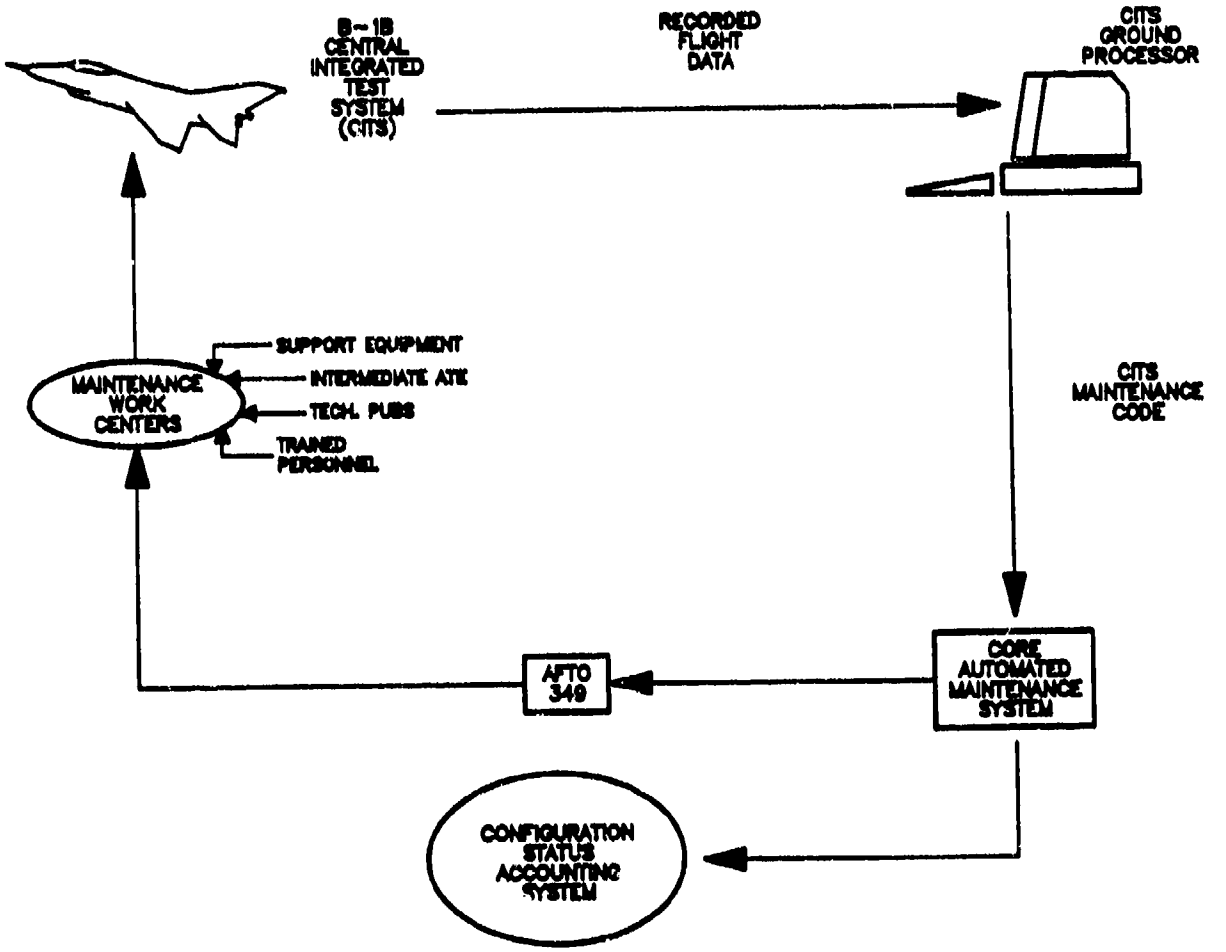


FIGURE 10. B-1B MAINTENANCE CONCEPT

CRITERIA FOR DESIGNING DIAGNOSTIC CAPABILITY	REQUIREMENT #3.2

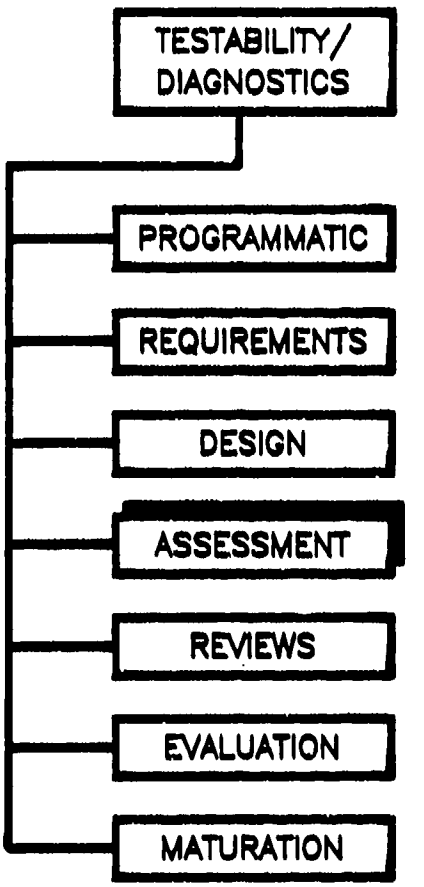
A detailed Failure Modes, Effects and Criticality Analysis (FMECA) provided the initial basis for selecting CITS tests. This was augmented by a structured LSA which identified all diagnostic tasks required to be accomplished. Part I Support Equipment Recommendation Data (SERD) was created documenting the need for all applicable support equipment. CITS personnel then evaluated each SERD and where possible made sure that the requirement was met by CITS. Where applicable, other visual inspections, etc., were also considered. All SERD requirements were eventually satisfied without the use of a significant amount of ground support equipment.

CRITERIA FOR DESIGNING DIAGNOSTIC CAPABILITY	REQUIREMENT #3.2

CHECKLIST

- ☑ Are available design criteria, rules, and other available guidance documentation being used?
- ☑ Is the integration of the various diagnostic elements being accomplished to provide a more effective and efficient diagnostic capability?
- ☑ Is design-for-test an integral part of the system design process?
- ☑ Has the designer chosen the testability/diagnostic design techniques appropriate for the level (or levels) of test?
- ☑ Have the general design considerations and the standard testability approaches been thoroughly evaluated and trade offs performed?
- ☑ Have appropriate fault isolation techniques been incorporated into the overall approach?
- ☑ Has the impact of the physical packaging of the system been thoroughly evaluated?
- ☑ Has a test and maintenance bus been considered for integration into the system?

ANALYSIS AND ASSESSMENT OF THE PERFORMANCE OF THE DIAGNOSTIC CAPABILITY



OVERVIEW

Throughout the design of the weapon system's diagnostic capability, it is essential to analyze, assess and demonstrate its performance. Such assessments are an integral part of logistics, reliability, maintainability, testability, human engineering, software and safety programs. The ability to properly conduct these analyses, assessments, and demonstrations is hampered by the lack of available techniques and tools to help, and the incompatibility of the available tools and techniques to function together. Thus both the program manager and the designer must have sufficient knowledge to understand the processes utilized and integrate these processes and tools to do the best possible job.

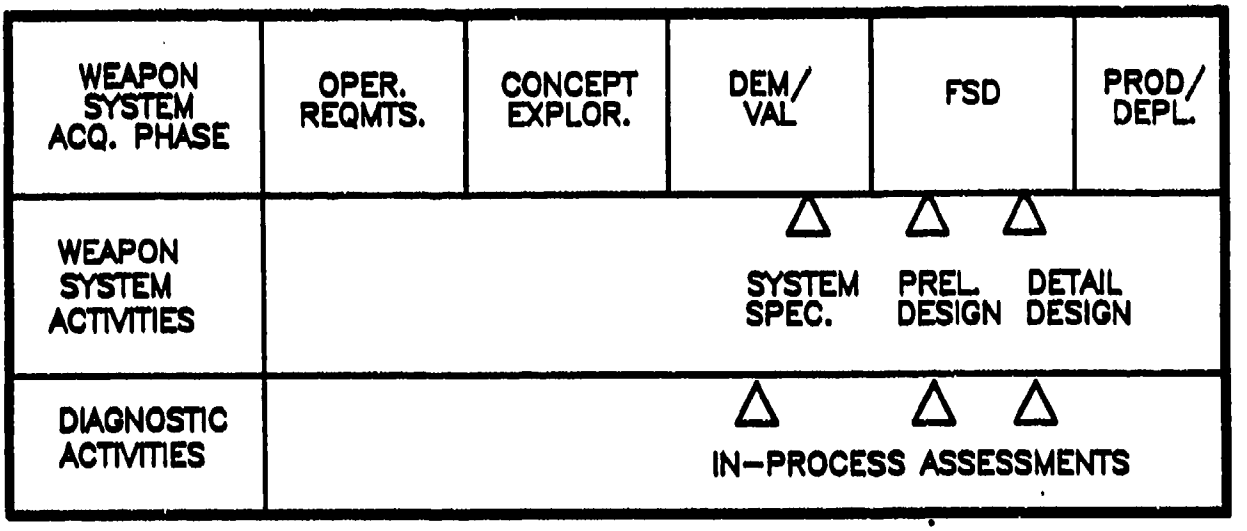
IMPORTANT CONSIDERATIONS TO BE ADDRESSED

Reqmt.

4.1 Analysis and assessment of the diagnostic capability should be performed for the entire diagnostic capability, as well as for each diagnostic element.

4.2 Maintainability demonstrations should be designed to verify that diagnostic requirements have been met.

IN–PROCESS TESTABILITY/DIAGNOSTIC ANALYSIS	REQUIREMENT #4.1



DIAGNOSTIC ACTIVITY

During Dem/Val and FSD, it is important to assess whether the testability/diagnostic requirements are being achieved. This activity encompasses all preliminary and full-scale engineering activities pertaining to both the embedded and external diagnostic capability.

PROCEDURE

In-process testability/diagnostic analyses can be conducted at most any time within Dem/Val and FSD. These in-process analyses are typically reviewed by the government at Preliminary Design Reviews and Critical Design Reviews. These analyses are, for the most part, implemented per MIL-STD-2165 (Task 202, Preliminary Design, and Task 203, Detail Design). Normally, these analyses will be the responsibility of the design or test engineer.

GUIDANCE

Basically, there are two types of in-process analyses. The first deals with the inherent testability of the hardware design and is independent of test stimuli and response data. The second type deals with the effectiveness of the diagnostic capability which deals with measures that include consideration of hardware design, embedded diagnostics, and external diagnostics. Diagnostic effectiveness measures include, but are not limited to, fault coverage, fault resolution, fault detection time, fault isolation time, and false alarm rate.

There are a number of techniques and tools available, both automatic and manual, which can be used to assist in these analyses. A thorough description of available techniques is contained in the Testability Analysis Handbook, which is referenced under Requirement #3.2. A description of available tools to assist in these analyses is contained in Appendix C.

INHERENT TESTABILITY

The first analysis deals with inherent testability. Inherent testability assessment is an evaluation of how well a design supports the testing process, whether built-in test or off-line test. The evaluation is preformed on the preliminary design and is performed before any test design is performed. It is, therefore, based solely upon the hardware design features, such as physical and electrical partitioning, controllability, observability, and test point placement, etc. The key to performing an inherent testability assessment is the identification of features which support or inhibit the diagnostic process early, at a point in time when the design can be changed relatively easily. The concept and the implementation of an inherent testability assessment can have great impact on overall system supportability.

In general, three generic groups of inherent testability predictive techniques are available, each with its unique advantages and disadvantages. Checklists are low cost, manual, and somewhat simplistic. Logic models utilize the actual circuit topology but often regard everything as a block, with inputs and outputs. The more detailed algorithmic approaches, such as Sandia Controllability/Observability Analysis Program (SCOAP), require libraries of the devices that most nearly simulate actual circuit devices.

Checklist approaches to inherent testability assessment have some very good characteristics. Checklists are manual approaches to testability assessment, yet are easily automated into an interactive format for the designer to input design features to allow testability grading. However, significant engineering analysis is still required. Two checklists of that type are the RADC PCB Testability Design Rating System and the MIL-STD-2165 Appendix B Checklist. The original RADC PCB rating system was limited to lower density digital board applications, whereas MIL-STD-2165 Appendix B covers analog, digital,a d hybrid applications from module to system level. An updated RADC PCB rating system now under development will be expanded to cover all modern digital,

analog and hybrid PCBs. The RADC rating system has fixed items of weighting, whereas MIL-STD-2165 Appendix B allows subjective treating of items and weighting values. Both checklists can be utilized early in design.

Logic models have considerable success and validity other than in support of the testability discipline, including logistics, fault isolation, integrated diagnostics, and maintainability. The logic model algorithms are of varying sophistication and validity, although the methodology for defining dependencies are similar.

Logic models systems for testability are applicable to analog, digital, and hybrid applications. They can be modeled at the component, board, or module subsystem and system level. One limitation of this broad approach is that every item is identified as a box with inputs and outputs. Thus, box complexity might range from and OR gate to a complete microprocessor. The same variations applies to the lines between boxes. Critical signals, such as a clock or a tri-state bus are not more important than any other line. Two of the more well-known models are Logic Modeling (LOGMOD) and System Testability And Maintenance Program (STAMP). Both are mature in nature, but each is tied to a specific vendor at the present time.

Algorithmic approaches are perhaps the most sophisticated approach. SCOAP seems to usually perform well, but has had some library limitations in the important area of CMOS primitives. Some CAE workstation vendors are including modified versions of SCOAP for up-front testability analyses. Daisy workstations include the Daisy Testability Analyzer (DTA) package, and GE/CALMA workstations include the Controllability-Observability-Predictability-Testability Report (COPTR) package. Both have improved on the basic SCOAP, via top-down modeling and large device model libraries of the more common IC types. GenRad also has a package called HITAP, which is based on the Computer-Aided Measure for Logistic Testability (CAMELOT) algorithm.

Another major issue surrounding inherent testability assessment is that many of the automated tools which exist are proprietary. This proprietary nature of the tools creates implementation problems from both a cost and a contractual point of view. Often, the best approach is to utilize a nonproprietary automated tool for inherent testability assessment.

Prior to the FSD phase, the design or test engineer should develop a total strategy for conducting inherent testability assessment on all systems, subsystems, etc. Based upon the availability and applicability of current inherent testability assessment approaches, it is anticipated that combination of tools and techniques will be required to form a totally comprehensive measurement capability in areas where an automated capability is not available, use the baseline of existing models to make modifications to provide the total capability required.

An evaluation criteria for inherent testability assessment tools and techniques should be developed based upon specific system and subsystem specific needs. The following list of evaluation criteria is recommended:

- o Automation; degree of automation
- o Proprietary nature
- o User friendliness
- o Automated link to design data base
- o Acceptability of output to the government
- o Cost of use
- o Availability (currently available or under development)
- o Quality
- o Sensitivity to key testability features
- o Feedback provided (does it recommend design fixes?)
- o Comprehensiveness (digital, analog, RF, VHSIC, mechanical, etc.)
- o General techniques; principles used
- o Link to test effectiveness prediction technique
- o Output reports
- o Scoring methodology
- o Applicability to chip, board, subsystem

TEST EFFECTIVENESS

The second type of analysis deals with test effectiveness. Traditional approaches to determining test effectiveness call for the generation of test sequences at the completion of the design phase and a measure or measures made of their effectiveness. The analysis need not wait on the completion of BIT and/or off-line TPS software. Modeling is encouraged, since this approach can analyze test effectiveness on a large number of postulated faults prior to incorporating the test stimulus in either an embedded or off-line program. The results of the analysis can feed forward, so as to

influence BIT or TPS software design, and feed backward to influence possible redesign of the primary system to improve its testability. Test effectiveness measures have traditionally included:

- Fault coverage
- Fault resolution
- Fault detection time
- Fault isolation time

Computer programs are used to input (via software) a large number of faults into the software model of the hardware item (UUT). The response of the simulated item to the test sequence is then evaluated, given the presence of the simulated faults. Fault detection, resolution, etc., are automatically ascertained. Most modern Automatic Test Program Generation (ATPG) and simulation systems have very efficient fault simulation capabilities. The HITS system, for example, runs a concurrent fault simulation to greatly speed the process. The usefulness of this approach in measuring test effectiveness depends on the adequacy of the models (hardware item model and fault model) to accurately reflect the real-world situation. Modeling must be achieved at a level of detail that allows all known and statistically significant failure modes to be included.

Although commonly accepted, the application of these measures is in various stages of maturity, based upon the equipment composition (i.e., digital, analog, radio frequency and/or mechanical). At this time, the application experience has been concentrated in the area of digital implementations. However, even in this area, significant additional effort will be required in order to relate these measures to operational performance. The degree of application of test effectiveness measurement techniques to the remainder of the listed equipment types has been quite limited to date. IDSS, the Navy's integrated diagnostics program, has recognized this need and has an active diagnostic tool development program underway. One of these tools, the Weapon System Testability Analyzer, is structured to address test effectiveness measurement, as well as inherent testability assessment.

Effective and realistic fault modeling is a key element in the development of the simulation capability needed to support the development of either an ATPG or an automated test effectiveness measurement tool. However, it is anticipated that no single fault model and/or simulator will be applicable to the broad range of equipments to be employed within a complex system; therefore, a combination of models will be required to meet the requirement for automated determination of fault detection and isolation levels.

For False Alarm estimation, a procedure which is documented in a report (RADC-TR-87-55) entitled "Predictors of Organizational - Level Testability Attributes" developed prediction equations for various testability related parameters. Rather than try

to develop an equation to predict False Alarm Rate (FAR) directly, an approach was taken to predict the CND rate since this parameter should closely track FAR. The following details a prediction method for CND rate. Note that this is a model based on empirical data from avionics equipment of the mid-1980's era and, as is the case with all models, care must be taken in using the model for new technology or applications.

The equation for CND rate follows:

CND RATE =-0.0028 + 0.375*FLRRATE
+2.8 E-5*TRANSIENT + 5.9E-11*TC7

The variable FLRRATE accounts for the LRU Failure Rate.

The variable TRANSIENT attempts to characterize the tendency of an LRU to exhibit intermittent failures resulting from marginal or degrading components.

The variable TC7 numerically characterizes the likelihood of a sneak circuit existing in an LRU.

TRANSIENT = (IC's + 2*RESISTORS + 41*RELAYS + 2*CAPACITORS
- 9*TRANSISTORS)/# of SRU's

TC7 = INTERCONNECTS*(IC's - 160*RELAYS
- 960*SWITCHES + 30*TRANSISTORS)

Where;

RELAYS = Total # of Relays in LRU.

CAPACITORS = Total # of Capacitors in LRU.

RESISTORS = Total # of Resistors, both fixed and variable, in LRU.

TRANSISTORS = Total # of discrete transistors, including FET's, Bipolar, etc. that are in LRU design.

IC's = Total # of integrated circuits in LRU.

SRU's = Total # of SRU's that compose an LRU.

INTERCONNECTS = Total # of electrical interconnects used to mate SRU's to the host LRU.

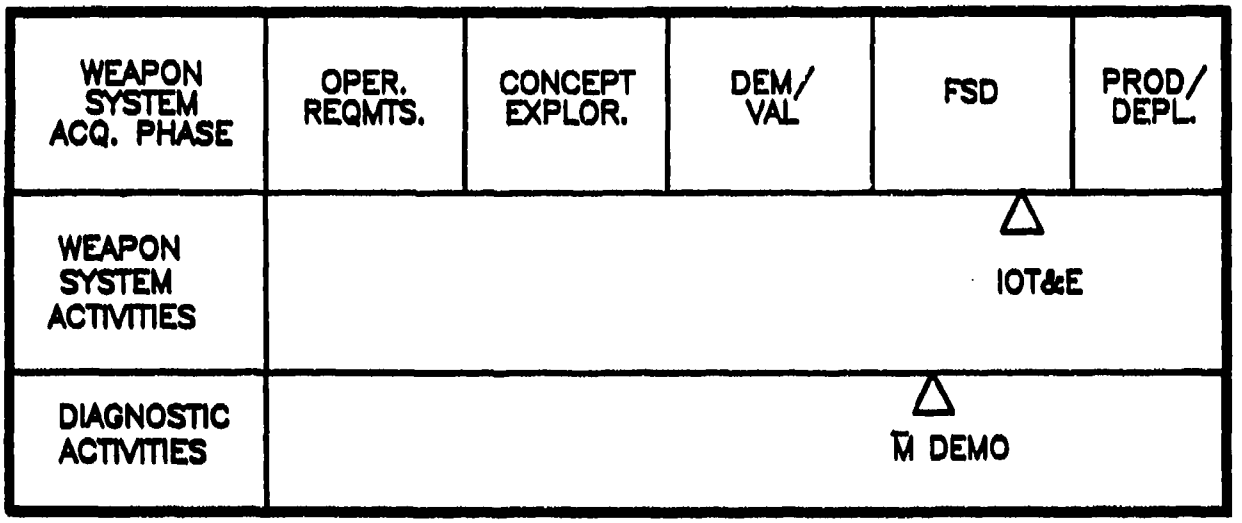
SWITCHES = Total # of switches in the LRU design.

Specifics about the development or application of this equation, or the estimation of the input parameters can be found in the above referenced report.

CHECKLIST

- ☑ Does the analysis of testability/diagnostic requirements address all major support disciplines?
  - Off–line ATE
  - Embedded diagnostics
  - Manpower required to support analysis outputs
  - Training requirements
  - Information requirements.
- ☑ Are all analyses complete and unambiguous? Do they meet specification requirements?
- ☑ Is the analysis integrated and cohesive? Are any requirements in conflict?
- ☑ Are the training, information, and manpower requirements adequately scoped and specified to support the technical complexity of the subject end item in its operational environment?
- ☑ Have available tools been selected and used?

MAINTAINABILITY DEMONSTRATIONS	REQUIREMENT #4.2



DIAGNOSTIC ACTIVITY

Maintainability Demonstrations are performed in accordance with the appropriate demonstration method contained in MIL-STD-471A. Notice 2 of MIL-STD-471A (USAF) contains requirements for demonstration and evaluation of system BIT/external test/fault isolation/testability attributes. This method will demonstrate the integration of the diagnostic capability for the system (e.g., integration of embedded test software and hardware techniques, automatic and manual test, BIT/SIT, training levels, human interfaces). The Maintainability Demonstrations evaluate the diagnostic performance of the system with respect to the diagnostic performance criteria and objectives established in accordance with MIL-STD-470 (Maintainability Program) and MIL-STD-2165 (Testability Program) and the requirements for an integrated diagnostics capability demonstration contained in the FSD SOW.

PROCEDURE

The integrated diagnostics process increases the scope of maintainability demonstrations. It is the Contractor Program Manager's responsibility to ensure that this increased scope is understood and implemented. It is the designers responsibility to design the demonstration, evaluate the results and take the necessary corrective actions.

The scope of Maintainability Demonstrations includes:

1. Demonstration of Testability Parameters
   - BIT Fault Detection
   - BIT Isolation Time

- BIT Fault Isolation Level (Ambiguity Group)

2. Demonstration of Test Effectiveness (ATE) (MIL-STD-2077)
   - ATE Fault Detection
   - ATE Fault Isolation Time
   - ATE Fault Isolation Level (Ambiguity Group)
   - UUT/ATE Compatibility

3. Demonstration of Technical Information
   - Technical Information Access Time
   - Technical Information Relative Access Ease
   - Technical Information Format
   - Technical Information Usability

4. Demonstration of Training/Skills
   - Relationship between maintenance procedures and skills
   - Relationship between formal training and actual maintenance job flow.

5. Demonstration of Vertical and Horizontal Integration
   - Compatibility and Consistency of diagnostic demonstration results between maintenance levels and among their respective diagnostic elements.

GUIDANCE

Unfortunately, the ability to carry out a single demonstration, or even a series of demonstrations, to prove/evaluate this level of diagnostic capability is dependent upon having all of the diagnostic elements available for the maintainability demonstration. While this should always be the goal, it may not be feasible for all of the above due to development schedules, UUT design instability, data availability and other overall program constraints. (Note that this is a primary reason for a Diagnostics Maturation Program.)

Typically, the contractor prepares a Maintainability Demonstration Plan early in the FSD Phase and that plan is subject to government review and approval. The designer should take advantage of this opportunity to design the Maintainability Demonstrations to include the factors cited above. This can have a significant cost-savings impact on the Diagnostics Maturation Program requirements. Maintainability Demonstrations represent the first major opportunity to evaluate the level of diagnostic capability achieved. Also, Maintainability Demonstrations can be conducted early enough to implement corrective action cost-effectively. Demonstrations are conducted while the system is still considered to be in the Development Phase. After the demonstrations are completed, the relative cost of identifying deficiencies and implementing corrective action is significantly increased.

A significant milestone of 'Government Acceptance' occurs upon satisfactory completion of Maintainability Demonstrations. After this milestone, costs for identification and resolution of diagnostic deficiencies may be subject to contract interpretation and/or negotiation. The total strategy for the test and evaluation of the diagnostic capability is placed on the TEMP, and detailed in the Integrated Test Plan.

Based upon the selected scope of the Maintainability Demonstration, procedures from MIL-STD-471 are utilized and adapted for the scope. These procedures are documented in the Maintainability Demonstration Plan. The results of the Maintainability Demonstration are documented in a technical report - Maintainability Demonstration Results.

Concurrent Demonstrations

The overall diagnostic capability is the sum of a variety of diagnostic elements. Therefore, a requirement should be established for early demonstration of the entire diagnostic capability produced by the integration of all of these diagnostic elements. This is referred to as concurrent demonstrations, where the timing of various diagnostic element demonstrations are planned and scheduled for concurrency so that the integrated capability can be assessed.

Each element of the diagnostic capability must be demonstrated, as well as the result of the combining or integration of the elements. For example, a demonstration of subsystem BIT may prove fault detection and isolation levels. A demonstration of ATE may prove external fault detection and isolation levels. A concurrent demonstration of these two diagnostic elements will prove the ability of the ATE to use BIT circuitry, to use BIT results, and the consistency of test results between BIT and ATE. By concurrent demonstration, the whole is greater than the sum of the parts. A significant set of factors related to the result of the integration of the diagnostic elements must be evaluated early.

CHECKLIST

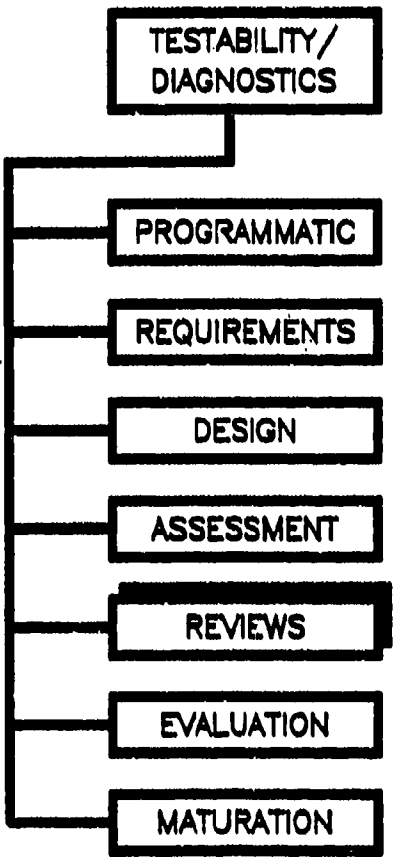
- ☑ Does the demonstration plan provide a 100% fault coverage capability across all levels of maintenance?
  - Organizational Level
  - Intermediate Level
  - Depot Level
- ☑ Are the failure modes to be demonstrated and criteria to be utilized adequately specified for each maintenance level? Will an adequate number of faults be inserted as required by MIL-STD-471 to statistically prove that FD/FI requirements are or are not met?
- ☑ Is the demonstation structured to provide an evaluation of the diagnostic capabilities as an integrated system?
- ☑ Do the subject plans demonstrate an integrated, cohesive maintenance flow in terms of demonstrating how a fault would be detected and repaired? Is a systems approach to the maintenance process taken in the overall approach to demonstration planning?

CONDUCTING DESIGN REVIEWS

OVERVIEW

During the acquisition of a weapon system there are at least eight formal technical reviews and audits, which may be conducted by the contractor for the Government Program Manager. As in the diagnostic design process, there is a tendency to conduct separate reviews and audits based upon the function being addressed. This particularly refers to logistics, reliability, maintainability, testability, human engineering, and safety. Integration of these reviews and audits to address diagnostic issues is a must. MIL-STD-1521 is the prime document which defines the issues to be addressed at each of these formal reviews. At present, these checklists are inadequate in terms of both testability and diagnostics and, thus, these reviews and audits may not serve their purpose. Additional guidance must be given to both the government and the contractor in order to alleviate this problem.

Informal reviews are often required. Guidance for these informal reviews can be drawn from formal review guidance.

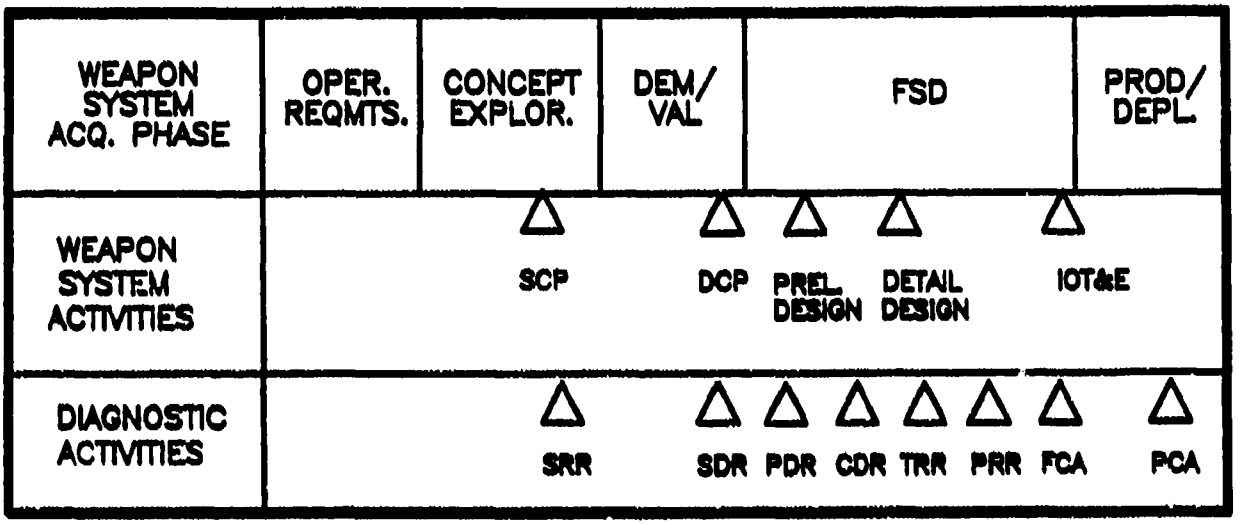


IMPORTANT CONSIDERATIONS TO BE ADDRESSED

Reqmt.

5.1 Technical reviews and audits must address all facets which affect the performance of the diagnostic capability.

CONDUCTING TECHNICAL REVIEWS AND AUDITS	REQUIREMENT #5.1



DIAGNOSTIC ACTIVITY

Technical reviews and audits are an important factor in assuring that the government is furnished with a weapon system which meets its requirements.

PROCEDURE

MIL-STD-1521 lists 10 formal technical reviews and audits. Of these 10, eight are considered critical in the achievement of a satisfactory diagnostic capability. The following guidance supplements and expands the guidance contained in MIL-STD-1521, Technical Reviews and Audits for Systems, Equipments, and Computer Software.

Although design reviews are recognized as being important to verify design before production, the lack of depth in these reviews is alarming. The cause of these inadequate reviews must be shared by both the contractors and the government. Contractually, the government rarely requires the contractor to do a comprehensive technical review, and the contractor does not do so unless required, even though it may be cost effective from his point of view. Even when the right words are used, the end results depend largely on corporate policy to allocate sufficient resources to perform a detailed analysis of the design and associated processes. The designer, obviously, has an important input to these reviews. Therefore, it follows that he must understand the objectives and scope of these reviews.

GUIDANCE

Guidance relating to these various reviews is contained in the appendices to MIL-STD-1521. Because these appendices do not address all aspects of testability and diagnostics, some supplemental information is included in the following paragraphs.

System Requirements Review (SRR)

The objective of this review is to ascertain the adequacy of the contractor's efforts in defining system requirements. It will be conducted when a significant portion of the system functional requirements has been established.

The diagnostic capability review portion of the SRR will analyze the system items that are related to diagnostics. The following items will be reviewed, as appropriate:

- Mission and Requirements Analysis
- Functional Flow Analysis
- Preliminary Requirements Allocation
- System/Cost Effectiveness Analysis
- Trade Studies
- Synthesis
- Logistic Support Analysis
- Specialty Discipline Studies
- Generation of Specifications
- Program Risk Analysis
- Integrated Test Planning
- Technical Performance Measurement Planning
- Engineering Integration
- System Safety
- Human Factors Analysis
- Life Cycle Cost Analysis
- Manpower Requirements/Personnel Analysis
- Milestone Schedules.

The diagnostic capability review should address the impact of the results of the items listed above on the diagnostic pieces listed below.

- Designed-In Reliability, Prognostics, and Testability
- Self-Test, Built-In Test, System Integrated Test
- Support Equipment, Maintenance Aids
- Technical Data
- Personnel Skill Requirements
- Training and Training Devices.

System Design Review (SDR)

This review shall be conducted to evaluate the optimization, correlation, completeness, and risks associated with the allocated technical requirements. Also included is a summary review of the system engineering process which produced the allocated technical requirements and of the engineering planning for the next phase of effort. Basic manufacturing considerations will be reviewed, and planning for production engineering in subsequent phases will be addressed. This review will be conducted when the system definition effort has proceeded to the point where system characteristics are defined and the Configuration Items (CI) are identified.

Specific diagnostic considerations relate to:

- o Optimizing the diagnostic capability (changes after Dem/Val usually are more costly and time consuming)
- o Preparation of a Maturation Plan
- o Preparation of a System Specification which provides a capability for addressing 100% FD/FI for each level of maintenance
- o Allocation of diagnostic requirements for each diagnostic element
- o Review of the software requirements specification to assure that embedded diagnostic software considerations are included.

Preliminary Design Review (PDR)

This review shall be conducted for each Configuration Item or aggregate of Configuration Items to: (1) evaluate the progress, technical adequacy, and risk resolution (on a technical, cost, and schedule basis) of the selected design approach; (2) determine its compatibility with performance and engineering specialty requirements of the Hardware Configuration Item (HWCI) development specification; (3) evaluate the degree of definition and assess the technical risk associated with the selected manufacturing methods/processes; and, (4) establish the existence and compatibility of the physical and functional interfaces among the Configuration Item and other items of equipment, facilities, computer software, and personnel. For CSCIs, this review will focus on: (1) the evaluation of the progress, consistency, and technical adequacy of the selected top-level design and test approach; (2) compatibility between software requirements and preliminary design; and, (3) on the preliminary version of the operation and support documents.

In addition, the following items in the diagnostic area should be presented at the appropriate depth for review.

- o Preliminary Failure Modes and Effects Analysis
- o Design data analyses for BIT/SIT integrated diagnostics, including requirements and preliminary design verification results
- o Maintenance concept for the portion of the system being reviewed and its traceability to the system maintenance concept
- o Operational maintenance functions
- o Results of the analysis of the inherent (intrinsic) testability of the preliminary design
- o Allocation of qualitative and quantitative requirements
- o Criteria for external diagnostic elements
- o Trade-off studies
- o Cost/System Effectiveness Modeling and Life Cycle Cost Analysis
- o Preliminary Logistic Support Analysis, including task analysis and related personnel and support equipment information
- o Evaluation of alternatives
- o Test and evaluation plans.

Critical Design Review (CDR)

This review shall be conducted for each Configuration Item when detail design is essentially complete. The purpose of this review will be to: (1) determine that the detail design of the Configuration Item under review satisfies the performance and engineering specialty requirements of the HWCI development specifications; (2) establish the detail design compatibility among the configuration and other items of equipment, facilities, computer software and personnel; (3) assess Configuration Item risk areas (on a technical, cost, and schedule basis); (4) assess the results of the producibility analyses conducted on system hardware; and, (5) review the preliminary hardware product specifications. For CSCIs, this review will focus on the determination of the acceptability of the detailed design, performance, and test characteristics of the design solution, and on the adequacy of the operation and support documents. The CDR shall be conducted on each Configuration Item prior to fabrication/production/coding release to ensure that the detail design solutions, as reflected in the Draft Hardware Product Specification, Software Detail Design Document (SDDD), Data Base Design Document(s) (DBDD), Interface Design Document(s) (IDD), and engineering drawings, satisfy requirements established by

by the Hardware Development Specification and Software Top-Level Design Document (STLDD). The CDR shall be held after the Computer Software Operator's Manual (CSOM), Software User's Manual (SUM), Computer System Diagnostic Manual (CSDM), Software Programmer's Manual (SPM), and Firmware Support Manual (FSM) have been updated or newly released.

It is desired at each CDR to provide as much assurance as practicable that all diagnostic requirements are satisfied: I. e., that 100% diagnostic capability will exist for each CI in the system. While it probably will not be practicable to certify that this will exist, the following data should be presented as an extension of the information presented at the PDR.

- Detailed fault detection/fault isolation analyses that identify the extent to which BIT/SIT detect and isolate faults and which identify those failures that will require support equipment and/or manual methods to detect and/or isolate.

- Diagnostic allocations in Part II CI specifications to the LRU and SRU level. Traceability of these requirements to the Part I CI System Specification should be demonstrated. Note: Flexibility to reallocate diagnostic allocations until product baseline is established at PCA should be provided within the envelope of system requirements.

- Definition of the maintenance plan/concept for the CI, together with supporting LSA documentation, including support requirement and level-of-repair analysis results. Logistic simulation results should be presented to substantiate the plan/concept.

- Presentation of testability analysis/assessment results for the CI design to substantiate the fault detection/ fault isolation analysis.

- Early program failure identification, prevention, and detection analyses applicable to the CI should be presented to assist in verifying diagnostic capability.

- Review detailed Maintainability Demonstration Plan for inclusion of diagnostic capability test requirements

- Appropriate updates to the items reviewed during the PDR.

Test Readiness Review (TRR)

This review is conducted for each CSCI to determine whether the software test procedures are complete and to assure that the contractor is prepared for formal CSCI testing. Software test procedures are evaluated for compliance with software test plans

and descriptions and for· in accomplishing test requirements. At TRR the contracting agency also reesults of informal software testing and any updates to the operation and support s. A successful TRR is predicated on the contracting agency's determination thware test procedures and informal test results form a satisfactory basis for proceformal CSCI testing.

The diagnostic s the system/CI TRR(s) shall be a formal review of the contractor's readiness to tal diagnostics-related CSCI testing. It is conducted after the software test proce available for diagnostics-related CSCI, such as CI BIT, System BIT, SIT, eter computer system component (CSC) integration testing is complete.

The items to be nclude:

1. Requirement --

   Any changeSIT, or testability requirements contained in the system/CI Sequirement Specification or Interface Requirements Specification not been approved and which impact CSCI testing.

2. Design Chan

   Any changes the BIT, SIT, or testability design parameters contained in tre Top-Level Design Document (STLDD), Software Detail Designt (SDDD), Interface Design Document(s) (IDD) since the PDR and h impact CSCI testing.

3. Software Testd Descriptions --

   Any changesibedded diagnostic element portion of the approved Software TestP) and Software Test Descriptions (STD).

4. Software Testes --

   Test proceduused in conducting BIT and/or SIT test effectiveness validation as e CSCI testing, including retest procedures for test anomalies anens.

5. Integration Te Procedures, and Results --

   Any embeddostic element CSC (e. g., BIT components, SIT components)on test cases, and procedures used in conducting informal diagment CSC integration tests and the test results.

6. Software Test Resources --

   Status of any software test resources that are required specifically for embedded diagnostic element CSCI testing. Such resources may include diagnostic test personnel and supporting test software and materials, including software test tool qualification and review of the traceability between requirements and their associated tests.

7. Test Limitation --

   Identification of all software test limitations associated with embedded diagnostic element CSCI/CSC testing.

8. Software Problems --

   Summary of embedded diagnostic element software problem status, including all known discrepancies of the CSCI and test support software.

9. Schedules --

   Schedules for the remaining embedded diagnostic element software milestones.

Production Readiness Review (PRR)

This review is intended to determine the status of completion of the specific actions which must be satisfactorily accomplished prior to executing a production go-ahead decision. The review is accomplished in an incremental fashion during the Full-Scale Development Phase--usually two initial reviews and one final review, to assess the risk in exercising the production go-ahead decision. In its earlier stages, the PRR concerns itself with gross-level manufacturing concerns, such as the need for identifying high-risk/low-yield manufacturing processes or materials or the requirement for manufacturing development effort to satisfy design requirements. The reviews become more refined as the design matures, dealing with such concerns as production planning, facilities allocation, incorporation of producibility-oriented changes, identification and fabrication of tools/test equipment, long-lead item acquisition, etc. Timing of the incremental PRRs is a function of program posture and is not specifically locked into other reviews. The diagnostic consideration concerns the use of any of the external diagnostic elements (e.g., ATE) in the production testing environment.

Functional Configuration Audit (FCA)

This is a formal audit to validate that the development of a Configuration Item has been completed satisfactorily and that the Configuration Item has achieved the performance and functional characteristics specified in the functional or allocated

configuration identification. In addition, the completed operation and support documents shall be reviewed.

The FCA is normally conducted on a prototype or preproduction item. The FCA validates that the item meets its specified performance requirements and is ready for production and acceptance into Air Force inventory. It is imperative that the diagnostic capability be validated against its specified performance requirements, so that any diagnostic capability deficiencies can be identified and corrected before the item proceeds into production and is then deployed.

Physical Configuration Audit (PCA)

This is a technical examination of a designated Configuration Item to verify that the Configuration Item "as built" conforms to the technical documentation which defines the Configuration Item.

After successful completion of the audit, all subsequent changes to the diagnostic elements are processed by an engineering change action. The PCA also determines that the diagnostic element acceptance testing prescribed by the documentation is adequate for acceptance of the production units by quality assurance activities. The procedures for conducting a PCA are contained in MIL-STD-1521, Appendix H. Sample PCA Certification Attachment Checklists are contained in MIL-STD-1521, Appendix I.

CHECKLIST

- ☑ Are the designers included in the reviews and audits so they can challenge the design and assess risks?

- ☑ Are the diagnostic reviews held as an integral part of the prime system review, but in a timely manner that allows change (if necessary) in the diagnostic equipment or process?

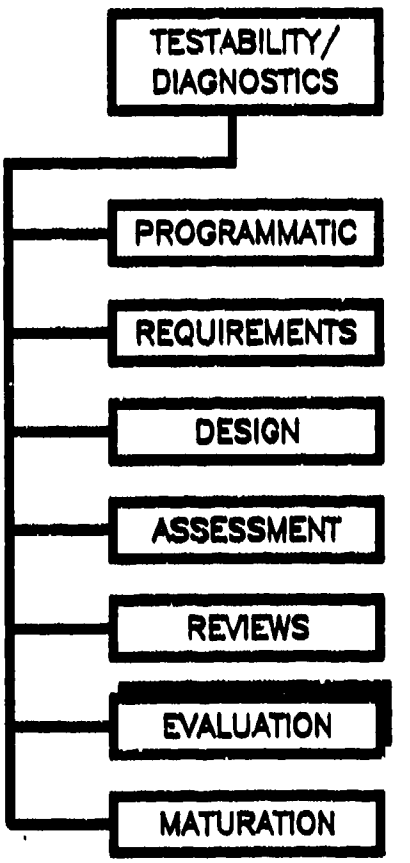
EVALUATION	REQUIREMENT #6

CONDUCTING TEST AND EVALUATION

OVERVIEW

During the development of a weapon system, a number of tests and evaluations are conducted by subcontractors, the prime contractor, and the government. Many of these tests address the performance of the diagnostic capability. It is not uncommon that these tests are conducted separately and, thus, do not address the entire diagnostic capability. Oftentimes the entire diagnostic capability is not delivered in time to test and evaluate the diagnostic capability as a whole. During the major tests and evaluations (e.g., DT&E, OT&E) as much as possible of the entire diagnostic capability should be included. Integrated demonstration, test, and evaluation is required.

Coordination of all test and evaluations, including demonstrations, can be accomplished through the preparation of an Integrated Test Plan.



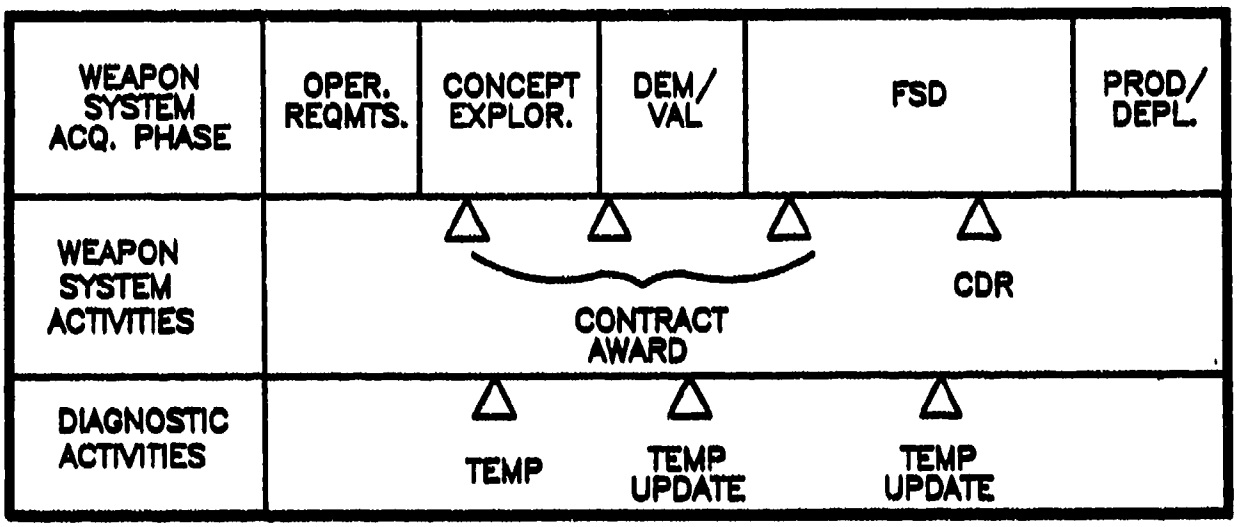
IMPORTANT CONSIDERATIONS BE ADDRESSED

Reqmt.

6.1 Provide input to the preparation of an Integrated Test Plan, which includes the requirements for a Test and Evaluation Master Plan.

6.2 Assure that formal test and evaluations address the entire diagnostic capability.

PREPARATION OF THE TEMP	REQUIREMENT #6.1



DIAGNOSTIC ACTIVITY

The requirements for diagnostics test and evaluation are identified, scheduled and integrated into the Test and Evaluation Master Plan (TEMP).

PROCEDURE

The TEMP is a living document normally prepared by the Contractor Program Manager. Its preparation goes through many iterations as the program proceeds through Concept Exploration Phase studies, Demonstration and Validation, Full-Scale Development, and Production. With each iteration, plans for diagnostic Test and Evaluation (T&E) should become firmer, better defined, and with target milestone dates attached.

Because test and evaluation is a major cost and schedule driver, adequate planning is essential long before it starts. Test planning between subcontractors, the prime contractor, and the government should start with program initiation. To ensure a successful integrated test program, close coordination is required between the government, the prime contractor, and all subcontractors. The designer should understand the scope and methods to be used in evaluating the product, and provide inputs to the TEMP, which promote realism in these tests.

PREPARATION OF THE TEMP	REQUIREMENT #6.1

GUIDANCE

DoD Directive 5000.3 requires the preparation of a Test and Evaluation Master Plan (TEMP). The TEMP is a broad plan relating test objectives to required system characteristics and critical issues, and is a top-level document used at major milestone reviews to assess the adequacy of planned test and evaluation. At minimum, it addresses both Development and Operational Test and Evaluation. It is important that as much as possible of the diagnostic capability be included in these T&Es.

Developmental Test and Evaluation (DT&E) is the T&E conducted throughout various phases of the acquisition process. This will ensure the acquisition and fielding of an effective and supportable system by assisting in the engineering design and development and verifying attainment of technical performance specifications, objectives and supportability.

Developmental Test and Evaluation also includes T&E of components, subsystems, preplanned product improvement (P^3I) changes, hardware-software integration and related software, as well as qualification and production acceptance testing. Test and evaluation of compatibility and interoperability with existing or planned equipment or systems is emphasized. This T&E encompasses the use of models, simulations and testbeds, as well as prototypes of Full-Scale Development models of the system. The diagnostic capability associated with component, assembly and subsystem DT&E should be included in these T&E activities.

Qualification Testing is the part of DT&E which verifies the design and the manufacturing process and provides a baseline for subsequent acceptance tests. This accomplishes two separate functions:

(1) Preproduction Qualification Tests are formal contractual tests that ensure design integrity over the specified operational and environmental range. These tests usually use prototype or preproduction hardware fabricated to the proposed production design specifications and drawings. Such tests include contractual reliability and maintainability demonstration tests required prior to production release. At a minimum, embedded diagnostics capabilities and the interfaces to external diagnostic elements should be tested and evaluated during preproduction qualification tests. As a goal, the capability of external diagnostic elements should also be tested and evaluated.

(2) Production Qualification Tests ensure the effectiveness of the manufacturing process, equipment and procedures. These tests are conducted on a sample lot taken at random from the first production lot, and are repeated as the process or design is changed significantly, and when a second or alternate source is brought on line. These tests are also conducted against contractual requirements. The utilization of diagnostic resources in the manufacturing process and the requirement for capture of diagnostic data from the manufacturing process should be evaluated during production qualification testing.

PREPARATION OF THE TEMP	REQUIREMENT #6.1

The completion of Preproduction Qualification Test and Evaluation before Milestone III decisions is essential and will be a critical factor in assessing the system's readiness for production.

Operational Test and Evaluation (OT&E) is the field test, under realistic conditions, of any item (or key component) of weapons, equipment or munitions for the purpose of determining the effectiveness and suitability for use in combat by typical military users; and the evaluation of the results of such tests. Operational testing is accomplished in an environment as operationally realistic as possible. The entire diagnostic capability should be assessed during OT&E as well as the integration of the diagnostic capability.

The TEMP must clearly specify development and operational test events. However, DT&E and OT&E are not necessarily serial phases in the evolution of a weapon system. During critical acquisition cycle transitions, elements of DT&E and OT&E may be combined or occur in parallel, but not at the expense of either development or operational test realism nor before sufficient DT&E can reasonably assure that the system is ready to enter dedicated operational testing. DT&E may continue into the Production and Deployment Phase, along with OT&E, to address system enhancements, correction of deficiencies, or modifications. In all cases, test planning for all test phases must be addressed in the system TEMP.

Test and evaluation planning is initiated at the inception of the development process to ensure adequate planning, programming and budgeting of test resources and to facilitate test scheduling to support major program decision milestones. Reliability assurance should be well underway before the initiation of system performance tests. System deficiencies must be addressed through a dynamic, well-documented, and tightly managed test-analyze-fix and retest program. The evaluation of embedded diagnostic elements should be injected into these reliability assurance tests.

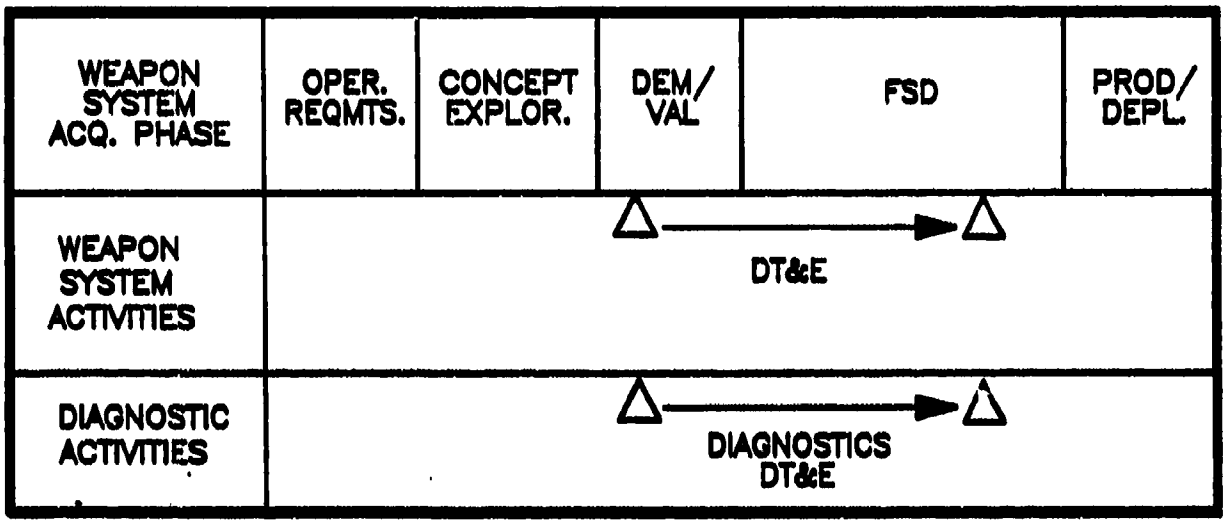
A TEMP is required for all major defense acquisition programs. The TEMP defines and integrates test objectives, critical issues, systems characteristics, responsibilities, resources and schedules for test and evaluation. Test resource requirements must be addressed in the TEMP, along with a critical analysis of any shortfalls that will impede the full test and evaluation of the system. The need for and the availability of the various diagnostic elements which make up the diagnostic capability is addressed in the TEMP. Plans to correct existing or anticipated test resource limitations are also included, as is a listing of evaluation criteria delineating critical parameters permitting continuous oversight and independent assessment.

DoD 5000.3-M-1 contains the guidelines for the preparation of the TEMP.

CHECKLIST

☑ Have T&E activities been realistically planned and scheduled to provide needed information on the performance of the entire diagnostic capability?

DEVELOPMENT TEST & EVALUATION	REQUIREMENT #6.2



DIAGNOSTIC ACTIVITY

Evaluate diagnostics performance characteristics during Development Test and Evaluation (DT&E) activities in order to determine diagnostic capabilities achieved and to identify deficiencies in the diagnostic capability. Diagnostics DT&E should also attend to the capability achieved by the integration of the various planned diagnostic elements (performance monitors, BIT/SIT, testing (automatic and manual), maintenance aids, technical information and training (skills)) into a comprehensive, cohesive, diagnostics subsystem.

PROCEDURE

Development Test and Evaluation is the T&E conducted throughout various phases of the acquisition process to ensure the acquisition and fielding of an effective and supportable system by assisting in the engineering design and development and verifying attainment of technical performance specifications, objectives and supportability.

Development Test and Evaluation also includes T&E of components, subsystems and preplanned product improvement (P^3I) changes, hardware-software integration and related software, as well as qualification and production acceptance testing. Test and evaluation of compatibility and interoperability with existing or planned equipment and systems is emphasized. Development Test and Evaluation encompasses the use of models, simulations, and testbeds, as well as prototypes or Full-Scale Development models of the system.

DEVELOPMENT TEST AND EVALUATION	REQUIREMENT #6.2

The designer should be as actively involved in diagnostics DT&E to ensure that valid tests are devised and performed, valid results documented, and valid data accumulated and to ensure that a closed-loop analytic approach is used to pinpoint and correct diagnostic deficiencies. The designer should ensure that every opportunity is being taken to evaluate diagnostics-related parameters. This may involve a wide range of test activities, including reliability tests, performance tests, human factor tests, etc. Basically whenever a system, subsystem or component is being operated, it is subject to a failure. The diagnostics requirements associated with dealing with the failure should be viewed as an opportunity to assess the diagnostic capability.

GUIDANCE

The thrust of the Integrated Diagnostics Process with respect to DT&E is to include/inject diagnostic performance evaluation into the mainstream of DT&E activities. This is done such that diagnostic performance can be evaluated, deficiencies pinpointed, and corrective action implemented while the system is still in development.

The diagnostics DT&E effort assists the diagnostic design and development process, and verifies attainment of diagnostic technical performance specifications, requirements, and objectives. As such, it is an integral part of the weapon system design process. Through the provision of diagnostics DT&E data, there is a feedback reiterative loop back into the integrated diagnostics activities in process, including the diagnostic system engineering analysis; diagnostic risk analysis; allocation of diagnostic goals; diagnostic trades for system optimization; diagnostic design trades; and, the identification and performance of diagnostic design tasks. Through this methodology, the diagnostic design is corrected, improved, or updated, and the diagnostic design matures.

Sufficient diagnostics DT&E must be accomplished before the Milestone III decision to proceed to production. This will ensure that the major specified diagnostics design and development requirements for the Full-Scale Development Phase have been met, with respect to performance requirements and specifications contained in program documents.

The scope of diagnostics T&E should include fault detection, isolation accuracy and timeliness provided by performance monitoring, BIT/SIT, automatic and manual testing, technical information and maintenance aids, maintenance procedures, manpower, personnel and skill levels at the system, subsystem, LRU/LRM, SRU levels across planned maintenance echelons (Organizational, Intermediate and Depot).

Any deviation from this full scope of T&E means that full confidence cannot be ascribed to the planned diagnostic capability.

The major approaches of DT&E for diagnostics include actions :

- To proceed in phase with the system and support equipment development, so that Built-In Test (BIT) is tested and evaluated concurrently with system performance; BIT and System Integration Test (SIT) tested and evaluated concurrently with subsystem integration and system testing; and, system integration and safety testing are concurrent with diagnostic testing of BIT and SIT features.

- To implement with the Diagnostics Maturation Program so that deficiencies, ambiguities, and additional failure modes identified during DT&E are recorded in a timely manner to ensure traceability and appropriate corrections are made to the integrated diagnostic procedures.

- To evaluate embedded diagnostic design as a separate entity in order to assure that it has been incorporated adequately as part of the system design.

- To evaluate for 100% diagnostic capability in selected critical areas of system design using fault evaluation.

- To analyze the system design hierarchy of test tolerances (e.g., between system BIT and LRU and SRU-level BIT) in order to minimize false alarms.

- To complete feasibility DT&E on prototype and preproduction units in order to assess technical risks and develop solutions to remedy deficiencies.

During FSD, specific diagnostic capability segments of DT&E efforts include the following requirements:

- When available, ATE shall be evaluated for initial use supporting build and check-out of systems. Manual procedures and associated operational prototypes shall be developed for support of test activities.

- Engineering evaluation of the diagnostic elements capability at subsystem and system levels shall be conducted in concert with system integration testing activities, including evaluation tests in the engineering laboratory and system integration test facilities.

- Effective development of a diagnostic capability requires that testing of diagnostic capabilities proceed concurrently with prime and support equipment development in an orderly and planned time-phased manner. The object of the following diagnostics testing approach is to provide a viable Organizational- and Intermediate-level diagnostic capability for use in support of flight and operational testing activities to provide for early maturation of the diagnostic capability. It should also be a program objective

to validate the diagnostic capability, as well as initial reliability and maintainability requirements before production.

- During early equipment development tests, built-in test features should be tested and evaluated concurrently with equipment performance testing. BIT performance is just as essential to overall weapon system performance as the usually emphasized aspects of equipment performance. Simulated equipment failures should be used to assist in BIT testing and evaluation.

- As equipment progresses to subsystem integration and performance testing, BIT and System Integrated Test (SIT) features should be concurrently tested, evaluated, and corrected. Simulated or emulated equipment failures should again be used for BIT/SIT testing and evaluation.

- System integration and safe-for-flight testing of equipment should include diagnostic testing of BIT and SIT features to assure readiness of this capability for Flight Test Support. Concurrently, Organizational-level support equipment required for diagnostic support should be tested to enable its use in the test program, together with Preliminary Maintenance Manuals for Initial Operational Test and Evaluation. Simulatior of equipment failures to evaluate diagnostic capabilities should be included in this testing effort.

- Qualification testing of both prime and support equipment shall include validation of diagnostic capability, which is a required aspect of both equipment and system performance. Simulated equipment failures should be included in the diagnostic validation test program. Evaluation of BIT/SIT should also be conducted during environmental extreme testing of the prime equipment and support equipment, to assure its proper functioning throughout the required equipment performance envelope.

CHECKLIST

- ☑ Does the Integrated Test Plan provide adequate detail concerning specific T&E procedures, data bases, models, test articles and scope of testing?
- ☑ Have critical or high risk items related to diagnostic capability been identified and highlighted?
- ☑ Are the necessary test articles available to conduct realistic, timely tests?
- ☑ Has every opportunity to evaluate diagnostics during DT&E activities been identified?

OPERATIONAL TEST AND EVALUATION	REQUIREMENT #6.3

WEAPON SYSTEM ACQ. PHASE	OPER. REQMTS.	CONCEPT EXPLOR.	DEM/ VAL	FSD	PROD/ DEPL.
WEAPON SYSTEM ACTIVITIES				△ IOT&E	△ FOT&E
DIAGNOSTIC ACTIVITIES				△ DIAGNOSTICS OT&E	△

DIAGNOSTIC ACTIVITY

Diagnostic performance characteristics must be evaluated in a realistic operational environment during Operational Test and Evaluation (OT&E) activities in order to determine diagnostic capabilities achieved and to identify deficiencies in the diagnostic capability. Diagnostics OT&E should focus on the capability achieved by the integration of the various planned diagnostic elements into a comprehensive, cohesive diagnostics subsystem.

PROCEDURE

Operational Test and Evaluation (OT&E) is the field test, under realistic conditions, for the purpose of determining the effectiveness and suitability of the system or equipment for use in combat by typical military users; and the evaluation of the results of such tests.

GUIDANCE

Operational Test and Evaluation (OT&E) activities include Initial OT&E (IOT&E) and Follow-on OT&E (FOT&E). The results of DT&E activities should be analyzed by the Contractor Program Manager with help from the designer to ensure consistency and continuity of T&E activities. Operational Test and Evaluation (OT&E) must be accomplished by a separate government facility prior to the Milestone III decision. Diagnostics OT&E is performed to provide a valid estimate of the operational effectiveness and suitability of the system's integrated diagnostics design and procedures using test items sufficiently representative of the expected production items.

Major approaches to diagnostics OT&E include:

- o Testing in an environment as operationally realistic as possible
- o OT&E initiated as early as possible during the FSD Phase
- o Testing for adherence to overall OT&E objectives, with respect to diagnostics
- o Continued coordination with the Diagnostics Maturation Program
- o Evaluation for 100% diagnostic capability in selected critical areas
- o Random diagnostics testing in noncritical areas
- o Further analysis of test tolerances related to the system hierarchy and embedded/external diagnostic procedures in order to minimize false alarms.

Testing (particularly operational tests) and data collection should focus on the diagnostics requirements. Testing and data collection should also evaluate the specified parameters; namely, identification of critical failures, the false alarm rate, the percentage of faults detected and isolated automatically or manually, associated repair times, the unnecessary removal rate, consistency of test results, and the adequacy of personnel skills considering all maintenance incidents.

During OT&E, system performance, operational suitability and supportability factors are evaluated in an operationally realistic environment. There are two types of information that can be obtained for Diagnostics T&E: 1) faults within the system and how those faults were identified (diagnosed); and, 2) faults/deficiencies within the diagnostic capability. For the latter, this includes evaluation of each element which contributes to the total diagnostic capability, as well as the capability, achieved by integration of the diagnostics elements. Focused, detailed T&E activities discussed in Requirement # 6.2 should be continued. The former type of data can be obtained as a result of Reliability Growth Testing. The following specific information should be evaluated for each fault occurrence.

1. How did the failure manifest itself?
2. Was the manifestation due to stressing of the system beyond normal operational limits?
3. If a BIT alarm occurred, was it the result of a confirmed failure?
4. What techniques were used to isolate the fault?

5. How long did fault isolation take using those techniques?

6. Was the failure mission or operation critical?

7. Was it a new or unplanned failure mode? Was BIT supposed to detect the failure? Did it?

8. Is this failure mode expected to be encountered in the operational system?

9. Should provisions be included in the diagnostic capability to deal with this failure mode?

10. Will this involve a modification/addition to BIT? ATE? Manual Test Equipment? Maintenance Procedures? Skill Levels? Technical Data? Maintenance Aids?

11. Is an ECP required?

12. Is further investigation required?
    If yes - What plans have been made?
    If no - Why not? (brief description)

13. Is correction of the diagnostic deficiency part of contractual requirements? Tied to incentive or warranty provisions?

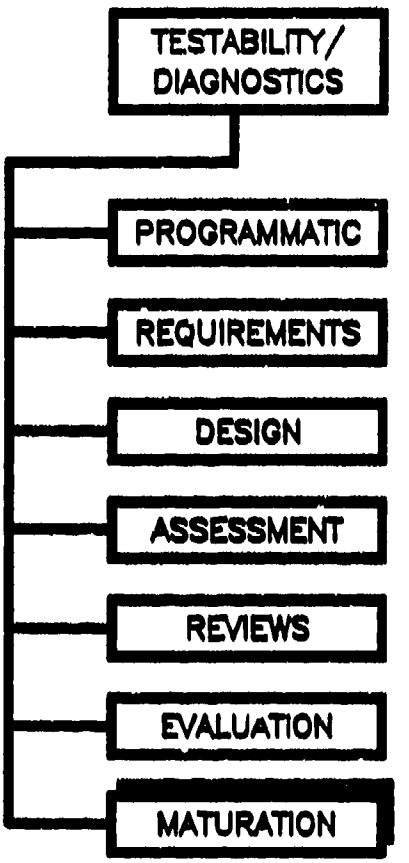
CHECKLIST

☑ Is the designer giving adequate support to OT&E activities?

MATURATION OF THE DIAGNOSTIC CAPABILITY

OVERVIEW

Historically, often a weapon system's diagnostic capability does not meet its performance requirements prior to deployment. The basic reason for this is that all faults cannot be predicted and, thus, adjustment of the diagnostic capability is required during the first few years after deployment. Escentially, this requires a well-planned maturation period, which allows for the growth of the diagnostic capability. Closely coupled with this maturation is the requirement for collection and analysis of data relating to the performance of this diagnostic capability, both in the field and in the factory. Care must be exercised by both the government and the contractor to assure that proper and detailed data is collected. Early planning for this maturation period is a must.



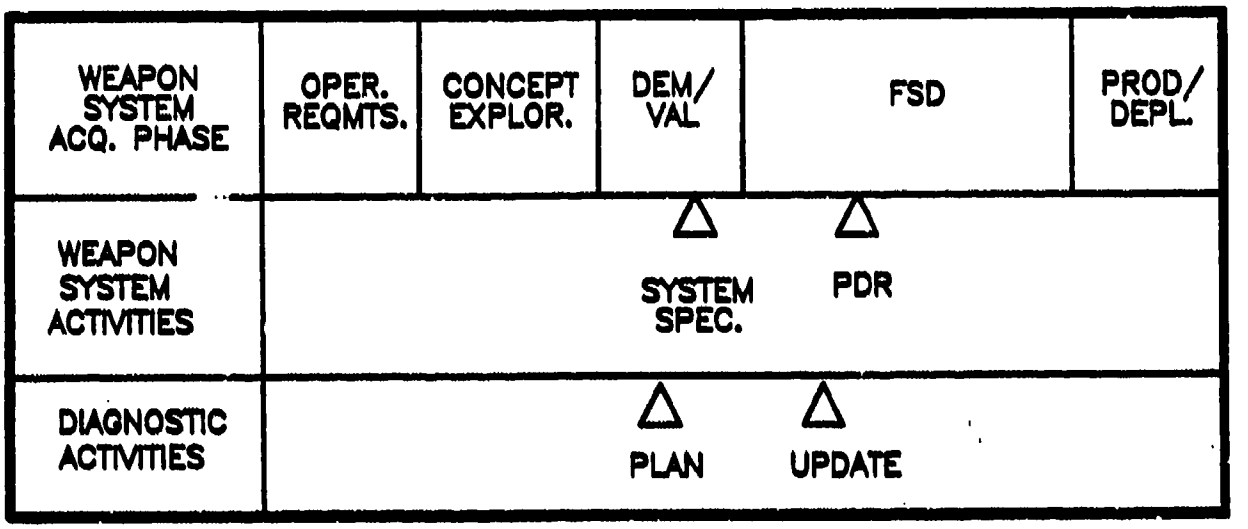
IMPORTANT CONSIDERATIONS TO BE ADDRESSED

Reqmt.

7.1 **A detailed Diagnostics Maturation Plan needs to be prepared early in the acquisition process.**

7.2 **A diagnostic data collection and analysis system must be established to provide for corrective measures.**

MATURATION PLANNING	REQUIREMENT #7.1



DIAGNOSTIC ACTIVITY

Most diagnostic implementations, no matter how well conceived, require a period of time for identification of problems and corrective action to reach specified performance levels. This requirement is established in order to formalize the diagnostics maturation and to allow the maturation to be initiated early in the test and evaluation process. This requirement is initiated early so that early identification, tracking, and correction of diagnostic problems are achieved. The planning for this activity is formalized by the development of a Diagnostic Maturation Plan or other appropriate document.

PROCEDURE

While it is the Contractor Program Managers responsibility to prepare the Diagnostic Maturation Plan, the designer should understand the scope and methods to be used in maturing the diagnostic capability to assure adequate corrective actions are planned and implemented.

The Contractor Program Manager must ensure that the plan is:

1. Comprehensive

   - o Across all diagnostic elements
   - o Includes the integration of the elements

2. Timely

- o Is initiated early to plan for the required resources and implement corrective actions
- o Maturation is completed by Milestone IV, per DoD-Instruction-5000.2

3. Coordinated

- o Includes coordinated activities from the "ilities"
- o Utilizes standard data collection systems

4. Cost Effective

- o Allows data collection to be transferable and usable by government (i.e., DT&E and production test data).

GUIDANCE

A program to mature the diagnostic capability should be planned for the early fielded production systems. A one-to-three-year maturation program should be planned for complex weapon systems with extensive automatic testing capability. For major weapon systems, the coordination with Milestone IV, Logistic Readiness and Support Review (DoD-Instruction-5000.2) is essential. This program should include provisions for on-site collection of diagnostic performance data, with engineering follow-up to provide corrective actions.

The plan should define an approach and methodology to assure that as development, test and evaluation, and early operational use of the system progress, problems presented by new failure modes, test voids, ambiguities, and test tolerance difficulties are recognized and defined, and their solutions are traceable to diagnostic software and manual procedure updates. The plan should recognize that such occurrences are expected and normal and, therefore, should concentrate on problem recognition, definition, and correction, with appropriate tracking for traceability.

The approach and methodology defined should recognize that a basic element of the integrated diagnostics concept is identification of the set of faults which are known or expected to occur. The methodology shall provide for definition of this set, initially through Failure Modes and Effects Analysis, Testability Analysis, and other tools and experience. Provision for growth of this set, as new failure modes are encountered during testing and deployment, should be incorporated in the plan, together with explicit criteria to be used in deciding whether or not a newly encountered fault shall be added to the set of faults for which explicit diagnostic procedures (as opposed to more general procedures) are provided for detection and isolation of the fault. The life cycle cost effectiveness of adding explicit diagnostic procedures for the newly encountered fault shall be one factor considered in the decision.

MATURATION PLANNING	REQUIREMENT #7.1

The plan should provide for an orderly development and maturation process for diagnostic software and manual procedures throughout the development, test and evaluation, and early operational use time periods of the weapon system and its subsystems. Methodology to assure timely and continuing technical support for this maturation process by both contractor and government activities, with a minimum of administrative delays, should be a feature of the plan. Orderly transition of technical responsibilities from the contractor to the government should also be addressed.

The plan should present milestones to be met. This will assure that the final system achieves the required degree of diagnostic capability. The plan shall show the time phasing of each task and its interrelationship with other tasks. It should identify required data review, verification, and utilization to accomplish the required tasks and to report progress, problems, and tradeoffs. The plan should assure the proper implementation of diagnostic design features by designers and subcontractors.

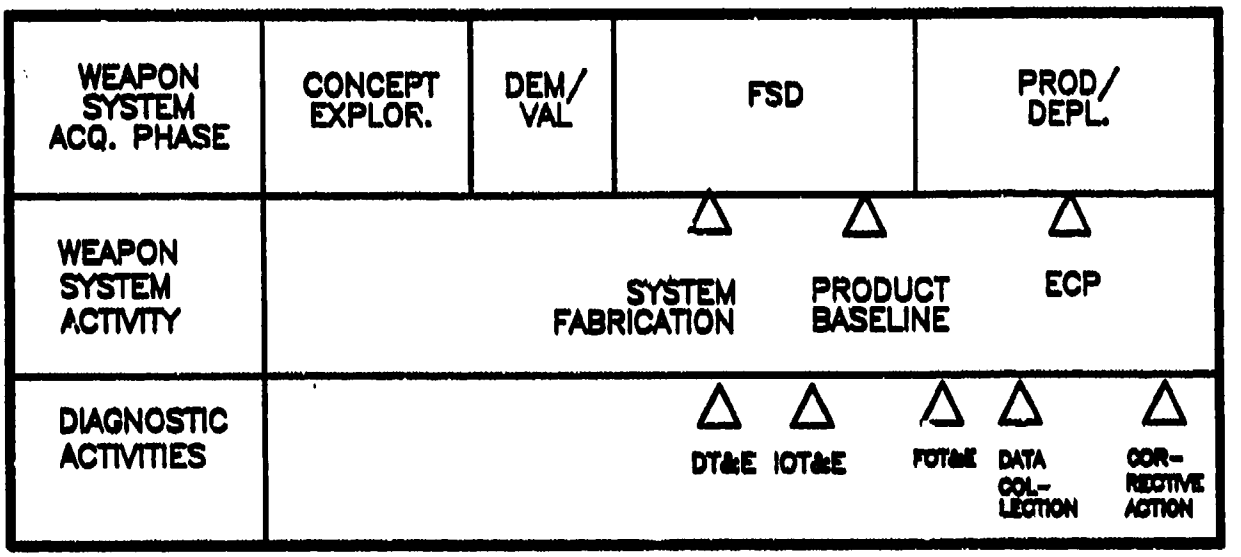
During the Dem/Val Phase, maturation planning is centered on preliminary planning for data collection, analysis and coordination with similar requirements for reliability, maintainability, logistics, data collection, analysis systems, etc. Specifically, this planning should identify potential data sources, such as:

- o Laboratory testing
- o Developmental testing
- o Operational test and evaluation
- o Acceptance testing
- o Preproduction testing
- o Production testing
- o Operation.

CHECKLIST

- ☑ Does the Diagnostics Maturation Plan include a strategy for the collection of diagnostic performance data through DT&E, OT&E, Production, Initial Operational Use, and Deployment?
- ☑ Is the diagnostic data collection plan in sufficient depth to allow adequate evaluation of diagnsotic capability?
- ☑ Does the plan include provisions for all diagnostic elements -- embedded and external -- as well as the integration of the diagnostic elements?
- ☑ Is the integration of the diagnostic elements planned for early enough to allow evaluation and cost-effective corrective action (e.g., prior to production go-ahead)?
- ☑ Does Maturation Planning include provisions for both:
  1. Adequacy of the diagnostic elements, with respect to the specified allocated capability, and
  2. Unplanned failure modes, which may arise throughout OT&E, DT&E, Production Test, and Field Use Test?

DATA COLLECTION AND FEEDBACK	REQUIREMENT #7.2



DIAGNOSTIC ACTIVITY

Data relating to the performance and effectiveness of the diagnostic capability must be collected during development, production, and operation. This data is used as the basis for the evaluation of diagnostics and for the correction of deficiencies.

The key thrust of this activity is definition of appropriate data to be collected, maximum use of data collected, coordination of data collection systems, and a structured approach to corrective action.

PROCEDURE

The Contractor Program Manager is responsible for the implementation of diagnostic data collection and feedback requirements. This includes development and implementation of a cradle-to-grave system for both contractor and government use. It is the designer's job to make sure these design corrections are implemented.

The earlier diagnostic performance deficiencies are identified, the sooner a more cost-effective solution can be implemented. Therefore, diagnostic data collection and feedback is initiated early in the test and evaluation process, continues through production test, and extends into the operational environment. Throughout these phases, different types of data are collected, different data collection procedures and methodologies are used, and different types of analysis technique are conducted.

GUIDANCE

There are no standard methods for data collection and analysis. As indicated under Requirement #7.1, Maturation Planning, the collection of this type of data is controlled by a number of military standards. The requirements for the standards which deal with logistics, reliability, maintainability, testability, human engineering, and safety overlap one another (many times data required by one may, indeed, be required by the other(s)). Thus close coordination among these various data requirements is needed. A single data base is desirable. Some tools are available to assist in the feedback and analysis task. These descriptions are contained in Appendix C.

The data collection procedures closely follow the test and evaluation functions. As explained in DoD Directive 5000.3, Test and Evaluation, the time periods and sequences for Development Test and Evaluation and Operational Test and Evaluation vary from program-to-program. They can overlap and even be done as a combined test and evaluation. Thus there are no standard guidelines that specify the exact points in the weapon system acquisition phase where data is to be collected. The system must be flexible to incorporate data as data is generated.

The Contractor Program Manager should ensure that the proper data is collected and that corrective actions are pursued. It is the job of the designer to make these corrections. Care must be taken to collect only that data required to assure that the diagnostic capabilities are performing as required. Automated data collection systems can be employed. Usually these are more effective, as they are less dependent on human motivation to supply the required information.

Corrective analysis and actions should be in a closed-loop system, so that each deficiency identified remains an open item until it is formally documented as being corrected.

The data collection and feedback system should be designed so that specific information is collected on the performance of the entire diagnostic capability, as well as for each of the diagnostic elements that make up the diagnostic capability. The information must be collected in quantitative form, if possible, and related to System Specification requirements. Thus the following guidelines on the type of data to be collected need to be tailored so that the information can be related to System Specification requirements and so that it is clearly apparent who is to supply the information and when this information is to be supplied. Examples of the type of data to be collected follow.

Diagnostic Data Feedback

- Effectiveness and efficiency of each diagnostic element
- Effectiveness and efficiency of the diagnostic elements as an integrated system
- Operational/support impact of the diagnostic deficiencies
- Corrective action(s) which should be taken or have been taken.

BIT Effectiveness

- Fault isolation time.

Tracking of False Alarms

- Type of alarm
- Frequency of alarm occurrence
- Cause (if known)
- Potential consequences of ignoring the alarm (crew safety, mission reliability)
- Operational costs of responding to false alarms (aborted missions, degraded mode operation, system down time)
- Support costs associated with the false alarm
- Operational environment when alarm occurred.

ATE Effectiveness Feedback

- Workarounds required to overcome mechanical or electrical deficiencies in the UUT/ATE interface
- Did the ATE system provide failure detection results consistent with those of initial failure detection by BIT?
- Were the ATE test results repeatable?
- Ambiguity size

- Fault isolation time.

Integration of Diagnostic Elements

- Are diagnostic resources provided consistent with the training/skill levels of assigned personnel?
- Effect of false alarms and unnecessary removals on operational availability and maintenance workload
- Shop throughput
- Wrong or inadequate technical information
- Logistic delay time
- BIT reliability
- ATE reliability.

Diagnostic data collection and diagnostic capability performance assessment may lead to the requirement for corrective action. Corrective action may involve redesign of prime equipment, test equipment, interface devices, maintenance documentation, built-in test circuits, diagnostic software, and ATE test programs. All changes must be made under strict configuration control.

The designer must recognize that modifications to the prime system/equipment may dictate modifications to the diagnostic capability as well.

CHECKLIST

- ☑ Is there direct communication back and forth between the person who reports a problem and the person who will be correcting the problem?

- ☑ Are all failures being analyzed to sufficient depth to identify failure causes and perform necessary corrective actions?

Table of Contents

1.0	INTRODUCTION	3
2.0	ESTABLISHMENT/INTERPRETATION OF REQUIREMENTS	4
3.0	STRUCTURING DESIGN CONCEPTS/CONSTRAINTS	7
4.0	MEANINGFUL PREDICTION AND ASSESSMENT METHODOLOGY	11
5.0	DESIGN REVIEWS	12
6.0	DEMONSTRATIONS	13
7.0	MATURATION	14
8.0	SUMMARY	16

1.0 INTRODUCTION AND BACKGROUND

The design engineer is the key to solving one of the most complex problems facing the military services in years. The problem is that today's weapon systems have become so sophisticated that the capability to maintain and repair them is a national priority. No longer can the design engineer be primarily concerned with the equipment meeting the operational requirements. Reliability and maintainability must be given equal consideration. Consideration can no longer be given only to varying environmental conditions under which the fielded product must operate. Equal consideration must now be given to the conditions under which repairs will be performed. Seemly short and simple tasks often become very time consuming when accomplished under extreme temperature conditions or in restrictive clothing such as chemical, biological or radiological attire.

The services, burdened with excessive maintenance problems, increasing demand for skilled manpower and skyrocketing costs, have given industry a clear message. The Air Force, for instance, has implemented a program which states basically that all new equipment will be designed to double reliability and reduce repair time by half. Reliability, maintenance, quality, and productivity in new equipment will be given as much attention as performance, program schedule and cost. The effects of this new program can be seen in recent development of a low-altitude navigational system. Performance at initial testing was with operational requirements. However, due to higher-than-anticipated vibration level, reliability requirements could not be demonstrated. The program was not permitted to continue to the next phase until reliability reached the required growth curves. This created a delay of approximately six months, placing the entire program in trouble.

Currently, diagnostics design is the major unknown in the reliability and maintainability arena. The statistics provided in this guide's introduction demonstrates the magnitude of this fact. This appendix presents additional experiences and the key learning points derived from them.

To introduce these lessons, a brief hypothetical scenario is provided regarding the start of a work day for an Air Force technician assigned to a modern bomber wing. This case is intended to provide some insight into what diagnostics programs may someday achieve.

Arriving at his duty station, the technician enters his code at a computer terminal and is provided a work order for the first task of the day. The work order concerns a malfunction which was detected during a flight completed just prior to his arrival at work. A quick glance at the work order reveals which system failed, what time it

occurred, and the Line Replaceable Unit (LRU) which is to be replaced to correct the problem. After a quick trip to a supply point for a serviceable LRU, with tool box and checklist in hand, he departs for the flight line. The defective LRU is changed within minutes after his arrival at work. A quick operational check, using the checklist and on-board test system, confirms that no other failures have occurred, and the system is declared operational.

Back at the intermediate shop, the flight line portion of the work order is closed out. This is quickly done, with a minimal amount of information input into the computer terminal regarding the work accomplished. The defective LRU is placed on its corresponding automatic test equipment (ATE). Keys within the LRU provided identification information to the computer contained within the ATE. Failure conditions and symptoms recorded on-aircraft at the time of the failure are also transferred to the ATE computer via the computer network. Rapidly the ATE goes through a set a tests specifically tailored to the reported failure conditions and the failed single component is identified. After the failed component is replaced, the LRU is checked again with the ATE to verify serviceability. Following serviceable testing, the LRU is given a quality control inspection and returned to the supply point, where it once again becomes a serviceable asset.

The above scenario (or parts thereof) has been a goal of the military services for many years. Great strides have been made toward achieving this objective, yet even total success in limited areas does not lay immediately at hand.

The reasons that success is fleeting are many. They include budget constraints, a relative lack of importance, political considerations, time, and the complexity of the task--just to mention a few. This appendix presents a few glimpses of activity on recent programs, results obtained, and lessons learned.

The information presented is a composite of experiences derived from B-1B and F-111 Aircraft, as well as the Minuteman, Peacekeeper, and Small ICBM Strategic Missiles. LSA examples from the AMRAAM and 30mm Gun PODS are also included.

2.0 ESTABLISHMENT/INTERPRETATION OF REQUIREMENTS

What is specified in the procurement specification and the contractual Statement of Work is what the government expects to receive. In the area of diagnostics, the government experience on past programs has not been the best. Diagnostic systems have proven to be incomplete, unable to to test to the desired level or simply do not as advertised. The basic foundation upon which any success will be directly dependent is the clear understanding of the actual requirements.

Need Statements and Work Statements

All of the programs surveyed in the preparation of this document seem to have an item in common dealing with their diagnostic requirements. That commonality factor is that the quantitative diagnostic requirements imposed are derived without a great deal of thought and analysis. Typically diagnostic requirements are more what has been judged by someone to be realistic values, rather than a product of studies performed to determine these requirements.

DoD-Instruction-5000.2 and other related documents describe a structured acquisition process beginning with, among other things, the development of a Mission Area Analysis and a Mission-Need Statement. Included in the Mission-Need Statement is a discussion of the Mission and Threat, Alternative Concepts, and Technology Involvement. Subsequently, during the Concept Exploration Phase, studies are conducted to develop a System Concept Paper which more thoroughly defines possible alternatives, and a selected concept. Many items are taken under consideration during this time frame including readiness, maintainability, manpower and training.

It is this process which generally drives the development of the procurement specifications. These functions are primarily the concern of Government Program Manager; however, inputs are sometimes requested from the contractor. Failure to consider testability when providing these inputs may limit chances for successful diagnostics later in the program. Overall diagnostics and testability, in general, should be given more concern at this early stage of development.

Understanding Requirements

Determining the proper diagnostic specifications necessary to meet the mission need is one thing. Describing them in such a way that they will be interpreted properly is another.

The following is one of the diagnostic requirements imposed on the B-1B. "The CITS shall provide an assurance of 95 percent to the aircrew that the system performance is operationally acceptable or that the indicated failure is valid during in-flight performance and ground readiness tests. The CITS shall provide fault isolation to an LRU with a certainty of at least 75 percent in the ground fault isolation mode."

Another requirement stated that "false alarms could not exceed 2 percent". Both seemingly good requirements, but two problems ensued in their accomplishment. First and foremost was the problem in the definition of the percent base. Percentages are

often used in defining requirements. But when so used, it must be stated as a percent of what. False alarms, as a percent of the possible alarms, gives one result. False alarms, as a percent of the number of total alarms indicated, gives another. When written, one must assume that achievement based on definition of the writer would meet mission needs. In reality, when achieved based on legal or implied definition, the results were far from those required by the operational command.

A second but in this case a lessor problem, was a conflict between the requirement. The first requirement above allows a 5 percent false alarm rate (100 minus the 95 percent accuracy). The second allows only 2 percent. Specification ambiguity leads to interpretation which will not necessarily end with the desired result. The design engineer can help eliminate these problems by informing program management when specification ambiguity is first encountered.

Logistic Support Analysis (LSA) Process

The LSA is not a direct function of the design engineer, however the process can influence many of the design requirements. MIL-STD-1388-1A defines basic objectives which are achieved when this standard is applied.

1. Cause supportability requirements to be an integral part of system requirements and design.

2. Define support requirements that are optimally related to the design and each other.

3. Define the required support necessary during the operational phase

These objectives are accomplished by the selective application of scientific and engineering efforts undertaken during the acquisition process, as part of system engineering. It is an iterative process of definition, synthesis, trade-off, test and evaluation. Many cases have been found which indicate that designers who successfully incorporate the results of these studies early will have a maintainable system when deployed in the field.

3.0 STRUCTURING DESIGN CONCEPTS/CONSTRAINTS

Controlling vs Constraining the Contractor

Today's trend in specification and contractor direction is to provide the contractors with the maximum leeway in meeting top-level requirements. The objective is to allow the contractor to define alternatives, select from the alternatives those which can best be accomplished and provide a product which meets all of the "real" requirements.

Existing systems covered by this document were all developed under a more structured specification approach. The previous school of thought was, generally speaking, that the more things which can be controlled by the specifications, the more chance the end product will be as desired. Experience with that approach has led to the more open trend. This is because the tighter approach did not allow the contractor to make maximum use of his many possible alternatives.

The customer is encouraging the design engineer to think in terms outside the realm of present diagnostics technology. Diagnostics technology in general has not developed to the point of satisfying the customer's requirements for maintaining complex weapon systems. An excellent example of the type diagnostics not desired is demonstrated by the built-in fault capability of a terrain-following radar. One of the line replaceable units contained within this system is the computer. Conveniently located in the aircraft nose compartment, this LRU contains the malfunction isolation switch and malfunction indicator on it's front panel. The malfunction indicator has nonvolatile flags set during flight at the occurrence of a malfunction. These were designed to serve as functional aids for maintenance personnel troubleshooting the terrain- following radar system. However, many of the malfunctions indicated are caused by associated subsystems that provide stimulus to the computer. This situation has caused unnecessary maintenance and supply cost, plus degraded operational readiness. The designer must ensure that all diagnostics will only be influenced by the parameter they are designed to monitor. The last thing the customer needs is a diagnostic system which increases the workload.

The Maintenance Concept

The Logistic Support Analysis tasks of MIL-STD-1388-1A which are concerned with the development of maintenance concepts and constraints are very important for the diagnostic community. The design engineer will benefit greatly by being involved with the LSA analyst and incorporating the results into the basic design at the earliest possible time. The MIL-STD-1388 tasks are structured to ensure consideration of existing

resources, compatibility with deployment and operational requirements, and maintenance personnel skill level.

The tie between the diagnostic method and the maintenance concept is bidirectional. They need to be established in unison. The maintenance concept is developed based on expected diagnostic capabilities. The diagnostic design ultimately forces the real maintenance concept. The designer who understands this concept recognizes the MIL-STD-1388 process as an aid to achieving the desired maintenance concept is success oriented. A lesson well learned is that when these tasks are used for historical purposes, instead of a tool for the designer, the desired maintenance concept is seldom achieved.

Established Air Force maintenance policies utilize system operation as the final determinant of the need for maintenance. If the system is functioning within tolerance, don't fix it. A unique situation has developed on the B-1B aircraft. Due to redundancies designed into the systems, overall operation appears normal, while some specific parts are not functioning. The diagnostics systems say the parts should be replaced. System operation indicates everything is functioning normally. To date, partially due to the lack of confidence in the aircraft diagnostics and partially due to established habits, these type malfunctions are not being repaired. Generally the diagnostics is believed faulty and no maintenance action is taken until another item malfunctions rendering the system inoperative. This experience shows that changing existing practices is slow. If it is confused with the lack of confidence in the diagnostics, the change is even more difficult.

The Unit Under Test (UUT) designer, ATE manager, and automatic test equipment designer are all vital elements in determining what off- equipment testing is required. Once the option for automated testing is confirmed, the ATE designers must convince the UUT designer to incorporate "Design To" criteria for maintainability, reliability and testability. Care must be taken to define the need for ATE, how the ATE is to be used, how the UUT will be designed for built-in test and interfacing abilities. ATE effectiveness is directly and immediately dependent upon this co-development with the unit under test. The following support trade-off factors must be considered when developing the UUT "Design To" criteria:

1. The maintenance concept and requirements imposed by the Repair Level Analysis of the system

2. Requirements of the built-in test for the UUT

3. The effectiveness of UUT functional partitioning

4. The ability to insert in the UUT test points

5. Design limits on reliability and maintainability.

Historically, probably the prime reason for dissatisfaction with a weapon system's diagnostic capability is the unnecessary removal of "good" items when conducting corrective maintenance. The designer must be aware of the causes for these unnecessary removals.

A field survey team (details of the survey are contained in RADC-TR-83-2 "Study of the Causes of Unnecessary Removals of Avionic Equipment") visited 12 AFB's in 1979 to determine the causes of this problem. When the survey was completed, a study analyzed the data and categorized the causes. The following major causes of unnecessary removals (URs) are listed along with the percentage of all URs for which they are responsible.

Ineffective BIT - 35%

This problem relates to built-in-test designs which provide incomplete or ambiguous information to aircrew and ground crew operators. Such incomplete information is the reason that operators must "interpret" BIT indications. Thus, there are instances when BIT indications are misinterpreted and an avionic equipment is erroneously reported as malfunctioning. Such "malfunctions" are termed false alarms and result in a CND or UR classification. These false alarms may either indicate a malfunction in a serviceable equipment when there is actually no malfunction in the system, or may not indicate a fault when one exists in the equipment.

Ineffective Supervision/Support - 25%

This problem involves control of the work habits of maintenance technicians. Although a lack of such support may be a result of the current short supply of middle management personnel, special attention of supervision is often necessary to maintain control of the UR rate.

Lack of adequate troubleshooting, incorrect use of test equipment, improper or inadequate documentation, and lack of historical tracking of aircraft and LRUs for intermittent problems all tend to point to the lack of effective direct supervision.

Management Directives - 11%

This problem relates to bypassing the normal standard troubleshooting procedures to obtain quick response turnaround times for priority sorties. There are times when turnaround time is most important and any supporting action is justified. However, this type of nonstandard action should be under regular surveillance by auditing personnel.

Test Equipment Differences - 10%

Test equipment differences between different levels of maintenance were noted by the survey team on relatively "new" equipment. A lack of commonality in the calibration of test equipment was also discerned by the field survey team at one repair facility. At one AFB, certain LRUs received from the repair depot are retested because of the lack of commonality between I-level test equipment and depot level test equipment.

Ineffective or Missing Test Equipment - 9%

This includes heavy or bulky test equipment. In most cases ineffective, heavy or unwieldy test equipment is the same as missing test equipment since it is not used. In this case, nonstandard troubleshooting is employed.

Inadequate Skill - 7%

Inadequate skill of maintenance technicians in the use of T.O.s, test equipment and troubleshooting procedures relates to the technicians' inability to completely cope with the relatively high technology of electronic equipment. This cause of URs is due to the technician not remembering details of his past training; be it formal, on-the-job training, technical readings or just familiarization with equipment and/or available diagnostic methods.

Inaccessibility

In addition to the above, the problem of inaccessibility cannot be overlooked. An inaccessibility problem can have a significant impact on the unnecessary removal rate. When LRUs are not readily accessible due to some restricted location, the removal of a suspect LRU may require the removal of one or more adjacent LRUs. Also, the difficulty in reaching a suspect LRU may preclude an on-equipment check, and the suspect LRU is removed and sent to the I-level shot for bench check.

4.0 MEANINGFUL PREDICTION AND ASSESSMENT METHODOLOGY

In-process assessment of diagnostics achievements has, in the past, been less than adequate. In fact, one of the most definitive and often repeated B-1B lessons is the need for an operational period to mature the diagnostic design. That lesson is described below in paragraph 7.0. Prediction and assessment techniques have, in the past, failed to provide sufficient information to uncover all of the inadequacies and shortcomings. Significant emphasis is currently being placed on testability analysis, reliability, and maintainability assessment tools under the umbrella of Computer-Aided Acquisition and Logistic Support (CALS). With that emphasis, one should expect great improvements in assessment techniques. The point for the design engineer on this subject is that the results of these predictions and assessments must be incorporated into the design so diagnostics requirements will be fulfilled.

Methodology

The CALS initiative would include diagnostics testing as an integral part of CAD design. The concept is that rules and techniques would be established in the CAD machine. As a specific item is designed, it is constantly checked for test access, built-in test capability, or whatever other rules that have been established.

This concept works fine for evaluating the diagnostic characteristics of a single electronic assembly. Evaluation of a weapon system's central test system is another question. For the B-1B, a complete integration lab was developed to test the diagnostics software in a functional environment. That process was useful, but still under the best of lab conditions some things could not be developed to the optimum level. An excellent example is the philosophy for checking the thrust of a jet engine. Simulated lab conditions equate more to an aircraft being on the ground. There thrust is compared to a reference schedule of gross thrust versus turbine blade temperature at two discrete operating points. These two points are the intermediate and maximum power settings. To develop an in-flight thrust check, a reference has to be calculated to monitor performance across the entire power range. This reference is obtained by comparing the engines in synchronization to one another in flight. This reference requirement, plus many preconditions necessary for calculating or examining thrust, dictated actual flight testing for development of a valid check.

Feedback Structure

Time is needed to assure that the design benefits form the assessment process. Logically, one does not need a whole lot of experience to understand this. However, it was proved once again on the B-1B aircraft that compressed schedules tend to eliminate this time. Concurrent Full-Scale Development and Production meant that the funding for studies and analysis occurred so late that results could not be implemented. When this happens, management direction is needed; however, management cannot take any action unless the problem is brought to their attention. The design engineer must notify management or the magnitude of the problem will increase with the passage of time.

Information Flow

A concern often expressed by many design engineers is the delay in receiving formal products generated with the MIL-STD-1388 process. Certainly this delay can create concerns. Who wants to think the design is stable only to discover major changes are required? The driving factors often necessitating these changes are studies performed for maintainability and testability. The design engineer who realizes this and develops a close working relationship with the personnel performing these studies will have fewer surprises. Experience has taught many times that it is much easier to communicate and incorporate changes during the initial design effort. Trying to make changes later is expensive, time consuming and often produces less than optimum results.

5.0 DESIGN REVIEWS

Formal Design Reviews provide the opportunity for the contractor to demonstrate to the customer the present design and what future design efforts hope to achieve. If the contractor can demonstrate that he is meeting the specifications, the customer can ask no more. It is the role of the design engineer to assure that sufficient design has been performed prior to Design Reviews, which can demonstrate with a degree of confidence, that diagnostic requirements are being fulfilled.

Scheduling

It's either too early or too late. Picking the optimum time for reviews is very important. Reviews need to be conducted after the design is sufficiently defined to make the evaluation but before it is too late to make design changes. The design engineer

needs to participate in schedule development to ensure that a reviewable product is available at the scheduled time.

The only identified lesson learned from experience is that the scheduling for formal reviews is typically determined at the beginning of the program. The stage of the design for the review is then whatever it is at the scheduled time. This is not necessarily bad, because typically the designers influence the work schedule toward having a reviewable product on the established schedule. Usually, reviews cannot be moved out without jeopardizing program schedules. Designers must guard against committing to a schedule with goals that are unrealistic.

Review Emphasis

Messages are sometimes sent to designers which can be misinterpreted, informing them where they should place emphasis. This misinterpretation is based on the importance an item is given in the reviews. If the Government Program Manager and his review team place little emphasis on diagnostics, designers get the message that diagnostics are "not important." This has often been done unintentionally in the past by quickly passing over the subject in the reviews, or otherwise indicating a minimal concern. The design engineer must not forget the importance of diagnostics, especially in cases where the Government review team has placed little emphasis on the subject.

6.0 DEMONSTRATIONS

Demonstrations are, in general, another form of a formal review. Thus, most of the points made in the previous section also apply here.

Timeliness

The opportune time for final demonstration of diagnostics does not exist, if a purpose of the demonstration is to identify corrective actions. Efforts to schedule demonstrations early enough to minimize the impact of "failure" have, in the past, resulted in the simulation of too many conditions and resources. To perform a complete diagnostics demonstration, all operational diagnostics tools must be in place. This includes support equipment (if appropriate), training, technical publications, and any other applicable diagnostic tool. Attempts to simulate or work around the absence of these operational items does not provide for a complete demonstration.

Simulated vs. Operational Conditions

This problem can be demonstrated by experience with a recent modification program on the F-111D Attack Radar. The modification was major--mainly made to improve reliability and maintainability. One significant portion of the modification was the re-work of the built-in test (BIT) capabilities.

The design job seemed to be done very well. Design Reviews were passed. Demonstrations of the new BIT performance in the laboratory exceeded the specifications and expectations. All looked like a job well done and the contract was considered complete.

The problem was that on the aircraft, in operational conditions, the BIT does not do so well. The BIT serves two functions, one being to advise the aircrew if the selected mode is operational, the other serving as a diagnostics aid to maintenance personnel. The aircrew function performs well, which is not surprising, being part of the basic operational requirement. However, the diagnostics portion of the software used in the fault isolation process has required extensive re-work. At first glance, one is led to believe that the simulated and operational conditions must differ greatly. This being the case, how does one explain that problems reported during field operations can later be demonstrated under laboratory conditions? Performing demonstrations with the primary objective of showing operational requirements are being fulfilled, with diagnostics given secondary concern, only delays finding problems in that area. An important point to remember is that diagnostics must be given equal consideration to operational requirements and the Demonstration phase is another chance to identify and start correcting diagnostic problems.

Providing for Resources

Scheduling/obtaining resources for the demonstration is an early function. This requirement has often been overlooked or minimized in the past. Design engineers involved in the demonstration process should be fully aware of the demonstration plan/requirements and assure that required assets are inputted for incorporation in top-level planning documents.

7.0 MATURATION

Maturation is a phase which has been identified as necessary primarily during development of new systems/technology for the embedded and external diagnostic capability. One especially critical area for these systems is the inherent requirement for

testing under actual operating conditions. Maturation becomes necessary to refine test method/fault limits/diagnostics logic embedded within the diagnostics software programs that operate these systems. The predicted operating characteristic of the various on-board systems must be compared to the actual operating characteristics of these systems as they interface with other systems under varying environmental conditions.

Early Planning

One thing learned on the B-1B is that the design engineer must keep management informed of the considerable time and resources necessary for maturation. The original B-1B development plan was to mature the diagnostics system (CITS) on 70 FSD flights. That would, it was thought, provide a mature system at the time of the first deployment of the B-1B to an Air Force Main Operating Base. Early in the Full-Scale Development Phase, it became evident that the plan would not be sufficient. A new plan was developed to use 468 SAC sorties over the years 1985 and 1986. The wing did not fly the required number of sorties over that time period and the program was extended through November of 1987. Additional aircraft deliveries and an increase in sortie generation rate produced a total of 1069 sorties by the end of that period. With that number of sorties, sufficient data was gathered to indicate an acceptable level of performance. At this point, it is estimated that as a general rule, at least 400 to 500 sorties will be required to mature an on-board test system like the CITS. Maturation time is difficult to estimate and as learned on the B-1B changes will have to be made as the process matures.

Operational or Flight Test Environment

How does one plan for 500 sorties prior to production? Is a plan to fly four FSD aircraft on the average of once every three calendar days for a year reasonable? Is a limited production block appropriate for maturation? These are questions which the design engineer must consider when advising management of schedules early in program planning.

Experience has identified one additional consideration to be included in making these decisions. That consideration is the impact a partially working diagnostics system has on the maintenance technician. If technicians lose confidence in a diagnostic aid, they will not use. Further, it is hard to convince them that the item has been improved and that now they can have confidence in it. Many maintenance technicians on the B-1B, F-111 and other systems who have been exposed to inaccurate diagnostic methods have never been convinced to use an "improved" version. All B-1B operating bases have the same current version of CITS. Field data shows, however, that the bases exposed to the earliest and poorest version of CITS continue to have the highest false alarm and cannot

duplicate maintenance rates. This is due to the lack of trust still carried from the early experience. Thus, it is important to accomplish maturation away from the majority of operational technicians, if possible.

Implementing Maintenance Concept

If the maintenance concept utilizing the planned diagnostics is significantly different from that with which the established technician is familiar, special training will need to provided. The B-1B conflict between using CITS or system performance to rule that a failure has occurred was discussed in paragraph 3.0 of this appendix. Trends are also in place today to isolate to and replace modules on the aircraft rather than the large "boxes" of the past. Utilizing the diagnostic indication produced during flight without further ground verification is also a current trend. Each of these "new" concepts must be thoroughly understood by the technicians, so that the maturation results are consistent with the planned fielded maintenance concept. Making changes is never an easy process and the maintenance technician is no exception to this concept.

8.0 SUMMARY

Diagnostics is not a simple matter and the perfect situation portrayed in the Introduction has yet to be achieved. Instead of the failure being identified to one LRU, often the ambiguity group is as many as four LRUs. The ATE which can isolate the failure to a single failed component would be the ideal solution but, more likely than not, it will only be one or sometimes several Shop Replaceable Units (SRUs) or a particular group of SRUs. The steps covered here are only some of the very basic ones required to insure good diagnostics. However, looking at many different programs, one finds even these simple steps have been omitted, or perhaps accomplished, at a time too late to have the desired results. The reasons are many: poor communication of needs or goals, time frame restrictions, money, and failure to properly consider the importance of diagnostics. To ensure diagnostics, it must be addressed at all phases and be given equal importance to other performance requirements. If the system cannot be maintained, it can never meet its operational requirements.

CHECKLIST

- ☑ Studies, analyses, and feedback take time. They need to be scheduled so that their results can influence the design.
- ☑ Test equipment designers need to have an input regarding the design requirements of the units to be tested.
- ☑ Proper priorities need to be demonstrated by both government and industry if diagnostics is to be properly implemented.
- ☑ Specifications must be well defined and represent exactly what is needed.
- ☑ Real operating time is required for maturation of the diagnostic system--lots of it.

LIST OF ACRONYMS

ABI	Avionics Bus Interface
ADA	Adaptive Diagnostic Authoring
ADS	Adaptive Diagnostic System
AFLC	Air Force Logistics Command
ADP	Automatic Data Processing
AFSC	Air Force Systems Command
AI	Artificial Intelligence
AIDA	Corporation - Santa Clara, CA
ALU	Arithmetic Logic Unit
AMC	Army Materiel Command
AMRAAM	Advanced Medium Range Air-to-Air Missile
ASIC	Application Specific Integrated Circuit
ASTEP	Advanced System Testability Analysis Program
ATE	Automatic Test Equipment
ATF	Advanced Tactical Fighter
ATG	Automatic Test Generator
ATLAS	Abbreviated Test Language for All Systems
ATPG	Automatic Test Pattern Generator
BCPE	Biphase Correlator Processing Element
BDL	Behavioral Design Language
BILBO	Built-In Logic Block Observation
BIST	Built-In Self Test
BIT	Built-In Test
BITE	Built-In Test Equipment
BLM	Behavioral Logic Model
BMM	Bulk Memory Module
C/ATLAS	Common Abbreviated Test Language for All Systems
CAD	Computer-Aided Design
CADAT	Computer-Aided Design & Test
CADAT 6	Computer-Aided Design & Test, Version 6
CADBIT	Computer-Aided Design for Built-In Test
CAE	Computer-Aided Engineering
CALS	Computer-Aided Acquisition & Logistics Support
CAMELOT	Computer-Aided Measure for Logistic Testability
CASS	Consolidated Automated Support System
CATS	Computer-Aided Test System
CDDB	Common Diagnostic Data Base
CDL	Circuit Description Language
CDR	Critical Design Review

CDRL	Contract Data Requirements List
CEP	Count Enable Parallel
CEPS	CITS Expert Parameter System
CET	Count Enable Trickle
CFE	Contractor Furnished Equipment
CI	Configuration Items
CITS	Central Integrated Test System
CLK	Clock
CLR	Clear
CMC	CITS Maintenance Code
CMOS	Complementary Metal Oxide Semi-Conductor
CML	Current Mode Logic
CMOS	Complimentary Metal Oxide Silicon
CND	Cannot Duplicate
CNO	Chief of Naval Operations
COPTR	Controllability-Observability-Predictability Testability Report
CPCI	Computer Program Configuration Item
CRC	Cyclic Redundancy Check
CSC	Computer System Component
CSCI	Computer Software Configuration Item
CSDM	Computer System Diagnostic Manual
CSI	CADAT Systems Interface
CSOM	Computer Software Operator's Manual
CTE	Commercial Test Equipment
CTF	Controllability Transfer Factor
CY	Controllability
D-Level	Depot Level
DAISY	Manufacturer Name - Mountain View, CA
DATPG	Digital Automatic Test Program Generator
DBDD	Data Base Design Document
DCP	Decision Coordinating Paper
Dem/Val	Demonstration and Validation (Phase)
DFT	Design For Testability
DIA	Defense Intelligence Agency
DID	Data Item Description
DIP	Dual In-line Package
DMUX	Demultiplexer
DoD	Department of Defense
DoD-D	DoD Directive
DoD-INST	DoD Instruction
DNE	Data Network Element
DT&E	Development Test and Evaluation

DTA	Daisy Testability Analyzer
EARS	Engineering Access Routine Set
ECL	Emitter Collector Logic
ECC	Error Correcting Code
ECL	Emitter-Coupled Logic
ECP	Engineering Change Proposal
EDIF	Electronic Design Interchange Format
EIA	Electronics Industry Association
ESU	Element Supervisor Unit
ETE	Electronic Test Equipment
FA	False Alarm
FA	Feedback Analysis
FCA	Functional Configuration Audit
FD	Fault Detection
FEFI	Fraction of Erroneous Fault Isolation Results
FFD	Fraction of Faults Detected
FFI	Fraction of Faults Isolated
FI	Fault Isolation
FIG	Fault Isolation Group
FIPAD	Failure Indentification, Prevention and Detection
FIS	Fault Isolation System
FLEX	Name (Navy Support Cost Model)
FMEA	Failure Modes and Effects Analysis
FMECA	Failure Modes, Effects and Criticality Analysis
FOM	Figure of Merit
FOT&E	Follow-On Operational Test & Evaluation
FPPE	Floating Point Processing Element
FRACAS	Failure Reporting, Analysis and Corrective Action
FSD	Full-Scale Development
FSM	Firmware Support Manual
FYDP	Five Year Defense Plan
GFE	Government Furnished Equipment
GIMADS	Generic Integrated Maintenance Diagnostics
GM	Global Memory
GPETE	General Purpose Electronic Test Equipment
GSE	Ground Support Equipment
HDL	Hardware Description Language
HITAP	Hi-Testability Analysis Program
HITS	Hierarchical Integrated Test Simulator

HSDB	High-Speed Data Bus
HW	Hardware
HWCI	Hardware Configuration Item
I-Level	Intermediate Level
I/O	Input/Output
IC	Integrated Circuit
ICE	Integrated Conceptual Environment
ICNIA	Integrated Communications, Navigation & Identification
ID	Interface Device
ID	Integrated Diagnostics
IDD	Interface Design Document
IDSS	Integrated Diagnostics Support System
IFTE	Intermediate Forward Test Equipment
IGES	Initial Graphics Exchange Specification
ILS	Integrated Logistic Support
ILSP	Integrated Logistic Support Plan
IMIS	Integrated Maintenance Information System
I/O	Input/Output
IOT&E	Initial Operational Test & Evaluation
IPS	Integrated Program Summary
IRST	Infrared Search and Track
ISPS	Instruction Set Processor Specification
ITC	International Test Conference
ITP	Integrated Test Plan
JTAG	Joint Test Action Group
KGM	Key Generator Module
LANA	Local Area Network Acceleration
LCC	Life Cycle Cost
LCCA TM	Life Cycle Cost Analysis
LCC Family of Models	Life Cycle Cost Family of Models
LFSR	Linear Feedback Shift Register
LCCC	Leadless Chip Carrier
LDCC	Leaded Chip Carrier
LED	Light Emitting Diode
LFSR	Linear Feedback Shift Register
LOGMOD	Logic Modeling
LRM	Line Replaceable Module

GUIDANCE

The guidance provided in this section is designed to permit maximum visibility into the diagnostic design process. The designer must understand the design process flow, timing, and data requirements which must be satisfied. In addition, it is important to recognize that current data item procurement practices may not always be supportive of the design activity in-process data needs. At times, the CDRL and associated DID do not adequately reflect these in-process needs. The high data item generation/revision costs generally experienced are strong motivators for delaying data item preparation to a point where the design has stabilized. Such motivation is in direct conflict with the utilization of the data to make design decisions. A complete, detailed data submittal indicating that the design is flawed is of little use after the design has been completed. The guidance that follows has been designed to provide the necessary insight into the design process, which will assist the designer in doing a better job.

Design Environment

The diagnostic design environment is an essential component of the overall diagnostic design activity, which has been established by the contractor in response to the RFP requirements. This environment encompasses both the implementation methodology and the specialty coordination associated with the diagnostic design process. Evidence of these should be apparent in the interim products of the design effort, which are made available to the government program management function (at both informal in-process reviews and formal system-level design reviews).

Diagnostic design is characterized by its iterative nature and a high degree of interdependence with the supportability engineering specialties (i. e., reliability, maintainability, integrated logistic support, testability, human engineering, and safety). The allocation of diagnostic resources must be based on inputs from these disciplines. Therefore, the timing and quality of data interchanges must be in accordance with the program plans. A breakdown in data availability and exchange can be responsible for program delays and shortfalls in the fielded diagnostic capability.

The data flow required to develop the composite diagnostic capability must be responsive to the diagnostic mix established for the specific system under consideration. Embedded diagnostic features, such as built-in test (BIT), built-in test equipment (BITE), system integrated test (SIT), performance monitoring, status monitoring, embedded training, embedded maintenance aiding, adaptive AI-based diagnostic systems, etc., are an integral part of the prime equipment design. Therefore, the diagnostic data flow associated with these features must be incremental and continue until the detail prime system Configuration Item designs are complete. For the external diagnostic elements, such as automatic test equipment and the associated test program sets, manual test equipment, portable maintenance aids, technical information (hard copy or electronic delivery), training, etc., the diagnostic data flows into the LSA process up to the point where the firm requirements for these diagnostic elements can be established. Once firm

requirements exist, the diagnostic design environment must facilitate a smooth transfer of data, which is sufficient in terms of detail and format to permit fabrication of the required external diagnostic capability.

Table 5 is a listing of the major military standards which influence the design of the diagnostic capability. Some of these military standards are programmatic in nature, in that they establish a specific program and describe the tasks which can be undertaken. The remainder of the standards are process or product-oriented. As can be seen, these standards influence various aspects in the design of the diagnostic capability, starting from establishing diagnostic requirements, through the design and assessment of the diagnostic capability. There is a sequence of tasks and procedures cited in these standards which can be applied to the diagnostic capability. The interfaces and relationships between these various activities are complex and cannot be easily diagrammed to promote understanding. Establishing diagnostic requirements is described in Requirement #2, and the assessment is described under Requirement #4. Thus the following guidance will address the design of the embedded and external diagnostic capability.

Design Integration

Figure 4 is a simplified diagram of the information flow in the design of the diagnostic capability. The design process begins with a maintenance concept and design data, such as specifications, block diagrams, and schematics. Establishing the system's architecture is the next step. System's architecture has a major impact on the design of the fielded diagnostic capability. The concept of fault tolerance supports the maintenance concept by promoting graceful degradation of the system's performance, thereby allowing the maintenance to be performed at the user's convenience rather than dictated by when faults occur. Design for testability concepts play an important part at this time. Partitioning especially is closely tied to fault tolerance, because the performance monitoring capability must be able to detect failed items in order that the capability of the system is known, that necessary switching to alternate means is facilitated, and that maintenance actions can be identified.

The Failure Modes and Effects Criticality Analysis (FMECA) utilizes the system's architecture and design data to determine the modes, causes and effects of item failures. This data drives the maintenance and safety requirements which in turn help to establish the diagnostic logic, test point selection, and test requirements. From this information, the diagnostic capability is designed and fabricated, including the testing, (built-in and external), technical information, training, and personnel capability. Obviously this entire process is iterative in nature - a factor not indicated in Figure 4.

TABLE 5. MILITARY STANDARDS APPLICABLE TO THE DESIGN OF THE DIAGNOSTIC CAPABILITY

		REQUIREMENT				DESIGN							ASSESS	
		ESTABLISH	ALLOCATE	OPTIMIZE	RISK ASSESS.	FAULT TOLERANT	INHERENT TSTAB	BIT/SIT	ATE/TPS	MANUAL TEST EQ	TECH. INFO.	PERS. & TRNG.	ANALYSIS	DEMONSTRATION
PROGRAMMATIC	MIL-STD-1388-1 Logistic Support Analysis	X	X	X	X								X	X
	MIL-STD-785 Reliability	X	X	X	X	X		X						
	MIL-STD-470 Maintainability	X	X	X	X			X	X	X	X	X	X	X
	MIL-STD-2165 Testability	X	X	X	X	X	X	X	X	X			X	X
	MIL-STD-882 Safety	X				X		X						
	MIL-STD-2167 Software Development	X	X	X	X	X		X	X				X	X
	MIL-H-46855 Human Engineering	X	X	X							X	X		
PRODUCT/PROCESS	MIL-STD-1591 Analysis		X	X										
	MIL-STD-415 Test Provisions						X	X	X					
	MIL-STD-1519 Preparation of -1345 Test Rqt. Doc.								X	X				
	MIL-STD-1629 Procedures for FMECA					X		X						
	MIL-STD-2077 Requirements for TPS								X					
	MIL-STD-471 Maintainability Demonstration													X
	MIL-STD-756 Reliability Modeling & Pred.		X											
	MIL-STD-1379 Contract Training Prog.											X		X

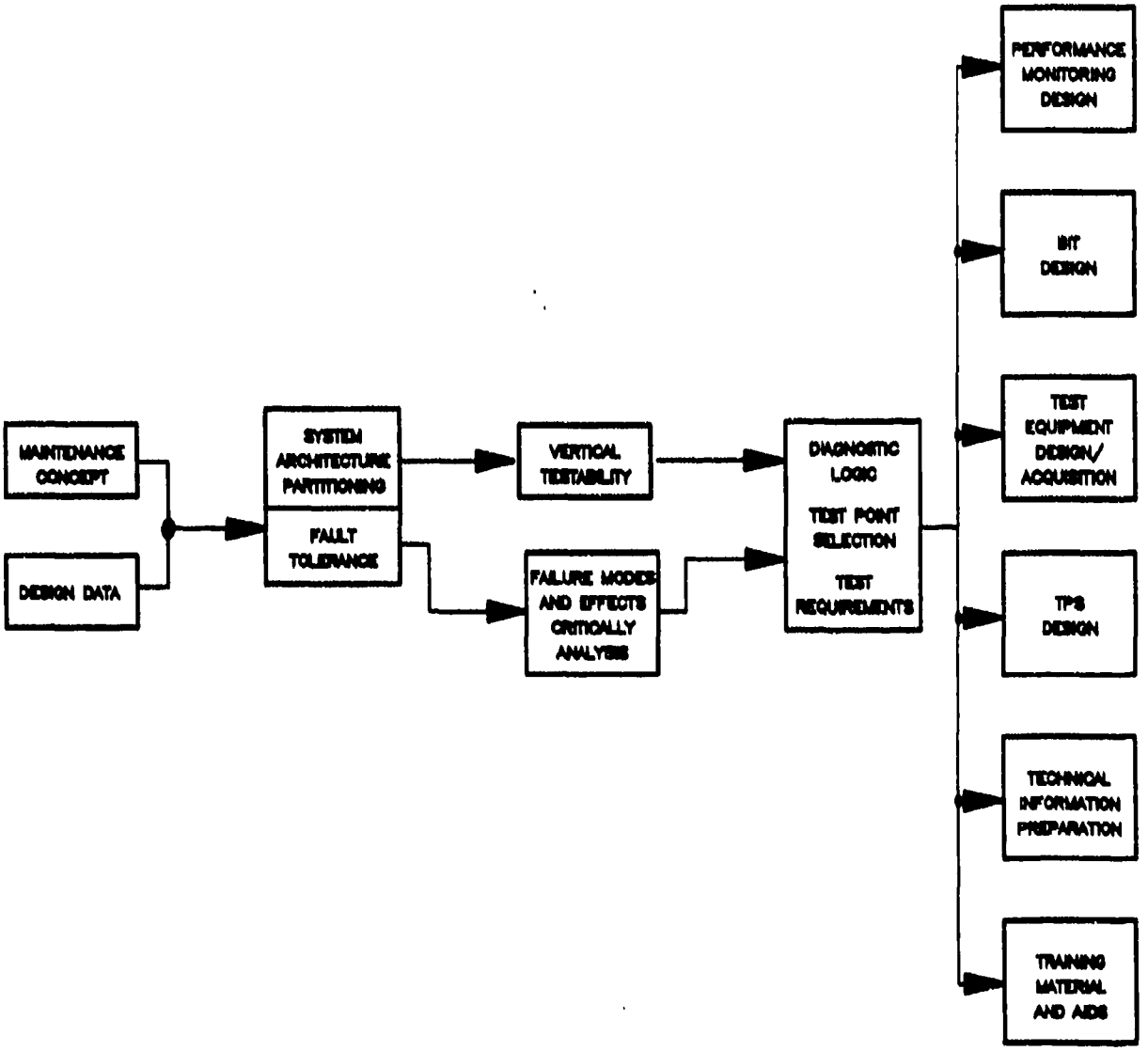


FIGURE 4. DESIGN INTEGRATION OF DIAGNOSTIC CAPABILITY

PROVIDING A COHESIVE DIAGNOSTIC DESIGN PROCESS REQUIREMENT #3.1

The concept of vertical testability was introduced years ago. In essence, this concept addressed the compatibility of testing among all levels of maintenance, including factory testing. The core of this concept is twofold. The first is the establishment of a Cone of Tolerance among these levels, and the second deals with the compatibility of environments under which these tests are performed.

The Cone of Tolerance concept is depicted in Figure 5, in which the testing tolerances are widened as the unit is tested closer to its operational environment.

The compatibility of testing environments can be implemented best through the use of common test equipment at Intermediate, Depot, and Factory Levels. This commonality of test equipment and any associated test programs is the method for implementing this compatibility.

The concept of vertical testability is key to the integration of the design of the diagnostic capability. Therefore additional guidance on vertical test methods is contained in Appendix D. This appendix also includes guidance on documenting the results of vertical testability analysis to assure this information will be integral to the diagnostic design process.

Extension of this vertical testability concept is recommended for the entire fielded diagnostic capability. Figure 6 depicts this concept, in which vertical testing is shown on the left and is joined by technical information and personnel and training compatibility requirements. Not only is this compatibility required vertically, but also horizontally. All elements that make up the diagnostic capability must be compatible at each maintenance level.

This concept of vertical and horizontal compatibility is key to the integration of diagnostic capability. The entire process is driven by the diagnostic logic which effects the requirements for all of the diagnostic elements. This diagnostic logic can be established by a variety of means including the use of maintenance dependency charts, fault trees, etc. To implement this concept, a series of matrices similar in format to Figure 6 can be prepared at various system hierarchy levels (e.g., system, subsystem, LRU, SRU). These matrices should be tailored to the unique requirements of a specific weapon system and may be used in conjunction with other required data deliverables (e.g., test requirements document).

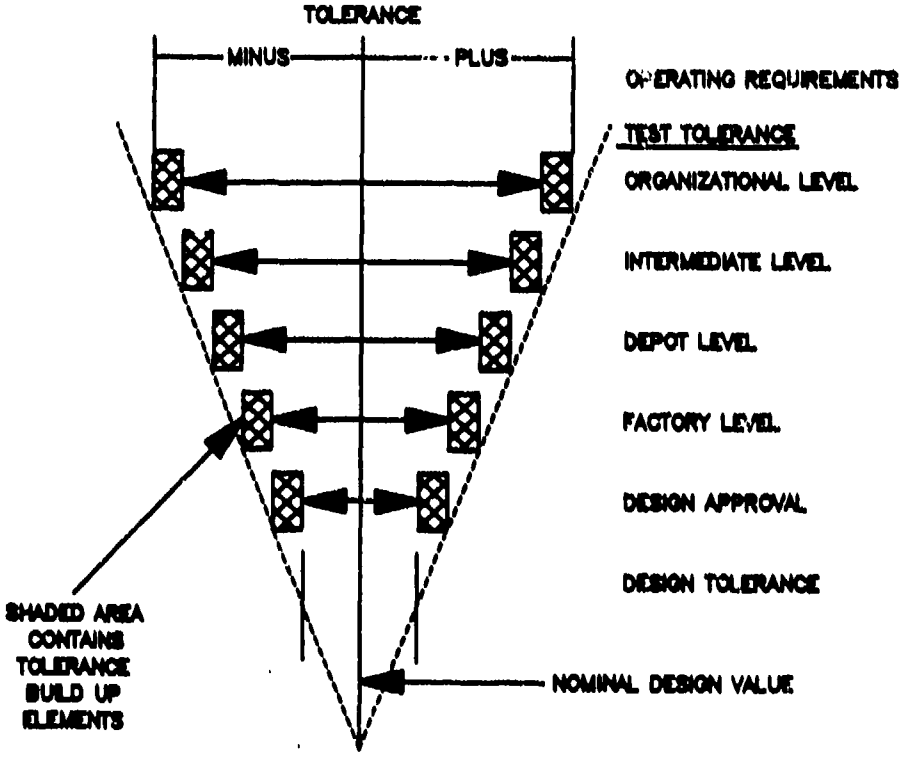


FIGURE 5. CONE OF TOLERANCE

PROVIDING A COHESIVE DIAGNOSTIC DESIGN PROCESS REQUIREMENT #3.1

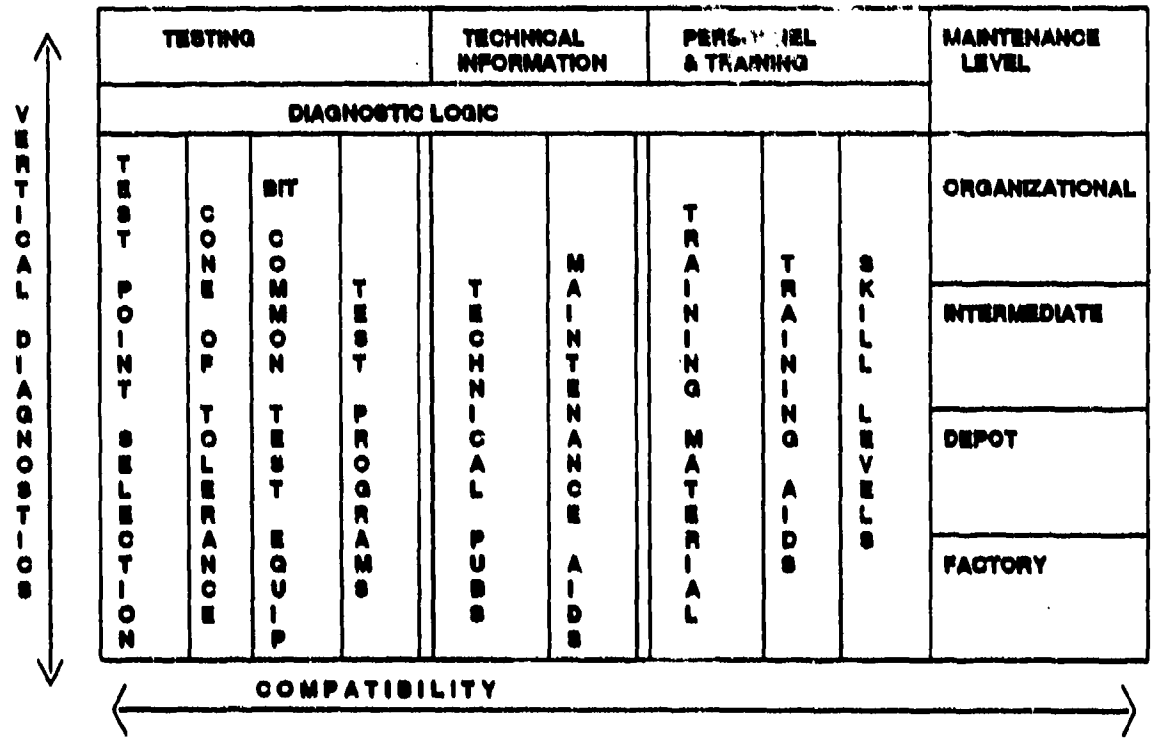


FIGURE 6. Design Integration of Diagnostic Capability

AUTOMATION OF THE DIAGNOSTIC DESIGN PROCESS

Automation of the diagnostic design process is encouraged to provide a more efficient and effective design process. The diagnostic design process should be an integral part of prime system computer-aided engineering and design.

The added efficiency and effectiveness in the use of automation is reflected in a number of ways. The effect of changes in either the diagnostic design or the prime system design can be readily ascertained as the design progresses. This iterative process then can give the designer information on whether or not the diagnostic specification requirements will be met. In addition, automation permits the concurrent development and evaluation of the entire diagnostic capability along with the remainder of the prime system.

Diagnostic Design Tools

Diagnostic design tools enhance the effectiveness and efficiency of the process. A description of available tools and processes is available in Appendix C. Appendix C identifies automated tools which can assist the designer in performing three major facets of the design process: system architecture design, implementation of design rules and practices, and diagnostic authoring.

The first element pertains to design automation tools that assist the designer in synthesizing a prime system functional capability, as well as providing an "environment" for developing a diagnostic capability concurrently with the prime weapon system development process. The architectural tools not only provide a capability to synthesize a functional capability, but also assist the designer in understanding systems methods of doing design work (i.e., operation).These tools generate documentation data bases, which are either explicitly or implicitly usable in the testing, technical information and/or personnel training disciplines.

The second element pertains to automated or manual tools which "capture" expert knowledge bases in diagnostic-related matrices for use by the designer. These knowledge bases may range from highly sophisticated and automated expert system software to unautomated, rudimentary checklists.

The final element pertains to tools and/or techniques which enable the designer to "author" (i.e., generate) diagnostic software routines and procedures utilizing prime system design data bases or heuristic information sources. These diagnostic authoring tools typically take the form of either expert system "knowledge bridges," which facilitate the extraction and/or generation of diagnostic-related procedures; or automatic test generators/fault simulators, which generate digital test vectors to fault detect/fault isolate an explicitly defined fault universe (i.e., stuck "1"/stuck "0"). In addition, time-tested analog and mixed mode simulators may be utilized not only as functional design tools, but also as diagnostic authoring tools in deriving and analyzing diagnostic test tolerances utilizing worst case or Monte Carlo analysis techniques.

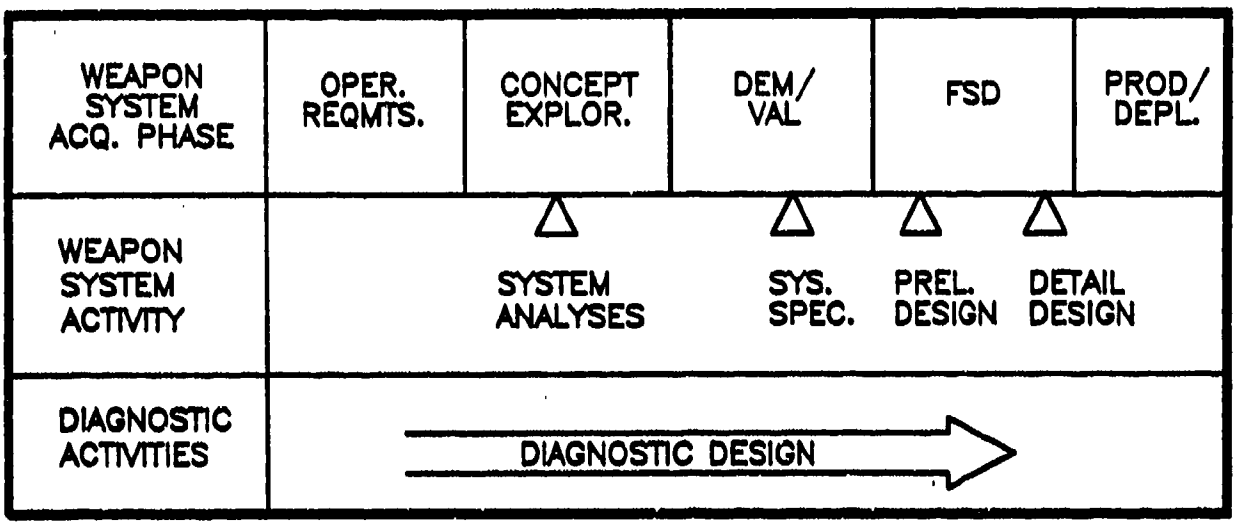
PROVIDING A COHESIVE DIAGNOSTIC DESIGN PROCESS	REQUIREMENT #3.1

CHECKLIST

- ☑ Has a concerted effort been made to assure vertical and horizontal integration and compatibility of all elements which comprise the diagnostic capability? Has this been documented for review?

- ☑ Have steps been taken to utilize automation of the diagnostic design process to enhance design efficiency and to improve the effectiveness of the fielded diagnostic capability? Have available design tools been utilized?

CRITERIA FOR DESIGNING DIAGNOSTIC CAPABILITY	REQUIREMENT #3.2



DIAGNOSTIC ACTIVITY

Design of the diagnostic capability and the elements that make up this capability are initiated early in weapon system development. It begins soon after initial analyses and allocation are completed and extends at least until Full-Scale Development has been completed. Design criteria and guidance need to be available for use as the diagnostic capability design progresses. Obviously, the bulk of this design guidance is utilized by the designer of the prime system and its support capability. He needs to be totally familiar with guidance that is available and be able to apply it appropriately.

PROCEDURE

Design criteria and guidance relating to the diagnostic capability and individual diagnostic elements are available from a number of sources, including standards, handbooks, and guides. Most often, this guidance is not a contractual requirement, except when a specific military standard is invoked. However, for the most part, the contractor should utilize this guidance material as he sees fit, as long as diagnostic requirements are met and interfaces are controlled. In addition, examples which depict the integration of the various diagnostic elements will be of value to both the manager and the designer.

Guidance to the designer consists of material contained in this section and identification of additional guidance where published material is not readily available. Tools to assist in the design process are described in Appendix C, 3.0.

GUIDANCE

The following are references to existing design criteria and guidance.

General Guidance

1. MIL-STD-454, Standard General Requirements for Electronic Equipment

   This standard covers the common requirements to be used in military specifications for electronic equipment. Reliability, maintainability, and human engineering requirements are included in this standard. However, for these types of engineering disciplines, the guidance stresses that this standard does not establish requirements and must not be referenced in contractual documents. These three requirement examples offer direction on what should be considered in preparing contractual documents.

2. MIL-STD-415, Design Criteria for Test Provisions for Electronic Systems and Associated Equipment

   This standard establishes design criteria for test provisions that permit the functional and static parameters of electronic systems and associated equipment to be monitored, evaluated, or isolated. The standard, in its present form, (Revision D) addresses older technologies and thus, if referenced in contractual documents, must be tailored to address only certain provisions in this standard.

3. The RADC Reliability Engineers Tool Kit

   The Tool Kit is intended for use by a practicing reliability and maintainability (R&M) engineer. Emphasis is placed on his role in the various R&M activities of an electronic systems development program. The Tool Kit is a compendium of useful R&M reference information to be used in everyday practice.

4. Study of the Causes of Unnecessary Removals of Avionics Equipment (RADC-TR-83-2)

   This study cites and verifies the causes of unnecessary removals of suspect items from avionics equipment and contains information on minimizing this problem.

System Architecture

Appendix E contains a compendium of testability and diagnostic design techniques, which provides designers various approaches and techniques for achieving improved testing of weapon systems. There are a number of other guides, which address the architecture of the system design, that promote improvements in the system's diagnostic capability. Included are:

1. Architecture Specification for PAVE PILLAR Avionics, January 1987, SPA90099001A

   This specification addresses the advanced avionics architecture which is specifically targeted for advanced tactical fighters and, in general, for all military aircraft applications. This architecture promotes a much-improved approach, which will foster an improved diagnostic capability. An example of this approach is contained later in this section.

2. Reliability, Testability Design Considerations for Fault Tolerant Systems (RADC-TR-84-57)

   Furnished reliability and testability evaluation and application guidance for fault-tolerant designs.

For fault tolerant systems, it is important that the design's inherent testability provisions include the ability to detect, identify, recover, and if necessary, reconfigure and report equipment malfunctions to operational personnel. In addition, because fault tolerant systems often are characterized by complex non-serial reliability block diagrams, a multitude of back-ups with non-zero switch over times, and imperfect fault detection, isolation, and recovery, it is imperative that the technical manager assure that effective testability provisions are incorporated in the system design concept. If not, the design when fielded will exhibit long troubleshooting times, high false alarm rates, and low levels of system readiness.

Fault tolerance and recovery strategies will have a significant impact on the degree to which testability is designed into the system. For example, when incorporating testability/diagnostic capability into the design, the penalties imposed by a fault tolerant system design which employs active redundancy and voting logic may be less than those imposed by a design employing standby redundancy. With active redundancy, the prime system hardware and software are more readily adaptable to perform multiple functions (including those required for testability). In active redundancy systems with voting logic, the performance/status monitoring function assures the operator that the equipment is working properly. This approach also simplifies the isolation of faults since the failure is easily isolated to the locked out branch, by the voting logic. In systems employing standby redundancy, test capability and diagnostic functions must often be designed into

each redundant or substitute functional path (both on-line and off-line) in order to determine their status.

Although the addition of redundancy is usually effective in improving system reliability, the technical manager is cautioned that the reliability improvement may be highly dependent on achievable FD/FI levels. In some cases, it is possible for imperfect FD/FI to actually cause system reliability to degrade as more redundant equipment is added. In general, the effect that varying levels of FD/FI have on system reliability can be evaluated by parametric analyses. The range of FD/FI values used in the analyses should be based on past experience with similar hardware/software systems and adjusted by evolutionary trends and expectations for state-of-the-art devices and designs.

Test Methodology for Fault Tolerant Systems - The following discusses a number of desirable design considerations for fault tolerant system testability.

- Comparison Method - An effective method for testing similar systems with similar inputs and outputs is to compare outputs and flag any gross disagreements. A means to determine which branch is faulted and an error log entry should be mandatory.

- Redundancy Verification - Each redundant path should be tested individually to prevent the mashing of faults in redundant items.

- Flexing of Spares - Periodically activate the built-in-test of the hot spares, log any errors found, and report status before these items are needed for system operation. This will prevent a faulty unit from being switched in when the system reconfigures.

- Voting Scheme Technique - A typical example of a voting scheme technique is to compare output values from three different sources. Confidence is placed in that value where at least two of the three sources agree. Errors found should be logged, and the source of the erroneous value should be recorded and corrected at an appropriate maintenance interval. Since diagnostic procedures are generally designed to locate a single fault, potential exists for the occurrence of multiple faults (e.g., a stuck-at-1 in multiple locations) than can go undetected. It may be necessary to add logic or test circuitry to ensure that each state, and each state transition, occurs correctly.

- Error Correction - Detection of degraded performance in states preceding an error correcting function is difficult since the error correcting function makes its preceding degraded state appear healthy. The error correcting functions should keep count of the number of times a correcting function had to be made and a record made in an error log. When a predetermined threshold

count is exceeded, a test signal may be injected to determine if the input stage is unacceptably degraded.

- Multiple Redundancy - In redundant systems, which are allowed to degrade gracefully through failures of redundant elements, a test should be established to verify that minimum acceptable system performance and redundancy levels are available at the start of a mission.

- Caution Indications - Fault tolerance can be applied to a variety of system types (i.e., electrical, mechanical, hydraulic, environmental, etc.). Regardless of the system type, it is customary to include a cautionary indication whenever a back-up system is called into service, especially for safety critical functions.

Fault Detection Latency Times - One of the most rigid demands imposed upon the testability design of fault tolerant systems is the quick response time necessary to reconfigure. Hence, the testability design process must take into account both spatial and temporal considerations for fault detection. The failure detection approach selection must be based upon the requirement for maximum acceptable failure latency. Continuous failure detection techniques should be used to monitor those functions that are mission critical and/or affect safety and where protection must be provided against the propagation of errors through the system. Periodic testing may be used for monitoring those functions which provide backup/standby capabilities or are not mission critical. Operator initiated testing is typically used for monitoring those functions which require operator interaction, sensor simulation, etc., or which are not easy, safe, or cost-effective to initiate automatically. The maximum permitted latency for failure detection determines the frequency at which diagnostic procedures should be run and must take into account function criticality, failure rate, possible wear out factors, and the overall maintenance concept.

Testability

There are a number of guidance documents which address testability issues. Some of these are listed below. These deal with the design techniques of controllability, observability, and partitioning. Controllability is a design attribute which describes the extent to which signals of interest may be controlled by the test process. It relates to difficulty of test generation, length of test sequence, and diagnostic resolution. Observability is another design attribute which describes the extent to which signals of interest may be observed by the test process. The emphasis is upon selection of the most appropriate test points. Partitioning deals with both the physical hardware and the functional partitioning of the circuitry. Test times and test generation costs escalate rapidly as partitioning size increases.

1. RADC Testability Notebook, Final Technical Report, June 1982

This notebook presents a consolidation of information relating to testability design techniques, procedures, cost trade-off tools, and the relationship of testability to other design disciplines and requirements. Specific examples of good testability design are contained in this document.

2. MIL-STD-2165, Testability Program for Electronic Systems and Equipments

   Appendix B of MIL-STD-2165 cites a series of factors which affect the inherent testability of a weapon system. This information can be used either as design guidance or, if weighted and scored, can actually provide a Figure of Merit for a specific system/unit.

3. Testability Analysis Handbook (Draft)

   At the time of printing the Contractor Program Managers Guide, the Testability Analysis Handbook was in draft form. Publishing is scheduled during FY89. The Preparing Activity is the Naval Sea Systems Command, CEL-DST. This handbook provides a systematic methodology for implementing testability analysis and design requirements, which are prescribed in MIL-STD-2165, Tasks 201, 202, and 203.

4. Predictions of Organizational Level Testability Attributes (RADC-TR-87-55)

   This report documents a methodology for predicting fraction of faults detected, fraction of faults isolated, and fraction of false alarms utilizing field measured data.

Built-In Test

1. Built-In Test Design Guide--Joint AMC/CNO/AFLC/AFSC Commanders, April 1987

   This Joint Service BIT Design Guide provides detailed guidelines on the implementation of BIT, including BIT design techniques at all levels within the weapon system.

2. Smart BIT (RADC-TR-85-148)

   Application of Artificial Intelligence techniques in the design of BIT, to minimize false alarms, retest OKs and non-required maintenance.

3. Sensor Handbook for Test, Monitoring, Diagnostic, and Control System Applications to Military Vehicles and Machinery, National Bureau of Standards

This handbook is intended as a guide for those who design, specify, use, and test weapon systems containing monitoring sensors. The Handbook addresses measures and principles of measurement, data acquisition, sensor calibration and testing, environmental considerations, stability, durability, reliability, and error assessment for various types of sensors.

4. Analysis of Built-In Test (BIT) False Alarm Conditions (RADC-TR-81-220)

   This study analyzes the root causes of the false alarm problem and provides design guidelines for avoiding BIT false alarms.

5. Design Guidelines and Optimization Procedures for Test Subsystem Design (RADC-TR-80-111)

   This study provides guidelines and procedures to optimize the design of built-in test.

6. BIT Verification Techniques (RADC-TR-86-241)

   This report covers practical verification techniques for formal and field demonstration of BIT effectiveness.

The problem of Built-In-Test False Alarms and Cannot Duplicates have plagued design for many years. These problems must receive the full attention of system designers. Future generations of BIT must include more emphasis on interpretation of detected system anomalies and better accounting for real world system operating conditions such as fielded system performance, environmental and operational factors.

In order for the BIT to reach, and remain at, its full potential in the field, it must be designed with sufficient flexibility, including the ability to easily adjust test limits and to change BIT software without affecting tactical software.

According to the above referenced document "Analysis of Built-In-Test (BIT) False Alarm Conditions", a common cause of false alarms are sudden environmental stresses such as momentary high temperatures, or a high "G" turn. The Rome Air Development Center, at the time of this printing, is developing a Time Stress Measurement Device (TSMD) chip which will monitor and categorize (in compacted form) data relative to the temperature, axial vibration and shock, and power quality that the equipment sees over time. A larger module has already been developed and flight tested which can monitor such characteristics. In the future, BIT indications can be correlated with TSMD data to help eliminate the occurrence of false alarms and CNDs. The integration of BIT indications, TSMD data, and smart (artificial intelligence) processing may also potentially yield even greater accuracy for onboard diagnostics.

Automatic Test Equipment (ATE)

1. Modular Automatic Test Equipment (MATE) Handbook

   Although Air Force-oriented, this handbook describe procedures and techniques for acquiring automatic test equipment.

2. MIL-STD-2077, General Requirements, Test Program Sets

   This standard covers the acquisition of test program sets for use with ATE. Design criteria is included, which addresses many detail requirements for TPSs.

Human Engineering

1. MIL-STD-1472, Human Engineering Design Criteria for Military Systems, Equipment, and Facilities

   This standard covers general human engineering design criteria which can be applied to any weapon system.

Technical Information

There are a variety of standards which address the preparation of technical publications. Most of these documents are directed at a specific military service. All address the delivery of paper-type documentation. There is no firm guidance relating to other, perhaps more, innovative means for generating and delivering technical information. In the past, many technical publications have been cited to have deficiencies. These deficiencies can best be described in the DoD Audit Report No. 87-115, April 3, 1987, "Summary Report on the Defense-Wide Audit on Acquisition of Technical Manuals and Related Data From Contractors."

Means should be sought for generating and delivering this technical information in a less costly manner, without compromising its quality. There are a number of tools available, or under development, which can assist the designer of this technical information in authoring the text, when electronic delivery of technical information is contemplated. Some guidelines and standards for automatic generation of technical information and its delivery electronically can be obtained from the Human Resources Laboratory at Wright-Patterson Air Force Base. This guidance information has been developed under the Integrated Maintenance Information System (IMIS) Program.

Innovative ideas for displaying this technical information are encouraged, as stipulated in Task 303, MIL-STD-1388-1. Care should be taken to provide for quick access to the required data. For electronic delivery of this data, formats may vary substantially from paper-based technical manuals. Previously specified access times and

information modification times should influence the type of generation and delivery methods. DoD-Instruction 4151.9 requires the services to plan and schedule the acquisition of technical manuals (technical information) to ensure their availability in final form before, or concurrently with, delivery of the system to the field. During design, final plans should be developed, along with the support equipment which is furnished.

Maintenance Aids

There is a need to present technical information and troubleshooting advice to the technician on location and readily available for his use. The maintenance aid, sometimes called a job performance aid, presents information generated by experts to assist the less-experienced technician.

The maintenance aid is a device, publication, or guide used on the job to facilitate performance of maintenance. It delivers:

- o Historical information on what fault was found when similar symptoms were experienced
- o Troubleshooting logic to assist in finding the fault
- o Procedural information which assists the technician in finding and correcting a failure.

Normally, a maintenance aid is used in conjunction with a testing capability. Maintenance aids could be paper-based or employ electronic delivery systems.

Electronic delivery of this type of information opens the door to solving some of the problems associated with paper maintenance aids. Two attributes of electronic delivery systems are:

- o Information can be available to the technician in a matter of seconds by carefully constructed menus, in lieu of the technician having to page through a paper document.
- o The collection of historical data and the subsequent modification to the software programs which deliver this technical information can be updated in a matter of seconds, instead of a matter of months.

This latter attribute lends itself to the introduction of expert systems, which often employ artificial intelligence technology. The expert system has the capability of combining various pieces of information to lead the technician to a logical decision on what is faulty and how it can be repaired.

Another important aspect of the maintenance aid is its ability to train technicians on the job. Training programs must be closely associated with the design and development of a maintenance aid.

Over the past 20 years, many maintenance aids have been designed, developed, and tested. These tests, for the most part, have proven successful. However, the transition of these maintenance aids into the field has not been accomplished to any great extent. One of the reasons is that specifications, standards, and guidance information on how to invoke this requirement are lacking.

A few important facts should be remembered when applying maintenance aids and expert systems.

- The design of the maintenance aids must be done with the user in mind. Once a working model of the equipment is available, there should be a dynamic interchange of information between the maintenance technician and the design engineer to ensure an effective and efficient man-machine interface is attained.

- User acceptance and adoption of maintenance aids will be facilitated in cases where potential users are given a trial period in which to become familiar with these devices prior to their formal implementation.

- A system must be devised to assure timely updating of information to correct errors and to add newly acquired information. Without such a system, the maintenance aid will quite rapidly become obsolete.

Manpower and Training

After personnel and training requirements/allocations have been made, the training curriculum needs to be established concurrently with the system detail design. This includes the formal schooling curriculum, as well as on-the-job training. One of the alternatives available, if electronic delivery of technical information is employed, is combining training aids with the delivered technical information. These two types of information (aiding and training) are somewhat similar in nature and, at times, indistinguishable. The training curriculum should be aimed at the user(s) and accessed in a manner which can be utilized by a variety of users.

These training devices can be freestanding or embedded in the prime system. They can serve as just maintenance training devices or they can be incorporated with operational training. Separate and distinct training devices (maintenance trainers) may be required to be developed for the formal schooling.

PROGNOSTICS

Although rarely considered in electronic system design, prognostics (incipient failure detection) techniques may have a significant impact in improving the operational readiness and mission success rate of future systems. Having the ability to test an equipment to see if any of its' critical components are soon to fail could radically change the repair philosophy for a system. An RADC study entitled "Marginal Checking" indicated that often prognostics techniques must be developed on an item by item basis. This being the case, it makes sense to concentrate first on developing techniques for detecting incipient failure of high failure rate critical components. The "Marginal Checking" study has identified potential prognostics techniques which are appropriate to cables, power supply bridge rectifiers and CMOS circuitry.

Integration of Diagnostic Elements

There are a variety of ways in which diagnostic elements can be integrated to produce a more effective and efficient diagnostic capability. Expert system technology can be incorporated in either ATE or BIT to supplement the basic testing capability. Fault-tolerant design and testability design can be introduced into prime system architecture to promote ease of testing, along with graceful degradation. Maintenance aiding and maintenance training can be combined to provide on-the-job maintenance and training information, utilizing a single portable device or embedded into the prime system. Two examples of this type of integration follow.

ADVANCED AVIONICS ARCHITECTURE EXAMPLE AS EMBODIED IN THE PAVE PILLAR

Advanced Modular Avionics Diagnostic Requirements

Mission availability and probability of mission success expectations for advanced modular avionic architectures such as PAVE PILLAR are totally dependent upon the embedded diagnostic capability that is implemented as an integral part of the design. The implication of this statement represents a significant departure from the traditional relationship that has existed between the circuit design and BIT/testability disciplines. An overview of the PAVE PILLAR modular avionics architecture with its integral diagnostic features is provided in the paragraphs that follow to facilitate an understanding of the new relationship that must be developed.

PAVE PILLAR Overview:

PAVE PILLAR is a highly modular flexible avionic system architecture which employs a common module set to exploit the commonality in air-to-air and air-to-ground missions. The major functional partitions of the avionics suite are: Digital Signal Processing, Mission Processing, Vehicle Management Processing, and Avionics System Control. Figure 7 depicts the enclosing boundaries (i.e., Digital Signal Processing, Mission Processing, and Vehicle Management Processing) for resource sharing, sparing, and substitutions. The unique characteristics of the functions within each of these boundaries preclude the utilization of resources across boundaries for the purpose of recovery or reconfiguration. As a consequence of this partitioning, the diagnostic process takes place within these boundaries and the associated results are communicated across the boundaries to provide the pilot with the required system status. The system architecture which supports the diagnostic process is described in the paragraphs below.

Diagnostic Strategy:

The PAVE PILLAR advanced system architecture employs a hierarchical approach that spans system elements from the individual chips to the entire system. Essential features of this hierarchical diagnostic architecture are shown in Figure 8. The incorporation of a test control function at each level of fault detection (i.e., chip, LRM, subsystem, and system) can facilitate both fault screening and test augmentation functions. The inherent flexibility provided by this architecture provides the system designer the ability to meet weapon system specific diagnostic requirements with a suite of standard modules.

It is essential that both the system and detail designer recognize the importance of implementing such a hierarchical diagnostic structure. A suite of standard line replaceable module (LRMs) will have individual fault detection, fault isolation and false alarm rate specifications that are not necessarily adequate to meet the system-level requirements imposed without fault screening and test augmentation. Advanced BIT concepts, which have been introduced through the "Smart BIT" project, are both compatible and dependent upon the availability of a hierarchical diagnostic architecture. A representative diagnostic implementation is provided in Figure 9.

LRM Bus Structure:

The bus structure associated with the diagnostic system implementation, shown in Figure 9, is dependent upon local maintenance and data buses, as well as system level multiplex data communications. At the chip and module level, the primary diagnostic control is provided over the VHSIC Phase 2 defined TM - Bus.

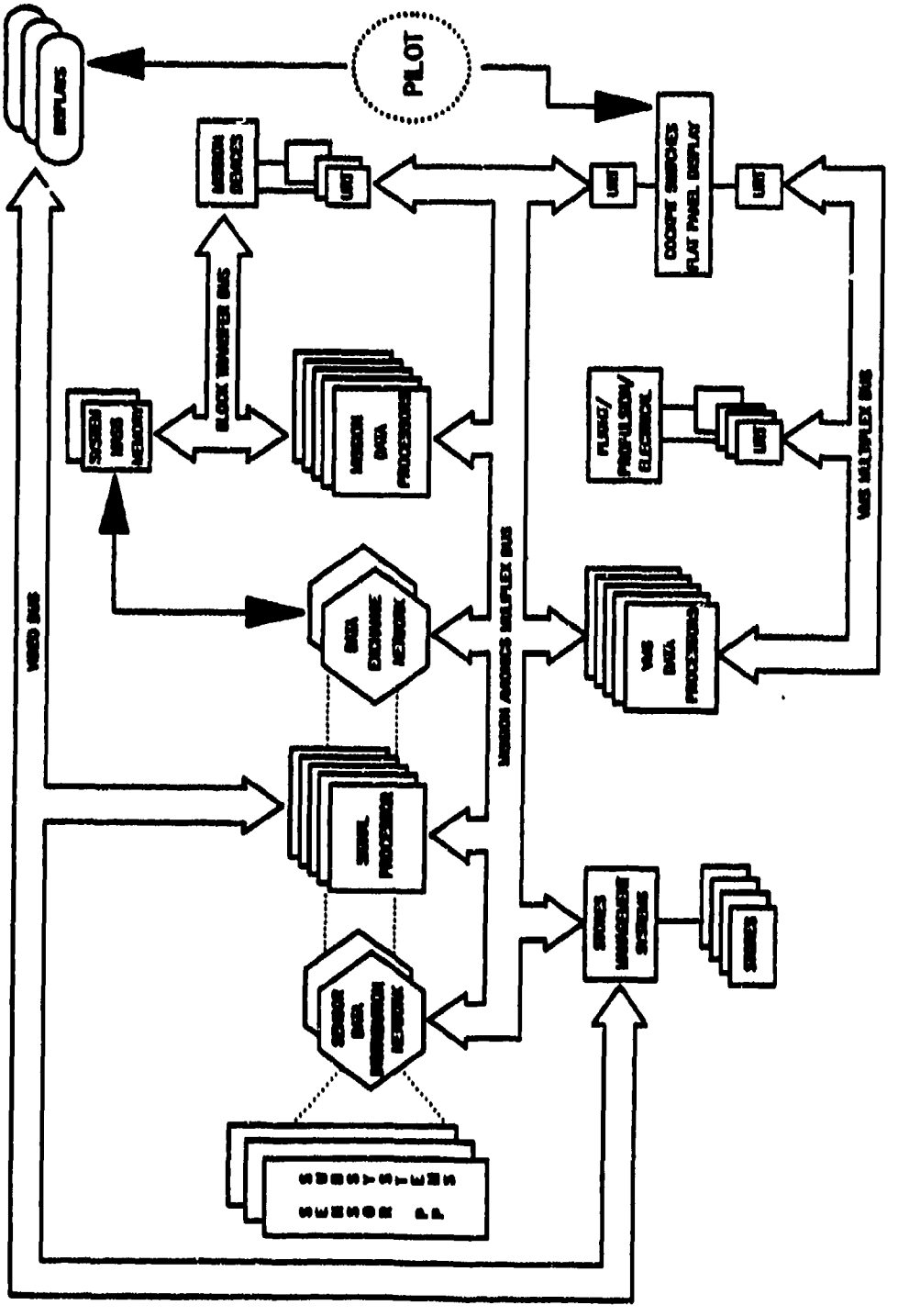


FIGURE 7. – COMMON ARCHITECTURE

HIERARCHICAL APPROACH USED FOR FAULT DETECTION, ISOLATION & SYSTEM/SUBSYSTEM STATUS REPORTING

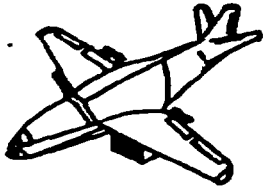
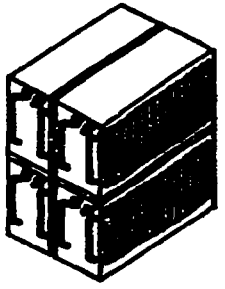
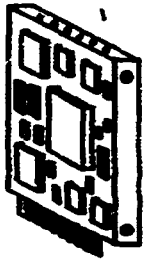
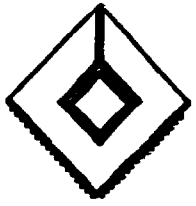
FAULT MANAGEMENT COMPUTER	SUBSYSTEM TEST CONTROLLER	MODULE TEST CONTROLLER	ON-CHIP TESTING
			
• MONITOR STATUS OF SUBSYSTEMS • ENABLES SUB-SYSTEM TESTING • PERFORMS INTER-CONNECT INTEGRITY TESTING • REPORT STATUS TO PILOT AND MAINTAINER	• MONITORS MODULE TEST STATUS • ENABLES MODULE TESTING • TESTS MAINTE-NANCE BUSSES • REPORTS SUB-SYSTEM STATUS TO FAULT MANAGEMENT COMPUTER	• MONITORS COMPONENT FAULT STATUS • INITIATES AND CONTROLS ALL MODULE TESTING • RETRYS BIT • REPORTS STATUS TO SUBSYSTEM TEST CONTROLLER	• CONTINUALLY TESTS ITSELF • FAULTS DETECTED AT LOWEST LEVEL • STATUS INDICATED TO MODULE TEST CONTROLLER

FIGURE 8. DIAGNOSTIC STRATEGY

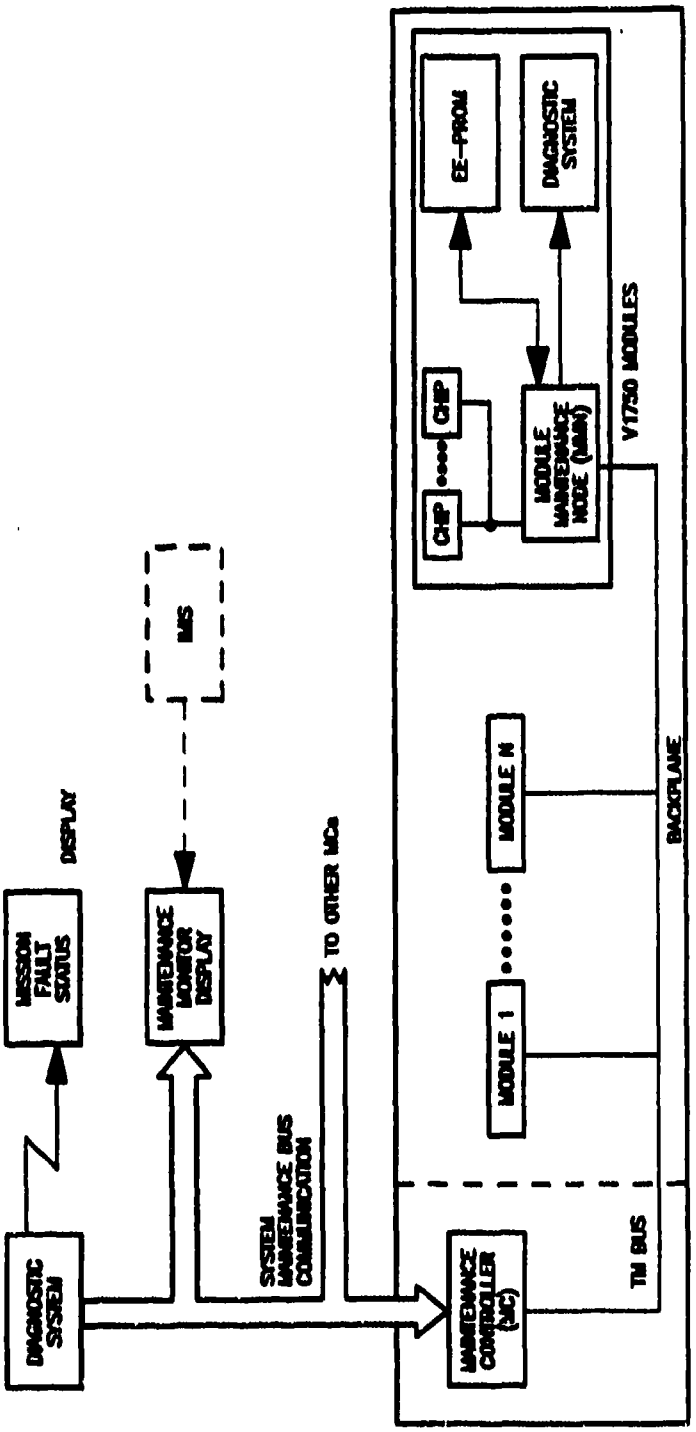


FIGURE 9. DIAGNOSTIC SYSTEM IMPLEMENTATION

Communication of diagnostic information between subsystems within the major PAVE PILLAR partitions (i.e., Signal Processing, Mission Processing, and Vehicle Management Processing) takes place over the respective partition Multiplex Bus. This configuration permits the use of a separate diagnostic/maintenance processor, or the incorporation of these functions within the mission processing function. This diagnostic system implementation is also compatible with the requirements for the System Status function, as currently defined under the Pilots Associate Program. (The Pilots Associate Program is sponsored by the Defense Advanced Research Projects Agency under Program Element Number 62301E. The performing activity is the Wright R&D Center (WRDC).

Diagnostic Applications, Summary and Conclusions:

The diagnostic architecture described above incorporates all the necessary functions to support the following operational and diagnostic/testability objectives:

Fault Detection
Fault Screening
Fault Isolation (Controllability, Observability)
Fault Recovery (Redundancy, Reconfiguration, or Graceful Degradation)
System Status Indication
Maintenance Data Recording
Repair Verification
Off-Board Maintenance Interface (IMIS)

In additional to the complete range of functions identified, the hierarchical diagnostic architecture offers sufficient flexibility for the system designer to achieve the weapon system supportability required. The layered access to the diagnostic capability within the system is essential for the application of artificial intelligence based BIT techniques being developed under programs such as: Smart BIT, Flight Control Maintenance Diagnostic System, Pilots Associate, etc. (These latter two programs are being performed by the Wright R&D Center (WRDC), WPAFB, Ohio.)

B-1B EXAMPLE OF DIAGNOSTIC INTEGRATION

The B-1B Bomber provides an example of many different facets of diagnostics deliberately combined to provide a total integrated diagnostics capability. The diagnostic elements include conventional and artificial intelligence diagnostic techniques, real-time in-flight performance monitoring and ground readiness testing, system performance monitoring for aircrew information and LRU fault isolation for maintenance personnel, detailed embedded data acquisition equipment and ground processing, standard inspection and other scheduled maintenance tasks (primarily in mechanical areas), and status information developed by the defensive and offensive avionics.

As shown in Figure 10, the core of the B-1B diagnostics is an on-board Central Integrated Test System (CITS). The general philosophy of CITS is to monitor and record activity on the aircraft equipment buses as well as performance status information generated by the defensive and offensive avionics system. Signal levels are compared to standards by the CITS computer. In the event of a failure, a CITS Maintenance Code (CMC) is generated identifying the faulty LRU. Both the CMC and measured signal levels are recorded for later analysis by a ground processor located in the intermediate shop.

Embedded equipment which makes up CITS include four data acquisition units, a CITS computer, an airborne printer, a CITS control and display panel, and the CITS maintenance recorder.

Associated ground equipment is the CITS Ground Processor. It is used for retrieving and interpreting the data recorded during each flight. The artificial intelligence portion of the diagnostics (CITS Expert Parameter System or CEPS) is also resident in a separate ground computer. The CITS Ground Processor is used to evaluate the maintenance codes recorded in flight and issue work orders directing the removal of the faulty LRU. CEPS is used when the CMC does not isolate the fault to single LRU. CEPS utilizes past history, expert diagnostic approaches, and monitored environmental data at the time of the failure to further break the CITS ambiguity groups for isolation to the single failed LRU.

Technical Orders (TOs) and crew training still play an important part in the overall diagnostics. Ground readiness tests are manually initiated following an LRU replacement. These tests are to assure proper system operation. They are performed per instructions in the TOs using the applicable tests of CITS. In addition there are limited physical inspections directed by the TOs covering the traditional but still effective monitoring of wear and fluid contamination.

The design process of integrating all of the above centered around three established disciplines. They are 1) a structured systems engineering approach, 2) a Failure Mode, Effects, and Criticality Analysis (FMECA), and 3) Logistic Support Analysis and Support Equipment Recommendation Data development.

CITS design manuals governed the systems engineering process. These manuals were created following MIL-Standard software development specifications and associated reviews. A basic document called CITS Autoflow was created for each system/subsystem which delineated the tests to be made for fault detection and isolation. The Autoflow identifies which inputs and outputs from each box are to be checked to assure that the problem is within the box and not caused externally.

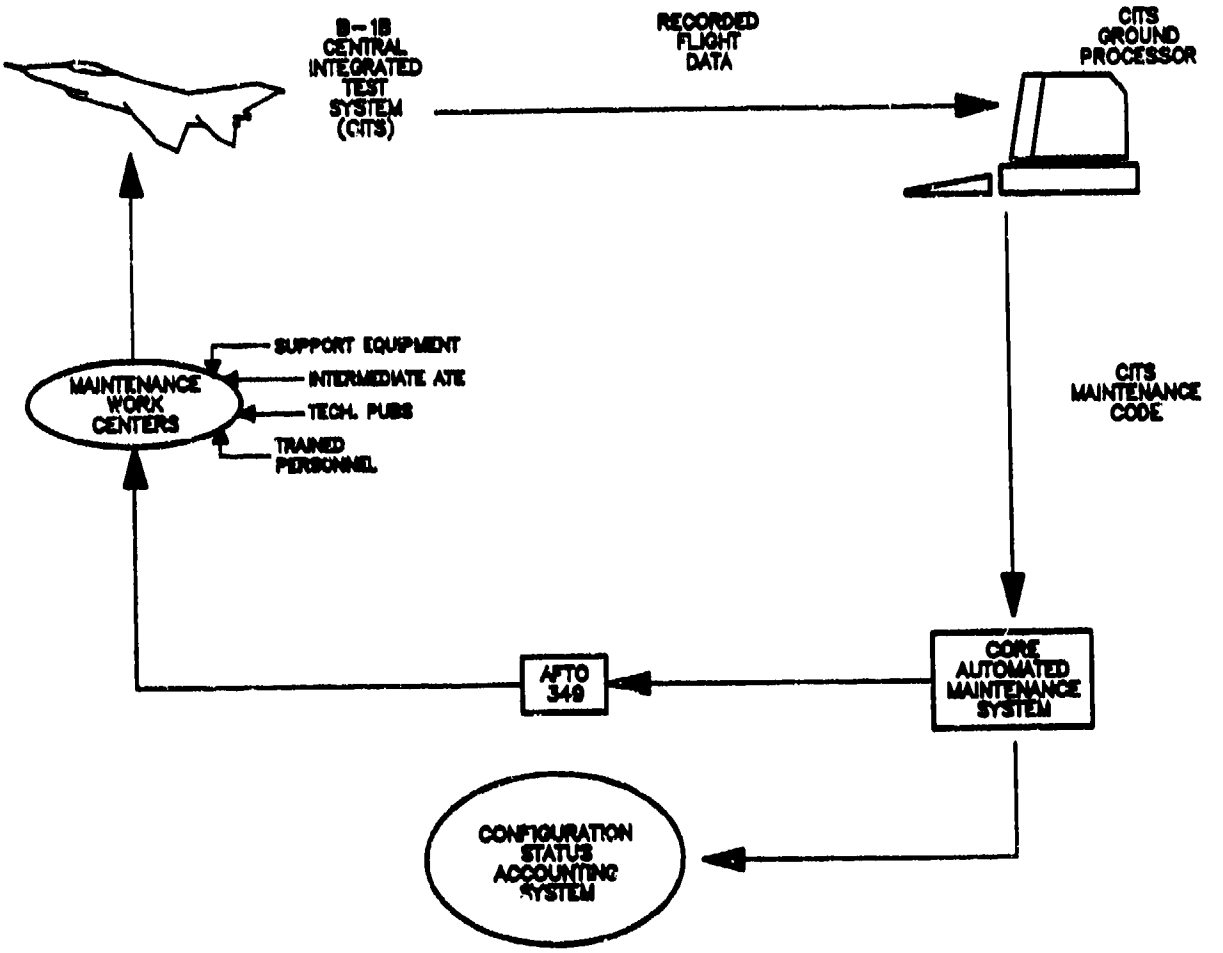


FIGURE 10. B-1B MAINTENANCE CONCEPT

CRITERIA FOR DESIGNING DIAGNOSTIC CAPABILITY	REQUIREMENT #3.2

A detailed Failure Modes, Effects and Criticality Analysis (FMECA) provided the initial basis for selecting CITS tests. This was augmented by a structured LSA which identified all diagnostic tasks required to be accomplished. Part I Support Equipment Recommendation Data (SERD) was created documenting the need for all applicable support equipment. CITS personnel then evaluated each SERD and where possible made sure that the requirement was met by CITS. Where applicable, other visual inspections, etc., were also considered. All SERD requirements were eventually satisfied without the use of a significant amount of ground support equipment.

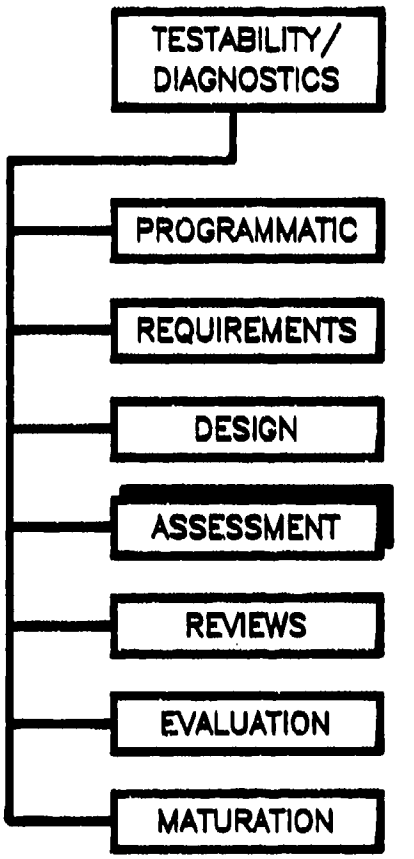
CRITERIA FOR DESIGNING
DIAGNOSTIC CAPABILITY

REQUIREMENT #3.2

CHECKLIST

- ☑ Are available design criteria, rules, and other available guidance documentation being used?
- ☑ Is the integration of the various diagnostic elements being accomplished to provide a more effective and efficient diagnostic capability?
- ☑ Is design-for-test an integral part of the system design process?
- ☑ Has the designer chosen the testability/diagnostic design techniques appropriate for the level (or levels) of test?
- ☑ Have the general design considerations and the standard testability approaches been thoroughly evaluated and trade offs performed?
- ☑ Have appropriate fault isolation techniques been incorporated into the overall approach?
- ☑ Has the impact of the physical packaging of the system been thoroughly evaluated?
- ☑ Has a test and maintenance bus been considered for integration into the system?

ANALYSIS AND ASSESSMENT OF THE PERFORMANCE OF THE DIAGNOSTIC CAPABILITY



OVERVIEW

Throughout the design of the weapon system's diagnostic capability, it is essential to analyze, assess and demonstrate its performance. Such assessments are an integral part of logistics, reliability, maintainability, testability, human engineering, software and safety programs. The ability to properly conduct these analyses, assessments, and demonstrations is hampered by the lack of available techniques and tools to help, and the incompatibility of the available tools and techniques to function together. Thus both the program manager and the designer must have sufficient knowledge to understand the processes utilized and integrate these processes and tools to do the best possible job.

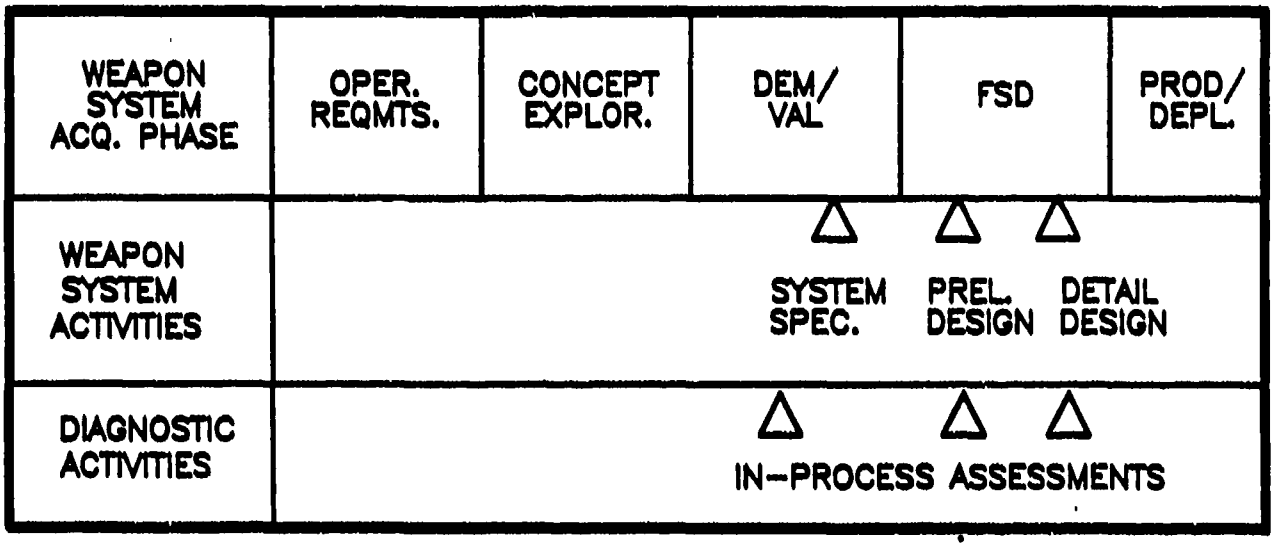
IMPORTANT CONSIDERATIONS TO BE ADDRESSED

Reqmt.

4.1 Analysis and assessment of the diagnostic capability should be performed for the entire diagnostic capability, as well as for each diagnostic element.

4.2 Maintainability demonstrations should be designed to verify that diagnostic requirements have been met.

IN–PROCESS TESTABILITY/DIAGNOSTIC ANALYSIS	REQUIREMENT #4.1



DIAGNOSTIC ACTIVITY

During Dem/Val and FSD, it is important to assess whether the testability/diagnostic requirements are being achieved. This activity encompasses all preliminary and full-scale engineering activities pertaining to both the embedded and external diagnostic capability.

PROCEDURE

In-process testability/diagnostic analyses can be conducted at most any time within Dem/Val and FSD. These in-process analyses are typically reviewed by the government at Preliminary Design Reviews and Critical Design Reviews. These analyses are, for the most part, implemented per MIL-STD-2165 (Task 202, Preliminary Design, and Task 203, Detail Design). Normally, these analyses will be the responsibility of the design or test engineer.

GUIDANCE

Basically, there are two types of in-process analyses. The first deals with the inherent testability of the hardware design and is independent of test stimuli and response data. The second type deals with the effectiveness of the diagnostic capability which deals with measures that include consideration of hardware design, embedded diagnostics, and external diagnostics. Diagnostic effectiveness measures include, but are not limited to, fault coverage, fault resolution, fault detection time, fault isolation time, and false alarm rate.

There are a number of techniques and tools available, both automatic and manual, which can be used to assist in these analyses. A thorough description of available techniques is contained in the Testability Analysis Handbook, which is referenced under Requirement #3.2. A description of available tools to assist in these analyses is contained in Appendix C.

INHERENT TESTABILITY

The first analysis deals with inherent testability. Inherent testability assessment is an evaluation of how well a design supports the testing process, whether built-in test or off-line test. The evaluation is preformed on the preliminary design and is performed before any test design is performed. It is, therefore, based solely upon the hardware design features, such as physical and electrical partitioning, controllability, observability, and test point placement, etc. The key to performing an inherent testability assessment is the identification of features which support or inhibit the diagnostic process early, at a point in time when the design can be changed relatively easily. The concept and the implementation of an inherent testability assessment can have great impact on overall system supportability.

In general, three generic groups of inherent testability predictive techniques are available, each with its unique advantages and disadvantages. Checklists are low cost, manual, and somewhat simplistic. Logic models utilize the actual circuit topology but often regard everything as a block, with inputs and outputs. The more detailed algorithmic approaches, such as Sandia Controllability/Observability Analysis Program (SCOAP), require libraries of the devices that most nearly simulate actual circuit devices.

Checklist approaches to inherent testability assessment have some very good characteristics. Checklists are manual approaches to testability assessment, yet are easily automated into an interactive format for the designer to input design features to allow testability grading. However, significant engineering analysis is still required. Two checklists of that type are the RADC PCB Testability Design Rating System and the MIL-STD-2165 Appendix B Checklist. The original RADC PCB rating system was limited to lower density digital board applications, whereas MIL-STD-2165 Appendix B covers analog, digital,a d hybrid applications from module to system level. An updated RADC PCB rating system now under development will be expanded to cover all modern digital,

analog and hybrid PCBs. The RADC rating system has fixed items of weighting, whereas MIL-STD-2165 Appendix B allows subjective treating of items and weighting values. Both checklists can be utilized early in design.

Logic models have considerable success and validity other than in support of the testability discipline, including logistics, fault isolation, integrated diagnostics, and maintainability. The logic model algorithms are of varying sophistication and validity, although the methodology for defining dependencies are similar.

Logic models systems for testability are applicable to analog, digital, and hybrid applications. They can be modeled at the component, board, or module subsystem and system level. One limitation of this broad approach is that every item is identified as a box with inputs and outputs. Thus, box complexity might range from and OR gate to a complete microprocessor. The same variations applies to the lines between boxes. Critical signals, such as a clock or a tri-state bus are not more important than any other line. Two of the more well-known models are Logic Modeling (LOGMOD) and System Testability And Maintenance Program (STAMP). Both are mature in nature, but each is tied to a specific vendor at the present time.

Algorithmic approaches are perhaps the most sophisticated approach. SCOAP seems to usually perform well, but has had some library limitations in the important area of CMOS primitives. Some CAE workstation vendors are including modified versions of SCOAP for up-front testability analyses. Daisy workstations include the Daisy Testability Analyzer (DTA) package, and GE/CALMA workstations include the Controllability-Observability-Predictability-Testability Report (COPTR) package. Both have improved on the basic SCOAP, via top-down modeling and large device model libraries of the more common IC types. GenRad also has a package called HITAP, which is based on the Computer-Aided Measure for Logistic Testability (CAMELOT) algorithm.

Another major issue surrounding inherent testability assessment is that many of the automated tools which exist are proprietary. This proprietary nature of the tools creates implementation problems from both a cost and a contractual point of view. Often, the best approach is to utilize a nonproprietary automated tool for inherent testability assessment.

Prior to the FSD phase, the design or test engineer should develop a total strategy for conducting inherent testability assessment on all systems, subsystems, etc. Based upon the availability and applicability of current inherent testability assessment approaches, it is anticipated that combination of tools and techniques will be required to form a totally comprehensive measurement capability in areas where an automated capability is not available, use the baseline of existing models to make modifications to provide the total capability required.

An evaluation criteria for inherent testability assessment tools and techniques should be developed based upon specific system and subsystem specific needs. The following list of evaluation criteria is recommended:

- o Automation; degree of automation
- o Proprietary nature
- o User friendliness
- o Automated link to design data base
- o Acceptability of output to the government
- o Cost of use
- o Availability (currently available or under development)
- o Quality
- o Sensitivity to key testability features
- o Feedback provided (does it recommend design fixes?)
- o Comprehensiveness (digital, analog, RF, VHSIC, mechanical, etc.)
- o General techniques; principles used
- o Link to test effectiveness prediction technique
- o Output reports
- o Scoring methodology
- o Applicability to chip, board, subsystem

TEST EFFECTIVENESS

The second type of analysis deals with test effectiveness. Traditional approaches to determining test effectiveness call for the generation of test sequences at the completion of the design phase and a measure or measures made of their effectiveness. The analysis need not wait on the completion of BIT and/or off-line TPS software. Modeling is encouraged, since this approach can analyze test effectiveness on a large number of postulated faults prior to incorporating the test stimulus in either an embedded or off-line program. The results of the analysis can feed forward, so as to

influence BIT or TPS software design, and feed backward to influence possible redesign of the primary system to improve its testability. Test effectiveness measures have traditionally included:

- o Fault coverage
- o Fault resolution
- o Fault detection time
- o Fault isolation time

Computer programs are used to input (via software) a large number of faults into the software model of the hardware item (UUT). The response of the simulated item to the test sequence is then evaluated, given the presence of the simulated faults. Fault detection, resolution, etc., are automatically ascertained. Most modern Automatic Test Program Generation (ATPG) and simulation systems have very efficient fault simulation capabilities. The HITS system, for example, runs a concurrent fault simulation to greatly speed the process. The usefulness of this approach in measuring test effectiveness depends on the adequacy of the models (hardware item model and fault model) to accurately reflect the real-world situation. Modeling must be achieved at a level of detail that allows all known and statistically significant failure modes to be included.

Although commonly accepted, the application of these measures is in various stages of maturity, based upon the equipment composition (i.e., digital, analog, radio frequency and/or mechanical). At this time, the application experience has been concentrated in the area of digital implementations. However, even in this area, significant additional effort will be required in order to relate these measures to operational performance. The degree of application of test effectiveness measurement techniques to the remainder of the listed equipment types has been quite limited to date. IDSS, the Navy's integrated diagnostics program, has recognized this need and has an active diagnostic tool development program underway. One of these tools, the Weapon System Testability Analyzer, is structured to address test effectiveness measurement, as well as inherent testability assessment.

Effective and realistic fault modeling is a key element in the development of the simulation capability needed to support the development of either an ATPG or an automated test effectiveness measurement tool. However, it is anticipated that no single fault model and/or simulator will be applicable to the broad range of equipments to be employed within a complex system; therefore, a combination of models will be required to meet the requirement for automated determination of fault detection and isolation levels.

For False Alarm estimation, a procedure which is documented in a report (RADC-TR-87-55) entitled "Predictors of Organizational - Level Testability Attributes" developed prediction equations for various testability related parameters. Rather than try

to develop an equation to predict False Alarm Rate (FAR) directly, an approach was taken to predict the CND rate since this parameter should closely track FAR. The following details a prediction method for CND rate. Note that this is a model based on empirical data from avionics equipment of the mid-1980's era and, as is the case with all models, care must be taken in using the model for new technology or applications.

The equation for CND rate follows:

CND RATE =-0.0028 + 0.375*FLRRATE
+2.6 E-5*TRANSIENT + 5.9E-11*TC7

The variable FLRRATE accounts for the LRU Failure Rate.

The variable TRANSIENT attempts to characterize the tendency of an LRU to exhibit intermittent failures resulting from marginal or degrading components.

The variable TC7 numerically characterizes the likelihood of a sneak circuit existing in an LRU.

TRANSIENT = (IC's + 2*RESISTORS + 41*RELAYS + 2*CAPACITORS
- 9*TRANSISTORS)/# of SRU's

TC7 = INTERCONNECTS*(IC's - 160*RELAYS
- 960*SWITCHES + 30*TRANSISTORS)

Where;

RELAYS = Total # of Relays in LRU.

CAPACITORS = Total # of Capacitors in LRU.

RESISTORS = Total # of Resistors, both fixed and variable, in LRU.

TRANSISTORS = Total # of discrete transistors, including FET's, Bipolar, etc. that are in LRU design.

IC's = Total # of integrated circuits in LRU.

SRU's = Total # of SRU's that compose an LRU.

INTERCONNECTS = Total # of electrical interconnects used to mate SRU's to the host LRU.

SWITCHES = Total # of switches in the LRU design.

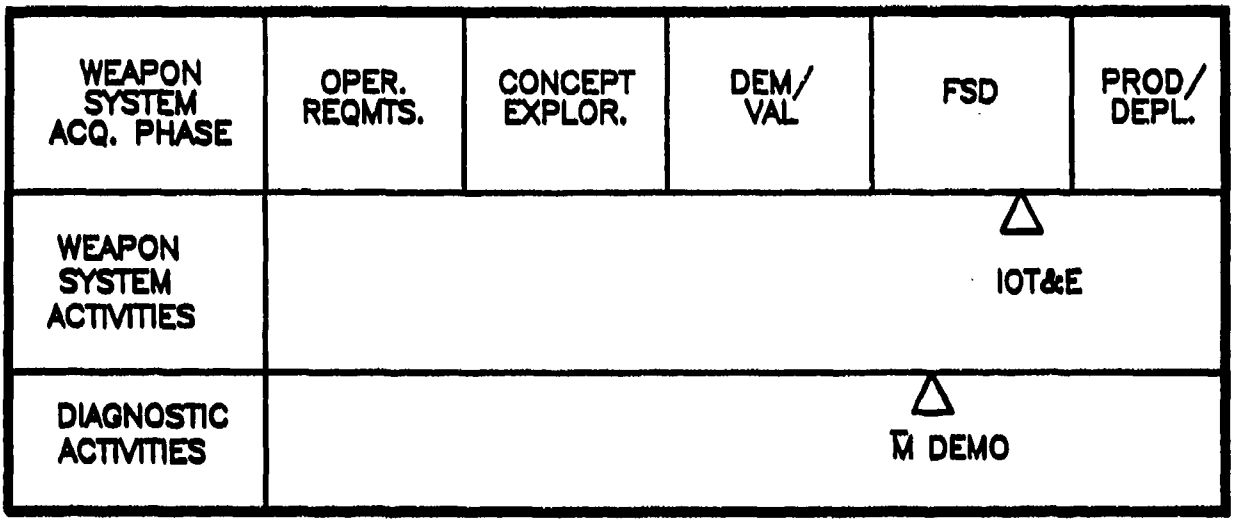
Specifics about the development or application of this equation, or the estimation of the input parameters can be found in the above referenced report.

IN–PROCESS TESTABILITY/DIAGNOSTIC ANALYSIS	REQUIREMENT #4.1

CHECKLIST

- ☑ Does the analysis of testability/diagnostic requirements address all major support disciplines?
  - Off–line ATE
  - Embedded diagnostics
  - Manpower required to support analysis outputs
  - Training requirements
  - Information requirements.
- ☑ Are all analyses complete and unambiguous? Do they meet specification requirements?
- ☑ Is the analysis integrated and cohesive? Are any requirements in conflict?
- ☑ Are the training, information, and manpower requirements adequately scoped and specified to support the technical complexity of the subject end item in its operational environment?
- ☑ Have available tools been selected and used?

MAINTAINABILITY DEMONSTRATIONS	REQUIREMENT #4.2



DIAGNOSTIC ACTIVITY

Maintainability Demonstrations are performed in accordance with the appropriate demonstration method contained in MIL-STD-471A. Notice 2 of MIL-STD-471A (USAF) contains requirements for demonstration and evaluation of system BIT/external test/fault isolation/testability attributes. This method will demonstrate the integration of the diagnostic capability for the system (e.g., integration of embedded test software and hardware techniques, automatic and manual test, BIT/SIT, training levels, human interfaces). The Maintainability Demonstrations evaluate the diagnostic performance of the system with respect to the diagnostic performance criteria and objectives established in accordance with MIL-STD-470 (Maintainability Program) and MIL-STD-2165 (Testability Program) and the requirements for an integrated diagnostics capability demonstration contained in the FSD SOW.

PROCEDURE

The integrated diagnostics process increases the scope of maintainability demonstrations. It is the Contractor Program Manager's responsibility to ensure that this increased scope is understood and implemented. It is the designers responsibility to design the demonstration, evaluate the results and take the necessary corrective actions.

The scope of Maintainability Demonstrations includes:

1. Demonstration of Testability Parameters
   - BIT Fault Detection
   - BIT Isolation Time

- BIT Fault Isolation Level (Ambiguity Group)

2. Demonstration of Test Effectiveness (ATE) (MIL-STD-2077)
   - ATE Fault Detection
   - ATE Fault Isolation Time
   - ATE Fault Isolation Level (Ambiguity Group)
   - UUT/ATE Compatibility

3. Demonstration of Technical Information
   - Technical Information Access Time
   - Technical Information Relative Access Ease
   - Technical Information Format
   - Technical Information Usability

4. Demonstration of Training/Skills
   - Relationship between maintenance procedures and skills
   - Relationship between formal training and actual maintenance job flow.

5. Demonstration of Vertical and Horizontal Integration
   - Compatibility and Consistency of diagnostic demonstration results between maintenance levels and among their respective diagnostic elements.

GUIDANCE

Unfortunately, the ability to carry out a single demonstration, or even a series of demonstrations, to prove/evaluate this level of diagnostic capability is dependent upon having all of the diagnostic elements available for the maintainability demonstration. While this should always be the goal, it may not be feasible for all of the above due to development schedules, UUT design instability, data availability and other overall program constraints. (Note that this is a primary reason for a Diagnostics Maturation Program.)

Typically, the contractor prepares a Maintainability Demonstration Plan early in the FSD Phase and that plan is subject to government review and approval. The designer should take advantage of this opportunity to design the Maintainability Demonstrations to include the factors cited above. This can have a significant cost-savings impact on the Diagnostics Maturation Program requirements. Maintainability Demonstrations represent the first major opportunity to evaluate the level of diagnostic capability achieved. Also, Maintainability Demonstrations can be conducted early enough to implement corrective action cost-effectively. Demonstrations are conducted while the system is still considered to be in the Development Phase. After the demonstrations are completed, the relative cost of identifying deficiencies and implementing corrective action is significantly increased.

A significant milestone of 'Government Acceptance' occurs upon satisfactory completion of Maintainability Demonstrations. After this milestone, costs for identification and resolution of diagnostic deficiencies may be subject to contract interpretation and/or negotiation. The total strategy for the test and evaluation of the diagnostic capability is placed on the TEMP, and detailed in the Integrated Test Plan.

Based upon the selected scope of the Maintainability Demonstration, procedures from MIL-STD-471 are utilized and adapted for the scope. These procedures are documented in the Maintainability Demonstration Plan. The results of the Maintainability Demonstration are documented in a technical report - Maintainability Demonstration Results.

Concurrent Demonstrations

The overall diagnostic capability is the sum of a variety of diagnostic elements. Therefore, a requirement should be established for early demonstration of the entire diagnostic capability produced by the integration of all of these diagnostic elements. This is referred to as concurrent demonstrations, where the timing of various diagnostic element demonstrations are planned and scheduled for concurrency so that the integrated capability can be assessed.

Each element of the diagnostic capability must be demonstrated, as well as the result of the combining or integration of the elements. For example, a demonstration of subsystem BIT may prove fault detection and isolation levels. A demonstration of ATE may prove external fault detection and isolation levels. A concurrent demonstration of these two diagnostic elements will prove the ability of the ATE to use BIT circuitry, to use BIT results, and the consistency of test results between BIT and ATE. By concurrent demonstration, the whole is greater than the sum of the parts. A significant set of factors related to the result of the integration of the diagnostic elements must be evaluated early.

CHECKLIST

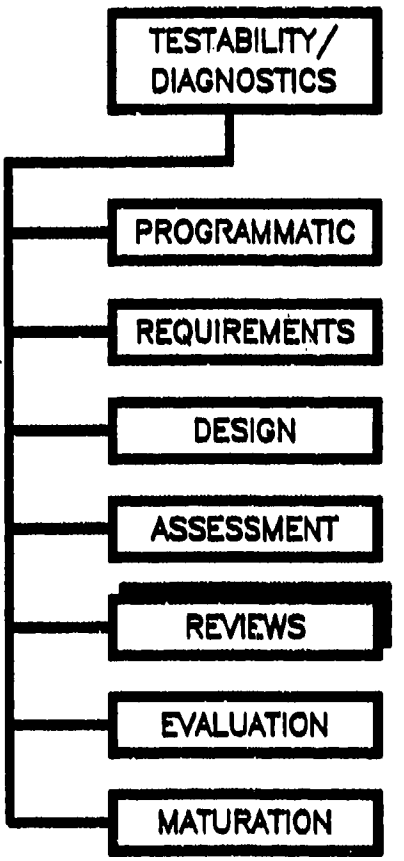
- ☑ Does the demonstration plan provide a 100% fault coverage capability across all levels of maintenance?
  - Organizational Level
  - Intermediate Level
  - Depot Level
- ☑ Are the failure modes to be demonstrated and criteria to be utilized adequately specified for each maintenance level? Will an adequate number of faults be inserted as required by MIL-STD-471 to statistically prove that FD/FI requirements are or are not met?
- ☑ Is the demonstation structured to provide an evaluation of the diagnostic capabilities as an integrated system?
- ☑ Do the subject plans demonstrate an integrated, cohesive maintenance flow in terms of demonstrating how a fault would be detected and repaired? Is a systems approach to the maintenance process taken in the overall approach to demonstration planning?

CONDUCTING DESIGN REVIEWS

OVERVIEW

During the acquisition of a weapon system there are at least eight formal technical reviews and audits, which may be conducted by the contractor for the Government Program Manager. As in the diagnostic design process, there is a tendency to conduct separate reviews and audits based upon the function being addressed. This particularly refers to logistics, reliability, maintainability, testability, human engineering, and safety. Integration of these reviews and audits to address diagnostic issues is a must. MIL-STD-1521 is the prime document which defines the issues to be addressed at each of these formal reviews. At present, these checklists are inadequate in terms of both testability and diagnostics and, thus, these reviews and audits may not serve their purpose. Additional guidance must be given to both the government and the contractor in order to alleviate this problem.

Informal reviews are often required. Guidance for these informal reviews can be drawn from formal review guidance.

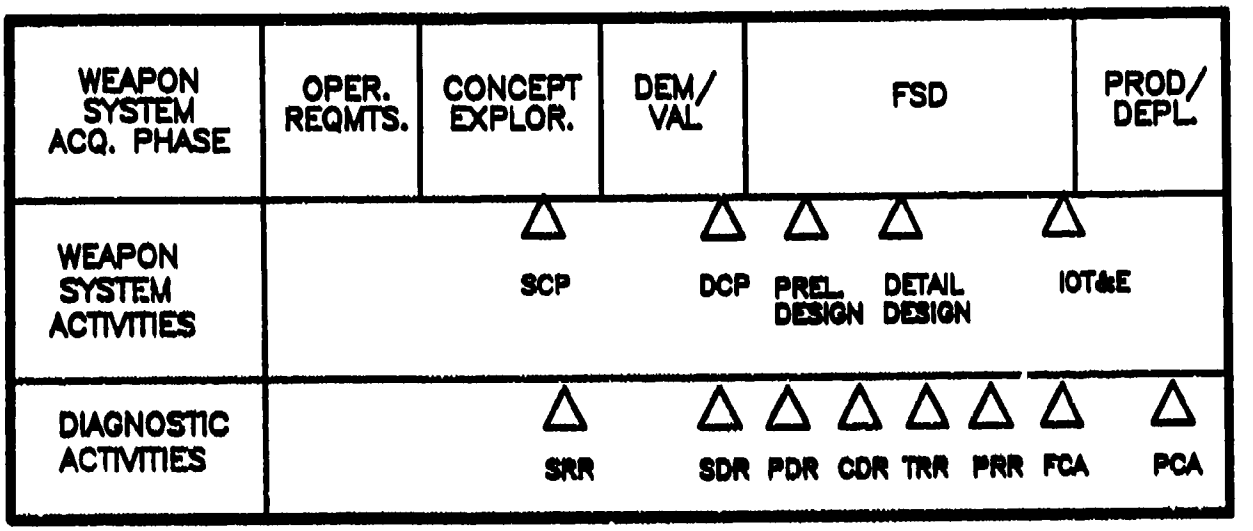


IMPORTANT CONSIDERATIONS TO BE ADDRESSED

Reqmt.

5.1 Technical reviews and audits must address all facets which affect the performance of the diagnostic capability.

CONDUCTING TECHNICAL REVIEWS AND AUDITS	REQUIREMENT #5.1



DIAGNOSTIC ACTIVITY

Technical reviews and audits are an important factor in assuring that the government is furnished with a weapon system which meets its requirements.

PROCEDURE

MIL-STD-1521 lists 10 formal technical reviews and audits. Of these 10, eight are considered critical in the achievement of a satisfactory diagnostic capability. The following guidance supplements and expands the guidance contained in MIL-STD-1521, Technical Reviews and Audits for Systems, Equipments, and Computer Software.

Although design reviews are recognized as being important to verify design before production, the lack of depth in these reviews is alarming. The cause of these inadequate reviews must be shared by both the contractors and the government. Contractually, the government rarely requires the contractor to do a comprehensive technical review, and the contractor does not do so unless required, even though it may be cost effective from his point of view. Even when the right words are used, the end results depend largely on corporate policy to allocate sufficient resources to perform a detailed analysis of the design and associated processes. The designer, obviously, has an important input to these reviews. Therefore, it follows that he must understand the objectives and scope of these reviews.

GUIDANCE

Guidance relating to these various reviews is contained in the appendices to MIL-STD-1521. Because these appendices do not address all aspects of testability and diagnostics, some supplemental information is included in the following paragraphs.

System Requirements Review (SRR)

The objective of this review is to ascertain the adequacy of the contractor's efforts in defining system requirements. It will be conducted when a significant portion of the system functional requirements has been established.

The diagnostic capability review portion of the SRR will analyze the system items that are related to diagnostics. The following items will be reviewed, as appropriate:

- Mission and Requirements Analysis
- Functional Flow Analysis
- Preliminary Requirements Allocation
- System/Cost Effectiveness Analysis
- Trade Studies
- Synthesis
- Logistic Support Analysis
- Specialty Discipline Studies
- Generation of Specifications
- Program Risk Analysis
- Integrated Test Planning
- Technical Performance Measurement Planning
- Engineering Integration
- System Safety
- Human Factors Analysis
- Life Cycle Cost Analysis
- Manpower Requirements/Personnel Analysis
- Milestone Schedules.

The diagnostic capability review should address the impact of the results of the items listed above on the diagnostic pieces listed below.

- Designed-In Reliability, Prognostics, and Testability
- Self-Test, Built-In Test, System Integrated Test
- Support Equipment, Maintenance Aids
- Technical Data
- Personnel Skill Requirements
- Training and Training Devices.

System Design Review (SDR)

This review shall be conducted to evaluate the optimization, correlation, completeness, and risks associated with the allocated technical requirements. Also included is a summary review of the system engineering process which produced the allocated technical requirements and of the engineering planning for the next phase of effort. Basic manufacturing considerations will be reviewed, and planning for production engineering in subsequent phases will be addressed. This review will be conducted when the system definition effort has proceeded to the point where system characteristics are defined and the Configuration Items (CI) are identified.

Specific diagnostic considerations relate to:

- o Optimizing the diagnostic capability (changes after Dem/Val usually are more costly and time consuming)
- o Preparation of a Maturation Plan
- o Preparation of a System Specification which provides a capability for addressing 100% FD/FI for each level of maintenance
- o Allocation of diagnostic requirements for each diagnostic element
- o Review of the software requirements specification to assure that embedded diagnostic software considerations are included.

Preliminary Design Review (PDR)

This review shall be conducted for each Configuration Item or aggregate of Configuration Items to: (1) evaluate the progress, technical adequacy, and risk resolution (on a technical, cost, and schedule basis) of the selected design approach; (2) determine its compatibility with performance and engineering specialty requirements of the Hardware Configuration Item (HWCI) development specification; (3) evaluate the degree of definition and assess the technical risk associated with the selected manufacturing methods/processes; and, (4) establish the existence and compatibility of the physical and functional interfaces among the Configuration Item and other items of equipment, facilities, computer software, and personnel. For CSCIs, this review will focus on: (1) the evaluation of the progress, consistency, and technical adequacy of the selected top-level design and test approach; (2) compatibility between software requirements and preliminary design; and, (3) on the preliminary version of the operation and support documents.

In addition, the following items in the diagnostic area should be presented at the appropriate depth for review.

- o Preliminary Failure Modes and Effects Analysis
- o Design data analyses for BIT/SIT integrated diagnostics, including requirements and preliminary design verification results
- o Maintenance concept for the portion of the system being reviewed and its traceability to the system maintenance concept
- o Operational maintenance functions
- o Results of the analysis of the inherent (intrinsic) testability of the preliminary design
- o Allocation of qualitative and quantitative requirements
- o Criteria for external diagnostic elements
- o Trade-off studies
- o Cost/System Effectiveness Modeling and Life Cycle Cost Analysis
- o Preliminary Logistic Support Analysis, including task analysis and related personnel and support equipment information
- o Evaluation of alternatives
- o Test and evaluation plans.

Critical Design Review (CDR)

This review shall be conducted for each Configuration Item when detail design is essentially complete. The purpose of this review will be to: (1) determine that the detail design of the Configuration Item under review satisfies the performance and engineering specialty requirements of the HWCI development specifications; (2) establish the detail design compatibility among the configuration and other items of equipment, facilities, computer software and personnel; (3) assess Configuration Item risk areas (on a technical, cost, and schedule basis); (4) assess the results of the producibility analyses conducted on system hardware; and, (5) review the preliminary hardware product specifications. For CSCIs, this review will focus on the determination of the acceptability of the detailed design, performance, and test characteristics of the design solution, and on the adequacy of the operation and support documents. The CDR shall be conducted on each Configuration Item prior to fabrication/production/coding release to ensure that the detail design solutions, as reflected in the Draft Hardware Product Specification, Software Detail Design Document (SDDD), Data Base Design Document(s) (DBDD), Interface Design Document(s) (IDD), and engineering drawings, satisfy requirements established by

by the Hardware Development Specification and Software Top-Level Design Document (STLDD). The CDR shall be held after the Computer Software Operator's Manual (CSOM), Software User's Manual (SUM), Computer System Diagnostic Manual (CSDM), Software Programmer's Manual (SPM), and Firmware Support Manual (FSM) have been updated or newly released.

It is desired at each CDR to provide as much assurance as practicable that all diagnostic requirements are satisfied: I. e., that 100% diagnostic capability will exist for each CI in the system. While it probably will not be practicable to certify that this will exist, the following data should be presented as an extension of the information presented at the PDR.

- Detailed fault detection/fault isolation analyses that identify the extent to which BIT/SIT detect and isolate faults and which identify those failures that will require support equipment and/or manual methods to detect and/or isolate.

- Diagnostic allocations in Part II CI specifications to the LRU and SRU level. Traceability of these requirements to the Part I CI System Specification should be demonstrated. Note: Flexibility to reallocate diagnostic allocations until product baseline is established at PCA should be provided within the envelope of system requirements.

- Definition of the maintenance plan/concept for the CI, together with supporting LSA documentation, including support requirement and level-of-repair analysis results. Logistic simulation results should be presented to substantiate the plan/concept.

- Presentation of testability analysis/assessment results for the CI design to substantiate the fault detection/ fault isolation analysis.

- Early program failure identification, prevention, and detection analyses applicable to the CI should be presented to assist in verifying diagnostic capability.

- Review detailed Maintainability Demonstration Plan for inclusion of diagnostic capability test requirements

- Appropriate updates to the items reviewed during the PDR.

Test Readiness Review (TRR)

This review is conducted for each CSCI to determine whether the software test procedures are complete and to assure that the contractor is prepared for formal CSCI testing. Software test procedures are evaluated for compliance with software test plans

and descriptions and for in accomplishing test requirements. At TRR the contracting agency also reesults of informal software testing and any updates to the operation and support s. A successful TRR is predicated on the contracting agency's determination thware test procedures and informal test results form a satisfactory basis for proceformal CSCI testing.

The diagnostic a the system/CI TRR(s) shall be a formal review of the contractor's readiness to tal diagnostics-related CSCI testing. It is conducted after the software test proce available for diagnostics-related CSCI, such as CI BIT, System BIT, SIT, eter computer system component (CSC) integration testing is complete.

The items to be nclude:

1. Requirement --

   Any changeSIT, or testability requirements contained in the system/CI Sequirement Specification or Interface Requirements Specification not been approved and which impact CSCI testing.

2. Design Chan

   Any changes the BIT, SIT, or testability design parameters contained in tre Top-Level Design Document (STLDD), Software Detail Designt (SDDD), Interface Design Document(s) (IDD) since the PDR and h impact CSCI testing.

3. Software Testd Descriptions --

   Any changesbedded diagnostic element portion of the approved Software TestTP) and Software Test Descriptions (STD).

4. Software Testes --

   Test proceduused in conducting BIT and/or SIT test effectiveness validation as e CSCI testing, including retest procedures for test anomalies anens.

5. Integration Te Procedures, and Results --

   Any embeddostic element CSC (e. g., BIT components, SIT componente)on test cases, and procedures used in conducting informal diagment CSC integration tests and the test results.

6. Software Test Resources --

   Status of any software test resources that are required specifically for embedded diagnostic element CSCI testing. Such resources may include diagnostic test personnel and supporting test software and materials, including software test tool qualification and review of the traceability between requirements and their associated tests.

7. Test Limitation --

   Identification of all software test limitations associated with embedded diagnostic element CSCI/CSC testing.

8. Software Problems --

   Summary of embedded diagnostic element software problem status, including all known discrepancies of the CSCI and test support software.

9. Schedules --

   Schedules for the remaining embedded diagnostic element software milestones.

Production Readiness Review (PRR)

This review is intended to determine the status of completion of the specific actions which must be satisfactorily accomplished prior to executing a production go-ahead decision. The review is accomplished in an incremental fashion during the Full-Scale Development Phase--usually two initial reviews and one final review, to assess the risk in exercising the production go-ahead decision. In its earlier stages, the PRR concerns itself with gross-level manufacturing concerns, such as the need for identifying high-risk/low-yield manufacturing processes or materials or the requirement for manufacturing development effort to satisfy design requirements. The reviews become more refined as the design matures, dealing with such concerns as production planning, facilities allocation, incorporation of producibility-oriented changes, identification and fabrication of tools/test equipment, long-lead item acquisition, etc. Timing of the incremental PRRs is a function of program posture and is not specifically locked into other reviews. The diagnostic consideration concerns the use of any of the external diagnostic elements (e.g., ATE) in the production testing environment.

Functional Configuration Audit (FCA)

This is a formal audit to validate that the development of a Configuration Item has been completed satisfactorily and that the Configuration Item has achieved the performance and functional characteristics specified in the functional or allocated

configuration identification. In addition, the completed operation and support documents shall be reviewed.

The FCA is normally conducted on a prototype or preproduction item. The FCA validates that the item meets its specified performance requirements and is ready for production and acceptance into Air Force inventory. It is imperative that the diagnostic capability be validated against its specified performance requirements, so that any diagnostic capability deficiencies can be identified and corrected before the item proceeds into production and is then deployed.

Physical Configuration Audit (PCA)

This is a technical examination of a designated Configuration Item to verify that the Configuration Item "as built" conforms to the technical documentation which defines the Configuration Item.

After successful completion of the audit, all subsequent changes to the diagnostic elements are processed by an engineering change action. The PCA also determines that the diagnostic element acceptance testing prescribed by the documentation is adequate for acceptance of the production units by quality assurance activities. The procedures for conducting a PCA are contained in MIL-STD-1521, Appendix H. Sample PCA Certification Attachment Checklists are contained in MIL-STD-1521, Appendix I.

CHECKLIST

- ☑ Are the designers included in the reviews and audits so they can challenge the design and assess risks?
- ☑ Are the diagnostic reviews held as an integral part of the prime system review, but in a timely manner that allows change (if necessary) in the diagnostic equipment or process?

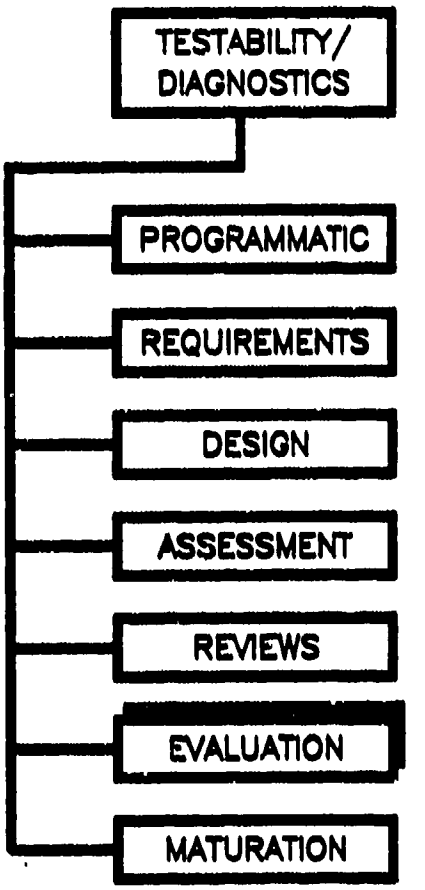
EVALUATION	REQUIREMENT #6

CONDUCTING TEST AND EVALUATION

OVERVIEW

During the development of a weapon system, a number of tests and evaluations are conducted by subcontractors, the prime contractor, and the government. Many of these tests address the performance of the diagnostic capability. It is not uncommon that these tests are conducted separately and, thus, do not address the entire diagnostic capability. Oftentimes the entire diagnostic capability is not delivered in time to test and evaluate the diagnostic capability as a whole. During the major tests and evaluations (e.g., DT&E, OT&E) as much as possible of the entire diagnostic capability should be included. Integrated demonstration, test, and evaluation is required.

Coordination of all test and evaluations, including demonstrations, can be accomplished through the preparation of an Integrated Test Plan.



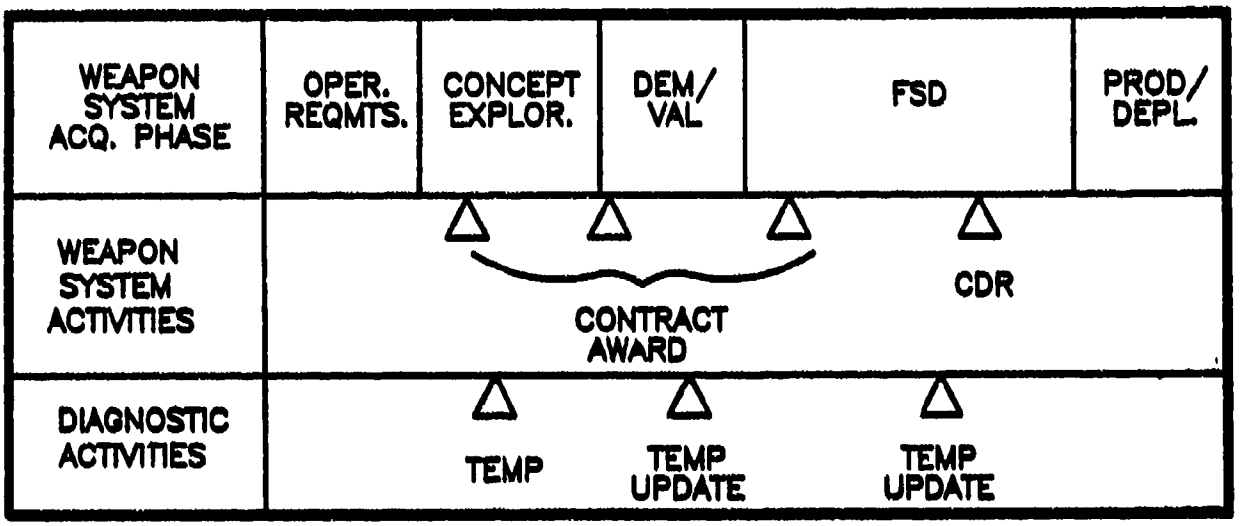
IMPORTANT CONSIDERATIONS BE ADDRESSED

Reqmt.

6.1 Provide input to the preparation of an Integrated Test Plan, which includes the requirements for a Test and Evaluation Master Plan.

6.2 Assure that formal test and evaluations address the entire diagnostic capability.

PREPARATION OF THE TEMP	REQUIREMENT #6.1



DIAGNOSTIC ACTIVITY

The requirements for diagnostics test and evaluation are identified, scheduled and integrated into the Test and Evaluation Master Plan (TEMP).

PROCEDURE

The TEMP is a living document normally prepared by the Contractor Program Manager. Its preparation goes through many iterations as the program proceeds through Concept Exploration Phase studies, Demonstration and Validation, Full-Scale Development,and Production. With each iteration, plans for diagnostic Test and Evaluation (T&E) should become firmer, better defined, and with target milestone dates attached.

Because test and evaluation is a major cost and schedule driver, adequate planning is essential long before it starts. Test planning between subcontractors, the prime contractor, and the government should start with program initiation. To ensure a successful integrated test program, close coordination is required between the government, the prime contractor, and all subcontractors. The designer should understand the scope and methods to be used in evaluating the product, and provide inputs to the TEMP, which promote realism in these tests.

PREPARATION OF THE TEMP	REQUIREMENT #6.1

GUIDANCE

DoD Directive 5000.3 requires the preparation of a Test and Evaluation Master Plan (TEMP). The TEMP is a broad plan relating test objectives to required system characteristics and critical issues, and is a top-level document used at major milestone reviews to assess the adequacy of planned test and evaluation. At minimum, it addresses both Development and Operational Test and Evaluation. It is important that as much as possible of the diagnostic capability be included in these T&Es.

Developmental Test and Evaluation (DT&E) is the T&E conducted throughout various phases of the acquisition process. This will ensure the acquisition and fielding of an effective and supportable system by assisting in the engineering design and development and verifying attainment of technical performance specifications, objectives and supportability.

Developmental Test and Evaluation also includes T&E of components, subsystems, preplanned product improvement (P^3I) changes, hardware-software integration and related software, as well as qualification and production acceptance testing. Test and evaluation of compatibility and interoperability with existing or planned equipment or systems is emphasized. This T&E encompasses the use of models, simulations and testbeds, as well as prototypes of Full-Scale Development models of the system. The diagnostic capability associated with component, assembly and subsystem DT&E should be included in these T&E activities.

Qualification Testing is the part of DT&E which verifies the design and the manufacturing process and provides a baseline for subsequent acceptance tests. This accomplishes two separate functions:

(1) Preproduction Qualification Tests are formal contractual tests that ensure design integrity over the specified operational and environmental range. These tests usually use prototype or preproduction hardware fabricated to the proposed production design specifications and drawings. Such tests include contractual reliability and maintainability demonstration tests required prior to production release. At a minimum, embedded diagnostics capabilities and the interfaces to external diagnostic elements should be tested and evaluated during preproduction qualification tests. As a goal, the capability of external diagnostic elements should also be tested and evaluated.

(2) Production Qualification Tests ensure the effectiveness of the manufacturing process, equipment and procedures. These tests are conducted on a sample lot taken at random from the first production lot, and are repeated as the process or design is changed significantly, and when a second or alternate source is brought on line. These tests are also conducted against contractual requirements. The utilization of diagnostic resources in the manufacturing process and the requirement for capture of diagnostic data from the manufacturing process should be evaluated during production qualification testing.

PREPARATION OF THE TEMP	REQUIREMENT #6.1

The completion of Preproduction Qualification Test and Evaluation before Milestone III decisions is essential and will be a critical factor in assessing the system's readiness for production.

Operational Test and Evaluation (OT&E) is the field test, under realistic conditions, of any item (or key component) of weapons, equipment or munitions for the purpose of determining the effectiveness and suitability for use in combat by typical military users; and the evaluation of the results of such tests. Operational testing is accomplished in an environment as operationally realistic as possible. The entire diagnostic capability should be assessed during OT&E as well as the integration of the diagnostic capability.

The TEMP must clearly specify development and operational test events. However, DT&E and OT&E are not necessarily serial phases in the evolution of a weapon system. During critical acquisition cycle transitions, elements of DT&E and OT&E may be combined or occur in parallel, but not at the expense of either development or operational test realism nor before sufficient DT&E can reasonably assure that the system is ready to enter dedicated operational testing. DT&E may continue into the Production and Deployment Phase, along with OT&E, to address system enhancements, correction of deficiencies, or modifications. In all cases, test planning for all test phases must be addressed in the system TEMP.

Test and evaluation planning is initiated at the inception of the development process to ensure adequate planning, programming and budgeting of test resources and to facilitate test scheduling to support major program decision milestones. Reliability assurance should be well underway before the initiation of system performance tests. System deficiencies must be addressed through a dynamic, well-documented, and tightly managed test-analyze-fix and retest program. The evaluation of embedded diagnostic elements should be injected into these reliability assurance tests.

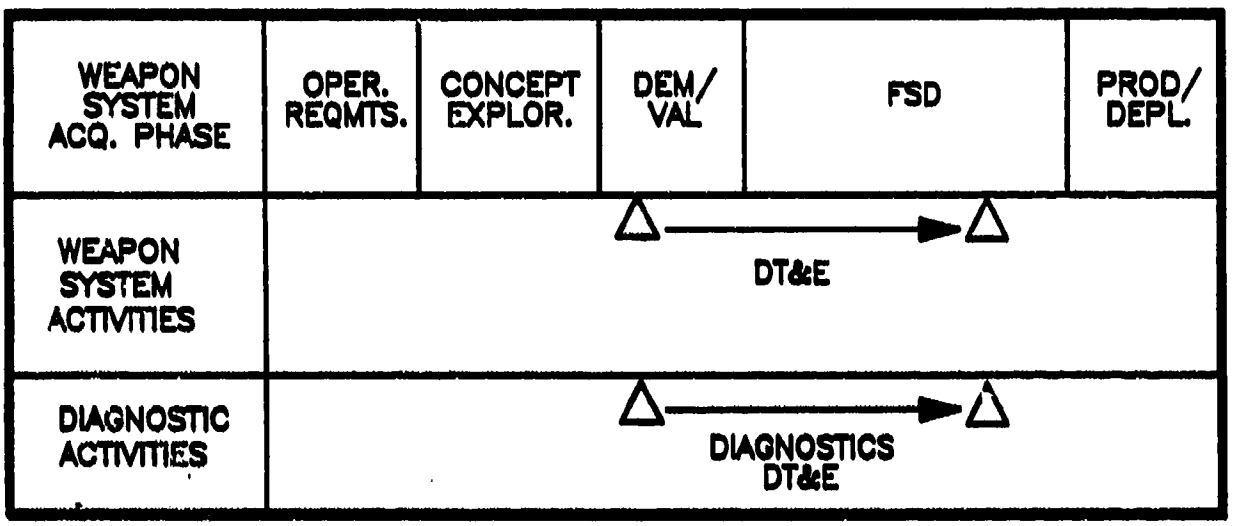
A TEMP is required for all major defense acquisition programs. The TEMP defines and integrates test objectives, critical issues, systems characteristics, responsibilities, resources and schedules for test and evaluation. Test resource requirements must be addressed in the TEMP, along with a critical analysis of any shortfalls that will impede the full test and evaluation of the system. The need for and the availability of the various diagnostic elements which make up the diagnostic capability is addressed in the TEMP. Plans to correct existing or anticipated test resource limitations are also included, as is a listing of evaluation criteria delineating critical parameters permitting continuous oversight and independent assessment.

DoD 5000.3-M-1 contains the guidelines for the preparation of the TEMP.

CHECKLIST

☑ Have T&E activities been realistically planned and scheduled to provide needed information on the performance of the entire diagnostic capability?

DEVELOPMENT TEST & EVALUATION	REQUIREMENT #6.2



DIAGNOSTIC ACTIVITY

Evaluate diagnostics performance characteristics during Development Test and Evaluation (DT&E) activities in order to determine diagnostic capabilities achieved and to identify deficiencies in the diagnostic capability. Diagnostics DT&E should also attend to the capability achieved by the integration of the various planned diagnostic elements (performance monitors, BIT/SIT, testing (automatic and manual), maintenance aids, technical information and training (skills)) into a comprehensive, cohesive, diagnostics subsystem.

PROCEDURE

Development Test and Evaluation is the T&E conducted throughout various phases of the acquisition process to ensure the acquisition and fielding of an effective and supportable system by assisting in the engineering design and development and verifying attainment of technical performance specifications, objectives and supportability.

Development Test and Evaluation also includes T&E of components, subsystems and preplanned product improvement (P^3I) changes, hardware-software integration and related software, as well as qualification and production acceptance testing. Test and evaluation of compatibility and interoperability with existing or planned equipment and systems is emphasized. Development Test and Evaluation encompasses the use of models, simulations, and testbeds, as well as prototypes or Full-Scale Development models of the system.

DEVELOPMENT TEST AND EVALUATION	REQUIREMENT #6.2

The designer should be as actively involved in diagnostics DT&E to ensure that valid tests are devised and performed, valid results documented, and valid data accumulated and to ensure that a closed-loop analytic approach is used to pinpoint and correct diagnostic deficiencies. The designer should ensure that every opportunity is being taken to evaluate diagnostics-related parameters. This may involve a wide range of test activities, including reliability tests, performance tests, human factor tests, etc. Basically whenever a system, subsystem or component is being operated, it is subject to a failure. The diagnostics requirements associated with dealing with the failure should be viewed as an opportunity to assess the diagnostic capability.

GUIDANCE

The thrust of the Integrated Diagnostics Process with respect to DT&E is to include/inject diagnostic performance evaluation into the mainstream of DT&E activities. This is done such that diagnostic performance can be evaluated, deficiencies pinpointed, and corrective action implemented while the system is still in development.

The diagnostics DT&E effort assists the diagnostic design and development process, and verifies attainment of diagnostic technical performance specifications, requirements, and objectives. As such, it is an integral part of the weapon system design process. Through the provision of diagnostics DT&E data, there is a feedback reiterative loop back into the integrated diagnostics activities in process, including the diagnostic system engineering analysis; diagnostic risk analysis; allocation of diagnostic goals; diagnostic trades for system optimization; diagnostic design trades; and, the identification and performance of diagnostic design tasks. Through this methodology, the diagnostic design is corrected, improved, or updated, and the diagnostic design matures.

Sufficient diagnostics DT&E must be accomplished before the Milestone III decision to proceed to production. This will ensure that the major specified diagnostics design and development requirements for the Full-Scale Development Phase have been met, with respect to performance requirements and specifications contained in program documents.

The scope of diagnostics T&E should include fault detection, isolation accuracy and timeliness provided by performance monitoring, BIT/SIT, automatic and manual testing, technical information and maintenance aids, maintenance procedures, manpower, personnel and skill levels at the system, subsystem, LRU/LRM, SRU levels across planned maintenance echelons (Organizational, Intermediate and Depot).

Any deviation from this full scope of T&E means that full confidence cannot be ascribed to the planned diagnostic capability.

The major approaches of DT&E for diagnostics include actions :

- To proceed in phase with the system and support equipment development, so that Built-in Test (BIT) is tested and evaluated concurrently with system performance; BIT and System Integration Test (SIT) tested and evaluated concurrently with subsystem integration and system testing; and, system integration and safety testing are concurrent with diagnostic testing of BIT and SIT features.

- To implement with the Diagnostics Maturation Program so that deficiencies, ambiguities, and additional failure modes identified during DT&E are recorded in a timely manner to ensure traceability and appropriate corrections are made to the integrated diagnostic procedures.

- To evaluate embedded diagnostic design as a separate entity in order to assure that it has been incorporated adequately as part of the system design.

- To evaluate for 100% diagnostic capability in selected critical areas of system design using fault evaluation.

- To analyze the system design hierarchy of test tolerances (e.g., between system BIT and LRU and SRU-level BIT) in order to minimize false alarms.

- To complete feasibility DT&E on prototype and preproduction units in order to assess technical risks and develop solutions to remedy deficiencies.

During FSD, specific diagnostic capability segments of DT&E efforts include the following requirements:

- When available, ATE shall be evaluated for initial use supporting build and check-out of systems. Manual procedures and associated operational prototypes shall be developed for support of test activities.

- Engineering evaluation of the diagnostic elements capability at subsystem and system levels shall be conducted in concert with system integration testing activities, including evaluation tests in the engineering laboratory and system integration test facilities.

- Effective development of a diagnostic capability requires that testing of diagnostic capabilities proceed concurrently with prime and support equipment development in an orderly and planned time-phased manner. The object of the following diagnostics testing approach is to provide a viable Organizational- and Intermediate-level diagnostic capability for use in support of flight and operational testing activities to provide for early maturation of the diagnostic capability. It should also be a program objective

to validate the diagnostic capability, as well as initial reliability and maintainability requirements before production.

- During early equipment development tests, built-in test features should be tested and evaluated concurrently with equipment performance testing. BIT performance is just as essential to overall weapon system performance as the usually emphasized aspects of equipment performance. Simulated equipment failures should be used to assist in BIT testing and evaluation.

- As equipment progresses to subsystem integration and performance testing, BIT and System Integrated Test (SIT) features should be concurrently tested, evaluated, and corrected. Simulated or emulated equipment failures should again be used for BIT/SIT testing and evaluation.

- System integration and safe-for-flight testing of equipment should include diagnostic testing of BIT and SIT features to assure readiness of this capability for Flight Test Support. Concurrently, Organizational-level support equipment required for diagnostic support should be tested to enable its use in the test program, together with Preliminary Maintenance Manuals for Initial Operational Test and Evaluation. Simulation of equipment failures to evaluate diagnostic capabilities should be included in this testing effort.

- Qualification testing of both prime and support equipment shall include validation of diagnostic capability, which is a required aspect of both equipment and system performance. Simulated equipment failures should be included in the diagnostic validation test program. Evaluation of BIT/SIT should also be conducted during environmental extreme testing of the prime equipment and support equipment, to assure its proper functioning throughout the required equipment performance envelope.

DEVELOPMENT TEST AND EVALUATION | REQUIREMENT #6.2

CHECKLIST

- ☑ Does the Integrated Test Plan provide adequate detail concerning specific T&E procedures, data bases, models, test articles and scope of testing?
- ☑ Have critical or high risk items related to diagnostic capability been identified and highlighted?
- ☑ Are the necessary test articles available to conduct realistic, timely tests?
- ☑ Has every opportunity to evaluate diagnostics during DT&E activities been identified?

OPERATIONAL TEST AND EVALUATION	REQUIREMENT #6.3

WEAPON SYSTEM ACQ. PHASE	OPER. REQMTS.	CONCEPT EXPLOR.	DEM/ VAL	FSD	PROD/ DEPL.
WEAPON SYSTEM ACTIVITIES				△ IOT&E	△ FOT&E
DIAGNOSTIC ACTIVITIES				△ DIAGNOSTICS OT&E	△

DIAGNOSTIC ACTIVITY

Diagnostic performance characteristics must be evaluated in a realistic operational environment during Operational Test and Evaluation (OT&E) activities in order to determine diagnostic capabilities achieved and to identify deficiencies in the diagnostic capability. Diagnostics OT&E should focus on the capability achieved by the integration of the various planned diagnostic elements into a comprehensive, cohesive diagnostics subsystem.

PROCEDURE

Operational Test and Evaluation (OT&E) is the field test, under realistic conditions, for the purpose of determining the effectiveness and suitability of the system or equipment for use in combat by typical military users; and the evaluation of the results of such tests.

GUIDANCE

Operational Test and Evaluation (OT&E) activities include Initial OT&E (IOT&E) and Follow-on OT&E (FOT&E). The results of DT&E activities should be analyzed by the Contractor Program Manager with help from the designer to ensure consistency and continuity of T&E activities. Operational Test and Evaluation (OT&E) must be accomplished by a separate government facility prior to the Milestone III decision. Diagnostics OT&E is performed to provide a valid estimate of the operational effectiveness and suitability of the system's integrated diagnostics design and procedures using test items sufficiently representative of the expected production items.

Major approaches to diagnostics OT&E include:

- o Testing in an environment as operationally realistic as possible
- o OT&E initiated as early as possible during the FSD Phase
- o Testing for adherence to overall OT&E objectives, with respect to diagnostics
- o Continued coordination with the Diagnostics Maturation Program
- o Evaluation for 100% diagnostic capability in selected critical areas
- o Random diagnostics testing in noncritical areas
- o Further analysis of test tolerances related to the system hierarchy and embedded/external diagnostic procedures in order to minimize false alarms.

Testing (particularly operational tests) and data collection should focus on the diagnostics requirements. Testing and data collection should also evaluate the specified parameters; namely, identification of critical failures, the false alarm rate, the percentage of faults detected and isolated automatically or manually, associated repair times, the unnecessary removal rate, consistency of test results, and the adequacy of personnel skills considering all maintenance incidents.

During OT&E, system performance, operational suitability and supportability factors are evaluated in an operationally realistic environment. There are two types of information that can be obtained for Diagnostics T&E: 1) faults within the system and how those faults were identified (diagnosed); and, 2) faults/deficiencies within the diagnostic capability. For the latter, this includes evaluation of each element which contributes to the total diagnostic capability, as well as the capability, achieved by integration of the diagnostics elements. Focused, detailed T&E activities discussed in Requirement # 6.2 should be continued. The former type of data can be obtained as a result of Reliability Growth Testing. The following specific information should be evaluated for each fault occurrence.

1. How did the failure manifest itself?
2. Was the manifestation due to stressing of the system beyond normal operational limits?
3. If a BIT alarm occurred, was it the result of a confirmed failure?
4. What techniques were used to isolate the fault?

5. How long did fault isolation take using those techniques?

6. Was the failure mission or operation critical?

7. Was it a new or unplanned failure mode? Was BIT supposed to detect the failure? Did it?

8. Is this failure mode expected to be encountered in the operational system?

9. Should provisions be included in the diagnostic capability to deal with this failure mode?

10. Will this involve a modification/addition to BIT? ATE? Manual Test Equipment? Maintenance Procedures? Skill Levels? Technical Data? Maintenance Aids?

11. Is an ECP required?

12. Is further investigation required?
    If yes - What plans have been made?
    If no - Why not? (brief description)

13. Is correction of the diagnostic deficiency part of contractual requirements? Tied to incentive or warranty provisions?

OPERATIONAL TEST AND EVALUATION	REQUIREMENT #6.3

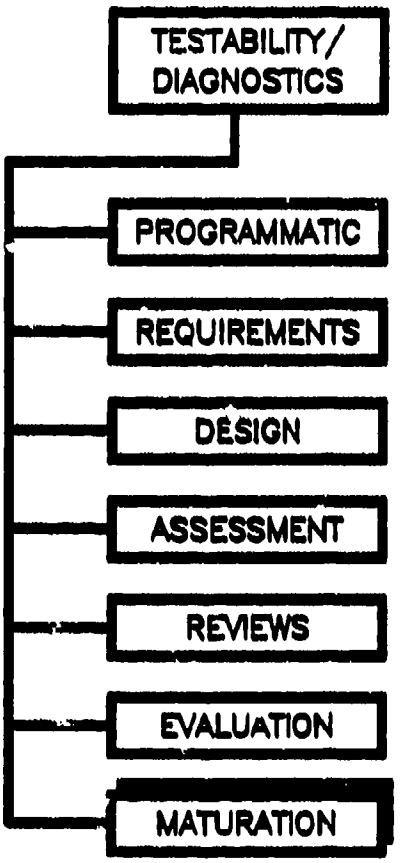
CHECKLIST

☑ Is the designer giving adequate support to OT&E activities?

MATURATION OF THE DIAGNOSTIC CAPABILITY

OVERVIEW

Historically, often a weapon system's diagnostic capability does not meet its performance requirements prior to deployment. The basic reason for this is that all faults cannot be predicted and, thus, adjustment of the diagnostic capability is required during the first few years after deployment. Essentially, this requires a well-planned maturation period, which allows for the growth of the diagnostic capability. Closely coupled with this maturation is the requirement for collection and analysis of data relating to the performance of this diagnostic capability, both in the field and in the factory. Care must be exercised by both the government and the contractor to assure that proper and detailed data is collected. Early planning for this maturation period is a must.



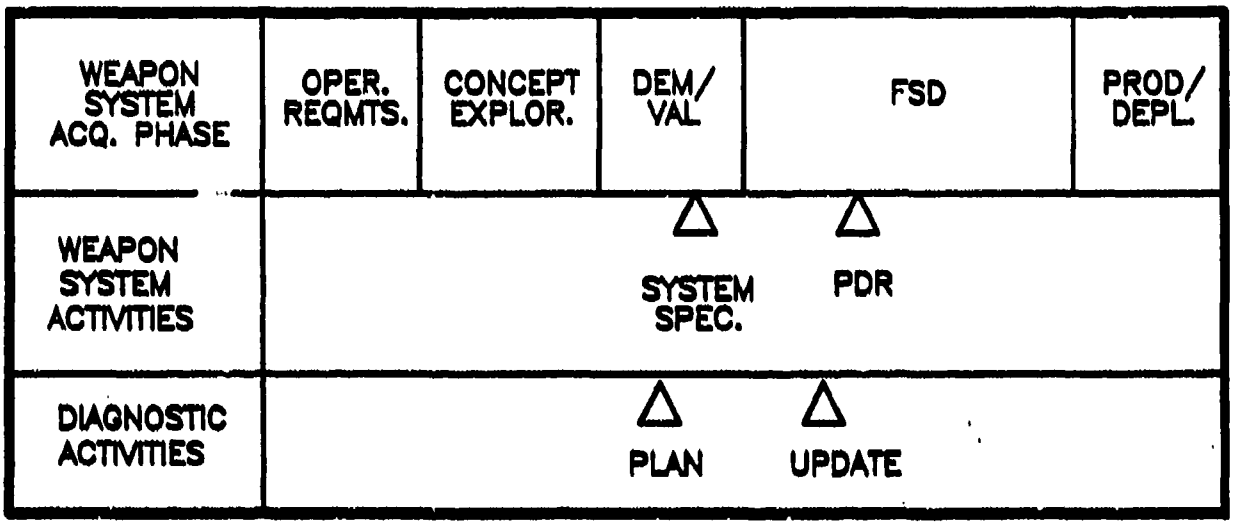
IMPORTANT CONSIDERATIONS TO BE ADDRESSED

Reqmt.

7.1 **A detailed Diagnostics Maturation Plan needs to be prepared early in the acquisition process.**

7.2 **A diagnostic data collection and analysis system must be established to provide for corrective measures.**

MATURATION PLANNING	REQUIREMENT #7.1



DIAGNOSTIC ACTIVITY

Most diagnostic implementations, no matter how well conceived, require a period of time for identification of problems and corrective action to reach specified performance levels. This requirement is established in order to formalize the diagnostics maturation and to allow the maturation to be initiated early in the test and evaluation process. This requirement is initiated early so that early identification, tracking, and correction of diagnostic problems are achieved. The planning for this activity is formalized by the development of a Diagnostic Maturation Plan or other appropriate document.

PROCEDURE

While it is the Contractor Program Managers responsibility to prepare the Diagnostic Maturation Plan, the designer should understand the scope and methods to be used in maturing the diagnostic capability to assure adequate corrective actions are planned and implemented.

The Contractor Program Manager must ensure that the plan is:

1. Comprehensive

   - o Across all diagnostic elements
   - o Includes the integration of the elements

2. Timely

- o Is initiated early to plan for the required resources and implement corrective actions
- o Maturation is completed by Milestone IV, per DoD-Instruction-5000.2

3. Coordinated

- o Includes coordinated activities from the "ilities"
- o Utilizes standard data collection systems

4. Cost Effective

- o Allows data collection to be transferable and usable by government (i.e., DT&E and production test data).

GUIDANCE

A program to mature the diagnostic capability should be planned for the early fielded production systems. A one-to-three-year maturation program should be planned for complex weapon systems with extensive automatic testing capability. For major weapon systems, the coordination with Milestone IV, Logistic Readiness and Support Review (DoD-Instruction-5000.2) is essential. This program should include provisions for on-site collection of diagnostic performance data, with engineering follow-up to provide corrective actions.

The plan should define an approach and methodology to assure that as development, test and evaluation, and early operational use of the system progress, problems presented by new failure modes, test voids, ambiguities, and test tolerance difficulties are recognized and defined, and their solutions are traceable to diagnostic software and manual procedure updates. The plan should recognize that such occurrences are expected and normal and, therefore, should concentrate on problem recognition, definition, and correction, with appropriate tracking for traceability.

The approach and methodology defined should recognize that a basic element of the integrated diagnostics concept is identification of the set of faults which are known or expected to occur. The methodology shall provide for definition of this set, initially through Failure Modes and Effects Analysis, Testability Analysis, and other tools and experience. Provision for growth of this set, as new failure modes are encountered during testing and deployment, should be incorporated in the plan, together with explicit criteria to be used in deciding whether or not a newly encountered fault shall be added to the set of faults for which explicit diagnostic procedures (as opposed to more general procedures) are provided for detection and isolation of the fault. The life cycle cost effectiveness of adding explicit diagnostic procedures for the newly encountered fault shall be one factor considered in the decision.

MATURATION PLANNING	REQUIREMENT #7.1

The plan should provide for an orderly development and maturation process for diagnostic software and manual procedures throughout the development, test and evaluation, and early operational use time periods of the weapon system and its subsystems. Methodology to assure timely and continuing technical support for this maturation process by both contractor and government activities, with a minimum of administrative delays, should be a feature of the plan. Orderly transition of technical responsibilities from the contractor to the government should also be addressed.

The plan should present milestones to be met. This will assure that the final system achieves the required degree of diagnostic capability. The plan shall show the time phasing of each task and its interrelationship with other tasks. It should identify required data review, verification, and utilization to accomplish the required tasks and to report progress, problems, and tradeoffs. The plan should assure the proper implementation of diagnostic design features by designers and subcontractors.

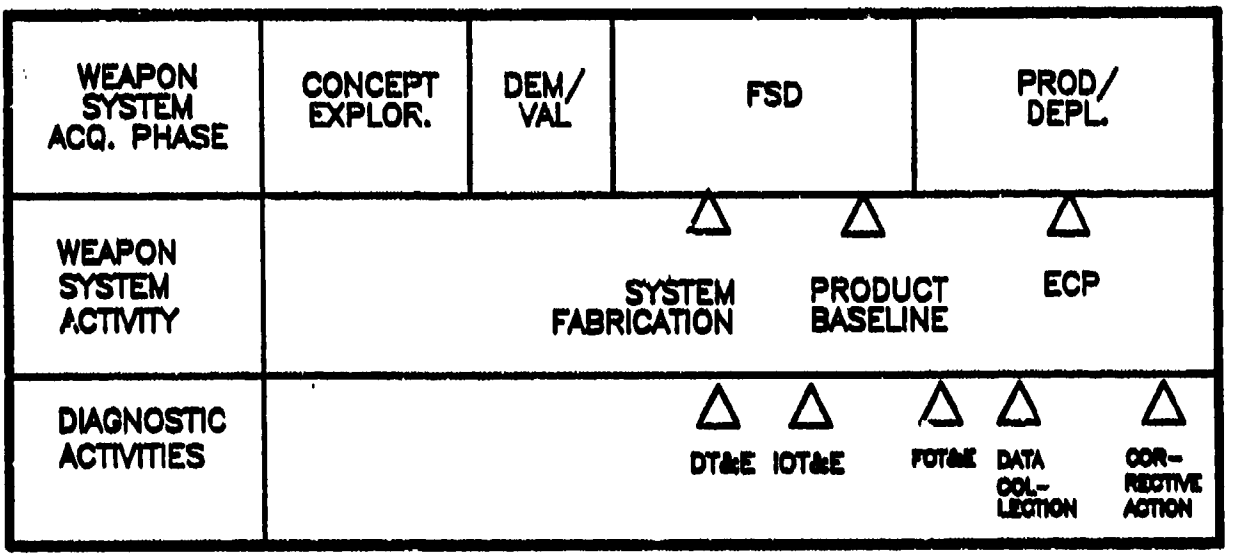
During the Dem/Val Phase, maturation planning is centered on preliminary planning for data collection, analysis and coordination with similar requirements for reliability, maintainability, logistics, data collection, analysis systems, etc. Specifically, this planning should identify potential data sources, such as:

- o Laboratory testing
- o Developmental testing
- o Operational test and evaluation
- o Acceptance testing
- o Preproduction testing
- o Production testing
- o Operation.

CHECKLIST

- ☑ Does the Diagnostics Maturation Plan include a strategy for the collection of diagnostic performance data through DT&E, OT&E, Production, Initial Operational Use, and Deployment?
- ☑ Is the diagnostic data collection plan in sufficient depth to allow adequate evaluation of diagnsotic capability?
- ☑ Does the plan include provisions for all diagnostic elements -- embedded and external -- as well as the integration of the diagnostic elements?
- ☑ Is the integration of the diagnostic elements planned for early enough to allow evaluation and cost-effective corrective action (e.g., prior to production go-ahead)?
- ☑ Does Maturation Planning include provisions for both:
  1. Adequacy of the diagnostic elements, with respect to the specified allocated capability, and
  2. Unplanned failure modes, which may arise throughout OT&E, DT&E, Production Test, and Field Use Test?

DATA COLLECTION AND FEEDBACK	REQUIREMENT #7.2



DIAGNOSTIC ACTIVITY

Data relating to the performance and effectiveness of the diagnostic capability must be collected during development, production, and operation. This data is used as the basis for the evaluation of diagnostics and for the correction of deficiencies.

The key thrust of this activity is definition of appropriate data to be collected, maximum use of data collected, coordination of data collection systems, and a structured approach to corrective action.

PROCEDURE

The Contractor Program Manager is responsible for the implementation of diagnostic data collection and feedback requirements. This includes development and implementation of a cradle-to-grave system for both contractor and government use. It is the designer's job to make sure these design corrections are implemented.

The earlier diagnostic performance deficiencies are identified, the sooner a more cost-effective solution can be implemented. Therefore, diagnostic data collection and feedback is initiated early in the test and evaluation process, continues through production test, and extends into the operational environment. Throughout these phases, different types of data are collected, different data collection procedures and methodologies are used, and different types of analysis technique are conducted.

GUIDANCE

There are no standard methods for data collection and analysis. As indicated under Requirement #7.1, Maturation Planning, the collection of this type of data is controlled by a number of military standards. The requirements for the standards which deal with logistics, reliability, maintainability, testability, human engineering, and safety overlap one another (many times data required by one may, indeed, be required by the other(s)). Thus close coordination among these various data requirements is needed. A single data base is desirable. Some tools are available to assist in the feedback and analysis task. These descriptions are contained in Appendix C.

The data collection procedures closely follow the test and evaluation functions. As explained in DoD Directive 5000.3, Test and Evaluation, the time periods and sequences for Development Test and Evaluation and Operational Test and Evaluation vary from program-to-program. They can overlap and even be done as a combined test and evaluation. Thus there are no standard guidelines that specify the exact points in the weapon system acquisition phase where data is to be collected. The system must be flexible to incorporate data as data is generated.

The Contractor Program Manager should ensure that the proper data is collected and that corrective actions are pursued. It is the job of the designer to make these corrections. Care must be taken to collect only that data required to assure that the diagnostic capabilities are performing as required. Automated data collection systems can be employed. Usually these are more effective, as they are less dependent on human motivation to supply the required information.

Corrective analysis and actions should be in a closed-loop system, so that each deficiency identified remains an open item until it is formally documented as being corrected.

The data collection and feedback system should be designed so that specific information is collected on the performance of the entire diagnostic capability, as well as for each of the diagnostic elements that make up the diagnostic capability. The information must be collected in quantitative form, if possible, and related to System Specification requirements. Thus the following guidelines on the type of data to be collected need to be tailored so that the information can be related to System Specification requirements and so that it is clearly apparent who is to supply the information and when this information is to be supplied. Examples of the type of data to be collected follow.

Diagnostic Data Feedback

- o Effectiveness and efficiency of each diagnostic element
- o Effectiveness and efficiency of the diagnostic elements as an integrated system
- o Operational/support impact of the diagnostic deficiencies
- o Corrective action(s) which should be taken or have been taken.

BIT Effectiveness

- o Fault isolation time.

Tracking of False Alarms

- o Type of alarm
- o Frequency of alarm occurrence
- o Cause (if known)
- o Potential consequences of ignoring the alarm (crew safety, mission reliability)
- o Operational costs of responding to false alarms (aborted missions, degraded mode operation, system down time)
- o Support costs associated with the false alarm
- o Operational environment when alarm occurred.

ATE Effectiveness Feedback

- o Workarounds required to overcome mechanical or electrical deficiencies in the UUT/ATE interface
- o Did the ATE system provide failure detection results consistent with those of initial failure detection by BIT?
- o Were the ATE test results repeatable?
- o Ambiguity size

- Fault isolation time.

Integration of Diagnostic Elements

- Are diagnostic resources provided consistent with the training/skill levels of assigned personnel?
- Effect of false alarms and unnecessary removals on operational availability and maintenance workload
- Shop throughput
- Wrong or inadequate technical information
- Logistic delay time
- BIT reliability
- ATE reliability.

Diagnostic data collection and diagnostic capability performance assessment may lead to the requirement for corrective action. Corrective action may involve redesign of prime equipment, test equipment, interface devices, maintenance documentation, built-in test circuits, diagnostic software, and ATE test programs. All changes must be made under strict configuration control.

The designer must recognize that modifications to the prime system/equipment may dictate modifications to the diagnostic capability as well.

CHECKLIST

- ☑ Is there direct communication back and forth between the person who reports a problem and the person who will be correcting the problem?
- ☑ Are all failures being analyzed to sufficient depth to identify failure causes and perform necessary corrective actions?

Table of Contents

1.0	INTRODUCTION	3
2.0	ESTABLISHMENT/INTERPRETATION OF REQUIREMENTS	4
3.0	STRUCTURING DESIGN CONCEPTS/CONSTRAINTS	7
4.0	MEANINGFUL PREDICTION AND ASSESSMENT METHODOLOGY	11
5.0	DESIGN REVIEWS	12
6.0	DEMONSTRATIONS	13
7.0	MATURATION	14
8.0	SUMMARY	16

1.0 INTRODUCTION AND BACKGROUND

The design engineer is the key to solving one of the most complex problems facing the military services in years. The problem is that today's weapon systems have become so sophisticated that the capability to maintain and repair them is a national priority. No longer can the design engineer be primarily concerned with the equipment meeting the operational requirements. Reliability and maintainability must be given equal consideration. Consideration can no longer be given only to varying environmental conditions under which the fielded product must operate. Equal consideration must now be given to the conditions under which repairs will be performed. Seemly short and simple tasks often become very time consuming when accomplished under extreme temperature conditions or in restrictive clothing such as chemical, biological or radiological attire.

The services, burdened with excessive maintenance problems, increasing demand for skilled manpower and skyrocketing costs, have given industry a clear message. The Air Force, for instance, has implemented a program which states basically that all new equipment will be designed to double reliability and reduce repair time by half. Reliability, maintenance, quality, and productivity in new equipment will be given as much attention as performance, program schedule and cost. The effects of this new program can be seen in recent development of a low-altitude navigational system. Performance at initial testing was with operational requirements. However, due to higher-than-anticipated vibration level, reliability requirements could not be demonstrated. The program was not permitted to continue to the next phase until reliability reached the required growth curves. This created a delay of approximately six months, placing the entire program in trouble.

Currently, diagnostics design is the major unknown in the reliability and maintainability arena. The statistics provided in this guide's introduction demonstrates the magnitude of this fact. This appendix presents additional experiences and the key learning points derived from them.

To introduce these lessons, a brief hypothetical scenario is provided regarding the start of a work day for an Air Force technician assigned to a modern bomber wing. This case is intended to provide some insight into what diagnostics programs may someday achieve.

Arriving at his duty station, the technician enters his code at a computer terminal and is provided a work order for the first task of the day. The work order concerns a malfunction which was detected during a flight completed just prior to his arrival at work. A quick glance at the work order reveals which system failed, what time it

occurred, and the Line Replaceable Unit (LRU) which is to be replaced to correct the problem. After a quick trip to a supply point for a serviceable LRU, with tool box and checklist in hand, he departs for the flight line. The defective LRU is changed within minutes after his arrival at work. A quick operational check, using the checklist and on-board test system, confirms that no other failures have occurred, and the system is declared operational.

Back at the intermediate shop, the flight line portion of the work order is closed out. This is quickly done, with a minimal amount of information input into the computer terminal regarding the work accomplished. The defective LRU is placed on its corresponding automatic test equipment (ATE). Keys within the LRU provided identification information to the computer contained within the ATE. Failure conditions and symptoms recorded on-aircraft at the time of the failure are also transferred to the ATE computer via the computer network. Rapidly the ATE goes through a set a tests specifically tailored to the reported failure conditions and the failed single component is identified. After the failed component is replaced, the LRU is checked again with the ATE to verify serviceability. Following serviceable testing, the LRU is given a quality control inspection and returned to the supply point, where it once again becomes a serviceable asset.

The above scenario (or parts thereof) has been a goal of the military services for many years. Great strides have been made toward achieving this objective, yet even total success in limited areas does not lay immediately at hand.

The reasons that success is fleeting are many. They include budget constraints, a relative lack of importance, political considerations, time, and the complexity of the task--just to mention a few. This appendix presents a few glimpses of activity on recent programs, results obtained, and lessons learned.

The information presented is a composite of experiences derived from B-1B and F-111 Aircraft, as well as the Minuteman, Peacekeeper, and Small ICBM Strategic Missiles. LSA examples from the AMRAAM and 30mm Gun PODS are also included.

2.0 ESTABLISHMENT/INTERPRETATION OF REQUIREMENTS

What is specified in the procurement specification and the contractual Statement of Work is what the government expects to receive. In the area of diagnostics, the government experience on past programs has not been the best. Diagnostic systems have proven to be incomplete, unable to to test to the desired level or simply do not as advertised. The basic foundation upon which any success will be directly dependent is the clear understanding of the actual requirements.

Need Statements and Work Statements

All of the programs surveyed in the preparation of this document seem to have an item in common dealing with their diagnostic requirements. That commonality factor is that the quantitative diagnostic requirements imposed are derived without a great deal of thought and analysis. Typically diagnostic requirements are more what has been judged by someone to be realistic values, rather than a product of studies performed to determine these requirements.

DoD-Instruction-5000.2 and other related documents describe a structured acquisition process beginning with, among other things, the development of a Mission Area Analysis and a Mission-Need Statement. Included in the Mission-Need Statement is a discussion of the Mission and Threat, Alternative Concepts, and Technology Involvement. Subsequently, during the Concept Exploration Phase, studies are conducted to develop a System Concept Paper which more thoroughly defines possible alternatives, and a selected concept. Many items are taken under consideration during this time frame including readiness, maintainability, manpower and training.

It is this process which generally drives the development of the procurement specifications. These functions are primarily the concern of Government Program Manager; however, inputs are sometimes requested from the contractor. Failure to consider testability when providing these inputs may limit chances for successful diagnostics later in the program. Overall diagnostics and testability, in general, should be given more concern at this early stage of development.

Understanding Requirements

Determining the proper diagnostic specifications necessary to meet the mission need is one thing. Describing them in such a way that they will be interpreted properly is another.

The following is one of the diagnostic requirements imposed on the B-1B. "The CITS shall provide an assurance of 95 percent to the aircrew that the system performance is operationally acceptable or that the indicated failure is valid during in-flight performance and ground readiness tests. The CITS shall provide fault isolation to an LRU with a certainty of at least 75 percent in the ground fault isolation mode."

Another requirement stated that "false alarms could not exceed 2 percent". Both seemingly good requirements, but two problems ensued in their accomplishment. First and foremost was the problem in the definition of the percent base. Percentages are

often used in defining requirements. But when so used, it must be stated as a percent of what. False alarms, as a percent of the possible alarms, gives one result. False alarms, as a percent of the number of total alarms indicated, gives another. When written, one must assume that achievement based on definition of the writer would meet mission needs. In reality, when achieved based on legal or implied definition, the results were far from those required by the operational command.

A second but in this case a lessor problem, was a conflict between the requirement. The first requirement above allows a 5 percent false alarm rate (100 minus the 95 percent accuracy). The second allows only 2 percent. Specification ambiguity leads to interpretation which will not necessarily end with the desired result. The design engineer can help eliminate these problems by informing program management when specification ambiguity is first encountered.

Logistic Support Analysis (LSA) Process

The LSA is not a direct function of the design engineer, however the process can influence many of the design requirements. MIL-STD-1388-1A defines basic objectives which are achieved when this standard is applied.

1. Cause supportability requirements to be an integral part of system requirements and design.

2. Define support requirements that are optimally related to the design and each other.

3. Define the required support necessary during the operational phase

These objectives are accomplished by the selective application of scientific and engineering efforts undertaken during the acquisition process, as part of system engineering. It is an iterative process of definition, synthesis, trade-off, test and evaluation. Many cases have been found which indicate that designers who successfully incorporate the results of these studies early will have a maintainable system when deployed in the field.

3.0 STRUCTURING DESIGN CONCEPTS/CONSTRAINTS

Controlling vs Constraining the Contractor

Today's trend in specification and contractor direction is to provide the contractors with the maximum leeway in meeting top-level requirements. The objective is to allow the contractor to define alternatives, select from the alternatives those which can best be accomplished and provide a product which meets all of the "real" requirements.

Existing systems covered by this document were all developed under a more structured specification approach. The previous school of thought was, generally speaking, that the more things which can be controlled by the specifications, the more chance the end product will be as desired. Experience with that approach has led to the more open trend. This is because the tighter approach did not allow the contractor to make maximum use of his many possible alternatives.

The customer is encouraging the design engineer to think in terms outside the realm of present diagnostics technology. Diagnostics technology in general has not developed to the point of satisfying the customer's requirements for maintaining complex weapon systems. An excellent example of the type diagnostics not desired is demonstrated by the built-in fault capability of a terrain-following radar. One of the line replaceable units contained within this system is the computer. Conveniently located in the aircraft nose compartment, this LRU contains the malfunction isolation switch and malfunction indicator on it's front panel. The malfunction indicator has nonvolatile flags set during flight at the occurrence of a malfunction. These were designed to serve as functional aids for maintenance personnel troubleshooting the terrain- following radar system. However, many of the malfunctions indicated are caused by associated subsystems that provide stimulus to the computer. This situation has caused unnecessary maintenance and supply cost, plus degraded operational readiness. The designer must ensure that all diagnostics will only be influenced by the parameter they are designed to monitor. The last thing the customer needs is a diagnostic system which increases the workload.

The Maintenance Concept

The Logistic Support Analysis tasks of MIL-STD-1388-1A which are concerned with the development of maintenance concepts and constraints are very important for the diagnostic community. The design engineer will benefit greatly by being involved with the LSA analyst and incorporating the results into the basic design at the earliest possible time. The MIL-STD-1388 tasks are structured to ensure consideration of existing

resources, compatibility with deployment and operational requirements, and maintenance personnel skill level.

The tie between the diagnostic method and the maintenance concept is bidirectional. They need to be established in unison. The maintenance concept is developed based on expected diagnostic capabilities. The diagnostic design ultimately forces the real maintenance concept. The designer who understands this concept recognizes the MIL-STD-1388 process as an aid to achieving the desired maintenance concept is success oriented. A lesson well learned is that when these tasks are used for historical purposes, instead of a tool for the designer, the desired maintenance concept is seldom achieved.

Established Air Force maintenance policies utilize system operation as the final determinant of the need for maintenance. If the system is functioning within tolerance, don't fix it. A unique situation has developed on the B-1B aircraft. Due to redundancies designed into the systems, overall operation appears normal, while some specific parts are not functioning. The diagnostics systems say the parts should be replaced. System operation indicates everything is functioning normally. To date, partially due to the lack of confidence in the aircraft diagnostics and partially due to established habits, these type malfunctions are not being repaired. Generally the diagnostics is believed faulty and no maintenance action is taken until another item malfunctions rendering the system inoperative. This experience shows that changing existing practices is slow. If it is confused with the lack of confidence in the diagnostics, the change is even more difficult.

The Unit Under Test (UUT) designer, ATE manager, and automatic test equipment designer are all vital elements in determining what off- equipment testing is required. Once the option for automated testing is confirmed, the ATE designers must convince the UUT designer to incorporate "Design To" criteria for maintainability, reliability and testability. Care must be taken to define the need for ATE, how the ATE is to be used, how the UUT will be designed for built-in test and interfacing abilities. ATE effectiveness is directly and immediately dependent upon this co-development with the unit under test. The following support trade-off factors must be considered when developing the UUT "Design To" criteria:

1. The maintenance concept and requirements imposed by the Repair Level Analysis of the system

2. Requirements of the built-in test for the UUT

3. The effectiveness of UUT functional partitioning

4. The ability to insert in the UUT test points

5. Design limits on reliability and maintainability.

Historically, probably the prime reason for dissatisfaction with a weapon system's diagnostic capability is the unnecessary removal of "good" items when conducting corrective maintenance. The designer must be aware of the causes for these unnecessary removals.

A field survey team (details of the survey are contained in RADC-TR-83-2 "Study of the Causes of Unnecessary Removals of Avionic Equipment") visited 12 AFB's in 1979 to determine the causes of this problem. When the survey was completed, a study analyzed the data and categorized the causes. The following major causes of unnecessary removals (URs) are listed along with the percentage of all URs for which they are responsible.

Ineffective BIT - 35%

This problem relates to built-in-test designs which provide incomplete or ambiguous information to aircrew and ground crew operators. Such incomplete information is the reason that operators must "interpret" BIT indications. Thus, there are instances when BIT indications are misinterpreted and an avionic equipment is erroneously reported as malfunctioning. Such "malfunctions" are termed false alarms and result in a CND or UR classification. These false alarms may either indicate a malfunction in a serviceable equipment when there is actually no malfunction in the system, or may not indicate a fault when one exists in the equipment.

Ineffective Supervision/Support - 25%

This problem involves control of the work habits of maintenance technicians. Although a lack of such support may be a result of the current short supply of middle management personnel, special attention of supervision is often necessary to maintain control of the UR rate.

Lack of adequate troubleshooting, incorrect use of test equipment, improper or inadequate documentation, and lack of historical tracking of aircraft and LRUs for intermittent problems all tend to point to the lack of effective direct supervision.

Management Directives - 11%

This problem relates to bypassing the normal standard troubleshooting procedures to obtain quick response turnaround times for priority sorties. There are times when turnaround time is most important and any supporting action is justified. However, this type of nonstandard action should be under regular surveillance by auditing personnel.

Test Equipment Differences - 10%

Test equipment differences between different levels of maintenance were noted by the survey team on relatively "new" equipment. A lack of commonality in the calibration of test equipment was also discerned by the field survey team at one repair facility. At one AFB, certain LRUs received from the repair depot are retested because of the lack of commonality between I-level test equipment and depot level test equipment.

Ineffective or Missing Test Equipment - 9%

This includes heavy or bulky test equipment. In most cases ineffective, heavy or unwieldy test equipment is the same as missing test equipment since it is not used. In this case, nonstandard troubleshooting is employed.

Inadequate Skill - 7%

Inadequate skill of maintenance technicians in the use of T.O.s, test equipment and troubleshooting procedures relates to the technicians' inability to completely cope with the relatively high technology of electronic equipment. This cause of URs is due to the technician not remembering details of his past training; be it formal, on-the-job training, technical readings or just familiarization with equipment and/or available diagnostic methods.

Inaccessibility

In addition to the above, the problem of inaccessibility cannot be overlooked. An inaccessibility problem can have a significant impact on the unnecessary removal rate. When LRUs are not readily accessible due to some restricted location, the removal of a suspect LRU may require the removal of one or more adjacent LRUs. Also, the difficulty in reaching a suspect LRU may preclude an on-equipment check, and the suspect LRU is removed and sent to the I-level shot for bench check.

4.0 MEANINGFUL PREDICTION AND ASSESSMENT METHODOLOGY

In-process assessment of diagnostics achievements has, in the past, been less than adequate. In fact, one of the most definitive and often repeated B-1B lessons is the need for an operational period to mature the diagnostic design. That lesson is described below in paragraph 7.0. Prediction and assessment techniques have, in the past, failed to provide sufficient information to uncover all of the inadequacies and shortcomings. Significant emphasis is currently being placed on testability analysis, reliability, and maintainability assessment tools under the umbrella of Computer-Aided Acquisition and Logistic Support (CALS). With that emphasis, one should expect great improvements in assessment techniques. The point for the design engineer on this subject is that the results of these predictions and assessments must be incorporated into the design so diagnostics requirements will be fulfilled.

Methodology

The CALS initiative would include diagnostics testing as an integral part of CAD design. The concept is that rules and techniques would be established in the CAD machine. As a specific item is designed, it is constantly checked for test access, built-in test capability, or whatever other rules that have been established.

This concept works fine for evaluating the diagnostic characteristics of a single electronic assembly. Evaluation of a weapon system's central test system is another question. For the B-1B, a complete integration lab was developed to test the diagnostics software in a functional environment. That process was useful, but still under the best of lab conditions some things could not be developed to the optimum level. An excellent example is the philosophy for checking the thrust of a jet engine. Simulated lab conditions equate more to an aircraft being on the ground. There thrust is compared to a reference schedule of gross thrust versus turbine blade temperature at two discrete operating points. These two points are the intermediate and maximum power settings. To develop an in-flight thrust check, a reference has to be calculated to monitor performance across the entire power range. This reference is obtained by comparing the engines in synchronization to one another in flight. This reference requirement, plus many preconditions necessary for calculating or examining thrust, dictated actual flight testing for development of a valid check.

Feedback Structure

Time is needed to assure that the design benefits form the assessment process. Logically, one does not need a whole lot of experience to understand this. However, it was proved once again on the B-1B aircraft that compressed schedules tend to eliminate this time. Concurrent Full-Scale Development and Production meant that the funding for studies and analysis occurred so late that results could not be implemented. When this happens, management direction is needed; however, management cannot take any action unless the problem is brought to their attention. The design engineer must notify management or the magnitude of the problem will increase with the passage of time.

Information Flow

A concern often expressed by many design engineers is the delay in receiving formal products generated with the MIL-STD-1388 process. Certainly this delay can create concerns. Who wants to think the design is stable only to discover major changes are required? The driving factors often necessitating these changes are studies performed for maintainability and testability. The design engineer who realizes this and develops a close working relationship with the personnel performing these studies will have fewer surprises. Experience has taught many times that it is much easier to communicate and incorporate changes during the initial design effort. Trying to make changes later is expensive, time consuming and often produces less than optimum results.

5.0 DESIGN REVIEWS

Formal Design Reviews provide the opportunity for the contractor to demonstrate to the customer the present design and what future design efforts hope to achieve. If the contractor can demonstrate that he is meeting the specifications, the customer can ask no more. It is the role of the design engineer to assure that sufficient design has been performed prior to Design Reviews, which can demonstrate with a degree of confidence, that diagnostic requirements are being fulfilled.

Scheduling

It's either too early or too late. Picking the optimum time for reviews is very important. Reviews need to be conducted after the design is sufficiently defined to make the evaluation but before it is too late to make design changes. The design engineer

needs to participate in schedule development to ensure that a reviewable product is available at the scheduled time.

The only identified lesson learned from experience is that the scheduling for formal reviews is typically determined at the beginning of the program. The stage of the design for the review is then whatever it is at the scheduled time. This is not necessarily bad, because typically the designers influence the work schedule toward having a reviewable product on the established schedule. Usually, reviews cannot be moved out without jeopardizing program schedules. Designers must guard against committing to a schedule with goals that are unrealistic.

Review Emphasis

Messages are sometimes sent to designers which can be misinterpreted, informing them where they should place emphasis. This misinterpretation is based on the importance an item is given in the reviews. If the Government Program Manager and his review team place little emphasis on diagnostics, designers get the message that diagnostics are "not important." This has often been done unintentionally in the past by quickly passing over the subject in the reviews, or otherwise indicating a minimal concern. The design engineer must not forget the importance of diagnostics, especially in cases where the Government review team has placed little emphasis on the subject.

6.0 DEMONSTRATIONS

Demonstrations are, in general, another form of a formal review. Thus, most of the points made in the previous section also apply here.

Timeliness

The opportune time for final demonstration of diagnostics does not exist, if a purpose of the demonstration is to identify corrective actions. Efforts to schedule demonstrations early enough to minimize the impact of "failure" have, in the past, resulted in the simulation of too many conditions and resources. To perform a complete diagnostics demonstration, all operational diagnostics tools must be in place. This includes support equipment (if appropriate), training, technical publications, and any other applicable diagnostic tool. Attempts to simulate or work around the absence of these operational items does not provide for a complete demonstration.

Simulated vs. Operational Conditions

This problem can be demonstrated by experience with a recent modification program on the F-111D Attack Radar. The modification was major--mainly made to improve reliability and maintainability. One significant portion of the modification was the re-work of the built-in test (BIT) capabilities.

The design job seemed to be done very well. Design Reviews were passed. Demonstrations of the new BIT performance in the laboratory exceeded the specifications and expectations. All looked like a job well done and the contract was considered complete.

The problem was that on the aircraft, in operational conditions, the BIT does not do so well. The BIT serves two functions, one being to advise the aircrew if the selected mode is operational, the other serving as a diagnostics aid to maintenance personnel. The aircrew function performs well, which is not surprising, being part of the basic operational requirement. However, the diagnostics portion of the software used in the fault isolation process has required extensive re-work. At first glance, one is led to believe that the simulated and operational conditions must differ greatly. This being the case, how does one explain that problems reported during field operations can later be demonstrated under laboratory conditions? Performing demonstrations with the primary objective of showing operational requirements are being fulfilled, with diagnostics given secondary concern, only delays finding problems in that area. An important point to remember is that diagnostics must be given equal consideration to operational requirements and the Demonstration phase is another chance to identify and start correcting diagnostic problems.

Providing for Resources

Scheduling/obtaining resources for the demonstration is an early function. This requirement has often been overlooked or minimized in the past. Design engineers involved in the demonstration process should be fully aware of the demonstration plan/requirements and assure that required assets are inputted for incorporation in top-level planning documents.

7.0 MATURATION

Maturation is a phase which has been identified as necessary primarily during development of new systems/technology for the embedded and external diagnostic capability. One especially critical area for these systems is the inherent requirement for

testing under actual operating conditions. Maturation becomes necessary to refine test method/fault limits/diagnostics logic embedded within the diagnostics software programs that operate these systems. The predicted operating characteristic of the various on-board systems must be compared to the actual operating characteristics of these systems as they interface with other systems under varying environmental conditions.

Early Planning

One thing learned on the B-1B is that the design engineer must keep management informed of the considerable time and resources necessary for maturation. The original B-1B development plan was to mature the diagnostics system (CITS) on 70 FSD flights. That would, it was thought, provide a mature system at the time of the first deployment of the B-1B to an Air Force Main Operating Base. Early in the Full-Scale Development Phase, it became evident that the plan would not be sufficient. A new plan was developed to use 468 SAC sorties over the years 1985 and 1986. The wing did not fly the required number of sorties over that time period and the program was extended through November of 1987. Additional aircraft deliveries and an increase in sortie generation rate produced a total of 1069 sorties by the end of that period. With that number of sorties, sufficient data was gathered to indicate an acceptable level of performance. At this point, it is estimated that as a general rule, at least 400 to 500 sorties will be required to mature an on-board test system like the CITS. Maturation time is difficult to estimate and as learned on the B-1B changes will have to be made as the process matures.

Operational or Flight Test Environment

How does one plan for 500 sorties prior to production? Is a plan to fly four FSD aircraft on the average of once every three calendar days for a year reasonable? Is a limited production block appropriate for maturation? These are questions which the design engineer must consider when advising management of schedules early in program planning.

Experience has identified one additional consideration to be included in making these decisions. That consideration is the impact a partially working diagnostics system has on the maintenance technician. If technicians lose confidence in a diagnostic aid, they will not use. Further, it is hard to convince them that the item has been improved and that now they can have confidence in it. Many maintenance technicians on the B-1B, F-111 and other systems who have been exposed to inaccurate diagnostic methods have never been convinced to use an "improved" version. All B-1B operating bases have the same current version of CITS. Field data shows, however, that the bases exposed to the earliest and poorest version of CITS continue to have the highest false alarm and cannot

duplicate maintenance rates. This is due to the lack of trust still carried from the early experience. Thus, it is important to accomplish maturation away from the majority of operational technicians, if possible.

Implementing Maintenance Concept

If the maintenance concept utilizing the planned diagnostics is significantly different from that with which the established technician is familiar, special training will need to provided. The B-1B conflict between using CITS or system performance to rule that a failure has occurred was discussed in paragraph 3.0 of this appendix. Trends are also in place today to isolate to and replace modules on the aircraft rather than the large "boxes" of the past. Utilizing the diagnostic indication produced during flight without further ground verification is also a current trend. Each of these "new" concepts must be thoroughly understood by the technicians, so that the maturation results are consistent with the planned fielded maintenance concept. Making changes is never an easy process and the maintenance technician is no exception to this concept.

8.0 SUMMARY

Diagnostics is not a simple matter and the perfect situation portrayed in the introduction has yet to be achieved. Instead of the failure being identified to one LRU, often the ambiguity group is as many as four LRUs. The ATE which can isolate the failure to a single failed component would be the ideal solution but, more likely than not, it will only be one or sometimes several Shop Replaceable Units (SRUs) or a particular group of SRUs. The steps covered here are only some of the very basic ones required to insure good diagnostics. However, looking at many different programs, one finds even these simple steps have been omitted, or perhaps accomplished, at a time too late to have the desired results. The reasons are many: poor communication of needs or goals, time frame restrictions, money, and failure to properly consider the importance of diagnostics. To ensure diagnostics, it must be addressed at all phases and be given equal importance to other performance requirements. If the system cannot be maintained, it can never meet its operational requirements.

CHECKLIST

- ☑ Studies, analyses, and feedback take time. They need to be scheduled so that their results can influence the design.
- ☑ Test equipment designers need to have an input regarding the design requirements of the units to be tested.
- ☑ Proper priorities need to be demonstrated by both government and industry if diagnostics is to be properly implemented.
- ☑ Specifications must be well defined and represent exactly what is needed.
- ☑ Real operating time is required for maturation of the diagnostic system--lots of it.

LIST OF ACRONYMS

ABI	Avionics Bus Interface
ADA	Adaptive Diagnostic Authoring
ADS	Adaptive Diagnostic System
AFLC	Air Force Logistics Command
ADP	Automatic Data Processing
AFSC	Air Force Systems Command
AI	Artificial Intelligence
AIDA	Corporation - Santa Clara, CA
ALU	Arithmetic Logic Unit
AMC	Army Materiel Command
AMRAAM	Advanced Medium Range Air-to-Air Missile
ASIC	Application Specific Integrated Circuit
ASTEP	Advanced System Testability Analysis Program
ATE	Automatic Test Equipment
ATF	Advanced Tactical Fighter
ATG	Automatic Test Generator
ATLAS	Abbreviated Test Language for All Systems
ATPG	Automatic Test Pattern Generator
BCPE	Biphase Correlator Processing Element
BDL	Behavioral Design Language
BILBO	Built-In Logic Block Observation
BIST	Built-In Self Test
BIT	Built-In Test
BITE	Built-In Test Equipment
BLM	Behavioral Logic Model
BMM	Bulk Memory Module
C/ATLAS	Common Abbreviated Test Language for All Systems
CAD	Computer-Aided Design
CADAT	Computer-Aided Design & Test
CADAT 6	Computer-Aided Design & Test, Version 6
CADBIT	Computer-Aided Design for Built-In Test
CAE	Computer-Aided Engineering
CALS	Computer-Aided Acquisition & Logistics Support
CAMELOT	Computer-Aided Measure for Logistic Testability
CASS	Consolidated Automated Support System
CATS	Computer-Aided Test System
CDDB	Common Diagnostic Data Base
CDL	Circuit Description Language
CDR	Critical Design Review

CDRL	Contract Data Requirements List
CEP	Count Enable Parallel
CEPS	CITS Expert Parameter System
CET	Count Enable Trickle
CFE	Contractor Furnished Equipment
CI	Configuration Items
CITS	Central Integrated Test System
CLK	Clock
CLR	Clear
CMC	CITS Maintenance Code
CMOS	Complementary Metal Oxide Semi-Conductor
CML	Current Mode Logic
CMOS	Complimentary Metal Oxide Silicon
CND	Cannot Duplicate
CNO	Chief of Naval Operations
COPTR	Controllability-Odservability-Predictability Testability Report
CPCI	Computer Program Configuration Item
CRC	Cyclic Redundancy Check
CSC	Computer System Component
CSCI	Computer Software Configuration Item
CSDM	Computer System Diagnostic Manual
CSI	CADAT Systems Interface
CSOM	Computer Software Operator's Manual
CTE	Commercial Test Equipment
CTF	Controllability Transfer Factor
CY	Controllability
D-Level	Depot Level
DAISY	Manufacturer Name - Mountain View, CA
DATPG	Digital Automatic Test Program Generator
DBDD	Data Base Design Document
DCP	Decision Coordinating Paper
Dem/Val	Demonstration and Validation (Phase)
DFT	Design For Testability
DIA	Defense Intelligence Agency
DID	Data Item Description
DIP	Dual In-line Package
DMUX	Demultiplexer
DoD	Department of Defense
DoD-D	DoD Directive
DoD-INST	DoD Instruction
DNE	Data Network Element
DT&E	Development Test and Evaluation

DTA	Daisy Testability Analyzer
EARS	Engineering Access Routine Set
ECL	Emitter Collector Logic
ECC	Error Correcting Code
ECL	Emitter-Coupled Logic
ECP	Engineering Change Proposal
EDIF	Electronic Design Interchange Format
EIA	Electronics Industry Association
ESU	Element Supervisor Unit
ETE	Electronic Test Equipment
FA	False Alarm
FA	Feedback Analysis
FCA	Functional Configuration Audit
FD	Fault Detection
FEFI	Fraction of Erroneous Fault Isolation Results
FFD	Fraction of Faults Detected
FFI	Fraction of Faults Isolated
FI	Fault Isolation
FIG	Fault Isolation Group
FIPAD	Failure Indentification, Prevention and Detection
FIS	Fault Isolation System
FLEX	Name (Navy Support Cost Model)
FMEA	Failure Modes and Effects Analysis
FMECA	Failure Modes, Effects and Criticality Analysis
FOM	Figure of Merit
FOT&E	Follow-On Operational Test & Evaluation
FPPE	Floating Point Processing Element
FRACAS	Failure Reporting, Analysis and Corrective Action
FSD	Full-Scale Development
FSM	Firmware Support Manual
FYDP	Five Year Defense Plan
GFE	Government Furnished Equipment
GIMADS	Generic Integrated Maintenance Diagnostics
GM	Global Memory
GPETE	General Purpose Electronic Test Equipment
GSE	Ground Support Equipment
HDL	Hardware Description Language
HITAP	HI-Testability Analysis Program
HITS	Hierarchical Integrated Test Simulator

HSDB	High-Speed Data Bus
HW	Hardware
HWCI	Hardware Configuration Item
I-Level	Intermediate Level
I/O	Input/Output
IC	Integrated Circuit
ICE	Integrated Conceptual Environment
ICNIA	Integrated Communications, Navigation & Identification
ID	Interface Device
ID	Integrated Diagnostics
IDD	Interface Design Document
IDSS	Integrated Diagnostics Support System
IFTE	Intermediate Forward Test Equipment
IGES	Initial Graphics Exchange Specification
ILS	Integrated Logistic Support
ILSP	Integrated Logistic Support Plan
IMIS	Integrated Maintenance Information System
I/O	Input/Output
IOT&E	Initial Operational Test & Evaluation
IPS	Integrated Program Summary
IRST	Infrared Search and Track
ISPS	Instruction Set Processor Specification
ITC	International Test Conference
ITP	Integrated Test Plan
JTAG	Joint Test Action Group
KGM	Key Generator Module
LANA	Local Area Network Acceleration
LCC	Life Cycle Cost
LCCA TM	Life Cycle Cost Analysis
LCC Family of Models	Life Cycle Cost Family of Models
LFSR	Linear Feedback Shift Register
LCCC	Leadless Chip Carrier
LDCC	Leaded Chip Carrier
LED	Light Emitting Diode
LFSR	Linear Feedback Shift Register
LOGMOD	Logic Modeling
LRM	Line Replaceable Module

LRU	Line Replaceable Unit
LSA	Logistic Support Analysis
LSAP	Logistic Support Analysis Plan
LSC	Logistic Support Cost
LSI	Large Scale Integration
LSL	Large Scale Linear
LSSD	Level Sensitive Scan Design
MASA	Modular Avionics System Architecture
MATE	Modular Automatic Test Equipment
McLDL	(Microelectronics Center) Logic Description Language
MCTBF	Mean Calendar Time Between Failures
MD	Maintainability Demonstration
MD	Multiplexed Data
MDT	Mean Down Time
MIDAS	Modular Integrated Design Automation System
MIL-STD	Military Standard
MIMOLA	Machine Independent Microprocessing Language
MIPS	Million Instructions Per Second
MISR	Multiple Input Signature Register
MMIC	Monolithic Microwave Integrated Circuit
MMS	Mission Management System
MMST	MIMOLA Module for Self Test
MNS	Mission Need Statement
MSI	Medium-Scale Integration
MTBF	Mean Time Between Failures
MTBUM	Mean Time Between Unscheduled Maintenance
MTE	Manual Test Equipment
MTTI	Mean Time to Isolate
MTTR	Mean Time To Repair
MUX	Multiplexer
NDI	Non-Developmental Items
NLFSR	Non-Linear Feedback Shift Register
NMOS	N-Channel Metal Oxide Semi-Conductor
NSIA	National Security Industrial Association
NSM	Network Switch Module
NVBMM	Non-Volatile Bulk Memory Module
O-Level	Organizational Level
OJT	On-the-Job Training
OSC	Oscillator
OT&E	Operational Test & Evaluation

OTF	Observability Transfer Factor
OUSD(A)	Office of the Under Secretary of Defense (Acquisition)
OY	Observability
P^3I	Preplanned Product Improvement
PAT&E	Production Acceptance Test & Evaluation
PCA	Physical Configuration Audit
PCB	Printed Circuit Board
PDR	Preliminary Design Review
PGA	Pin Grid Array
PIA	Programmable Interface Assembly
PLA	Programmable Logic Array
PLCC/PCC	Plastic Leadless Chip Carrier, Plastic Chip Carrier
PMRT	Program Management Responsibility Transfer
PPBS	Planning, Programming and Budgeting System
PROFILE	Name
PROD/DEP	Production/Deployment (Phase)
PRR	Production Readiness Review
RADC	Rome Air Development Center
RADSS	Random Access Dynamic Set Scan
RAM	Random Access Memory
RDGT	Reliability Development/Growth Test
RF	Radio Frequency
RFP	Request for Proposal
RISE	Readiness Improvement Through System Engineering
ROC	Required Operational Capability
ROM	Read Only Memory
RTL	Register Transfer Language
RTOK	Retest OK
SAE	Society of Automotive Engineers
SCOAP	Sandia Controllability/Observability Analysis
SCP	System Coordinating Paper
SDDD	Software Detail Design Document
SDI	Scan Data In
SDLC	Sychronous Data Link Control
SDO	Shift Data Out
SDR	System Design Review
SDS	Schematic Design System
SEMP	System Engineering Management Plan
SERD	Support Equipment Recommendation Data

SHARP	Standard Hardware Acquisition Requirement Process
SI	Sensor Interface
SILVER LISCO	Corporation - Menlo Park, CA
SIT	System Integrated Test
SMD	Surface Mounted Device
SMS	System Management System
SO	Small Outline
SON	Statement of Need
SOW	Statement of Work
SPICE/PSPICE	Program Name
SPM	Software Programmer's Manual
SRA	Shop Replaceable Assembly
SRR	System Requirements Review
SRU	Shop Replaceable Unit
SSI	Small-Scale Integration
STATGRADE	Name - Statistical Fault Analysis Gateway Design
STAMP	System Testability Analysis Maintenance Program
STD	Software Test Descriptions
STLDD	Software Top-Level Design Document
SUM	Software User's Manual
SW	Software
T/D	Testability/Diagnostics
T&E	Test & Evaluation
T	Testability
TAB	Tape Automated Bonding
TAH	Testability Analysis Handbook
TBD	To Be Determined
TCG	Timing and Control Generator
TEMS	Turbine Engine Monitoring System
TEMP	Test & Evaluation Master Plan
TESTGRADE	Name - Test Vector Grading
TFOM	Testability Figure of Merit
THESEUS	Name
TI	Technical Information
TIATA	Technical Information & Training Authoring
TISSS	Tester Independent Support Software System
TM	Test and Maintenance
TO	Technical Orders
TP	Test Point
TPI	Test Program Instruction
TPS	Test Program Set
TRD	Test Requirements Document

TRITAC	Tri-Service Tactical Comm. Program (Joint Services C&C)
TRR	Test Readiness Review
TTL	Transistor-Transistor Logic
TY	Testability
UMDP	Univeral Mask Data Preparation
UPE	Universal Pin Electronics
URR	Ultra-Reliable Radar
UUT	Unit Under Test
VHDL	VHSIC Hardware Descriptive Language
VHSIC	Very High Speed Integrated Circuit
VLSI	Very Large Scale Integration
VMS	Vehicle Management System
VPE	Vector Processing Element
WBS	Work Breakdown Structure
WRA	Weapon Replaceable Assembly
WSTA	Weapon System Testability Analyzer
XOR	Exclusive-or (gate)
ZIF	Zero Insertion Force
ZYCAD	Company - St. Paul, MN

Table of Contents

1.0	OVERVIEW	5
2.0	REQUIREMENT TOOLS	10
2.0.1	A/LICE - ADA/LATTICE Integrated Conceptual Environment	11
2.1	Establishing Requirements	13
2.1.1	ACCLOGTROM - Army Communications Command Logistics Trade-Off Model	19
2.1.2	ARLCAP - Army Logistics Capability Assessments	19
2.1.3	ASOAR - Achieving A System Operational Availability Requirement Methodology	20
2.1.4	COVERS - Combat Vehicle RAM Simulation	20
2.1.5	ERAMS - Electronic RAM Simulation	20
2.1.6	LEAD - Logistics Engineering Analysis of Design	20
2.1.7	LSSAS - Logistics Support Simulator for Aircraft Systems	21
2.1.8	OREM - Operational Readiness Evaluation Model	21
2.1.9	FLEX - Name (Navy Support Cost Model)	23
2.1.10	LSC - Logistic Support Cost	23
2.1.11	LCCA TM - Life Cycle Cost Analysis	23
2.1.12	LCC Family of Models - Life Cycle Cost Family of Models	23
2.1.13	TRITAC - Tri-Service Tactical Comm. Program (Joint Services C&C)	23
2.1.14	COSTPRO - Cost Projection Management Information System for Life Cycle Costs	23
2.1.15	LOCAM 5 -Logistics Cost Analysis Model -Version 5	24
2.1.16	LOGAM - Logistics Analysis Model	24
2.1.17	PRICE - Parametric Review of Information for Costing and Evaluation	25
2.1.18	MOD III, LOR - Level of Repair Analysis Model	27
2.1.19	NRLA - Network Repair Level Analysis	27
2.1.20	OASES - Operations and Support Environment Synthesis	27
2.1.21	OSAMM - Optimum Supply and Maintenance Manual	27
2.1.22	SESAME - Selected Essential Item Stockage for Available Method	28
2.1.23	CALSA - Computer-Aided Logistic Support Analysis	29
2.1.24	LEADS - Logistic Entry and Analysis Data Systems	29
2.2	Allocation Tools	29
2.3	Optimization Tools	31

2.4	Risk Assessment Tools	31
2.4.1	RISENET - Risk Network	32
2.4.2	TRACE - Total Risk Assessing Cost Estimate	32
2.4.3	TRACE-P - Total Risk Assessing Cost Estimate - Production	32
3.0	DESIGN IMPLEMENTATION TOOLS	35
3.1	Architecture Tools	36
3.1.1	ADAS - Architecture Design and Assessment System	39
3.1.2	A/LICE-ADA/Lattice Integrated Conceptual Environment	42
3.1.3	CAD/BIT - Computer-Aided Design for Built-In Test	43
3.1.4	CADAT - 6 -Computer-Aided Design and Test, Version 6	45
3.1.5	DAISY - Manufacturer Name - Mountain View, CA	48
3.1.6	IDSS - Integrated Diagnostic Support System	49
3.1.7	LASAR VERSION 6	52
3.1.8	MENTOR GRAPHICS	54
3.1.9	MIDAS - Modular Integrated Design Automation System	55
3.1.10	MIMOLA MST - MIMOLA Module for Self Test	57
3.1.11	SILVAR LISCO - Corporation - Menlo Park, CA	59
3.1.12	SPICE/PSPICE - Program Name	63
3.2	Design Rules and Practices	65
3.2.1	IDSS-ADS - Adaptive Diagnostic System	67
3.2.2	TDES - Testable Design Expert System	70
3.2.3	TEA - Test Engineers Assistant	72
3.2.4	TISSS - Tester Independent Support Software System	74
3.2.5	Printed Circuit Board Testability Design Guide and Rating System	77
3.3	Diagnostic Authoring Tools	78
3.3.1	Generic Diagnostic Authoring Tools	78
3.3.1.1	CATA - Computer-Aided Testability Analysis	80
3.3.1.2	GIMADS Diagnostic Library - Generic Integrated Maintenance Diagnostics Program	81
3.3.1.3	IDSS-ADA - Adaptive Diagnostic Authoring	83
3.3.1.4	IDSS-TIATA - Technical Information and Training Authoring	85
3.3.1.5	TGIR - Test Generator Inferred Reasoning	87
3.3.1.6	AI-TEST - Artificial Intelligence Test	89
3.3.2	Automatic Test Generation Authoring Tools	92
3.3.2.1	AIDA - Corporation - Santa Clara, CA	95
3.3.2.2	HITS - Hierarchical Integrated Test Simulator	96
3.3.2.3	LASAR VER 6; PROSECUTER	97
3.3.2.4	SOCRATES - Structure-Oriented, Cost Reducing, Automatic Test Pattern Generation System	98
3.3.2.5	THESEUS	101
3.3.2.6	ZYCAD; Next Gen	103

4.0	ASSESSMENT TOOLS	105
4.1	Inherent Testability Analysis Tools	105
4.1.1	CAFIT - Computer-Aided Fault Isolation Testability	108
4.1.2	CAMELOT-Computer-Aided Measure for Logic Testability	110
4.1.3	COMET - Controllability and Observability Measurement for Testability	112
4.1.4	COP - Controllability and Observability Program	114
4.1.5	COPTR - Controllability - Observability - Predictability - Testability Report	116
4.1.6	DTA - Daisy Testability Analyzer	118
4.1.7	FACE - Fault Coverage Estimation	120
4.1.8	HECTOR - Heuristic Controllability and Observability Analysis	122
4.1.9	HITAP - HI-Testability Analysis Program	124
4.1.10	IDSS-WSTA - Weapon System Testability Analyzer	126
4.1.11	ITTAP - Interactive Testability Analysis Program	127
4.1.12	PROTEST - Probabilistic Testability Analysis	129
4.1.13	SCOAP - Scandia Controllability/Observability Analysis Program	131
4.1.14	STAMP - System Testability and Maintenance Program	133
4.1.15	TY CHECKLIST - Testability Checklist	134
4.1.16	THESEUS - Name	136
4.1.17	TMEAS - Testability Measurement	137
4.1.18	VICTOR - VLSI Identifier of Controllability, Testability, Observability, and Redundanc	139
4.2	Diagnostic Effectiveness Tools	140
4.2.1	Test Strategy Tools	140
4.2.1.1	ACE - Apt Computational Environment	143
4.2.1.2	ASTEP - Advanced System Testability Evaluation	145
4.2.1.3	I-CAT - Intelligent Computer-Aided Test	147
4.2.1.4	IDSS - WSTA - Weapon System Testability Analyzer	149
4.2.1.5	LOGMOD - Logic Model	152
4.2.1.6	PROFILE - Name	154
4.2.1.7	STAMP - System Testability and Maintenance Program	156
4.2.1.8	TESAP - Test Strategy Assessment Program	159
4.2.1.9	TIME - Testability Interfaced Maintainability Estimates	161
4.2.2	Fault Simulation Tools	163
4.2.2.1	AIDA - Corporation, Santa Clara, CA	165
4.2.2.2	BITGRADE - Build-In Test Grade	167
4.2.2.3	CADAT 6 - Computer-Aided Design & Test, Version 6	169
4.2.2.4	HITS - Hierarchical Integrated Test Simulator	171
4.2.2.5	IKOS 800	174
4.2.2.6	LASAR VER 6: JUDGE	177
4.2.2.7	QUICKFAULT	178

4.2.2.8	STAFAN - Statistical Fault Analysis	181
4.2.2.9	STATGRADE-Statistical Fault Analysis Gateway Design	183
4.2.2.10	TESTGRADE - Test Vector Grading	185
4.2.2.11	ZYCAD - Company, St. Paul, MN	187
5.0	DEMONSTRATION TOOLS	190
5.1	MIL-STD-471A - Maintainability Verification/Demonstration/Evaluation	191
6.0	MATURATION TOOLS	194
6.1	CITS/CEPS - Central Integrated Test System/CITS Expert Parameter System	195
6.2	IDSS-FA - Feedback Analysis	197

1.0 OVERVIEW

This appendix provides information concerning many types of tools that are useful in performing a multitude of tasks required to include testability/diagnostic capabilities into functional designs throughout each and every phase of the acquisition process.

Many types of tasks require many types of tools. Some tools provide a framework so that other tools may operate. We have called this type, "Tools for Tools." There are two "tools" of this type included in this document.

There are guidance type tools, such as military standards. There are parametric models which are tools that are algorithmic and calculative in nature. One puts various parameters into a parametric model and obtains certain parameters of interest from it. Models of interest of these types are Availability, Life Cycle Cost (LCC) models, Level of Repair (LOR), and Logistic Support Analysis Records (LSAR).

Mention must be made to the software design systems that perform on engineering workstations as well as to the software utilities that compose these systems. Therefore, both types were included into this document and may appear next to one another when they are listed.

There are a variety of types of software utilities. There are digital simulators, analog simulators, and simulators with capabilities to simulate both digital and analog, which we call mixed mode.

There are dependency models that serve in defining and assessing testability/diagnostic capabilities, some of which come in the form of a software utility. Some expert systems, another type of tool, use dependency modeling within their framework. Expert systems, a branch of Artificial Intelligence (AI), are finding applications in defining and assessing the testability/diagnostic capability arena and a few of them are included in this document.

There are software utilities available that provide and display information and others that process information. Still others provide test engineering assistance such as calculating controllability and observability figures.

Last, but not least, is the checklist, useful both in establishing goals and/or requirements and assuring that these requirements have been met.

A fact sheet, for characterizing in a normalized manner each of the software tools surveyed, is provided in Table 1.

Table 1
Fact Sheet Utilized to Characterize Each
Software Tool Surveyed

NAME: The product's acronym followed by the full name.

YEAR: The year the product was first introduced and/or the year of the present version.

FUNCTION: A brief description or listing of the functions, propose, or capabilities of the tool.

CAPABILITY: The maximum or typical size of the circuit or system the tool is performed on.

CPU TIME: A statement to provide an idea of how much CPU time will be consumed when processing a circuit or system of the above capacity. Any other fact relating to process speed can also be stated here.

APPLICATION: VLSI PCB SUBSYS SYSTEM

The level or levels of integration for which this tool can provide any assistance is underlined. Most of these fact sheets have been reviewed and corrected by the developer. Developers tend to be more optimistic on how many levels of integration their tool will apply.

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

The phase or phases during which this tool is useful is underlined. See above note.

PUBLIC DOMAIN? Y/N

A "N" means the public cannot obtain the tool. A few tools have been included with "N" underscored for academic purposes. Whenever a "Y" has been underscored, there follows either "(GFE)", "(UNIV)", or "(PRTY)". GFE stands for Government Furnished Equipment, UNIV stands for "university", and PRTY stands for "priority". GFE software is distributed by the U.S. Government at a minimal price. UNIV software is distributed by the university responsible for its development at a minimal price. PRTY software is published to make a profit for the company that developed it. Although GFE or UNIV software does have a minimal price attached, it is considered to be "FREE" and PRTY software is considered "FOR SALE".

SPICE (not PSPICE) is an example of a university developed tool that is available at a minimal price. TDES is an example of a university developed tool that is not available. FACE, VICTOR, and PROTEST are three university developed tools wherein the information as to their public access is not available at this time. The availability of ITTAP, developed by ITT-LSI Technology Center, is also unknown at this time.

DESIGN ENV: Stated here are what computer platforms the tool will run on, what design system the tool is a part of, and/or what language the program is written in.

CIRCUIT DESCRIPTION: All tools of the type included in this section require that the circuit or system first be described either in a particular HDL, a netlist, a functional block diagram, etc. A statement as to how it is described is provided here.

USE PREREQUISITE: Listed here are requirements that must be met prior to using the tool other than describing the circuit. Here is also where the input parameters would be listed.

DEVELOPER: The name of the company or individual responsible for developing the tool.

COMMENTS: Statements or facts not yet mentioned that would be important to someone who may want to use the tool. What is most unique about the product is included here.

REFERENCES: Sources of more information and/or where to obtain the tool is listed.

All of the tools of all of the tool types mentioned above have been categorized by function (i.e., Requirement, Design, Assessment and Maturation). These categories are listed horizontally in Table 2 and the various tool types are listed vertically. The tools themselves are listed in this matrix. To find when a tool is used or for what task, look up to find its column heading. To find what type a tool is, look to the left to find its row heading.

In the chart, you see arrows to the left and to the right of most of the tools. This signifies that the tool, or group of tools, is useful in performing more categories of tasks than the ones delegated to it in this document. Positioning these tools on the chart was difficult and it may, unintentionally, generate some disagreement. Bear in mind that the intent of the chart is simply to provide a broad overview of the tools to help distinguish the forest from the trees.

Often there are many tools of the same type that are used for the same tasks and it would have been impossible to fit them all in the proper location on the chart. For this reason, numbers have been assigned to the tools according to how they have been categorized in this document. For example, 2.1.1 would be the first tool described in the first subheading, "Establishing Requirements," for Section 2, "Requirements." The numbers 3.3.2.3 refer to the third tool listed in the second subheading, "ATPG's," which is part of the third subheading, "Diagnostic Authoring," which is part of Section 3, "Design Implementation."

TABLE 2. TYPES OF TESTABILITY/DIAGNOSTIC TOOLS vs. AREAS OF APPLICATION

APPLICATION / TOOL TYPE	REQUIREMENTS	DESIGN IMPLEMENTATION	ASSESSMENT	MATURATION
	EST. ALLOC. OPT. RISK	ARCH. RULES AUTH.	INH.T. EFF. DEMO.	FDBK. & ANAL.
I. TOOL FOR TOOLS	A/LICE	3.1.2 3.2.4 TISSS		
II. MIL-STDS	1388-2A		471 (5.1)	
III. PARAMETRIC MODEL				
a. AVAILABILITY	2.1.1/ 2.1.8			
b. LCC	2.1.9/ 2.1.17	4.1/ 2.4.3		
c. LOK	2.1.18/ 2.1.22			
d. LSAR	2.1.23/ 2.1.24			

APPLICATION / TOOL TYPE	REQUIREMENTS	DESIGN IMPLEMENTATION	ASSESSMENT	MATURATION
	EST. ALLOC. OPT. RISK	ARCH. RULES AUTH.	INH.T. EFF. DEMO.	FDBK. & ANAL.
I. SOFTWARE DESIGN		3.1.1 3.1.4/ 3.1.11		6.2
a. DIGITAL SIM.		3.3.3.1/ 3.3.3.5	4.2.2.1/ 4.2.2.11	
b. ANALOG SIM.		3.1.12		
c. MIXED MODE			4.2.1.8 4.2.1.2 4.2.1.6	
		3.3.1.1		
d. DEPENDENCY MODEL			4.2.11 4.1.10/4.2.1.4 4.2.1.3 4.2.1.5 4.1.14/4.2.1.7	
e. EXPERT SYS.		3.2.1		
		3.2.2 3.3.1.5		
f. INFO DISPLAY		3.1.3		
		3.3.1.2 3.2.3		
g. INFO PROCESS		3.3.1.3 3.3.1.4		6.1
h. CONT/OBS CALC			4.1.1/ 4.1.9.4.1.11/4.1.13	
V. CHECKLIST			4.1.15	

2.0 REQUIREMENT TOOLS

This section lists models and standard procedures, as well as software tools that are either available or under development, to aid in establishing optimized diagnostic requirements that are properly allocated to the various repair levels and depict minimum risk to both mission success and life cycle costs.

These requirement tools have been categorized with the following four subheadings:

- o Establishing requirements
- o Allocation
- o Optimization
- o Risk.

The entries will be presented in the above order.

A major Government Program established to assure proper technological integration of these tools is knows as, "Computer Aided Acquisition and Logistic Support (CALS)." A/LICE is a software program that has been developed to provide a framework for the implementation of the CALS philosophy and will be included first because of its nature. It is a tool for tools. Another tool for tools, "TISSS," is briefly described in Section 3.0, "Design Implementation Tools," under the subheading, "Design Rules and Practices."

The criteria for a tool being listed in this section is established by answering the following question:

> "According to engineering judgment, does this tool markedly aid in the task of establishing optimized requirements of an effective diagnostic system?"

There are no claims made that this is an all inclusive list. There are perhaps dozens of tools that are not included.

2.0.1 A/LICE

NAME: **A/LICE**; ADA/LATTICE INTEGRATED CONCEPTUAL ENVIRONMENT

YEAR: 1985 (first operational version)

FUNCTION: The CALS philosophy is to pool the knowledge of all the departmental experts so that planners of a new weapon system get as much of the big picture as possible and, thereby, greatly enhance system efficiency and the probability of success. The CALS knowledge base requires "super" representation schema due to the fact that it must contain diverse specialties and often competing "ilities," including, Reliability, Maintainability, Availability, ILS, Support, Design/Build, Planning and Technical Documentation, and others. This super representation schema is called an Integrated Conceptual Environment (ICE).

The immediate problem of implementing such a philosophy is trying to standardize upon the knowledge representing formats, knowledge that must somehow be linked to text and graphics and also to the various emerging expert systems. A/LICE is an Ada coded program with knowledge lattice extension operators to form an outline for such an ICE and to provide a method of machine hosting ICE morphological operators in Ada. A/LICE thus presents a solution to processing the massive and diverse knowledge requirements of the CALS program. A/LICE has the following features:

- A/LICE can interface with diverse expert systems regardless of knowledge format, the language, or structures involved.
- At the meta-level, A/LICE sees knowledge objects in lattice arrays which can be processed using standard math array processors.
- The A/LICE high-level instruction set will interface with the emerging electro-optical analog computers that will use direct capture "image as knowledge" processing techniques.

CAPACITY: N/A

CPU TIME: N/A

APPLICATION: VLSI PCB SUBSYS SYSTEM SYSTEM OF SYSTEMS

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N

DESIGN ENV: Personal computers using standard math array processors.

INFORMATION BUNDLING TECHNIQUE: The highest level operator set are the "machine instructions" anticipated in the advanced analog optical computers being developed by DOD. In practice, Ada derived hardware description languages (like VHDL) or Lisp extensions (such as EDIF) are simultaneously accommodated. A/LICE conforms to the TISSS data typing protocol.

USE PREREQUISITE: Generic "CALS" input is anticipated.

DEVELOPER: Sirius, Incorporated (now a division of Science Applications International Corp.).

COMMENTS: Knowledge of any kind can be fused by simple procedures if the source calculus is derived from the universal structure employed. Image data, graphic modeling information, engineering data, and performance data can all be integrated, or fused, onto a common knowledge data structure.

It is good to note some of the virtues of Ada. Ada is portable, maintainable, reliable, modular, and easily upgradable.

REFERENCES:

1. H. T. Goranson, "An Approach to Knowledge Structuring For Advanced Phases of the Technical and Management Information System," 1st International Conference on Ada in the Space Station, June 1986.

2. H. T. Goranson, "CALS Knowledge Bases For Assurance Technologies," RAM Symposium Jan. 1987.

3. H. T. Goranson, "Research and Development Strategy For Advanced CALS," Internal Document, Defense Advanced Research Projects Agency, 1988.

4. Sirius Incorporated
Attn: H.T. Goranson
1976 Munden Point
Virginia Beach, VA 23457
(804) 463-9110

2.1 Establishing Requirements

The requirement tools of this section have been categorized with the following subheading:

- Establishing Requirements

and have been categorized as follows:

Logistic Support Analysis (MIL-STD-1388-1A)

A) Readiness or Availability Models

B) Life Cycle Cost Models

a) Cost Models

b) Level of Repair Analysis Models

c) LSAR Data Base Generation

LOGISTICS SUPPORT ANALYSIS (LSA)

DIAGNOSTIC ACTIVITY

The LSA process identifies and defines system diagnostic requirements and assesses diagnostic capabilities.

An LSA activity is the selective application of scientific and engineering efforts undertaken during the acquisition process, as part of the system engineering and design process, to assist in complying with supportability and other Integrated Logistics Support (ILS) objectives. LSA is a regulatory requirement and is required in all material acquisition programs without exception.

The LSA process, in accordance with MIL-STD-1388-1A, consists of the following series of tasks that are tailored to the particular system by the Statement of Work (SOW):

100 Series:

- Program Planning & Control - The LSA Plan documents the LSA management structure, i.e., what LSA tasks are to accomplished,

when each task will be completed, who will perform them, and how the tasks are integrated, and how results are used.

200 Series:

- Mission & Support Systems Definition - Establish supportability objectives and supportability-related design goals, thresholds, and constraints through comparison with existing systems and analyses of supportability, cost, and readiness drivers. This task is predominantly done manually and takes many months to complete.

300 Series:

- Preparation and Evaluation of Alternatives - Optimize the support system for the new item and develop a system which achieves the best balance between cost, schedule, performance, and supportability.

400 Series:

- Determination of Logistic Support Resource Requirements - Identify tools, test equipment, spares, personnel skills, etc. These tasks are spreadsheet intensive.

500 Series:

- Supportability Assessment - Supportability test, evaluation, and verification.

The LSA Record (LSAR), which is prepared in accordance with MIL-STD-1388-2A, documents LSA results and, in conjunction with the Joint Service LSAR Data Systems, provides specific output reports.

A contractor develops and submits LSA/LSAR data based on tailoring of LSA/LSAR in Contract Data Requirements Lists (CDRL), Data Item Descriptions (DID), and ILS Statements of Work (SOW). LSA data and an LSA Plan (LSAP) are submitted pursuant to MIL-STD-1388-1A and MIL-STD-1388-2A. A copy of the LSAP is also provided to a readiness support responsible agency that either provides the contractor with the "Joint Service LSAR ADP System" as GFP or provides validation for the contractor's LSAR Data System via a "Joint Service Team." The contractor then develops LSA/LSAR data submissions in support of the following program elements:

- System/Equipment Design Program
- System/Equipment Reliability Program

- System/Equipment Maintainability Program
- Human Engineering Program
- Standardization Program
- Parts Control Program
- System Safety Program
- Packaging, Handling, Storage, and Transportability Program
- Initial Provisioning Program
- System/Equipment Testability Program
- Survivability Program
- Technical Publications Program
- Facilities Program
- Support Equipment Program
- Test and Evaluation Program
- Life Cycle Cost Program.

The 200 Series of tasks are necessary for establishing requirements. The 300 Series of tasks are necessary to optimize the design. A novice desires to know which tool to use for each of the tasks. An expert knows that these tasks involve many disciplines, are fractionated, and that there are a vast number of tools and models from which to choose. Trade studies carried out within the systems engineering process involve a team of design engineers, ILS engineers, and specialists from various disciplines as required by the specific subject of the study. Furthermore, one model often will be used during the tasks of establishing requirements and also used when the design is optimized. In fact, this is so often the case, that any model included in this document under the section for establishing requirements is also included in the section for optimization.

WHICH MODEL TO USE?

Which model to use depends on:

- The models input and output parameters. Each program has a different set of priorities and, therefore, the different model parameters will have different weights of importance. Certain models apply more readily to some parameters than others.

- Budget. Certain models are software intensive and are more expensive to obtain than others. Others are labor intensive.

- Resources. Some models require specific host computers.

MIL-STD-1388-1A, Appendix A, paragraph 40.6 recommends using simple models during Concept Phase and more complex models with more

detailed parameters as the design becomes better defined and a support concept is established.

WHAT TRADEOFFS IMPACT DIAGNOSTIC REQUIREMENT GOALS?

Upon reviewing all the programs required for LSA/LSAR data submission, those most related to diagnostic capabilities are:

Reliability, Maintainability, Testability, and Support.

What these four programs really provide is system readiness or availability. It is always desirable for mission equipment to be ready and available. The major offset to availability is cost. Availability versus cost is the most dominant tradeoff in establishing diagnostic requirement goals. It is always desirable for mission equipment to cost little. What the customer ideally wants, therefore, is for his mission equipment to always be ready and available and cost effective.

Both availability and cost have always been difficult measures to accurately predict due to the complexity of all the myriads of factors that compose a real system. Remember the battle that was lost for lack of a horseshoe nail? This is even true for multi-million dollar programs. To achieve accurate predictions, more and more parameters enter into the equations and the computations become excessive. There have been many conscientious attempts, however, to include most relevant factors without being too computationally burdensome. Also, much of the computational burden is eased through use of the computer.

HOW IS AVAILABILITY MODELED?

Availability, due to its complexity, is rarely modeled the same way in every detail by different programs. Consider the following definition for Availability (Ao) provided by NAVMATINST 3000.2 summarized below. (This model differs, by the way, from the general model provided by RADC-TR-79-309 which substitutes MTTR for MDT.)

GENERAL MODEL: $Ao = \frac{MTBF}{MTBF + MDT}$

MDT = MDTs + MTTR + MLDT + MDToa + MDTt + MDTor

MDT = Mean Down Time

MDTs = Mean Down Time due to Scheduled Maintenance

MTTR = Mean Time To Repair

MLDT = Mean Logistics Delay Time

MDToa = Average Down Time Waiting for Outside Assistance

MDTd = Average Down Time due to lack of Documentation

MDTt = Average Down Time due to lack of Training

MDTor = Average Down Time due to Other Reasons

The agency responsible for establishing BIT/ETE effectiveness goals should be keenly aware that a system designed with excellent diagnostic capabilities will substantially reduce:

MDTs - More confidence in system checkout.

MTTR - Reduce FD/FI times.

MDToa - Less time for outside engineering assistance.

MDTd - Less need for documentation.

MDTt - Less skill level required.

The trend is to embed more and more diagnostic capability into the BIT function. Incorporating BIT into a system results in varied amounts of increased size, weight, power, and software requirements depending upon the BIT methodology used. These factors must be considered when making cost and performance tradeoffs for alternate test systems.

Some other important parameters and concepts to consider are spare stockage levels, the logistics, maintenance, and local repair concepts, the degree of modularization, etc. One quickly sees that one analysis using one model is not going to accurately establish requirement goals. Each parameter in the above model must be separately analyzed for its total content and how this total is calculated. Assumptions made must be documented so that when changes in the assumptions occur, changes in the calculations can be understood. Numerous operational availability sensitivity analyses must be performed to ensure a high and stable Ao.

Although MIL-STD-1388-1A defines the LSA program requirements, it does not define the procedures/approaches for LSA task accomplishment. Ref[1] has been developed to strengthen the LSA program and assist in the

accomplishment of those LSA tasks set forth in MIL-STD-1388-1A. This document catalogues numerous methodologies, both manual and automated, that exist within the DoD and industry which can be used to satisfy many of the LSA task requirements in total, or in part.

The techniques, models, and programs that compose Ref[1] will all in some way apply to the overall task of integrating a diagnostic capability into a function; however, of particular interest are those models that directly include a BIT/ETE FOM either as an input or an output. Some of these models are referenced in the guidance section of the BIT/ETE FOMs article when the given FOM is used either as an input or an output to the model. BIT/ETE FOM parameters most often used in the LSA process are Availability, MTTR, and MTBF. (BIT/ETE affect MTBF predominantly when fault tolerance is deployed in the functional design.)

The following four pages contain a model listing that provides some examples of models used in part to establish Availability or System Readiness goals during the LSA process. Both the model's acronym and its full name are given. To better understand what the model will do, a brief list of inputs and outputs is also provided. More background information may be obtained in Ref[1] which also provides sources for even further information. Ref[7] is a good source for Availability equations for redundant systems.

ACRONYM	FULL NAME	INPUTS	OUTPUTS
2.1.1 ACCLOGTROM	Army Communications Command Logistics Tradeoff Model	Operational availability goal • • Equipment reliability block diagrams • • LRU data consisting of each LRUs cost, failure rate, density, mean time to restore on-site, repair location, washout rates, and production lead time • • Supply information consisting of order and ship times from depot and area supports to site and off-site order fill rates desired	Least cost set of LRUs required for sparing at each site which meets the desired operational availability goal • • Set of LRU spares required off-site to achieve off-site order fill rates desired • • Total cost of LRUs spared per site, at area supports, and depot
2.1.2 ARLCAP	Army Logistics Capability Assessment	Flying program file which has 66 inputs which include mission length, daily flying hours per aircraft, etc. Parts data base which has 116 inputs which includes such information as unit cost, administrative lead time, order and ship time, repair costs, etc. • • Force file which has 7 inputs required which is number of aircraft in a unit, description of unit, etc.	Mission success rate • • Repair cost • • Aircraft availability • • Repairable spare maintainability reliability analysis • • One year capability purchase analysis

ACRONYM	FULL NAME	INPUTS	OUTPUTS
2.1.3 ASOAR	Achieving a System Operational Availability Requirement Methodology	System operational availability requirement • • System configuration end item • • System support concepts planned • • Mean time to obtain an LRU spare • • End Item Mean Calendar Time Between Failures (MCTBF) • • End item mean restored time • • End item cost estimate	Whether the system design and support planned will achieve the system operational availability requirements • • Gross estimation of the relative cost for secondary item spares • • Operational availability goals for each different type of critical end item • • System Mean Calendar Time Between Failures • • System mean restoral time • • Effective MCTBF of a redundant network
2.1.4 COVERS	Combat Vehicle RAM Simulation	Mean Time To Repair • • Parts information • • Maintenance manpower information • • Scenario usage rates • • Scenario damage rates • • Mean time between failures	Parts needed • • Down time • • Operational availability
2.1.5 ERAMS	Electronic RAM Simulation	Failure rates • • Repair times • • Maintenance manpower availability • • Usage rates	Manpower requirements • • Parts requirements • • Down times • • Operational availability
2.1.6 LEAD	Logistics Engineering Analysis of Design	Reliability block diagram with each block's failure rates • • MTTR • • Preventive maintenance policy • • Number of hours per day the system operates	System, subsystem, and block reliability predictions for user specified time intervals • • System MTBF • • System Mean Time Between Unscheduled Maintenance MTBUM • • Availability

ACRONYM	FULL NAME	INPUTS	OUTPUTS
2.1.7 LSSAS (fill)	Logistics Support Simulator for Aircraft Systems	The model can be exercised in a parametric fashion over a wide range of input data, including: • • Variations of MTBF, MTTR, Not Operationally Ready Supply (NORS) rates, and ATE capability • • Pre-flight duration, launch activities, taxi and end-of-runway check, probability of ground and air abort, sortie and post-flight duration • • Scheduled maintenance interval and duration • • Spare data	In terms of sortie rate in response to variations of input data
2.1.8 OREM	Operational Readiness Evaluation Model	Reliability, maintainability, support, and mission estimates	Daily listing of aircraft readiness data in terms of availability, ability to satisfy the mission requirements, and sortie rates. • • Time lost due to: - parts delay - aircraft not fully equipped - delays in providing Ground Support Equipment - time spent in maintenance shops

HOW MUCH WILL IT COST ?

The grand-daddy cost to consider is the Life Cycle Cost (LCC) or the total cost to acquire and maintain mission equipment throughout its life cycle. LCC studies are performed against a fixed availability goal. LCC elements are divided between development cost elements and operating and support (O&S) cost elements. The following are typical development LCC elements:

- Equipment
- System Test & Evaluation
- System Engineer/Program Management (non-ILS)
- Data (non-ILS)
- ILS
- Industrial Facilities
- Test Program Sets.

The following are typical O&S LCC elements:

- Personnel
- On-Site Equipment Material
- Direct Depot Maintenance
- Sustaining Investment
- Software Support
- Contractor Sustained On-Site Support
- Recurring Publications Cost
- Indirect Equipment Costs.

LCC models can prove to be very valuable cost saving decision making tools. Ref[6] uses the USAF Logistics Support Cost (LCS) model to make a comparison of test effectiveness with and without a testability program. This reference shows an example of a $4,400,000.00 savings when a 15-year, 50-unit, continuous operation is deployed with testability considered up front as opposed to not considering it.

Ref[5] is a first source for LCC models. Each reference will contain LCC model description information. The following three pages provide some information to some LCC models available. Please note that the criteria for listing them is simply that information concerning them was readily available and that they are listed only to provide examples.

ACRONYM	FULL NAME	INPUTS	OUTPUTS
2.1.9 FLEX 9 • • • • 2.1.10 LSC • • • • 2.1.11 LCCA TM [ADA: LCCA] • • • • 2.1.12 FAMILY OF LCC MODELS: LCC -2 -2A, -10, -AT, Etc. • • • • 2.1.13 TRITAC	These five logistic support cost modeling systems were respectfully developed by: - Navy - AFALD/XTXC - Analytical Sciences Corp. (TASC) - Dept. of Air Force - Strategic Financial Planning Systems	Approximately 50 weapon and system input variables Approx. 25 input variables per flight line unit	Total LSC with various options as to how they are classified and displayed
2.1.14 COSTPRO	Cost Projection Management Information System for Life Cycle Costs	Support item cost and maintenance codes • • Mean Time To Repair • • Mean Time Between Failures • • Repair turnaround time • • Supply and maintenance hierarchy • • Indenturing hierarchy • • Repair labor costs • • Support equipment training and documentation costs	Summary by appropriation • • Cost and funding responsibility report. All reports producible in base year or inflated dollars • • Cost summary report

ACRONYM	FULL NAME	INPUTS	OUTPUTS
2.1.15 LOCAM 5	Logistics cost Analysis Model Version 5	Approximately 325 LRU, deployment, and common standard cost factors	Life Cycle Cost of ownership for individual LRUs • • Cost totals for operating and maintenance • • Inherent and operational availability at both LRU and system levels • • Manpower requirements • • Provisioning requirements • • Test equipment requirements
2.1.16 LOGAM	Logistics Analysis Model	Approximately 325 LRU, deployment, and common • • Up to 200 Table of Organization (TOE) related variables depending on application may be input to obtain O&S costs • • Repair time • • Failure rate • • False failure • • Attrition rate • • Manpower by type and rates • • Supply factors and support costs	Life Cycle Cost for the system and individual LRUs • • O&M costs • • O&S costs • • Availability • • Manpower requirements • • Provisioning requirements and support equipment usage are provided in the outputs

ACRONYM	FULL NAME	INPUTS	OUTPUTS
2.1.17 PRICE; FOUR RELATED PROGRAMS: • H • S • HL • SL	Parametric Review of Information for costing and Evaluation (PRICE) H: hardware HL: Hardware LCC Model S: Software SL: Software LCC Model [Developed by RCA]	PRICE H inputs: design concept description: size, weight, component type, power dissipation, etc., classified according to weighting factors (i.e., critical, important) PRICE HL inputs: • Equipment Dependent *MTBF *Mean Repair Times *HDWE costs, qtys., learning curves for acq. & supply *Cost of test equip. * # of modules & parts per unit • Deployment dependent (U.S., Europe, SE Asia; also must specify the # of installations) • Employment dependent (Repair frequency/ frequency of use) • Organization dependent * # maintenance and supply locations * Specific concepts possible for HDWE maintenance * Unique factors that apply to the specific operation of an organization (i.e., transportation, resupply, time, scrap, labor rates, etc.)	Estimates LCC costs for a variety of systems PRICE H: cost estimation of the development, production, and/or modification of hardware PRICE HL: cost estimation of supply and maintenance of hardware systems PRICE S: cost estimation of the development, production, and/or modification of software PRICE SL: Cost estimation of supply and maintenance of software systems
PRICE S & SL		Data sets for one PRICE Program can be carried over to other PRICE versions.	

HOW MUCH SUPPORT EQUIPMENT AND MANPOWER WILL WE NEED?

A major task of the LCC analyses is known as the Level of Repair (LOR) analysis. LOR analysis results in establishing requirements in two major areas, sparing and manpower. The TFOM requirement, which identifies ambiguities (false pulls), will also impact these areas. Typical tradeoffs made while doing an LOR analysis are:

- Support equipment vs sparing requirements
- Where is the repair made?
- Repair and Replace (R/R) task descriptions.

There are many models from which to choose for performing an LOR analysis. The following two pages provide some information to some LOR analysis models available. Note again that the criteria for listing these particular models is simply that information concerning them was readily available and that they are listed only to provide examples.

ACRONYM	FULL NAME	INPUTS	OUTPUTS
2.1.18 MOD III, LOR	Level of Repair Analysis Model Developed to fulfill MIL-STD-1390B Appendix A Developed by NAVAIR	Parameters of WRA's: MTBF, MTTR, weight, etc.	I/Depot level logistic support costs
2.1.19 NRLA	Network Repair Level Analysis (AFALD/XRS - [WPAFB])	Weapon system data (i.e., # bases, systems/bases, operating hours), logistics data (i.e., labor rates, item transportation factors) support equipment data (i.e., cost, availability) LRU & SRU factors (i.e., failure rate, repair time)	Repair level decision recommendations, to minimize cost, detailed repair level cost, sensitivity analysis information
2.1.20 OASES	Operations and Support Environment Synthesis	A/C flight hours • • WRA/SRA R&M • • Personnel skill requirements • • SE requirements • • Available skill level and SE • • Site scenario • • Spares	WRA/SRA TAT • • Availability • • Resource utilization • • Spares availability
2.1.21 OSAMM	Optimum Supply And Maintenance Manual	R/M data (MTBF, Repair Time) • • Support equipment cost • • Equipment weight and cost • • Available target • • Military Occupational Specialty (MOS) skill level and labor rate	Maintenance & repair task distribution • • Availability • • Sparing levels

ACRONYM	FULL NAME	INPUTS	OUTPUTS
2.1.22 SESAME	Selected Essential - Item Stockage for Available Method U.S. Army CECOM	System deployment by year and location • • Turnaround time • • Repair task distribution • • Maintenance task distribution • • Order ship time • • Repair time • • Failure factor	Sparing levels at each support echelon • • Total spares cost (yearly) • • Availability achieved

HOW IS THE LOGISTIC SUPPORT DATA PROCESSED?

A major task of LCC analyses is maintaining and updating a data base with the abundance of data generated during the process. This means that, sooner or later, personnel must sit down and record a large number of the relevant facts onto a spreadsheet. The spreadsheet may physically be made of paper, or it may be computer processed. The computer-assisted versions offer convenient data access, which make them much more desirable. Of what use is an important, decision-making piece of information that is buried under a stack of spreadsheets or in some dusty file cabinet?

To maintain consistent communications between departments and programs, the government has established MIL-STD-1388-2A which provides the formats to record LSA data.

Two popular means of fulfilling MIL-STD-1388-2A requirements via programs that offer computerized assistance are listed below.

ACRONYM	FULL NAME	INPUTS	OUTPUTS
2.1.23 CALSA	Computer Aided Logistic Support Analysis	MIL-STD-1388-2A input data sheets	All MIL-STD-1388-2A report formats with capability for modifying and expanding
2.1.24 LEADS	Logistic Entry and Analysis Data Systems	MIL-STD-1388-2A data elements	Sheet 2

2.2 Allocation Tools

The requirement tools of this section have be categorized with the following subheading:

- Allocation.

Presently the predominant tools for this task are military standards. Reference is first made in Table 5 (Requirement 3.1) which mentions four military standards that help provide the definitions of what is meant by allocating diagnostic capabilities. They are: MIL-STDs-499, -785, -470, and -2165.

Since the essence of system readiness is assuring that a fault-free item is available, many of the decisions within the assurance branch of engineering are based on the failure rate. The failure rate is therefore predicted first and all other assurance goals and parameters follow. We look to MIL-STD-756B for direction as to how to derive the failure rate. Of particular interest in this standard to those concerned with allocating testability/diagnostic requirements, is Task 102. This task defines and discusses how to construct a mission reliability model (2.3 & 2.4) and lays the foundation for the following modeling concerns:

- A definition of performance parameters, as well as physical and functional boundaries.
- A definition of what constitutes a failure, including mission critical thresholds.
- A definition of the environmental conditions and the periods of operation. Policies concerning use of redundant equipment are introduced here.
- A definition of mission time.
- A definition of the reliability variable of the item elements.
- The Mission Reliability block diagram, a pictorial form of a statement of what is required for mission success. Several block diagrams may be developed when requirements are not firm.

Referring once again to the 200 & 300 Series of tasks of MIL-STD-1388-1A, it is apparent that there are certain subtasks that are included to make baseline comparisons of similar systems and perform a Level of Repair analysis. Diagnostic requirements should be allocated in a manner that is familiar to the user. This is accomplished (in a roundabout way) by performing a baseline comparison. Juxtaposed to making baseline comparisons are the alternative and LOR tradeoffs. Determining at which level the diagnostics are to be performed is similar to determining where the repair is made. In other words, tradeoffs made in order to properly allocate the diagnostic capabilities to the various repair levels are similar to many of the tradeoffs performed during an LOR analysis. Reference will, therefore, be made to the guidance provided in Section 2.1 of this appendix, titled, "Establishing Requirements". Many of the models mentioned in that section will also help allocate diagnostic requirements.

Currently the Rome Air Development Center, in conjunction with the Army (CECOM), is developing methodology to perform diagnostic allocation while accounting for factors such as cost, weight and complexity. Final documentation of this program will be available from RADC in early 1990.

2.3 Optimization Tools

The requirement tools of this section have been categorized with the following subheading:

- Optimization.

  Refer to the section containing requirement tools with the subheading, "Establishing Requirements" (Section 2.1 of this Appendix). The same tools are used to optimize the design.

2.4 Risk Assessment Tools

The requirement tools of this section have been categorized with the following subheading:

- Risk.

Upon reviewing the 200 & 300 Series of tasks of MIL-STD-1388-1A, it is apparent that there are certain subtasks which are included to assess risk. The two major aspects at risk to any program are the risk of a successful mission for which the program was intended, and the risk of depleting the finances that support the program. Careful attention to minimize one of these major risk factors could directly cause the other to approach a maximum. In essence, minimizing risks is performing Availability versus Life Cycle Cost tradeoffs in relation to development schedules. Reference will therefore be made to the guidance provided in the section with the subheading "Establishing Requirements" because many of the models mentioned in that section will also help minimize risk.

Ref[1] does, however, list three models that were specifically developed to assess risk. They are listed below.

ACRONYM	FULL NAME	INPUTS	OUTPUTS
2.4.1 RISNET	Risk Network	Two activity parameters: - Time - Cost Node logic for combining activities	Probability of accomplishing the programs effort vs. time/cost - a total program effort or a segment of an effort Critical path for schedule and cost or other parameters
2.4.2 TRACE	Total Risk Assessing Cost Estimate	Cost estimation probabilities and data	Probability of risks
2.4.3 TRACE-P	Total Risk Assessing Cost Estimate P = Production	Production related risk factors (i.e., schedules, costs, performance, etc.)	Cost distributions

REFERENCES

1. AD-A179 008/8/XAB, "Logistics Support Analysis Techniques Guide," Army Materiel Command, Alexandria, Va., Corp. Source Codes: 040838000: 039150, Report No. : AMC-P-700-4, 15 MAR 85 227 p

2. Levy, Girard W., et al., "Final Report on Improved Maintenance Procedures for Inertial Guidance Systems," PRAM Program Office, AFSC, Aeronautical Systems Division: Wright-Patterson AFB (1976).

3. Bogard, D.R. et al., "Operation and Support Cost Characteristics of Testers and Test Subsystems," RADC-TR-79-334, 1980.

4. Ferrell, B.L., et al., "Tools For Integrated Diagnostics," RADC-TR-86-195, Dec. 1986, Rome, N.Y., Chapters 3.1 - 3.4.

5. Mary Eddins Earles, "Factors, Formulas, and Structures For LCC," a Mary Eddins Earles publication.

6. Contardi, R. L., "Early Test-Program Start Pays Off In Cost Savings," Test & Measurement World, June 1987.

7. RADC, Capt. Jo Sato, RADC-TM-87-11,"Availability Equations For Redundant Systems, Both Single And Multiple Repair," 1987.

3.0 DESIGN IMPLEMENTATION TOOLS

This section lists software design systems as well as a variety of other tools that are either available or under development to aid in the design and development of an effective diagnostic system.

These design implementation tools have been categorized with the following three subheadings:

- Architecture
- Design Rules and Practices
- Diagnostic Authoring
  - Diagnostic Test Strategies
  - Automatic Test Generation (ATG) or Automatic Test Program Generation (ATPG) [Generating Digital Test Vectors]

There are no claims made that this is an all inclusive list of the tools that in some way aid the task of designing and developing an effective diagnostic system. There are, perhaps, dozens of tools that are not included.

It may be well to note here also that SIT/BIT circuitry is designed and developed almost exactly in the same fashion that functional circuitry is. Therefore, the same tool aids that are targeted for functional design and development may be listed in this section. A feature that SIT/BIT designers seek, however, is the capability to design hierarchically.

A summary and listing of all tools included in each section is provided at the beginning of each section.

The entries that follow in each summary are listed in alphabetical order.

3.1 Architecture Tools

Tools that aid in the architectural design and development of an effective diagnostic capability listed in this section all have the following features in common:

- They are useful during more than one acquisition phase and apply to more than one level of integration.
- They apply to a variety of hardware technologies, including analog and digital circuitry.
- They are all versatile and capable of many design assist functions including assisting in the design and evaluation of a diagnostic capability.

A summary description of tools that aid in the architecture design of a diagnostic capability follows.

ACRONYM	APPLICATION	PRIMARY DIAGNOSTIC DESIGN ASSIST FUNCTION	COMMENTS
ADAS	VLSI/PCB/ Subsystem/ System	BIT/BIT optimization including hierarchical hardware & software design	Part of VSC (VHSIC Silicon Compiler) Tool Set; TEA was developed to be used with ADAS
A/LICE	VLSI/PCB/ Subsystem/ System/ System of Systems	Forms an outline for an Integrated Conceptual Environment (ICE) required to implement CALS philosophy	A/LICE can interface with diverse expert systems regardless of knowledge format, language or structures involved
CAD/BIT	PCB	Implementing self test onto PCBs composed of functional circuitry	The CAD/BIT library contains techniques that can be: - structured or Ad Hoc - Digital or analog - HW, SW, or both - Concurrent or non-concurrent
CADAT 6	VLSI/PCB	Fault Simulation; THESEUS ATG; supports digital HW BIT design	Post process stimuli for target tester compatibility; much speed increase with CATS accelerator
DAISY	VLSI/PCB	DTA; supports digital hierarchical HW BIT design	Popular
IDSS	VLSI/PCB/ Subsystem/ System	Provides those tools necessary to ensure an optimized diagnostic capability	Provides the capability to design a complete support system for the full life cycle of a functional design
LASAR VER 6	VLSI/PCB	JUDGE & PROSECUTOR; supports hierarchical digital HW BIT design	Post process of ATG stimuli for target tester compatibility
MENTOR GRAPHICS	VLSI/PCB/ Subsystem/ System	QUICKFAULT; supports hierarchical digital HW BIT design	Popular

ACRONYM	APPLICATION	PRIMARY DIAGNOSTIC DESIGN ASSIST FUNCTION	COMMENTS
MIDAS	VLSI/PCB/ Subsystem/ System	STAFAN; supports bottom up design of digital system diagnostics using best technology	Best is similar to & compatible with boundary scan technology
MIMOLA: MMST	VLSI/PCB/ Subsystem/ System	Automatic Self Test Program Generation in micro or machine code	Optimizes testability and Cost tradeoffs; reports poor CY & OY
SILVAR-LISCO	VLSI/PCB/ Subsystem/ System	Supports hierarchical SIT/BIT design	Thorough software tool support throughout the design process; applicable to digital and/or analog design
SPICE/PSPICE	VLSI/PCB	Supports Ad Hoc analog HW BIT design	Popular

3.1.1 ADAS

NAME: **ADAS** - Architecture Design and Assessment System

YEAR: 1988 (basic phase prototype)

FUNCTION: ADAS provides high-level system design and analysis support with capabilities to functionally model both hardware and software together. Using the ADAS graphical interface, the software and hardware designs are created together, analyzed, and refined until the system goals are satisfied. In addition to a graph editor and a consistency checker, the ADAS tool set consists of the following seven modules:

- **Software Data Flow Graph:** The logic and data flow of the conceptual algorithm is represented with colorful squares and rectangles (nodes) connected by arcs (data flow). Individual nodes can be expanded in the same manner, supporting hierarchical software design.

- **Petri Net Simulator:** Simulates the Software Data Flow Graph verifying that the sequence of execution, the amount of data generated, and the execution rates are all do-able.

- **HDL Generator:** Constructs an HDL (hardware description language) program to simulate the hardware design using the structure imposed by the Hardware Configuration Graph and the behavioral information contained in the library modules. Versions are currently available for Helix and ISPS. A VHDL version is under development.

- **Hardware Configuration Graph:** Supports hierarchical hardware design with a unique ability to model a hardware implementation of a software algorithm.

- **Petri Net Analyzer:** Generates a report containing performance statistics enabling one to determine if the design meets performance goals defined in the system specification.

- **Software Functional Simulators:** Constructs a program that allows simulation of software routines associated with nodal structure imposed by the Software Data Flow Graph. During graph simulation, each node will be able to process inputs and provide outputs. Versions are currently available for C and Ada.

- **Software to Hardware Mapping:** Promotes efficient hardware design by mapping the software graph onto the hardware graph, exposing any deficiencies or excessiveness.

CAPACITY: N/A

CPU TIME: N/A

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (PRTY)

DESIGN ENV: ADAS is complemented within a complete graphics environment and also requires:

- a tape drive for initial loading of ADAS system distribution.
- a recommended minimum working memory space of 3 Mbytes.
- a recommended minimum disk storage capacity of 10 Mbytes.

The following hardware platforms provide most elements required for the ADAS environment:

Hardware:	Operating System:
any VAX or MicroVAX	VMS, UNIX BSD 4.3, or ULTRIX (Berkeley 4.2 UNIX)
Gould PowerNode 9000	UTX 2.0
Apollo model 560 (and others)	DOMAIN/IX Version 9.5
Sun-3/160C	UNIX 3.2
VAXstation II/GPX	VMS 4.4 or higher or ULTRIX 1.2
Color Vax Stn 2000	VMS or ULTRIX

ADAS marketing can provide information on cost-effective and functional system configurations and can assist with hardware procurement.

CIRCUIT DESCRIPTION: The design is created and captured on a hierarchical graph display that can be printed out in graphical or tabular form.

USE PREREQUISITE: The user must provide conceptualized high-level inputs of the function to be designed.

DEVELOPER: Research Triangle Institute (RTI)

COMMENTS: ADAS is part of an even larger tool set known as VSC; VHSIC (Very High Speed Integrated Circuits) Silicon Compiler. VSC is predominantly composed of ADAS, VHDL, and Genesil, and aids in the design process from system specifications to mask making in the fabrication stage.

TEA, Test Engineers Assistant, listed in this section is also a product of RTI and was developed to be used in conjunction with ADAS.

REFERENCES:

ADAS Marketing Coordinator
Center for Digital Systems Research
Research Triangle Institute
P.O. Box 12194
Research Triangle Park, NC 27709
(919) 541-7436

3.1.2 A/LICE

NAME: **A/LICE** - ADA/LATTICE Integrated Conceptual Environment

YEAR: 1985 (first operational version)

FUNCTION: The CALS philosophy is to pool the knowledge of all the departmental experts so that planners of a new weapon system get as much of the big picture as possible and thereby greatly enhance system efficiency and the probability of success. The CALS knowledge base requires "super" representation scheme due to the fact that it must contain diverse specialties and often competing "ilities," including, RMA, ILS, Support, Design/Build, Planning and Technical Documentation, and others. This super representation scheme is called an Integrated Conceptual Environment (ICE).

The immediate problem of implementing such a philosophy is trying to standardize upon the knowledge representing formats, knowledge that must somehow be linked to text and graphics and also to the various expert systems emerging. A/LICE is an Ada coded program with knowledge lattice extension operators to form an outline for such an ICE and to provide a method of machine hosting ICE morphological operators in Ada. A/LICE thus presents a solution to processing the massive and diverse knowledge requirements of the CALS program. A/LICE has the following features:

- A/LICE can interface with diverse expert systems regardless of knowledge format, the language, or structures involved.
- At the meta-level, A/LICE sees knowledge objects in lattice arrays which can be processed using standard math array processors.
- The A/LICE high-level instruction set will interface with the emerging electro-optical analog computers that will use direct capture "image as knowledge" processing techniques.

Refer to the section 2.0 of this appendix for testability/diagnostic requirement tools.

3.1.3 CAD/BIT

NAME: **CAD/BIT** - Computer Aided Design For Built-In Test

YEAR: 1988 (first demonstration)

FUNCTION: A transportable, user friendly, menu driven CAD/CAE software program that automates much of the process linking BIT to the functional design. CAD/BIT consists of the following three major sections:

- **The Tutorial Phase:** Any of the CAD/BIT techniques contained in the library can be either briefly or fully described.
- **The Selection Phase:** Based on information provided by the designer, the CAD/BIT software will display suitable BIT techniques and rank them according to various Figures of Merit.
- **The Implementation Phase:** After a BIT technique has been determined, it is displayed in an implementation diagram. The designer is provided with sufficient information to add BIT circuitry similar to the way he would add functional circuitry. All circuitry added specifically for BIT is totalled for additional informational output.

CAPACITY:

CPU TIME:

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (GFE)

DESIGN ENV: CAD/BIT is written in C, operates on UNIX, and uses the IGES transfer protocol.

CIRCUIT DESCRIPTION: Information on the functional circuit block to be tested is derived from responses to questions asked the designer.

USE PREREQUISITE: Knowledge of how the system will be partitioned into physical PCBs and what functional blocks of circuitry will reside on the PCB to which BIT circuitry is to be added.

DEVELOPER: Grumman under contract to RADC.

COMMENTS: CAD/BIT, as demonstrated, is predominantly geared for PCB/module designers using discrete, off-the-shelf components. For those implementing their PCB/module design with VLSI ASICs (full or semi-custom standard cells, gate arrays, PALs, silicon compiled devices, etc.), its effectiveness tapers off; however, the tutorials, BIT selection logic, and implementation instruction still basically apply. In fact, much of CAD/BIT is quite relevant to the VLSI design process.

The CAD/BIT library contains both structured and ad hoc BIT techniques that pertain to both digital and analog circuitry. The techniques entail use of both hardware and software and can be applied to both concurrent and non-concurrent BIT execution.

CAD/BIT supports the maintenance philosophy that mandates for each PCB in the system to have the capability of testing itself and be able to report a go/no-go signal after test. Much confidence must be placed in the reliability of this go/no-go signal, particularly in military systems. When done properly and reliably, a PCB self test system dramatically impacts the following:

- Fault isolation time/ system readiness.
- Factory- and Depot-level system verification testing complexity.
- Organizational maintenance personnel skill level.

REFERENCES:

1. Lt T.W. Oxford, "Computer Aided Design For Built-In Test (CADBIT)," ATE & Instrumentation Conference, June 1987.

2. Air Force Point of Contact
RADC/RBES
Griffiss AFB, NY 13441-5700

3.1.4 CADAT

NAME: **CADAT 6**

YEAR:

FUNCTION: CADAT 6 is composed of the following related tools operating from a single user data base:

- LOGIC SIMULATION; to verify the base functionality of the designer's concept and detailed logic design.
- PERFORMANCE SIMULATION; to analyze the circuit propagation characteristics of the design under worst case timing conditions.
- FAULT SIMULATION; to evaluate the effectiveness of test vectors developed by the designer or the test department for manufacturing test of the final design.

The CADAT Fault Simulation option utilizes a functional Concurrent Fault Simulation algorithm to analyze the impact of potential manufacturing errors and circuit failures. The algorithm optimizes simulation throughput for all levels of device modeling, from switch level to hardware.

CAPACITY:

CPU TIME:

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (PRTY)

DESIGN ENV: CADAT 6 will run on the following hardware platforms:

Apollo/AEGIS with AUX
IBM PC-AT/PC-DOS (Personal Cadat)
IBM MVS
IBM VM/CMS
SUN Microsystems UNIX
VAX ULTRIX
VAX VMS

LANA (Local Area Network Acceleration) is also available.

CIRCUIT DESCRIPTION: Behavioral Design Language (BDL)

USE PREREQUISITE: Behavioral description or a circuit netlist must be input.

DEVELOPER: HHB Systems

COMMENTS: The following are CADAT related HHB products:

Personal CADAT	An integrated logic, timing and fault simulator for use on the IBM Personal Computer.
CATS Modeler	Enables designers to incorporate physical model of LSI and VLSI chips into logic, timing and fault simulations.
CATS Accelerator	High-speed simulation on a special purpose system (300 times faster than a VAX/780).
THESEUS	Automated test generation for ICs; combinational,sequential, and scan path circuits.
Extensive Device Library	Software models of SSI/MSI devices, LSI Logic gate arrays, and Standard Microsystems' standard cells. Hardware models of LSI/VLSI devices from Intel, Motorola, TI, Zilog, and others.

In order to maximize the utilization of relevant test data created during the design process, post processor links to a growing range of test systems have been developed. These range from go/no-go links with IC test systems, such as Sentry, to complete diagnostic data base links with leading high performance PCB test systems, such as those from Factron, Teradyne and Computer Automation.

In keeping with industry trends toward user integration and open system architecture, CADAT accepts network information developed on any system having an EDIF output format. Implementation of this industry standard is further enhanced in CADAT 6 by the addition of a totally new data base access package known as the CADAT Systems Interface (CSI).

CSI allows the user to access all aspects of the simulation data base such as stimulus responses, and topology information, with a series of high-level call functions. These can be used to derive output data from CADAT in any desired format incorporating any parts of the internal data base developed during simulation.

REFERENCES:

HHB Systems
Attn: Mr. Kenneth Lipston
1000 Wyckoff Ave.
Mahwah, N.J. 07430
(201) 848-8000

3.1.5 DAISY

NAME: **DAISY**

YEAR:

FUNCTION: Daisy Logician Workstation is Intel 80286 based and supports DTA along with GATEMASTER, CHIPMASTER, BOARDMASTER, and MEGAGATEMASTER. Validation by Daisy Logic Simulator ensures the design is ready for DTA. Daisy's recommended design sequence is: DANCE - DRINK - SIFT - SOM - DTA.

Refer to Daisy Testability Analyzer (4.1.6).

3.1.6 IDSS

NAME: **IDSS** - Integrated Diagnostic Support System

YEAR: 1989

FUNCTION: The Navy's Integrated Diagnostic Support System is developing an innovative approach to diagnostics via the development of an integrated set of software tools. All of these tools function within the context of a standard common diagnostic data base (CDDB). The CDDB is a predefined set of logistic and technical data elements derived from weapon system design and logistics data. Preprocessors are required to input CAE and LSA data into the CDDB. Tools currently under development are:

- Weapon System Testability Analyzer (WSTA)
- Adaptive Diagnostic Subsystem (ADS)
- Adaptive Diagnostic Authoring Tool (ADA)
- Feedback Analyzer (FA)
- Technical Information and Training Authoring (TIATA) Tool

A brief description of each software tool is provided below.

WEAPON SYSTEM TESTABILITY ANALYZER (WSTA)

The WSTA requires as inputs unit under test digital or analog topology and logistic support analysis (LSA) data. WSTA then generates a test strategy which is very near optimal in terms of minimizing average test times or test costs. A primary function of WSTA is to provide static (topological) testability figures of merit, such as average inherent ambiguity group size and feedback loop characteristics. WSTA also provides dynamic (test strategy based) figures of merit, such as mean or maximum time to fault isolate. WSTA provides guidance to the designer on the optimal placement of test points based on the fault isolation data each test point can provide. WSTA utilizes a system dependency model and the time-efficient sequencer of tests (TEST) algorithm to generate an optimal test strategy. WSTA is currently in BETA site testing and is scheduled for distribution to interested government and contractor organizations during 1989.

ADAPTIVE DIAGNOSTIC SUBSYSTEM (ADS)

The ADS is an intelligent troubleshooting aid which provides a recommended "next-best" test or a recommended repair action. The ADS contains logic to utilize the diagnostic resources (e.g., BIT, augmented BIT, guided manual test, technician inputs, etc.) and diagnostic reasoners (e.g., optimal WSTA strategy, original dependency data, production rules, etc.) which will have the most likelihood of success based on the diagnostic session results up to that point. The

ADS applies increasingly sophisticated "level of intelligence" to the problem, but only as needed. It remembers its successes and failures and automatically adapts its test strategy to changing failure rates and test environments. The ADS may be utilized as a diagnostic software shell for both embedded and off-line test program set (TPS) applications.

ADAPTIVE DIAGNOSTIC AUTHORING (ADA)

The ADA validates the model and optimal strategy generated by WSTA; organizes weapon system-specific data from the IDSS data base; adds production rules to update existing strategy, based on the analysis of on-going weapon system performance; and authors the diagnostic program, which is then provided as input to the ADS.

FEEDBACK ANALYZER (FA)

The FA is a software aid that gathers global feedback failure data (e.g., type of failure, fault symptoms, environmental conditions, etc.) and performs an analysis to determine the significance of the failure as it relates to enhancements of fault detection and isolation strategy. The ADA utilizes the output of the FA to aid the user in authoring new production rules for addition to the existing diagnostic procedures.

TECHNICAL INFORMATION AND TRAINING AUTHORING TOOL (TIATA)

The TIATA is a software aid which uses weapon-specific data from an IDSS data base to generate maintenance procedures, tutorials, circuit schematics, or parts breakdown diagrams needed by the maintenance technician as part of the diagnostic process. This tool also has the capability to generate tailored training sessions, enhance maintenance reporting, and facilitate user interaction with the IDSS data bases.

CAPACITY: N/A

CPI TIME: N/A

APPLICATION: VLSI PCB SUBSYS SYSTEMS

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN? Y/N (GFE)

DESIGN ENV: To properly execute the Berkeley CAE preprocessor, the LSA preprocessor and Weapon System Testability Analyzer Program requires either a Sun 3/160 computer, with UNIX 42 (Berkeley version) operating system, or a

Digital Equipment Corporation Micro Vax II, with a Virtual Memory System. A minimum of four Mbytes of main memory is required, with 16 Mbytes of virtual memory for the Sun, and four Mbytes main/16 Mbytes virtual memory for the Micro Vax II.

Additional information concerning hardware platform requirements needed to properly execute the remainder of the IDSS tool set will be provided upon product release of each individual tool.

CIRCUIT DESCRIPTION: All technical information, CAD/CAE data, and ILS data are processed into standard CDDB format and are accessible to each of the four IDSS design tools.

USE PREREQUISITE: CDDB data elements required as data inputs for each particular tool must be provided.

DEVELOPER: Harris Corporation under contract from the US Navy.

COMMENTS: A key design concept of IDSS is the delivery of complete support over the full life cycle of the weapon system.

REFERENCES:

1. Dr. Bruce J. Rosenburg, "The Navy Integrated Diagnostic Support System - System Overview, Architecture and Interfaces," IEEE AUTOTESTCON 1987.

2. Navy Point of Contact
NAVSEA - Code CEL-DS
Washington, DC 20362-5101
(202) 692-2035/2036

3.1.7 LASAR VERSION 6

NAME: **LASAR VERSION 6**

YEAR: 1982, origin 1978

FUNCTION: It provides a CAD simulation system for design verification and test program generation incorporating the testability subprograms called Judge & Prosecutor. Refer to Section 3.3.2.

CAPACITY:

CPU TIME:

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (PRTY)

DESIGN ENV: VAX; VALID S320; TERADYNE DATA SERVER; IBM PC AT; also supports design entry.

CIRCUIT DESCRIPTION: TML - an optional register transfer language.

USE PREREQUISITE: LASAR Version 6 netlist format; translators available to convert leading CAD/CAE circuit data bases.

DEVELOPER: Teradyne

COMMENTS: When LASAR is used for both design verification and test program generation, test programming time shrinks by 50% or more. There's no need to recreate the circuit model or stimulus vectors. Only one set of models must be maintained. Test development can begin immediately after design, in parallel with prototype production and verification.

LASAR VERSION 6 supports the following modeling features:

There is a library of over 4,000 SSI, MSI, and LSI device models and gate array macros, including manufacturer's timing, input loading delay factors, and current drive strength specifications.

Behavioral modeling language.

Hardware modeling with behavioral software enhancement for repeatable simulation, min/max timing analysis, and fault simulation.

RAMGEN, ROMGEN, PLAGEN, PLAGEN programs for rapid structural modeling of repetitive logic devices.

Tester characteristics are taken into account in parallel with circuit test generation and are, therefore, easily integrated with ATE hardware, minimizing the time required to debug tests.

Post-processors translate stimulus and response data into the symbolic language of the target test system for go/no-go test and fault diagnosis.

REFERENCES:

1. Teradyne, Inc.
   Attn: Fred Grant
   321 Harrison Ave.
   Boston, Ma. 02118
   617-482-2700

2. Mary Wasilowiki, "Simulation Modeling of the Test System Environment to Speed Board-Test Program Debugging and Tester Integration," International Test Conference 1987

3.1.8 MENTOR GRAPHICS

NAME: **MENTOR GRAPHICS**

YEAR:

FUNCTION: Mentor Graphics provides a design system integrated with software support tools that relate to design of function as well as to design of the diagnostic capabilities of that function. In particular, it provides a deterministic fault simulator. Refer to the Tools For ATG/Fault Simulation section.

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (PRTY)

DESIGN ENV: Mentor Graphics engineering workstations

CIRCUIT DESCRIPTION: See above for all Mentor Graphics model types

USE PREREQUISITE: Schematic must be captured using a Mentor Graphics model type; use with QUICKSIM.

DEVELOPER: Mentor Graphics Corporation

COMMENTS:

REFERENCES:

Mentor Graphics Corporation
8500 S.W. Creekside Place
Beaverton, Oregon 97005-7191
(503) 626-7000

Frank Binnendyk
Product Manager
Design and Analysis Div

3.1.9 MIDAS

NAME: **MIDAS** - Modular Integrated Design Automation System

YEAR: 1985 (Periodic updates)

FUNCTION: Full simulation support for selected semiconductor technologies (CMOS gate arrays, TTL MSI logic, etc.), from gate level simulation to subsystem and system scale. MIDAS supports logic, timing and fault simulation.

In 1987 N.2 was added for architectural and RTL level simulation. This tool has a graphic interface on Apollo workstation. This interface is a set of block symbols, which can be used to draw the RTL level diagrams. The netlist is produced automatically on the workstation.

Using the MIDAS simulation tools with N.2 as a front end, the simulation process is as follows:

- Define architecture and timing constraints using N.2
- Define subsystems using N.2
- Enable software simulator using the subsystem level models
- Break subsystems into individual chips
- Verify chip designs using logic, timing and fault simulators (MIDAS)
- Verify chips in subsystem environment (MIDAS)
- Verify subsystem in system environment (MIDAS)

CAPACITY: Targeted for one chip through large designs.

CPU TIME: Depends on system size and simulation methods used.

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (PRTY)

DESIGN ENV: CYBER mainframes with Apollo workstation front end

CIRCUIT DESCRIPTION: MIDAS Logic Interconnect Language netlist. (VHDL in the future.)

USE PREREQUISITE: Ample definition of a function to be designed and implemented with gate arrays.

DEVELOPER: MIDAS - Control Data Corporation
N.2 - ENDOT, Inc.

COMMENTS: STAFAN is a part of the MIDAS toolset. See Ref[1] and refer to the section: Tools That Aid in Fault Simulation Tasks.

A significant feature of the consultant services offered by MIDAS personnel, is that they will offer instruction on how to implement BIST (Built-In Self Test) into the functional design. Control Data has a number of implementations for BIST, under the names of OCMS (for 6000-gate arrays), BEST (for 20,000-gate arrays) and VISTA (for standard cell).

These built-in self test circuits support chip as well as system level-test. See Refs[2]&[3].

REFERENCES:

1. Jain, S.K., Agrawal, V. D., "STAFAN: An Alternative to Fault Simulation, Design Automation Conference, 1984

2. Ron Lake, "A Fast 20K Gate Array With On-Chip Test System," VLSI Systems Design (magazine), June 1986.

3. David R. Resnick, "Testability and Maintainability with a New 6K Gate Array," VLSI Design (magazine) Mar/April 1983.

4. Control Data Corporation
ADAM Marketing, Minneapolis, MN
Attn: Robert Biggs HQM274
(612) 853-3117.

3.1.10 MMST

NAME: MMST - Mimola Module For Self Test
MIMOLA - Machine Independent Microprogramming Language

YEAR: 1979

FUNCTION:

- Automatic self test program generation which can be implemented in micro or machine code.
- Optimizes TY and cost tradeoffs.
- Reports poor CY and OY.

CAPACITY: Circuit nodes on the MIMOLA design level normally are register-transfer modules, that is complete ALUs, RAM, ROM, registers, multiplexers, buses, etc. Only in special cases modules are simple logic gates. The current implementation allows several hundred of such circuit nodes. This number could be increased, but up to now even in the most complex applications it has never been a limitation.

CPU TIME: (Example) 3,000 fault free instructions took 20 min.

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (PRTY)

DESIGN ENV: MIMOLA is written in pure standard Pascal and runs on any machine and operating system that has a Pascal compiler and at least one megabyte of unsegmented main memory. Among others, applications are running on VAX (VMS, UNIX), Sun, and Apollo workstations.

CIRCUIT DESCRIPTION: CHDL; part of MIMOLA

USE PREREQUISITE: Functional model of the target circuit.

DEVELOPER: Institut fur Informatik u. Prakt. Math.

COMMENTS: Most unique about MSST is that it allows automatic generation of self test programs in binary code for complete processor systems, including paths of high sequential depth. Such self test programs for system-level test are usually still written by hand and are tedious, requiring highly skilled specialists.

The MIMOLA Design System is a product funded by the German government. It is generally available but a contribution to the development cost is expected if it is to be used commercially.

REFERENCES:

- Kruger, G., "Automatic Generation of Self Test Programs - A New Feature of the MIMOLA Design System," IEEE DAC 1986.

- For a trial version contact:

Dr. Peter Marwedel
Institut fur Informatik u. Prakt. Math.
Universitat Kiel
Olshausenstr. 40-60
D-2300 Kiel 1

3.1.11 SILVAR-LISCO

NAME: **SILVAR-LISCO**

YEAR: 1981

FUNCTION: The Silvar-Lisco CAE Software Environment is an all inclusive design aid, assisting in almost every design stage from schematic capture to the final step before manufacturing. The software environment includes:

- **SDS** (Schematic Design System): SDS creates or "captures" the design data base, which stores the basic logic information used throughout the Integrated Design Environment using a multi-window technique which allows the designer to view various schematics and to work on several levels of design simultaneously.

- **HELIX**: A hierarchical, top-down, behavioral logic simulator. HELIX allows designers to prove the principle of their conceptual design at its block diagram stage. When and if satisfied, the details continue to be worked out at the register level and then the gate level.

- **ANDI** (Analog-Digital Simulation): Provides functional time domain simulation of mixed analog/digital sampled data systems. ANDI's analog primitives are:

  -MOS transistor switches
  -Bipolar transistors
  -Diodes
  -Independent and controlled voltage and current sources
  -Inductors
  -Resistors
  -Capacitors

- In addition to the above analog primitives, ANDI services common digital primitives such as gates, ROM & RAM, PLAs, flip-flops, etc., as well as A/D and D/A converters.

- **SWAP**: Provides time and frequency domain simulation of switched-capacitors networks. Non-linear transistor models are simplified to MOS switches. Frequency domain calculations are made directly in the frequency domain, with no need to be followed by a Fast Fourier Transform (FFT). SWAP is conducive to a broad range of analyses including sensitivity, distortion, band-scan, and noise analyses.

- For mathematical reasons all analyses, except distortion analysis, are limited to switched-capacitor networks with only linear components. For efficiency sake, all high-level analyses, including distortion analysis, require that the circuit does not have any time constants caused by resistors or inductors.

CAPACITY:

CPU TIME:

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (PRTY)

DESIGN ENV: In addition to the above utilities, Silvar-Lisco also provides a printed circuit board design system utility as well as the following numerous utilities in the IC development realm:

IC Development

PRINCESS: A full custom physical design system.

GARDS: A gate array design system.

CAL-MP: A standard cell design system.

Verification

UDRC: A universal design rule checker.

Electrical Verification: A set of verification utilities including six checking utilities, a circuit extractor, and a SPICE netlist generator.

YIELD: A geometric yield analyzer.

Mask Data Preparation

UMDP: A universal mask data preparation utility.

Silvar-Lisco CAE products operate on Apollo, IBM, Sun, and VAX computing systems, however, not every utility runs on each of these four platforms.

CIRCUIT DESCRIPTION: HELIX Hardware Description Language (HHDL) provides the ability to write behavior for components, describe parts with complex behavior, and model concurrent design functions.

USE PREREQUISITE: See SDS above. SDS translates design data into netlist format, or its functional equivalent. See comments for various interfaces available.

DEVELOPER: Silvar-Lisco

COMMENTS: The Silvar-Lisco CAE software environment is ideal for System Integrated Test (SIT) designers in that they can simulate a conceptualized SIT and work out many of the typical problems (i.e., maintenance bus interfacing, ETE compatibility, BIT logic flow, test sequencing, etc.) and then allocate that portion of the overall scheme to the embedded BIT designer at the PCB level, who can similarly pass on his scheme to the VLSI people who in turn can play their own BIT games. Furthermore, its capabilities of handling both digital and analog circuitry, as well as synchronous and asynchronous designs, make it even more attractive.

Optional plotting packages are available for generating documentation on widely used plotters.

An optional software package, Engineering Access Routine Set (EARS), provides direct access to the central data base promoting the ability to write interfaces to proprietary software. A report-generation utility can be made for example.

Optional interfaces support the following simulators and systems:

- TEGAS
- SPICE
- SALOGS
- LOGCAP
- ILOGS
- SCI-CARDS system
- HILO
- SUPER-COMPACT
- Applicon system
- CBDS system
- REDAC-CADET system.

Users must purchase Fault Simulators, ATPGs, and/or Testability Analyzers from outside sources as they are not provided by Silvar-Lisco.

Profile (see Prediction Tools) uses the ANDI simulator during its process.

REFERENCES:

Silvar-Lisco
Corporate Headquarters
1080 Marsh Road
Menlo Park, CA 94025
(415) 324-0700

3.1.12 SPICE/PSPICE

NAME: **SPICE/PSPICE**

YEAR: 1984 (PSPICE)

FUNCTION: Although SPICE does not provide design aid specific to the task of integrating diagnostic capabilities into a functional design, it is one of the few software tools available for analog design simulation. Soft simulation before a hard prototype is a prime advantage of being able to assess and predict the merits of a function's diagnostics.

There are various versions available of the analog simulator known as SPICE. SPICE 2 was developed at the University of California at Berkeley in the late 60's and early 70's. PSPICE originated from SPICE 2 and is the chosen version for this document due to its popularity. This fact sheet will concern facts about PSPICE.

CAPACITY:

CPU TIME:

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (PRTY)

DESIGN ENV: IBM PC and PS2 compatibles (DOS) (Aug 1 OS/2), Macintosh II, Sun 3/4, VAX (Micro, supermini, and mainframe).

CIRCUIT DESCRIPTION: Compatible with a variety of commercially available schematic capture programs.

USE PREREQUISITE: Circuit devices must be modeled using either the model library of parts included or developed from data sheet information using "Parts," an optional utility.

DEVELOPER: PSPICE is developed by MicroSim Corp.

COMMENTS: PSPICE performs the following types of analyses:

- DC, or bias point, voltages and currents of the circuit.
- AC, or frequency, response of the circuit.
- Noise behavior of the circuit over frequency.

PSPICE is complemented with the following available options:

- **Device Equations:** Used to change the equations which translate model parameter values and terminal voltages into the device's currents and capacitances.
- **Probe:** Functions as a software oscilloscope with the capability to produce a hardcopy printout.
- **Parts:** Semi-automates the process of creating model libraries.
- **Monte Carlo Analysis:** After inputting various component tolerances, PSPICE analyses are performed using random values that are somewhere in the range of tolerance.
- **Digital Files:** Allows one to run a digital simulator and use the results as input to PSPICE or vice versa.

It must be emphasized that PSPICE is not marketed as a design aid peculiar to the inclusion of the TESTABILITY/DIAGNOSTIC capability. However, one must remember that much of analog BIT design is accomplished quite successfully with the following ad hoc techniques: peak detector or window comparator, wraparound, substitution of a known value, etc. There is little guidance required to describe how to implement these techniques, however, simulators such as PSPICE enable one to optimize the design before it is breadboarded.

Furthermore, although much more tedious than using a fault simulator of the digital world, the Device Equations option is useful in determining if BIT will detect a component failure. The Probe option is handy for documenting troubleshooting manuals. Considering that BIT data processing is almost always done digitally, the Digital Files option should prove worthwhile.

REFERENCES:

MicroSim Corporation
Attn: Michael Taggart
20 Fairbanks
Irvine, CA 92718
(714) 770 - 3022; (800) 826 - 8603

3.2 Design Rules and Practices

Tools that aid in incorporating design rules and practices into a diagnostic capability that are listed in this section all have the following features in common:

- They are useful during more than one acquisition phase and, with the exception of TDES, app to more than one level of integration.
- They apply to a variety of hardware technologies, including analog and digital circuitry. [TDES applies only to digital.]
- They are all versatile and capable of many design assist functions. Those functions relative to this document being: assuring standard design for testability practices, deploying rules that implement a diagnostic capability, and standardizing diagnostic data and tester information transport.

A summary description of tools that aid in incorporating design rules and practices into a diagnostic system is provided in the following Table:

ACRONYM	APPLICATION	PRIMARY DIAGNOSTIC DESIGN ASSIST FUNCTION	COMMENTS
IDSS-ADS	SUBSYSTEM/ SYSTEM	SELECTS & INTERPRETS TESTS FOR FD & FI	MAY BE APPLIED TO MANY DIFFERENT CLASSES OF SYSTEMS INCLUDING: - ANALOG AND/OR DIGITAL - MECHANICAL - HYDRAULIC, ETC.
TDES	VLSI	PROVIDES SYSTEMATIC DIGITAL DFT METHODOLOGY	TDES IS AN EXPERT, KNOWLEDGE-BASED SYSTEM THAT ISA PART OF ADVANCED DESIGN AUTOMATION (ADAM) SYSTEM OF TOOLS
TEA	PCB/SUBSYSTEM SYSTEM	DFT GUIDANCE; BIT HW RECOMMENDATIONS; BIT HW COST ASSESSMENT; BIT HW PLACEMENT RECOMMENDATIONS	USED IN CONJUNCTION WITH ADAS; DEVELOPED VERSION OF NOT MUCH USE; ENHANCED TEA IS YET TO BE DEVELOPED
TISS	VLSI	PROVIDES A CAPABILITY FOR INTERDEPARTMENT ACCESS INTO ONE, TEST RELATED, DEVICE INFORMATION DATA BASE	PROMOTES EFFICIENT INFORMATION ACCESS DURING THE TESTABILITY/ DIAGNOSTIC DESIGN PROCESS
TESTABILITY CHECKLIST	PCB/SUBSYSTEM	PROVIDES STANDARD DESIGN FOR TESTABILITY PRACTICES IN THE FORM OF A CHECKLIST	QUICK AND INEXPENSIVE

3.2.1 IDSS-ADS

NAME: **IDSS - ADS** - Adaptive Diagnostic System

YEAR:

FUNCTION: A knowledge-based, expert system that assists in weapon system fault detection and isolation, by selecting and interpreting tests. The ADS also updates the data after the fault isolation session is over. The ADS has the following attributes:

- The ADS employs a generic diagnostic process capable of operating upon a variety of application-specific knowledge bases.
- The ADS minimizes on-line computational load and the speed of fault isolation by first utilizing the simplest system model and then progress, if necessary, to more complex models. See the third comment below.
- It maintains a data base of fault histories which is used to bias future test and repair recommendations. This fault base is also used to flag occurrences when system behavior is inconsistent with the system model.

CAPACITY:

CPU TIME:

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (GFE)

DESIGN ENV:

CIRCUIT DESCRIPTION: N/A

USE PREREQUISITE: The ADS will utilize the following type of data:

- Logistic data including component failure rates, test execution costs, etc.
- Various forms of design and testability analysis data, as well as functional interdependencies developed during the design process.
- Knowledge concerning the operational environment and situations experienced during operation.
- Technical information via a portable maintenance aid interface.
- BIT initiation and test result information.
- Fault history information updated after every fault isolation session.

DEVELOPER: Harris Corporation under contract from the US Navy.

COMMENTS: After reviewing the type of knowledge upon which the ADS operates, one realizes that these application-specific knowledge bases can be largely developed from the structured design process promoted by the IDSS design philosophy. Thus, the costs traditionally associated with the development of knowledge bases required for expert systems are greatly minimized.

The generic nature of the ADS enables it to be applied to many different classes of systems, including digital and/or analog avionics, mechanical, hydraulic, and many hybrid type systems.

The ADS pre-computed test tree is built using the Time-Efficient Sequencer of Tests (TEST). If this proves to be inadequate, ADS will deploy logic modeling. If this also proves to be inadequate, ADS will rely on local cause and effect rules, Ref[2].

This tool has been categorized under the subheading "Design Rules and Practices" because it is a rule based expert system that is composed of standard techniques and practices used to diagnose weapon system faults. One may argue that ADS is more a troubleshooting tool than a tool to insure standard diagnostic procedures are used, however, it seemed appropriate to include it in this section.

REFERENCES:

1. Dr. Bruce J. Rosenburg, "The Navy Integrated Diagnostic Support System - System Overview, Architecture and Interfaces," IEEE AUTOTESTCON 1987.

2. Magliero, A., R. Leong, and R. Bethel, "ADS - The IDSS Adaptive Diagnostic System," IEEE AUTOTESTCON 1987.

3. Navy Point of Contact
NAVSEA - Code CEL-DS
Washington, DC 20362-5101
(202) 692-2035/2036

3.2.2 TDES

NAME: **TDES** - Testable Design Expert System

YEAR: 1985 (prototype)

FUNCTION: A knowledge-based expert system dedicated to the following design assistant services:

- Provide a systematic DFT methodology.
- Apply measures and attributes (ie., fault coverage, area overhead, etc.) to various test strategies.
- Partition the total circuit into testable blocks. Logic is divided into three basic structures, combinational logic, registers, and RAMs. The basic structures are further divided into design styles. For example, the design styles of combinational logic are PLAs, ROMs, and random logic.
- Match a particular block of circuitry or kernel that has a particular design style with an effective test strategy.
- Add circuitry as required to establish uniformly structured testable circuits.

CAPACITY: Large VLSI circuits; from 5K to 100K transistors

CPU TIME: 1 - 2 hours; not including ATPG or fault simulation.

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N

DESIGN ENV: The prototype has been implemented in Lisp and runs on a Sun and DEC-20. It is part of the Advanced Design Automation system, ADAM, see Ref [3].

CIRCUIT DESCRIPTION: The design is input into TDES using Register Transfer Level (RTL) descriptions (PLAs, registers, buses, RAMs, ROMs, etc.).

USE PREREQUISITE: Design goals and constraints are input into TDES along with the RTL descriptions.

DEVELOPER: University of Southern California

COMMENTS: TDES applies to digital circuitry.

REFERENCES:

1. Abadir, M.S., and M.A. Breuer, "A Knowledge-Based System for Designing Testable VLSI Chips," IEEE Design and Test of Computers, August 1985.

2. M. A. Breuer, "A Methodology for the Design of Testable VLSI Chips," Proc., IEEE Workshop on Test Environments, pp. 105 - 114, Sept. 17, 18, 1985.

3. Abadir, M. and M. A. Breuer, "Test Schedules For VLSI Circuits," IEEE Trans. on Computers, vol. c-35, pp. 361 - 367, April 1986.

3.2.3 TEA

NAME: **TEA** - Test Engineers Assistant

YEAR: 1988 (basic phase prototype)

FUNCTION: TEA provides high-level system testability design support in conjunction with high-level functional design assistance provided by ADAS (Architecture Design and Assessment System). TEA consists of the following five modules:

- **Design for Testability Guideline Checker:** Identifies violations of DFT guidelines.
- **BIT Hardware Recommendation:** Provides advice as to which BIT method, deterministic or pseudorandom, is most adequate for the circuit at hand.
- **BIT Cost Assessment Module:** Predicts the general cost of implementing a particular BIT technique in terms of additional PCB real estate and I/O ports.
- **BIT Placement Recommendation:** Guides the designer as to where BIT components and testpoints should be placed and then calculates the costs associated with such an addition.
- **Testability Facilities Cost Assessment:** Accounts for each incremental change included for testability and provides the total cost of implementing them.

CAPACITY:

CPU TIME:

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (GFE)

DESIGN ENV: TEA was developed to be used with ADAS (Architecture Design and Assessment System). See ADAS for design environment requirements.

CIRCUIT DESCRIPTION: VHDL

USE PREREQUISITE: High-level description of the functional circuitry that requires testability.

DEVELOPER: Research Triangle Institute (RTI)

COMMENTS: Usefulness of TEA will increase substantially upon the completion of Enhanced TEA.

Enhanced TEA conceptually has the following additions:

- Fully populated DFT rules, BIT techniques, and BIT modules data base.
- Fully automated AI-based module interfacing.
- Fault-tolerance design capability
- Design for prognostic capability.
- Extension of TEA for Monolithic Microwave Integrated Circuits (MMIC) to be known as TEAM.

Interfaces to TISSS (included in this section).

REFERENCES:

1. ADAS Marketing Coordinator
Center for Digital Systems Research
Research Triangle Institute
P.O. Box 12194
Research Triangle Park, NC 27709
(919) 541-7436

2. Army Point of Contact
U.S. Army - CECOM
Attention: AMCPM-TMDE-LT
Ft. Monmouth, NJ 07703
(201) 532-1447

3.2.4 TISSS

NAME: **TISSS** - Tester In Support Software System

YEAR: 1988

FUNCTION: The primary f TISSS is to automate the generation and maintenance of product ans and test programs for Very High-Speed Integrated Circuits (VHSIC) complex Very Large-Scale Integrated (VLSI) devices.

To accomplish ti provides a means of capturing computer-aicied design (CAD) data oduct specifications. TISSS also aids in the generation of these data seaptured, the data is automatically loaded into the TISSS data base, whentained in a standardized, transportable, and computer-accessible formdata sets are accessed by TISSS to generate product specifications and .ms. TISSS also provides tools for validating the data sets to ensure cor of data, proper syntax, semantics, and data intent.

CAPACITY:

CPU TIME:

APPLICATION: VLSI PCB

ACQUISITION PHASE: CONCEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN? Y/N (GI

DESIGN ENV: TISSS is aystem written in Ada. Although designed to be portable, with minimal ns, TISSS was developed on, and uses, the features of the Digital Eorporation VAX computer systems. The minimum TISSS hardware/quirements are:

- Micro Vax IIh floating point coprocessor, or a VMS-compatible DIh higher performance
- Necessary cxtures
- 5 MB main m
- KDA50 disk c
- TK50 95 MB t

- o DHV11 8-line asynchronous multiplexer
- o Diagnostics and hardware
- o RA81-EA three disks in second 40" cabinet; disk capacity 456 MB each
- o VT240 graphics terminal
- o MicroVMs
- o Ada License (DEC)
- o Oracle Data Base Management System License (Oracle Corp.)
- o Test Description Macro Skeleton License (DEC)
- o PALETTE grahics license (Palette Corp.)

OPTIONAL:

- o HITS (government furnished)
- o HILO (GenRad Corp.)
- o VHDL Support Environment (government furnished).

An estimate of the on-line disk storage necessary for an operational TISSS system is based on the following table:

STORAGE SIZE IN BLOCKS
(Block=512 bytes)

Item	Required	Optional
TISSS executable code	21,000	
MI executable	18,000	
On-line software users manual	5,000	
All TISSS documentation		73,000
HILO simulator executable		34,000
HITS simulator executable		19,000
VHDL analyzer executable		16,000
Oracle software library	22,500	
PALETTE executable code	6,600	
TDMS	23,000	

TISSS data base working area (20K gates, one device)	550,000	
VHDL design library (transient)		20,000
Test vector (transient)		50,000
Paging space (20K gate device, single user)	300,000	
TOTAL	1,148,6000 Blocks	202,000 Blocks
	or	
	574.3 Mb	108 Mb

REFERENCES:

Point of Contact:
Bill Russell, RADC/RBR
Griffiss AFB, NY
315/330-3974

3.2.5 Printed Circuit Board Testability Design Guide and Rating System

NAME: **Printed Circuit Board Testability Design Guide and Rating System**

YEAR:

FUNCTION: A methodology was developed during an Air Force-sponsored study that accurately evaluates the testability merits of a printed circuit board (PCB). This is accomplished through a "Figure of Merit" rating system that weights the "difficult to test" and "easy to test" aspects of a circuit design.

The principal output of this study is an extensive Testability Design Guide that describes how testability problems associated with circuit structure can be corrected. The design guide works hand-in-hand with the rating system so that the rating system identifies the nature and extent of the current testability problem, and the guide provides the means to correct the design deficiencies.

CAPACITY: N/A

CPU TIME: N/A

APPLICATION: VLSI PCB SUBSYS SYSTEM

PUBLIC DOMAIN? Y/N

DESIGN ENV: N/A

REFERENCES:

Consolia, W.M., Danner, F.G., "An Objective Printed Circuit Board Testability Design Guide and Rating System", Rome Air Development Center (RADC) Technical Report TR-79-327, January 1980.

3.3 Diagnostic Authoring Tools

This section lists software design systems, as well as software tools, that are either available or under development to aid in the design and development of an effective diagnostic system.

Diagnostic Authoring Tools may be further subdivided into two (2) major categories: Generic Diagnostic Authoring Tools and Automatic Test Generation Tools. Information pertaining to each of these tool types is provided in the ensuing paragraphs.

3.3.1 Generic Diagnostic Authoring Tools

Those types of tools are predominantly concerned with authoring optimized test sequences, techniques, procedures, or technical information in support of the diagnostic design process.

Tools that aid in diagnostics authoring during design and development have the following features in common:

- They are useful during more than one acquisition phase and apply to more than one level of integration.
- They apply to a variety of hardware technologies, including analog and digital circuitry.
- They are all versatile and capable of many design assist functions. That function capability relative to this document being: assisting in the various authoring tasks required to design and evaluate a diagnostic capability.

A summary description of tools that aid in diagnostic authoring during design are provided an the following Table:

ACRONYM	APPLICATION	PRIMARY DIAGNOSTIC DESIGN ASSIST FUNCTION	COMMENTS
CATA	VLSI/PCB/ SUBSYSTEM	DFT GUIDANCE; FI STRATEGY; TY ANALYSIS AND ATG	APPLICABLE TO: - INCOMING INST - MFG TESTING - SYSTEM MFG TEST - PRDCTN VERIFICATION - SYS MAINTAINABILITY
GIMADS DIAGNOSTIC LIBRARY	VLSI/PCB/ SUBSYSTEM	REFERENCE SOURCE FOR FI STRATEGIES	IDEAL TO USE IN CONJUNCTION WITH FMEA DEVELOPMENT
IDSS - ADA	SUBSYSTEM/ SYSTEM	AUTHOR DIAGNOSTIC RULES AND PROCESS AND DEVELOP TEST PROCEDURES	FUNCTIONS AS A BRIDGE TO PASS INFO FROM DESIGN TO SUPPORT
IDSS - TIATA	PCB/SUBSYSTEM SYSTEM	PROVIDES INTERFACE FOR AUTHOR/EDITOR FOR CREATING OF TECHNICAL TRAINING MATERIAL	PROMOTES PAPERLESS DISSEMINATION OF TECHNICAL INFORMATION
TGIR/FIS	PCB/SUBSYSTEM	PROVIDES 'BEST NEXT TEST' RECOMMENDATION FOR TPS DEVELOPMENT	TGIR IS A PROGRAM UTILIZING FIS, A GENERAL DIAGNOSTIC KNOWLEDGE-BASED EXPERT SYSTEM

3.3.1.1 CATA

NAME: **CATA** - Computer Aided Testability Analysis

YEAR: 1983

FUNCTION: CATA provides DFT guidance, FI strategy, and automatic generation of test list after TY evaluation.

CATA is applicable to incoming inspection, manufacturing test, system manufacturing test, production verification, and system maintainability.

CAPACITY: (Typical) 5 to 15 PCBs with 400 ICs of SSI to LSI complexity.

CPU TIME: 30 min for above capacity

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N

DESIGN ENV: VAX 11-780; written in Pascal

CIRCUIT DESCRIPTION:

USE PREREQUISITE:

DEVELOPER: Etudes et Productions Schlumberger

COMMENTS: CATA applies to digital and analog circuits.

New developments of the system were terminated in 1986. CATA is an internal testability analysis program and is not marketed as a product by Schlumberger.

REFERENCES:

1. Robach, Ch., P. Malecha, and G. Michel, "Computer-Aided Testability Evaluation and Test Generation," IEEE International Test Conference 1983

2. Robach, Ch., P. Malecha, and G. Michel, " CATA: A Computer Aided Test Analysis System," IEEE D&T of Computers, May, 1984

3.3.1.2 GIMADS Diagnostic Library

NAME: **GIMADS DIAGNOSTIC LIBRARY**
GIMADS - Generic Integrated Maintenance Diagnostics Program

YEAR:

FUNCTION: An effective method of classifying and referencing diagnostic techniques making a wealth of diagnostic strategies readily available to planners of new systems. The library is in spreadsheet format and looks very similar to an FMEA. In fact, a description of the component and its corresponding failure mode are two of the five columns provided on the spreadsheet. The other three columns, which refer to the diagnostic techniques associated with detecting and isolating the described component in each of its given failure modes, are as follows:

- DIAGNOSTIC TECHNIQUE: A small selection of possible test techniques is provided in this column.
- IMPACTS: This column relates the impact (ie., skill level, reliability, cost, etc.) that each particular technique has on the system.
- COMMENTS: Included in this column are pertinent facts that may help the designer during the selection process.

CAPACITY: N/A

CPU TIME: N/A

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (GFE)

DESIGN ENV: Pencil, paper, and calculator.

CIRCUIT DESCRIPTION: N/A

USE PREREQUISITE: Ample definition or description of the system requiring diagnostic authoring.

DEVELOPER: GIMADS - General Dynamics team under contract from US Air Force.

COMMENTS: It would be a good idea to fill in some FMEA data while working with the GIMADS Diagnostic Library. If, therefore, a diagnostic technique is selected from the library for a particular failure mode of a particular component, then 50 percent of the FMEA process would also be completed at the same time.

Engineering analysis is still required to make sure that the best selection offered by the library is the best selection for the system being developed.

Often a test technique will detect many failure modes of many components. In this case much of the information in the library need not be referenced.

REFERENCES:

1. GIMADS Team
   General Dynamics
   MZ1406
   P.O. Box 748
   Ft. Worth, Tx. 76101
   (817)762-2204

2. Air Force Point of Contact
   ASD/ENE (GIMADS)
   Wright-Patterson AFB, Ohio 45433
   (513) 255-2509/4428

3.3.1.3 IDSS - ADA

NAME: **IDSS - ADA** - Adaptive Diagnostic Authoring

YEAR:

FUNCTION: Performs those functions necessary to ensure the conversion of test strategies, procedures and heuristics information into a form suitable for use by the on-line diagnostician. The ADA will perform the following tasks:

- Process WSTA output for entry into the weapon system knowledge and data bases.
- Receive fault pattern information from a test engineer and add it to the fault-symptom table of the weapon system knowledge base, as well as author additional diagnostic rules.
- Process developed test procedures. This includes developing result descriptions of the test procedures for the weapon system data base and adding the test procedure code to the test procedure library that is accessed during fault isolation.

CAPACITY:

CPU TIME:

APPLICATION: VLSI PCB. SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (GFE)

DESIGN ENV:

CIRCUIT DESCRIPTION: N/A

USE PREREQUISITE: The ADA will process the following type of inputs:

- WSTA outputs including, dependency model representation, candidate search strategies, and logistic data (e.g., MTBF).
- Test procedures. All non-IDSS test procedure generators must provide executable code and a description of the results of the tests.
- Fault pattern information.

DEVELOPER: Harris Corporation under contract from the US Navy.

COMMENTS: Together, the ADA and TIATA provide a bridge for passing information between the design and support clusters.

REFERENCES:

1. Dr. Bruce J. Rosenburg, "The Navy Integrated Diagnostic Support System - System Overview, Architecture and Interfaces," IEEE AUTOTESTCON 1987.
2. Navy Point of Contact
NAVSEA - Code CEL-DS
Washington, DC 20362-5101
(202) 692-2035/2036

3.3.1.4 IDSS -TIATA

NAME: **IDSS - TIATA** - Technical Information And Training Authoring

YEAR:

FUNCTION: Ensures the conversion of design information into teachable format suitable for use by the on-line diagnostician. The TIATA is dedicated to the following tasks:

- Paperless dissemination of technical information.
- Provide an interactive interface for an author/editor, which can be used to create, view and edit technical information and training tutorials consisting of a combination of text, graphics, and audio-visual material. The technical information and training tutorials can be hierarchically structured which promotes thoroughness of instruction while reducing redundancy in the training process.

CAPACITY:

CPU TIME:

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (GFE)

DESIGN ENV:

CIRCUIT DESCRIPTION: All technical information is in IGES.

USE PREREQUISITE: The TIATA will process various documentation sources.

DEVELOPER: Harris Corporation under contract from the US Navy.

COMMENTS: Together, the ADA and TIATA provide a bridge for passing information between the design organization and the user in the field.

The TIATA provides two types of instruction, quick instruction for step-by-step guidance during an on-line diagnostic procedure, and broader instruction for study and maintenance skill development.

REFERENCES:

1. Dr. Bruce J. Rosenburg, "The Navy Integrated Diagnostic Support System - System Overview, Architecture and Interfaces," IEEE AUTOTESTCON 1987.

2. Navy Point of Contact
NAVSEA - Code CEL-DS
Washington, DC 20362-5101
(202) 692-2035/2036

3.3.1.5 TGIR

NAME: **TGIR** -Test Generator Inferred Reasoning
FIS - Fault Isolation Shell

YEAR: 1987 (prototype)

FUNCTION: TGIR is a Navy sponsored program seeking for AI solutions to TPS development. The prototype system developed by the TGIR program uses FIS (a software system developed for general diagnosis Ref[1]) for the generation of test procedures, typically represented as diagnostic flowcharts. Although the system needs to be refined, a knowledge-based expert system has been demonstrated with the following characteristics:

- The initial knowledge base is comprised of collected information or facts which describe the specific UUT (may be attained from CAD data base) and the UUT problems or symptoms.
- The inference engine continues to develop the knowledge base by conducting an interactive dialogue with the test operator and/or the ATE system.
- The inference engine performs a search and pattern match to select the appropriate test or fault diagnosis. When the fault diagnosis routine stalls due to incomplete, ambiguous, or contradictory data, additional data is requested from the test operator or ATE system.
- The rule base will be partitioned into several parts. One part may be a generic troubleshooting procedure and one part a specific procedure as delegated by the UUT's TRD. Other parts may be the UUT's failure history or a functional description of the UUT.

CAPACITY: N/A

CPU TIME: The time to derive a 'best next test' recommendation can take seconds to minutes, depending on the complexity of the UUT.

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (GFE most likely)

DESIGN ENV: The prototype has been implemented in LISP. FIS runs on Sun and Vax workstations with planned capability to run on Symbolics and ISI workstations.

CIRCUIT DESCRIPTION: N/A

USE PREREQUISITE: FIS requires a list of tests before it can recommend what test to perform next. A 'go chain' is also required in order to verify normal operation.

DEVELOPER: NAVAIR Engineering Center is developing TIGR and Naval Research Lab's Navy Center for Applied Research in Artificial Intelligence is developing FIS.

COMMENTS: TIGR/FIS applies to analog/digital circuitry.

TIGR blackboard architecture needs to be developed to determine next best test to perform.

There are 15 to 20 other programs besides TIGR utilizing FIS as a diagnostic reasoning aid.

Causal modeling and heuristic search are the dominant inference techniques used by TIGR/FIS.

The techniques used by knowledge engineers will have a great impact in determining whether or not AI will provide successful solutions sought by the ATE community. See Ref[3].

REFERENCES:

1. F. Pipitone, "The FIS Electronic Troubleshooting System," IEEE Computer magazine, July 1986, pp 68-76.

2. Kenneth A. Porter, Jr., "AI Applications to Automatic Testing: Trend For the Future," AUTOTESTCON 87 pp 377-382.

3. Jerry L. Kunert, "Knowledge Engineering," AUTOTESTCON 86 pp 159-164.

4. Navy Point of Contact
   Navy Air Engineering Center
   Code 9013E
   Lakehurst, New Jersey 08733-5000
   (201) 323-2462/2648

3.3.1.6 AI-TEST

NAME: **AI-TEST** - Artificial Intelligence Test

YEAR: 1987 (Commercially available)

FUNCTION: AI-TEST (formally ATEX - see Ref[2]) is an expert system that offers the following assistance during the testing process:

- Ranks modules according to their likelihood of being the cause of failure.
- Suggests diagnostic goals for the next stage of testing.
- Identifies and evaluates tests which may be used in achieving the diagnostic goals and proposes the most cost effective test.

CAPACITY: MS-DOS version up to 100 modules; UNIX version up to 1000 modules.

CPU TIME: The time to derive a 'best next test' recommendation can take seconds to minutes depending on the complexity of the UUT.

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (PRTY)

DESIGN ENV: AI-TEST has been implemented in the "C" language (a part of the inference engine was prototyped in PROLOG) and can run on an MS-DOS based computer (e.g., an IBM PC/XT or AT, HP Vectra, Compaq) which may be interfaced to an ATE minicomputer, be embedded in a UNIX based ATE minicomputer (HP 350) or be operated independently in manual testing.

CIRCUIT DESCRIPTION: N/A

USE PREREQUISITE: The user must create a UUT-specific knowledge base which includes the unique structure and characteristics of a given UUT, and the test available to diagnose it. Specifically, for any given UUT one must supply the following information in order to generate the UUT-specific knowledge base:

- The block diagram, which consists of elements (modules, ports, junctions, test-points) and links (wires, buses) between these

elements. For each module, several parameters are needed including its failure rate (if known), degree of criticality, and whether it belongs to one of the families in the AI-TEST library.

- The set of tests defined for the UUT. For each test, the user provides a description of its path and the measurements taken at its output points.

- For a unit with 100 modules and 80 - 100 tests, UUT preparation may take 120 hours.

DEVELOPER: Intelligent Electronics, Inc. (See Ref.)

COMMENTS: AI-TEST applies to analog/digital circuitry.

Besides the UUT-specified knowledge base which is built by the user, AI-TEST contains a general purpose knowledge base. This base encompasses knowledge related to electronic theory, including knowledge about different kinds of measurements (DC voltage, frequency, etc.), and a library of functional level element families (A/D converters, amplifiers, etc.).

The user may request AI-TEST to explain its line of reasoning.

Summary reports may be viewed on the display, printed out in hard copy, or stored in a file. Such files are later used for learning purposes in order to improve the UUT knowledge base. Data base tools are included for data management of UUTs.

In an ATE environment, AI-TEST may be interfaced to the ATE computer via RS-232 or IEEE-488 interface. Through this link, a fully automatic test system may be obtained where AI-TEST communicates with the test executive.

As time goes on, AI-TEST will reach higher levels of comprehensive and correct diagnostic assessment.

REFERENCES:

1. M. Ben-Basset, et al, "A Case Study With AI-TEST: An Expert System For Electronic Troubleshooting," IEEE AUTOTESTCON, 1987.

2. M. Ben-Basset, "Expert Systems for ATE the ATEX Approach," IEEE AUTOTESTCON, 1985.

3. IET - Intelligent Electronics Inc.
14 Eeser Tachanot St.
Ramat Machayal, Tel Aviv
Israel

3.3.2 Automatic Test Generation Authoring Tools

This section contains software tools available that are a subset of tools that aid in diagnostic authoring. In particular, they aid in authoring or generating digital test vectors.

They are called Automatic Test Generators (ATGs) or Automatic Test Pattern Generators (ATPGs).

Automatic Test Generation (ATG) and fault simulators go hand-in-hand and sometimes are one in the same tool. However, for classification purposes, fault simulators can be found in the Section for Testability/Diagnostic Assessment Tools (Section 4.2) under the subheading called Diagnostic Effectiveness (Prediction).

A simulator is a software program which provides a simulation of the normal (i.e., no-fault) behavior of internal circuit nodes and primary outputs in response to stimuli applied to its inputs. Simulators are very important design tools and are one instrumental aid in the design/development of modern digital systems. Most simulators provide a fault simulation capability. Fault simulation is a simulation of a circuit in the presence of a fault. The fault introduced is chosen to be compatible with the particular characteristics specified by a fault model for a particular device (i.e., stuck at "0" or stuck at "1"). Most units under test contain many fault possibilities. Most fault simulation systems will, on demand, automatically inject faults one-by-one into a UUT, so that the user can ascertain the effect of these faults in response to a given test vector stimulus input to the UUT. A fault is considered "detected" if the response of the circuit is visibly different from the good circuit with respect to the stimuli applied. Implicit in fault simulation is the capability to compare a good circuit simulation against faulty circuit behavior.

Automatic Test Generators (ATGs) or Automatic Test Pattern Generators (ATPGs), as they are sometimes referred to, typically employ both fault simulation and test pattern generation capability. After a fault simulation is run in response to a given test vector stimulus, the ATPG examines which faults remain undetected and tries to determine what new test vectors need to be applied to the UUT to detect those previously undetected failure modes. The process of "generating" new test vectors is called test generation. ATPGs/ATGs are used to automate this process. However, even relatively simple sequential circuits can be extremely difficult for an ATPG/ATG algorithm to analyze and, subsequently, to generate test patterns. Practically speaking, most tests are for the most part derived "manually," often by trial and error, utilizing a fault simulator for data feedback analysis.

The digital test vector "authoring process" concludes when 100 percent of the faults in the UUT fault list are detected.

Both types of tools (i.e., ATPG/ATG and fault simulators) are essential to the design and evaluation of an effective diagnostic system.

Many systems that do not plan to include a particular diagnostic capability as part of the design process, deploy ATGs simply because they make the production validation process economically feasible. However, these tool capabilities very much serve in the assistance of including diagnostic capabilities. For example, they may be useful during the Deployment Phase within the repair cycle, for developing BIT routines, or for fault-tolerant design and evaluation.

Tools that aid in automatic test generation all have the following features in common:

- They would be deployed during the Full Scale Development Phase for the purpose of deriving an effective and optimized set of test vectors necessary to perform the task of validating the functional design during production.

- They all apply to digital circuitry and primarily process faults at the gate level.

- Although each tool is capable of being run separately, ATGs and fault simulators are almost always run together. The ATG provides the test vector set. The fault simulator, either statistically or deterministically, evaluates what percentage of the total faults considered would be detected by such a test vector set.

- Speed enhancement in this family of tools prime competitive sport and each has a unique way of remaining in the arena. Refer to the following chart with the column headed, "INCREASED SPEED METHOD."

A summary description of tools that aid in test generation authoring during design are provided in the Table below:

ACRONYM	APPLICATION	INCREASED SPEED METHOD	COMMENTS
AIDA	VLSI/PCB/ Subsystem	A RISC CO simulator is mounted into the workstation	Provides excellent scan design assistance
HITS	VLSI/PCB	Uses concurrent simulation	Assists in TPS development
LASAR VER 6: PROSECUTOR	VLSI/PCB	Concurrent Simulation	Post processing available to make stimuli compatible with target tester
SOCRATES	VLSI	Uses an improved ATG fan algorithm & an efficient fault simulation approach	Carries information heuristically from TY Analysis to ATG process
THESEUS	VLSI/PCB	After each test vector, all detected faults are no longer considered	Provides an alternative to scan design
ZYCAD: NextGen	VLSI/PCB/ Subsystem	Simulation implemented in hardware employing parallel pipeline processing techniques	ZYCAD's Next Gen is an accelerated ATG

3.3.2.1 AIDA

NAME: **AIDA** ATPG

YEAR: 1987

FUNCTION: As part of an integrated set of testability tools, the AIDA ATPG works in conjunction with the AIDA fault simulator and automatic scan generator to reduce the "testability overhead" during SCAN design and speeds up the generation of test vectors for manufacture. ATPG creates vectors to detect manufacturing defects and can achieve 100 percent coverage of the detectable single stuck-at fault in a SCAN design. When used in conjunction with the AIDA fault simulator, the set of required test vectors can be reduced to a small number. Often only a few hundred test vectors are required to adequately test a 30K gate design.

Refer to Section 4.2.2.1 for more information on AIDA.

3.3.2.2 HITS

NAME: **HITS** - Hierarchical Integrated Test Simulator

YEAR: 1987 - HITS 14

FUNCTION: HITS is a Digital Automatic Test Program Generator (DATPG) software system. Its primary function as a software tool is to assist in the development of digital Test Program Sets (TPS) and as a means to evaluate/verify digital designs.

Refer to the section on fault simulators which can be found in appendix (4.2.2.4) Testability/Diagnostic Assessment Tools.

3.3.2.3 LASAR VERSION 6 - PROSECUTOR

NAME: **LASAR VERSION 6 - PROSECUTOR**

YEAR: 1982, origin 1978

FUNCTION: LASAR Version 6 provides a CAD simulation system for design verification and test program generation incorporating the following testability subprogram:

PROSECUTOR - An optional component of LASAR Version 6, it automatically generates test vectors for CMOS, NMOS, TTL, and ECL gate arrays, standard SSI/MSI parts, fuse-programmable logic arrays and sequencers, and other digital parts of similar size and complexity. It uses a critical path sensitization technique, generating self-initializing stimulus vectors which cause circuit node faults to propagate to primary output pins.

Testability problems such as non-initializable latches, redundant circuitry, and tristate outputs which need pull-up resistors are uncovered in the process. These problems are reported so they can be evaluated by the circuit designer who can interact if necessary.

Refer to the section on fault simulators which can be found in this appendix (4.2.2.6) for Testability/Diagnostic Assessment Tools under the subheading called Testability/Diagnostic Effectiveness (Prediction) and also to this appendix for Testability/Diagnostic Design Implementation under the subheading called Architecture (3.1).

3.3.2.4 SOCRATES

NAME: **SOCRATES** - Structure Oriented, Cost Reducing, Automatic Test Pattern Generation System

YEAR: 1988

FUNCTION: An ATG system providing random or deterministic test patterns for VLSI with scan and combinational circuits. It promises significant cost reductions by combining the following technical improvements to its ATG process:

- It uses a highly efficient fault simulation approach, see Ref[4].
- It improves upon an already efficient FAN algorithm, Ref[2], using special techniques that reduce the number of backtracings and recognize conflicts early.
- It heuristically carries over information derived from the testability analysis and applies it to the ATG process [refer to HECTOR].

CAPACITY: Over 100,000 primitives on an Apollo Domain DN 4000

CPU TIME: A sample circuit (c7552 of Ref [3]) achieved a 98.25% fault coverage when 231 test vectors (both random and deterministic) were applied. The total CPU process time took 284.3 seconds, of which 86.6 seconds were due to fault simulation and the remaining time due to generating test vectors.

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (PRTY)

DESIGN ENV: Part of the SITEST CAD/CAT system. The above example was run on an Apollo DN 3000 workstation.

CIRCUIT DESCRIPTION: Circuits can be described at gate level, including XOR and XNOR gates, or can be described with high-level primitives, e.g., adders, multiplexers, demultiplexers, encoders, decoders, etc. Circuit description is based on the SMILE Simulator, Ref[5].

USE PREREQUISITE: SMILE simulator or (at least) SMILE circuit compiler.

DEVELOPER: Siemens - AG, W. Germany

COMMENTS: SOCRATES will provide the following outputs:

- List of faults of interest
  - aborted
  - redundant
  - undetected
- List of generated test patterns
- The backtrack distribution file
- A summary file providing information such as CPU time, fault coverage, etc.

SOCRATES provides an option to generate tests for only a given set of faults.

The developers of SOCRATES are investigating more possibilities of improvement, for example:

- Find alternatives to random patterns, particularly for random pattern resistant circuits.
- Minimize multiple backtraces while maintaining SOCRATES performance in terms of backtrackings and aborted faults.
- Develop techniques that would automatically choose the testability measure that would achieve good performance with minimal conflicts.
- Determine if dynamic testability measures would prove beneficial.
- Further improve upon the heuristics employed.

REFERENCES:

1. Schulz, M.H., E. Trischler, and T.M. Sarfert, "SOCRATES: A Highly Efficient Automatic Test Pattern Generation System," IEEE ITC 1987, pp. 1016 -1026
2. Fujiwara, Hideo and Takeshi Shimono, "On the Acceleration of Test Generation Algorithms," IEEE Transactions on Computers, vol. C - 32, No.3, pp. 1137 - 1144, Dec. 1983.
3. Brglez, F. and H. Fujiwara, "A Neutral Netlist of 10 Combinational Benchmark Circuits and a Target Translator in Fortran," Proc. IEEE Int. Sym. on Circuits and Systems; Special Session on ATPG and Fault Simulation, pp. 663 - 698, June 1985.

4. Antreich, K.J. and M.H. Schulz, "Fast Fault Simulation in Combinational Circuits," IEEE Int. Conf. on Computer Aided Design, ICCAD - 1986, Nov. 86.

5. Gonauser, M., Egger, F., Frantz, D., "SMILE - A Multilevel Simulation System," Proc. 1984 IEEE ICCD, pp. 188 - 193.

3.3.2.5 THESEUS

NAME: **THESEUS** - ATG With Inherent Testability Analyzer

YEAR: 1986

FUNCTION: An ATG system capable of high fault coverage for complex sequential circuits without need to change design for testability. There is an optional interactive testability analyzer.

CAPACITY: 250 K nodes/chip

CPU TIME: (Example) 3,585 vectors for a highly sequential, function controller circuit took 116 min.

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (PRTY)

DESIGN ENV: VAX 11/785; part of CADAT simulation toolset

CIRCUIT DESCRIPTION: HHB System unique

USE PREREQUISITE: The circuit description is a simple netlist file describing the circuit's parts and interconnections. This file is easily generated from schematic capture systems using netlist translators.

DEVELOPER: HHB Systems, Mahwah, N.J.

COMMENTS: THESEUS is a practical alternative to scan path techniques or complex Testability analyzers.

Testability reports are produced which contain the following information:

- List of zero, one, and tri-state CY for each node in the circuit, sorted by node or value
- Histogram displaying nodal CY
- List of nodes that cannot be controlled
- List of sources of un-CY

- List of feedback loops that cannot be initialized

Test vectors produced by THESEUS are text files containing the stimuli to the circuit. This file can be transported to the user's in-house simulator; or using the CADAT simulator option, the complete test program consisting of stimulus and response data can by generated.

Upon completion of the test generation process, THESEUS provides a variety of fault analysis reports which consist of:

- Listing of all faults including the detection step and status of each
- Listing of faults detected by test step
- Listing of any undetected faults
- Listing of test generator performance for each fault selected

THESEUS takes advantage of the fact that a test for one fault frequently detects many other faults. All detected faults are eliminated from the data base. Control is then returned to the ATG engine which generates vectors for the remaining undetected faults. This process continues until all the faults have been detected and the fault data base is exhausted.

THESEUS allows reconvergent fan-out structure paths to be simultaneously sensitized without excessively burdening the test generation process.

REFERENCES:

1. Marlett, Dr. R.A., "An Effective Test Generation System For Sequential Circuits," IEEE Design Automation Conference, 1986.
2. Marlett, Dr. R.A., " A Comprehensive Test Generation Technique For Highly Sequential Circuits," IEEE Design Automation Conference, 1978.
3. Also see CADAT 6

3.3.2.6 ZYCAD NextGen

NAME: **ZYCAD** NextGen

YEAR: 1985

FUNCTION: Perform ATG tasks. It implements the simulation in hardware employing parallel pipeline processing techniques. This affords greater speed than possible with software routines which must retrieve instructions from external memory and execute them sequentially.

COMMENTS : NextGen, an accelerated ATG, features an enhanced implementation of extended backtrace algorithm. It can be applied to devices containing any mix of combinational and sequential logic and up to 65,000 gates.

NextGen allows for fault dictionary or guided probe fault isolation and enables the user to focus on those parts of the design that may be untestable, helping to achieve optimum testability.

NextGen II replaces the original NextGen and includes many enhancements as follows:

- Speed 10 - 20 times original release of NextGen
- Fault coverage significantly extended
- Features added, user interface enhanced.

NextGen does a testability analysis of the network prior to beginning test generation. With the addition of "Dynamic Heuristics," NextGen augments this information and continues learning about the circuit during the test building process. This allows NextGen to better choose and sensitize paths through the design, improving performance and minimizing test generation time.

Use of ZILOS, a friendly simulation environment, promotes the use of the same data base files for logic simulation, fault simulation, test analysis, and test generation. Refer to the section on fault simulators (4.2.2.11) in this appendix.

4.0 ASSESSMENT TOOLS

There are two categories of tools that lie within the assessment umbrella. Inherent testability analysis tools and diagnostic test effectiveness tools.

4.1 Inherent Testability Analysis Tools

This section contains various tools available to aid in the task of performing an inherent testability analysis.

There are no claims made that this is an all inclusive list. There are perhaps dozens of tools that are not included that perform better or equally as well as some of those described here.

The inherent testability analysis tools listed in this section all have the following features in common:

- They would be used as inherent testability analysis tools during the Full Scale Development Phase, with the possibility of being useful during the Dem/Val Phase.
- With the exception of IDSS/WSTA and STAMP, they all apply to digital circuitry only.
- They measure a circuit's testability by determining controllability and observability factors.

No attempt has been made to rank these tools. Therefore, they are listed in alphabetical order.

A summary description of tools that aid in assisting inherent testability is provided in the following Table.

The following explains the column headings of the chart:

TEST PNT ANAL/REC?: A "yes" indicates that test point analysis and recommendations are included in the program.

STATISTIC ANAL?: A "yes" indicates that statistical or probabilistic analysis is performed on the circuit. Statistical analysis vs deterministic analysis is a tradeoff between speed and accuracy.

ENHANCE ATG?: All testability analysis will enhance test program generation, however, a "NO" here indicates that the program is not

closely coupled to the ATG process and it does not have a direct impact on tests generated. Tools that do significantly enhance the ATG process were either labeled with the word "VECTORS" or "STRTGY". This was done to distinguish between ATG's that produce a myriad of digital test vectors and test generation that produces an overall test strategy. A test strategy may or may not include digital test vectors.

CHNG/ RE-ANA?: A "yes" indicates that one is able to run the analysis and detect the areas that require changes and then re-run the analysis while remaining within the program.

ACRONYM	APPLICATION	LEVEL OF ANALYSIS	*TEST PNT ANAL/REC	*STATISTIC ANALYSIS?	*ENHANCE ATG?	*CHNG/ RE-ANA
CAFIT	VLSI/PCB/	GATE	YES	NO	NO	NO
CAMELOT	VLSI	GATE	NO	NO	NO	NO
COMET	VLSI	GATE	YES	NO	NO	YES
COP	VLSI	GATE	NO	NO	VECTORS	NO
COPTR	VLSI/PCB	GATE	NO	NO	VECTORS	NO
DTA	VLSI/PCB	GATE/REG/ MACRO	YES	NO	NO	YES
FACE	VLSI	TRNSTR/ GATE/BLOCK	NO	YES	NO	NO
HECTOR	VLSI	GATE	NO	NO	VECTORS	YES
HITAP	VLSI	GATE	NO	NO	NO	NO
IDSS/WSTA	PCB/SUBSYS SYSTEM	COMPONENT/ DEPENDENCY MODELING	YES	NO	STRTGY	NO
ITTAP	VLSI	GATE	NO	NO	NO	YES
PROTEST	VLSI	TRNSTR/ GATE	NO	YES	VECTORS	YES
STAMP	ALL LEVELS	COMPONENT/ DEPENDENCY MODELING	YES	NO	STRATEGY	NO
SCOAP	VLSI	GATE	YES	NO	NO	NO
TESTABILITY CHECKLIST	PCB/SUBSYS	COMPONENT	NO	NO	NO	NO
THESEUS	VLSI/PCB	TRNSTR/ GATE	NO	NO	VECTORS	YES
TMEAS	VLSI/PCB	REGISTER	YES	NO	VECTORS	NO
VICTOR	VLSI	GATE	YES	NO	NO	NO

* See previous page for explanation of these categories.

4.1.1 CAFIT

NAME: **CAFIT** - Computer-Aided Fault Isolation Testability

YEAR: 1987

FUNCTION:

- Identifies specific circuitry which inhibits or defeats fault detection
- Chooses optimal input and output test points
- Estimates maximum possible fault coverage (this analysis does not require any test vector generation)
- Identifies feedback loops and the fewest number of breakpoints
- Provides an index of fault isolation test program complexity based on the number of signals

CAPACITY: 3500 gates on an IBM PC-AT w/640 K. Memory requirement for larger networks 1 Mbyte RAM for each 5,000 network equivalent gates. Capacity on Mentor Graphics Idea Station is 100,000 gates.

CPU TIME: Full Analysis of 3500 gate design on an IBM PC-AT is 4 hours. Run time dependency on model size is slightly higher than linear. Run time inversely proportional to MIPS.

APPLICATION: VLSI, PCB, SUBSYS, SYSTEM

ACQUISITION PHASE: CONCEPT, DEM/VAL, FSD, PRDCTN, DPLYMNT

PUBLIC DOMAIN: Y/N GFE

DESIGN ENV: IBM PC-AT and Mentor Graphics Idea Station. It is written in FORTRAN 77.

CIRCUIT DESCRIPTION: Netlist

USE PREREQUISITE: No training is required. Device library must be created based on function table descriptions of devices. No special language has to be learned.

DEVELOPER: ATAC, Mountain View, CA

COMMENTS:

- Identifies specific circuitry which inhibits or defeats fault detection
- Can generate reports in terms of color-coded schematics
- Able to process bi-directional signals

REFERENCES:

1. ATAC
   ATTN: Brad Ashmore
   1200 Villa Street
   Mountain View, CA 94041
   (415)965-8801

2. Naval Ocean Systems Center, "Testability Analysis Tools On A Military System", Technical Report, September, 1987.

3. Navy Point of Contact
   Naval Ocean Systems Center
   Code 936(B)
   San Diego, CA 92152-5000
   (619) 553-3261

4.1.2 CAMELOT

NAME: **CAMELOT** - Computer Aided Measure for Logic Testability

YEAR: 1980

FUNCTION: Assigns CY and OY values for every node in the circuit and calculates Testability. Unlike SCOAP, CAMELOT can compute Testability around feedback paths or reconvergent circuits.

CAPACITY:

CPU TIME:

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N; see HITAP (Tool No. 4.1.9)

DESIGN ENV: The first draft was written in Pascal and used a subset of a circuit image written for the TEGAS logic simulator, but is not restricted to this form of input.

CIRCUIT DESCRIPTION:

USE PREREQUISITE: One must read in and check circuit connectivity description. Also CY and OY transfer factors, CTF & OTF, must first be computed for each node.

DEVELOPER: Cirrus Computers, UK, for the British Post Office

COMMENTS: The ultimate measure of Testability is the total cost of producing and evaluating a test program. CAMELOT provides a quick, early evaluation to guard against cost overruns.

- Provides interactive design for Testability.
- For reconvergent paths of unequal length, CAMELOT selects the shortest path.
- For reconvergent paths of equal length, CAMELOT computes the OY for both paths and retains the higher value.

- A simple measure of Testability, for each line is obtained in CAMELOT as:

  Testability(line) = CY(line) * OY(line)
  where 0<Testability<1 for 0<CY<1 and 0<OY<1.

- The overall Testability of the circuit is computed as the arithmetic mean of the individual line's Testability:

$$\text{Testability(circuit)} = \frac{\text{Total Sum of All TY(lines)}}{\text{No. of lines}}$$

- CAMELOT may be used in a test generation strategy because of the path sensitizing approach included in the CAMELOT algorithm.

- Because CAMELOT does not provide for automatic re-analysis of the circuit, obtaining a circuit design optimized for testing can be an exhaustive process.

- For very large scale circuits the computations necessary to derive CTF and OTF become uneconomical.

REFERENCES:

1. Bennetts, R.G., et al, "CAMELOT, A Computer-Aided Measure For Logic Testability," IEEE International Conference on Computers and Circuits, 1980, p 1162-1165.

2. Bennetts, R.G., "Design Of Testable Logic Circuits" Addison Wesley, Reading, MA.

3. Also see HITAP

4.1.3 COMET

NAME: **COMET** - Controllability and Observability Measurement for Testability

YEAR: 1982

FUNCTION:

- Measure Testability from controllability (Cy) and observability (Oy) measurements for each node, as well as an overall Testability statistic.
- A graphic statistics option is available. The statistical analysis of the circuit Testability measures, such as the combinational CY/OY mean and standard deviation, are provided.
- Re-design and re-analysis within COMET capability.
- Automated test point and logic inserters available.
- A measure of ease of initializing the circuit for test is one of COMET's outputs.

CAPACITY: N/A

CPU TIME: N/A

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N

DESIGN ENV: Part of Highland Design CAD System; VAX 11/780

CIRCUIT DESCRIPTION: McLDL

USE PREREQUISITE:

DEVELOPER: United Technologies Microelectronics Center

COMMENTS:

- Based on SCOAP (gate level)
- Designers can experiment with different methods without having to change the circuit description or recompile.
- COMET can handle circuit characteristics such as gate inputs tied directly to power and ground, bidirectional signals, and three state buses.
- The user enters a single line of code to describe, for example, an Arithmetic Logic Unit (ALU).
- Fan-out nodes are listed and their OY's are given as the minimum of the fan-out branch OYs. This information is important when running fault simulation to determine the fastest propagation path.
- COMET is not used to predict test patterns or to aid in fault simulation.

REFERENCES:

- Berg, W.L., Hess, R.D., "COMET: A Testability Analysis And Design Modification Package," IEEE International Test Conference, 1982

4.1.4 COP

NAME: **COP** - Controllability and Observability Program

YEAR: 1984

FUNCTION:

- Estimate fault coverage when pseudo random patterns are applied
- Heuristics carry over for ATP generation.
- Testability assessment with or without test patterns.
- I/O with Testability signatures can aid in design verification.

CAPACITY:

CPU TIME: Order of magnitude faster than traditional fault simulators.

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N

DESIGN ENV: IBM

CIRCUIT DESCRIPTION: Neutral file

USE PREREQUISITE: Levelize and rewrite using only one type netlist; recommend using a program for this task

DEVELOPER: Bell Northern Research

COMMENTS:

- Uses Boolean algebra to determine fault detection.
- Does not detect redundantly masked faults.
- Fault simulation method.
- Critical delay path trace capability.
- Gate level representation.
- The combinational circuitry is partitioned into sets of overlapping structural cones which also helps to enhance the ATPG process and the critical delay path tracing algorithm.
- Similar to TESTSCREEN (not described in this document).
- Used by Texas Inst.

REFERENCES:

- Brglez, F., Pownall, P., Hum, R., " Applications Of Testability Analysis: From ATPG To Critical Delay Path Tracing," 1984 IEEE International Test Conference.

4.1.5 COPTR

NAME: **COPTR** - Controllability - Observability - Predictability - Testability Report

YEAR: 1982

FUNCTION:

- CY, OY, And Testability analysis for each node as well as for the entire circuit.
- Closely coupled with ATG.
- It points out changes that would make a circuit testable.

CAPACITY: (Example) 5,894 signals

CPU TIME: The above example was compiled in 2 min, linked in 10 min, the fault generation took 3 min, and the COPTR computation time was 20 min.

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (PRTY)

DESIGN ENV: The TEGAS - 5 software family for design and test simulation; CALMA workstations; 32 BIT Apollo workstation; supervised by Taskmaster.

CIRCUIT DESCRIPTION: TEGAS Description Language (TDL)

USE PREREQUISITE: Code network description in TDL, compile and link. No test pattern or simulation is required.

DEVELOPER: CALMA Company

COMMENTS:

- Many CALMA menus enable variations of COPTR printout.
- COPTR is hierarchical with a good model library.
- Use with TCAT, CALMA's fault simulator and ATPG.

REFERENCES:

1. Kirkland, T., Flores, V.,"Software Checks Testability and Generates Tests of VLSI Design," ," Electronics Mag., March 10, 1983.

2. CALMA Company;
   Attn: Thomas Poos
   Milpatas, Ca. 95035 - 7489
   (408) 434-4870

3. Naval Ocean Systems Center, J.C. Bussert, "Testability Measures On a State-of-the-Art Circuit," Technical Document 835, Feb, 1986

4.1.6 DTA

NAME: **DTA** - DAISY Testability Analyzer

FUNCTION:

- Computes six CY and OY values for each node.
- Evaluates Boolean expressions, ROMS, RAMS, and PLAs.
- Manual or automatic test point insertion.
- Allows for re-evaluation in software mode.

CAPACITY: (Example) 122,110 gate equivalent circuit

CPU TIME: 8 MIPS; 11 min for above example.

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (PRTY)

DESIGN ENV: Daisy Logician Workstation, Intel 80286 based. Used with GATEMASTER, CHIPMASTER, BOARDMASTER, and MEGAGATEMASTER. Written in PL/M with 12,000 lines of code.

CIRCUIT DESCRIPTION:

USE PREREQUISITE: Validation by Daisy Logic Simulator ensures the design is ready for DTA.

DEVELOPER: Daisy Systems Corporation

COMMENTS:

- DTA is event directed and compiler driven, therefore requires no library and can process macro cells.
- DTA is SCOAP based.

REFERENCES:

1. Wang, L.T., Law, E., "An Enhanced Daisy Testability Analyzer (DTA)," AUTOTESTCON, 1985.

2. Daisy Systems Corp.
   Attn: Mark Fuccio
   700 Middlefield Road
   Mountain View, Ca. 94039
   (415)960-7168

3. Naval Ocean Systems Center ,"Testability Analysis Tools On A Military System ", Technical Report, September, 1987.

4.1.7 FACE

NAME: **FACE** - Fault Coverage Estimation

YEAR: 1986

FUNCTION:

- Statistical fault analysis.
- Applicable to mixed levels; MOS transistor level, gate level, and combinational functional block level.
- Its logic simulator deploys both event driven and circuit leveling techniques.

CAPACITY: Information not available.

CPU TIME: Information not available.

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N UNIV

DESIGN ENV: Unknown

CIRCUIT DESCRIPTION: See above.

USE PREREQUISITE: Information not available.

DEVELOPER: University of California; Berkeley, CA

COMMENTS: The ability of FACE to process both gate level and transistor level is particularly useful when designing with CMOS technology. Stuck open CMOS transistors cause the combinational logic they are part of to exhibit sequential behavior, which is an order of magnitude more difficult to analyze.

One is able to process Testability analysis while the design is only described in functional blocks.

REFERENCES:

- Ma, H.K., Sangiovanni - Vincentelli, A.L., "Mixed Level Fault Coverage Estimation," IEEE Design Automation Conference, 1986

4.1.8 HECTOR

NAME: **HECTOR** - Heuristic Controllability And Observability Analysis

YEAR: 1984

FUNCTION:

- Creates a weighted AND/OR graph based upon the circuit description.
- Calculates CY and OY (identical to SCOAP) and Testability measures.
- CY and OY measures are assigned to the hyperarcs connecting the parent node with a set of successor nodes.
- Hyperarcs are ordered according to CY and OY measures.
- Creates a tree search data base to run ATWIG, Ref[2]&[4], and EDIT, Ref[1].
- Provides guidance at each decision node for ATWIG.
- Provides guidance on how to select flip-flops to be included into an incomplete scan path.

CAPACITY: 10,000 nodes.

CPU TIME: Example circuit requires 371 sec on VAX 11/780; for other benchmark circuits, see Ref. [3].

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N

DESIGN ENV: Written in C with UNIX; part of IDAS: Integrated Design for testability and Automatic TPG System. See COMMENTS and Ref [1].

CIRCUIT DESCRIPTION: CADAT circuit file input description.

USE PREREQUISITE: Keyboard command transforms circuit description from Daisy Logician database to CADAT to run IDAS.

DEVELOPER: Siemens Corporate Research and Support, Princeton, NJ.

COMMENTS: The IDAS design system consists of the following:

CADAT - a fault simulator
HECTOR - testability analyzer
PRETEST - predicts the cost of an ATG run. See Ref[5].
EDIT - A set of tools to evaluate, display and improve Testability (e.g., to automatically include an incomplete scan path into a given circuit).
ATWIG - Automatic TPG With Inherent Guidance for combinational and sequential circuits. See Ref[2]&[4].

The IDAS system was developed for research purposes only, many of its features are used in SOCRATES (3.2.2.4), an ATPG, see Ref. [7].

REFERENCES:

1. Trishler, E., "An Integrated Design for Testability and Automatic Test Pattern Generation System: An Overview," Proc. 21st Design Automation Conference 1984, pp. 209-215.

2. Trischler, E., "ATWIG, An Automatic Test Pattern Generator With Inherent Guidance," Proc. 1984 IEEE Int. Test Conf., pp. 80-87.

3. Trischler, E., Schulz, M., "Applications to Testability Analysis to ATG: Methods and Experimental Results", Proc. 1985 IEEE Int. Symp. on Circuits and Systems, PP. 691-694.

4. Trischler, E., "Guided Inconsistent Path Sensitization: Method and Experimental Results", Proc. 1985 IEEE Int. Test Conf., ppl 79-86.

5. Trischler, E., "A Methodology for Statistical Evaluation of Estimated and Real Testability Measures," Siemens Forsch. - u. Entwickl.-Ber., Vol. 16, No. 1, 1987, pp. 1-8.

6. Trischler, E., "Incomplete Scan Path with an Automatic Test Pattern Generation Methodology," Proc. 1980 IEEE Test Conf., pp. 153-162.

7. Schulz, M.H., E. Trischler, and T.M. Sarfert, "SOCRATES: A Highly Efficient Automatic Test Pattern Generation System," IEEE ITC 1987, pp. 1016 -1026

4.1.9 HITAP

NAME: **HITAP** - HI-Testability Analysis Program

YEAR: 1985

FUNCTION: HITAP is a testability analysis program for gate array and standard cell designs. HITAP is compatible with GenRad's HILO Universal Logic Simulation System. It allows the design engineer to integrate testability into the design process. With HITAP, areas that are difficult to test are revealed during, rather than after, the design process. HITAP provides the user with a relative figure of merit for each circuit node. The figures of merit are expressed in terms of observability (the ability to observe a node at a primary output) and controllability (the ability to control the node from a primary input). HITAP calculates a testability figure of merit for each item analyzed, as the product of the observability and controllability factors. These factors then provide a relative measure of the testability of the design.

CAPACITY:

CPU TIME:

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (PRTY)

DESIGN ENV: DEC 32 bit machine; used in conjunction with HILO, HIPOST, HICHIP, HITEST; written in Pascal.

CIRCUIT DESCRIPTION: Hardware Description Language (HDL)

USE PREREQUISITE: Requires netlist in HILO format, HDL.

DEVELOPER: GenRad

COMMENTS: HITAP first finds CY, then OY, then Testability.

REFERENCES:

GenRad Inc.
Attn: Michael Busch
37 Main St.
Bolton, Mass 01740
(508) 779-6271

4.1.10 IDSS - WSTA

NAME: **IDSS - WSTA - Weapon System Testability Analyzer**

YEAR:

FUNCTION: The WSTA provides the following testability analyzer capabilities:

- Controllability/Observability calculations for each test point or I/O contained in the UUT.
- Test point utilization data. A measure of how often a test is used in a test strategy.
- Test point criticality. A measure relating the test point to the criticality of the circuitry involved.
- Provide to the test designer a prioritized list of test points which must be monitored to detect the presence or absence of a fault in the weapon system under test.

Refer to the Tools That Aid Testability/Diagnostic Prediction for more details and more capabilities of the IDSS/WSTA tool (4.2.1.4).

4.1.11 ITTAP

NAME: ITTAP - Interactive Testability Analysis Program

YEAR: 1982

FUNCTION:

- Testability analysis program.
- Measures are provided for 2 groups: difficulty to control and observe, and test length.
- Contains standard logic elements as primitives within the library as well as the ability to define CY and OY of any block of logic.
- Interactive.

CAPACITY:

CPU TIME: It uses a selective trace algorithm which saves 90 - 98% CPU time compared to trying to evaluate every node for every test vector.

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N

DESIGN ENV: The Prime 750 was used in the example.

CIRCUIT DESCRIPTION:

USE PREREQUISITE: Describe circuit as interconnections of standard cell elements found in library or Fortran subroutine.

DEVELOPER: ITT-LSI Technology Center

COMMENTS:

- An order of magnitude run time improvement over SCOAP.
- Various interactive commands for assessing Testability are available.

REFERENCES:

- Goel, C.K., McDermott, R.M., "An Interative Testability Analysis Program - ITTAP," IEEE Design Automation Conference 1982

4.1.12 PROTEST

NAME: **PROTEST** - Probabilistic Testability Analysis

YEAR: 1985

FUNCTION:

- Using signal probabilities as input, the program will output the probability a fault will be detected.
- Determines the required test length to obtain specified fault coverage.

CAPACITY:

CPU TIME: (Example) 26,450 transistors with 32,000 test patterns took 23 seconds of CPU time. To optimize to 1,778 test patterns took 2,181 seconds of CPU time.

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N

DESIGN ENV: Carlsruher Digital Design System (CADDY) [University of Carlsruher, Fed Rep of Germany]; written in Pascal. The example was run on a Siemens 7561 computer.

CIRCUIT DESCRIPTION:

USE PREREQUISITE: A description of the combinational circuit.

DEVELOPER: Univ of Karlsruhe, Germany

COMMENTS:

- When using BILBO, PROTEST determines test length.
- When using NLFSR, PROTEST determines optimal input signal probabilities.
- It reduces computing time of ATPGs by providing optimized pattern sets.

REFERENCES:

- Hans-Joachim Wunderlich, "PROTEST: A Tool For Probabilistic Testability Analysis," IEEE Design Automation Conference 1985.

4.1.13 SCOAP

NAME: **SCOAP** - Sandia Controllability/Observability Analysis Program

YEAR: 1980

FUNCTION:

- Calculates six functions that characterize CY & OY properties of digital circuits.
- Identifies poor Testability nodes.
- Makes test point recommendations.
- Evaluates design modifications.

CAPACITY: (In 1980) More than 10,000 standard cells or more.

CPU TIME: The worst case structure of a circuit with 250 cells took 91 seconds.

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (UNIV)

DESIGN ENV: A DEC SYS 10 Computer was used in the example. It is written in structured FORTRAN and is reasonably portable. The source code = 3,000 lines.

CIRCUIT DESCRIPTION: No special language.

USE PREREQUISITE: The circuit must be described as a netlist of elements contained in the library.

DEVELOPER: Sandia Laboratories

COMMENTS:

- SCOAP consists of 6 modules performing the following functions: control, preprocess, translation, calculation, sorting, and graphing
- SCOAP has widespread popularity. Other testability analysis tools are either based on it or compare themselves to it.
- Further analysis by the designer is often required after SCOAP has processed such circuit elements as reconvergent fan-outs, redundant nodes, power and ground lines, tied nodes, and bidirectional devices.
- Users have sought to improve SCOAP by including the ability to provide design modifications, reduce test generation costs, increase fault coverage, identify redundancy, etc.

REFERENCES:

- Goldstein, L.H., Thigpen, E.L., "SCOAP: Scandia Controllability And Observability Analysis Program," IEEE DAC, 1980.

4.1.14 STAMP

NAME: **STAMP** - System Testability And Maintenance Program

YEAR: 1980

FUNCTION: STAMP provides the following testability analyzer capabilities:

- Testability improvement recommendations.
- Test point evaluation.
- BIT effectiveness.

Refer to the Tools That Aid Testability/Diagnostic Prediction section for more details and more capabilities of STAMP (4.2.1.7).

4.1.15 Testability Checklist

NAME: **Testability Checklist**

YEAR: 1979

FUNCTION: A quick and inexpensive way of evaluating testability by requiring the user to review his design by answering a list of generic questions and estimate a score of how well his design for testability is. The two reference below are provided as sources for examples of Testability Checklist.

CAPACITY: N/A

CPU TIME: N/A

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (GFE)

DESIGN ENV: Pencil, paper, and calculator.

CIRCUIT DESCRIPTION: N/A

USE PREREQUISITE: Ample definition or description of the system requiring testability.

DEVELOPER: MIL-STD-2165/RADC -TR -79 -327

COMMENTS: Computers are potentially able to make the checklist method more powerful.

Extensive engineering analysis is still required after using the checklist approach.

The Testability Checklist of Ref[2] has fixed items of weighting and applies only to digital boards, whereas the one provided in Ref[1] allows subjective treating of items and weighting values and applies to digital/analog circuitry from module to system level design. As of the publication date of this document, both tools were under revision.

- The Testability Checklists of both references are "free" and can be utilized well in selective applications.

REFERENCES:

1. MIL-STD 2165 Appendix B; "Testability Program for Systems and Equipments," Publications & Forms Center

   Available through:

   Attn: NPFC 1032
   5801 Tabor Ave.
   Philadelphia, Pa. 19120

2. RADC - TR - 79 - 327; "An Objective Printed Circuit Board Testability Design Guide And Rating System" January, 1980.

   Available through:

   DTIC
   Report AD 082329
   Cameron Station
   Alexandria, VA 22304-6145
   (202) 274-7633

3. Naval Ocean Systems Center, "Testability Analysis Tools on a Military System," Technical Report, September, 1987.

4.1.16 THESEUS

NAME: THESEUS - ATG With Inherent Testability Analyzer

YEAR: 1986

FUNCTION: An ATG system capable of high fault coverage for complex sequential circuits without need to change design for Testability. There is an optional interactive Testability analyzer.

THESEUS provides the following testability analyzer capabilities:

- List of zero, one, and tri-state CY for each node in the circuit, sorted by node or value
- Histogram displaying nodal CY
- List of nodes that cannot be controlled
- List of sources of un-CY
- List of feedback loops that cannot be initialized

Refer to the Tools That Aid Testability/Diagnostic Prediction (ATG or Fault Simulation) section for more details and more capabilities of THESEUS (3.3.2.5).

4.1.17 TMEAS

NAME: **TMEAS** - Testability Measurement

YEAR: 1976

FUNCTION: Measures CY & OY and thus derives Testability of each node as well as the total circuit.

Identifies poor Testability locations.

Provides test point selection.

Aids test generation.

CAPACITY: (Example) PCB with 20 - 70 ICs of SSI & LSI complexity.

CPU TIME: (Example) 400 signal paths took 3 seconds to process.

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N

DESIGN ENV: (Example) IBM TSS/370; Amdahl 470 V/7

CIRCUIT DESCRIPTION: LSL Local

USE PREREQUISITE: Translate circuit description of LSL Local into a Testability model.

DEVELOPER: AT&T Bell Labs

COMMENTS: It models at the register transfer level.

The algorithm can be gleaned from the two reference articles and home grown implemented.

REFERENCES:

1. J. E. Stephenson and J. Grason, "A Testability Measure For Register Transfer Level Digital Circuits," Proceedings of 1976 International Symposium on Fault Tolerant Computing, June 1976, pp. 101-107.

2. John Grason, "TMEAS, A Testability Measurement Program," IEEE DAC 1979

3. AT&T Bell Laboratories
Attn: John Grason
IL417
Holmdel, New Jersey 07733
(201)949-3000 ext-3086

4.1.18 VICTOR

NAME: **VICTOR** - VLSI Identifier Of Controllability, Testability, Observability, And Redundancy

YEAR: 1982

FUNCTION: See title.

CAPACITY:

CPU TIME:

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N

DESIGN ENV: VICTOR algorithm was implemented with 3500 lines of ANSI FORTRAN 77

CIRCUIT DESCRIPTION:

USE PREREQUISITE:

DEVELOPER: Electronics Research Lab, University of California, at Berkeley.

COMMENTS:

REFERENCES:

- Ratiu, I.M., et al, "VICTOR: A Fast VLSI Testability Analysis Program," IEEE International Test Conference, 1982.

4.2 Diagnostic Effectiveness Tools

There are two classes or categories of tools that lie under the diagnostic test effectiveness tool umbrella: test strategy tools and fault simulation tools. Both categories of tools assess testability/diagnostics via employing a test effectiveness measure to the assessment methodology utilized.

4.2.1 Test Strategy Tools

This section contains software tools available to aid in the task of predicting/assessing the effectiveness and strategy of system diagnostics via a Testability Figure of Merit approach.

There are not claims made that this is an all inclusive list. There are perhaps dozens of tools that are not included that perform better or equally as well as some of those described here.

The tools are arranged in alphabetical order.

Tools that aid in the prediction of the effectiveness of the diagnostic capability and assist in test strategy formulation, listed in this section, all have the following features in common:

- They are useful during more than one acquisition phase and apply to more than one level of integration.
- They apply to a variety of hardware technologies, including analog and digital circuitry.
- They are all dedicated to assisting in the design and evaluation of fault isolation strategies.

A summary description of tools that aid in assessing diagnostic effectiveness and test strategy functions are provided in the following table.

ACRONYM	APPLICATION	ACQUISITION PHASE	TFOM	COST PARAMETER	COMMENTS
ACE	VLSI/PCB/ SUBSYSTEM/ SYSTEM	DEM/VAL/FSD PRDCTN	AMBIGUITestability GROUP SIZE AND NUMBER	REQUIRED NUMBER OF TESTS/TPS COST	PROVIDES TEST DECISION TREE REPORT
ASTEI'	PCB/ SUBSYSTEM/ SYSTEM	CONCEPT/ DEM/VAL FSD/PRDCTN	FD/DET FI/ FI RESOLUTION	TEST COST/ SITE OVERHEAD	CONCERNED WITH TEST TIMES
I-CAT (IN-ATE)	PCB/SUBSYS/ SYSTEM	DEM/VAL FSD/PRDCTN	AMBIGUITY GROUP SIZE	COST TO DIAGNOSE/ REPLACEMENT COST	TEST STRATEGY REPORT/TEST POINT ASSESSMENT/ REL DATA/ EDIF CAD/CA NETLIST & SCHEMATIC/ TEST PROGRAM IN ATLAS OR BASIC
IDSS-WSTA	PCB/ SUBSYS/ SYSTEM	CONCEPT/ DEM/VAL FSD	STATIC: AMB GROUP DISTRIBUTION/ INHERENT FI LEVELS/ COMPONENT INVOLVEMENT RATIOS DYNAMIC: MTTI/MTTR	MEAN COST TO ISOLATE/ MEAN COST TO REPAIR/COST TO REPLACE VS. COST TO FURTHER TEST	PERFORMS A COMPLETE TESTABILITY ANALYSIS FUNCTION/ GENERATES FAULT TREES & OPTIMUM TEST STRATEGIES
LOGMOD	VLSI/PCB/ SUBSYSTEM/ SYSTEM	DEM/VAL FSD/ PRDCTN/	MTTI/MTTR		TEST STRATEGY RECOMMENDATION BIT PREFERENCE
PROFILE	VLSI/PCB/ SUBSYSTEM SYSTEM	CONCEPT DEM/VAL FSD/ PRDCTN/ DEPLOYMENT	MTTR	MAINTENANCE PRODUCTION COSTS	ANALYSIS OF INDICATORS, TEST POINTS & FALSE REPLACEMENT SUMMARY OF MAINTENANCE ACTIONS/ MAINTENANCE TRAINING & DIAGNOSTIC AID TOOL

ACRONYM	APPLICATION	ACQUISITION PHASE	TFOM	COST PARAMETER	COMMENTS
STAMP	VLSI/PCB/ SUBSYSTEM/ SYSTEM	CONCEPT DEM/VAL FSD/	23 DIFFERENT TFOMS	INCLUDED IN THE 23 TFOMS	BIT ASSESSMENT FI DESIGN & EVALUATION/ TESTABILITY ANALYSIS/SW SELF TEST VALIDATION/ FAULT TREE GENERATION
TESAP	PCB/ SUBSYSTEM	PRDCTN/ DEPLOYMENT		COSTS OF VARIOUS TESTING STRATEGIES	DETERMINES WHAT COMBINATION OF TESTS ARE BEST
TIME	SUBSYSTEM SYSTEM	DEM/VAL FSD	MAINTENANCE TASK TIMES		UNDER DEVELOPMENT

4.2.1.1 ACE

NAME: ACE - APT Computational Environment
APT - ALPHATECH Program For Testability

YEAR: 1987 for phase 1 prototype.

FUNCTION: Provides a test decision tree report.

Provides histograms with size and number of occurrence of ambiguity groups for individual components and for all components.

Gives cost in terms of required number of tests and non-terminal decision nodes.

Gives cost relating to Test Program Sets (TPS).

CAPACITY:

CPU TIME:

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (PRTY)

DESIGN ENV: Sun 3/160 C Workstation

CIRCUIT DESCRIPTION:

USE PREREQUISITE: Model the system similar to a schematic diagram.

DEVELOPER: Alphatech Inc.

COMMENTS: ACE is still being developed.

REFERENCES:

1. Alphatech Inc.
   Attn: Robert Tenney
   111 Middlesex Turnpike
   Burlington, Ma 01803
   (617) 273 3388

2. Naval Ocean Systems Center, "Testability Analysis Tools On A Military System" Technical Report, September, 1987.

4.2.1.2 ASTEP

NAME: **ASTEP** - Advanced System Testability Evaluation Program

YEAR: 1988

FUNCTION: Generate prioritized, failure rate weighted, Fault Isolation Group (FIG) lists (i.e., fault dictionary or ambiguity lists) and generate performance predictions of the following common diagnostic test characteristics:

- Fault Detection
- Test Execution/Detect Times
- Detected Faults Isolated
- BITE Overhead
- Fault Isolation Resolution (mean & discrete list sizes)
- Test Cost

CAPACITY:

CPU TIME:

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (PRTY)

DESIGN ENV: IBM PC/AT

CIRCUIT DESCRIPTION: ASTEP uses a compiled DB III, which is not required by users.

USE PREREQUISITE: One must input system failure rate and test coverage estimates or measurements.

DEVELOPER: BITE INC., Manassas, Va.

COMMENTS: Test performance can be tracked for any hierarchical level; hardware partition, functional/logical partition, or test partition.

- Applicable to all hardware technologies.
- ASTEP is a design aid. The quality of the output is dependent upon the quality of the input.

REFERENCES:

1. BITE INC.
   Attn: John Cunningham
   9254 Center St.
   Manassas, VA 22110
   (703)361-7050

2. Naval Ocean Systems Center, "Testability Analysis Tools On A Military System," Technical Report, September, 1987.

4.2.1.3 I-CAT

NAME: **I-CAT** - Intelligent Computer-Aided Test

YEAR: 1984

FUNCTION: Provides test strategy report in the form of a test and replace flow diagram report.

Provides a Testability analysis report which includes the average:

- Cost to diagnose
- Replacement cost
- Ambiguity group size
- Reports on test point effectiveness.

Provides printouts of the following information types:

- Reliability Data
- EDIF CAD/CAM Netlist
- EDIF CAD/CAM Graphical Schematic

A test program is automatically generated in BASIC or ATLAS.

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (PRTY)

DESIGN ENV: Macintosh Plus PC; Apollo and Sun workstation compatibility in development.

CIRCUIT DESCRIPTION: Draw box with Mac Draw then click in information with mouse.

USE PREREQUISITE: Enter information such as :

- voltage or current values
- expert rules that apply
- presets such as switch settings
- failure rates

DEVELOPER: Automated Reasoning Corp.

COMMENTS: I-CAT seems to have proven itself to be a successful AI application.

- I-CAT formally called IN-ATE.

REFERENCES:

1. Automated Reasoning Corp.
   Attn: Richard Cantone
   290 W. 12th St., Suite 1-D
   New York, NY 10014
   (212) 206-6331

2. Naval Ocean Systems Center, "Testability Analysis Tools On A Military System," Technical Report, September, 1987.

4.2.1.4 IDSS - WSTA

NAME: **IDSS - WSTA** - Weapon System Testability Analyzer

YEAR: 1989

FUNCTION: The WSTA provides the following capabilities:

Grade the testability of a weapon system with both static and dynamic TFOMs and make recommendations for improvement. The following are the static TFOMs:

- Ambiguity group distribution.
- Inherent fault isolation levels.
- Component involvement ratios. This is a measure of the number of times a component appears in any ambiguity group in relation to the total number of possible ambiguity groups.
- Identification of all feedback loops.

TFOMs that are based on the actual fault diagnostic strategy are called dynamic TFOMs. The following are the dynamic TFOMs:

- Isolation penalties (MTTI and Mean Cost to Isolate).
- Repair penalties (MTTR and Mean Cost to Repair).
- Replacement/isolation tradeoffs. Data used to determine when further testing is preferred to the repair of an ambiguity group.
- Test point utilization data. A measure of how often a test is used in a test strategy.
- Test point criticality. A measure relating the test point to the criticality of the circuitry involved.

Generate fault trees and provide an optimum test strategy with additional reports/recommendations for an improved test strategy.

Provide a dependency model and test strategy/fault tree for use during on-line troubleshooting.

Provide to the hardware designer a prioritized list of test points which must be monitored to isolate a fault in the weapon system under test.

CAPACITY:

CPU TIME:

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (GFE)

DESIGN ENV: WSTA requires either a Sun 3/160 computer, with UNIX 4.2 bsd operating system, or a Digital Equipment Corporation MICRO VAX II, with a VMS operating system. A minimum of 4 Mbytes of virtual memory for the Sun and 4 Mbytes/16 Mbytes virtual memory for the MICRO VAX II.

CIRCUIT DESCRIPTION: Either VHDL and LSAR or first order dependencies.

USE PREREQUISITE:

DEVELOPER: Harris Corp. under contract to the US Navy.

COMMENTS: WSTA may be applied to digital, analog, hybrid, and/or electro-mechanical systems. Furthermore, WSTA is not limited to weapon systems but also applies to space, avionics, and support design.

The principal sequencing technique in test strategy generation is based upon the Time Efficient Sequence of Tests (TEST) algorithm composed of a top-down search that integrates concepts from information theory and AI techniques. See Ref[2].

The measurable TFOMs provided by WSTA are consistent with the operational scenarios used to detect and isolate faults in the field.

Refer to the Tools That Aid Testability/Diagnostic Design section for detailed descriptions on the rest of IDSS tools.

It is important to note that although WSTA is categorized here as a prediction tool, it is intended to be used during the design process.

REFERENCES:

1. Franco, JR, and JM Scott, "WSTA The IDSS Weapon System Testability Analyzer," IEEE AUTOTESTCON 1987.

2. Pattipati, KR, Alexandrias, MG, Deckert, JC, "Time Efficient Sequencer of Tests (TEST)," IEEE AUTOTESTCON 1985.

3. Dr. Bruce J. Rosenburg, "The Navy Integrated Diagnostic Support System-System Overview, Architecture and Interfaces," IEEE AUTOTESTCON 1987.

4. Navy Point of Contact
NAVSEA - Code CEL-DS
Washington, DC 20362-5101
(202) 692-2035/2036

4.2.1.5 LOGMOD

NAME: **LOGMOD** - Logic Model

YEAR: 1970

FUNCTION: Testability evaluations, automatic testability report with TFOMs, test strategy recommendations, FI support, part of expert support, hard copy logic model, battle damage assessment, BIT preference, maintainability information, test strategies, MTTFI, MTTR, and a validation file output similar to FMEA.

CAPACITY:

CPU TIME:

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (PRTY)

DESIGN ENV: Any mainframe or minicomputer with FORTRAN 77 compiler; IBM PC & WICAT-150 workstation

CIRCUIT DESCRIPTION:

USE PREREQUISITE: Input schematic or system block diagram

DEVELOPER: DETEX

COMMENTS: With available DETEX training, the user can enter and interpret results.

LOGMOD treats all signals equally, but human logic will weigh certain signal states to have more relevance than others.

REFERENCES:

1. DETEX Systems Inc.,
Attn: Ralph DePaul
17871 Santiago Blvd, Suite 221
Villa Park, CA 92667
(714) 637-9325

2. Naval Ocean Systems Center, "Testability Analysis Tools On A Military System,," Technical Report, September, 1987.

4.2.1.6 PROFILE

NAME: **PROFILE** - A Generic Expert Diagnostician

YEAR: 1982, 1987

FUNCTION: When used as a design analysis tool, PROFILE projects the maintenance performance required for each of a sample of failures, and keeps track of the reasons for excessive fault resolution time. Among its summary results are the following:

- The distribution of repair times, with mean time to repair
- An analysis of the utilities of all indicators and test points. This can highlight maintenance features which are redundant or of marginal value, considering their production cost.
- An analysis of false replacements, indicating those components which are likely to be consumed in quantities greater than their failure rates would indicate. This also focuses attention on needs for additional indicators and test points, to discriminate between parts which produce identical symptoms under the current design.
- A summary of the types and frequencies of maintenance actions required to resolve the sample of faults, and the proportion of time spent performing those functions.

CAPACITY: Multi-unit systems

CPU TIME: Intensive

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (UNIV) See last comment.

DESIGN ENV: Apollo, Sun, and VAX compatibility. Written in Pascal.

CIRCUIT DESCRIPTION: Mentor Graphics CAD interface capability available.

USE PREREQUISITE: Data concerning manual operations time consumption and reliability estimates.

DEVELOPER: Behavioral Technology Labs, University of Southern California, supported by the Office of Naval Research

COMMENTS: PROFILE is also useful as a maintenance training system and a diagnostic aid tool.

- There is an interest in the success of PROFILE by government officials so that MTTR times can be more effectively verified.
- PROFILE can be used in conjunction with ANDI, a simulator with analog and digital capabilities, offered by the Silvar Lisco Corporation.
- In order to use PROFILE for subsystem/system level design, one must manually input the system model. Simulators, similar to ANDI for system-level design, are helpful however.
- At present, the US Government has unrestricted rights to the software and can distribute it to whomever it wants. An agency has not yet been established to do this, however. Until then, one may obtain PROFILE at the point of contact listed below.

REFERENCES:

1. Towne, D. M., "A Generic Expert Diagnostician," Proceedings of AF Workshop on AI Applications for ID, University of Colorado, July, 1986.
2. Towne, D. M., Johnson, M. C., and Corwin, W. H., "A Performance Based Technique For Assessing Equipment Maintainability," Los Angeles, CA, Behavioral Technology Laboratories, University of Southern California, Report No. TR-102.
3. Behavioral Technology Laboratories
1845 South Elena Ave. Fourth Floor
Redondo Beach, CA 90277
(213) 540-3654

4.2.1.7 STAMP

NAME: **STAMP** - System Testability And Maintenance Program

YEAR: 1980

FUNCTION: BIT effectiveness; FI evaluation; testability improvement recommendations; test recommendations; software self-test design validation; testability measures; multiple failure detection; test point evaluation

CAPACITY: 2000 + nodes

CPU TIME: Fault trees may take an hour or more to compute at the 2000 node level.

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N ; only by special arrangement, see first comment.

DESIGN ENV: Apple (earlier 200-node version); HP-1000/A900

CIRCUIT DESCRIPTION: Logic Modeling, knowledge base development

USE PREREQUISITE: A functional block diagram, including test points, is the basic input to STAMP. The program will make use of test cost, failure frequencies, skill level and other data, as well as modifications and overrides to logic and inference.

DEVELOPER: ARINC Research Corp.

COMMENTS: The company policy is not to sell or lease STAMP, but rather to sell and provide testability services. STAMP is an essential tool that ARINC Research employees will use themselves in order to provide these services.

STAMP's wide range of applications and useful outputs make it a highly desired tool. The analysis process is fully hierarchical.

STAMP has 23 different numerical TFOM measures that require contractor assistance (Ref[2]). In addition, it will directly compute specification compliance numbers, it will generate a number of user specified analyses, and generate both single and multiple failure FMEAs.

STAMP will also prepare fault isolation strategies that may be optimized on a number of user input factors, including the generation of multiple failure and/or replaceable unit fault trees.

The fault trees that are generated are useful in the development of:

- UUT diagnostic software either for BIST or ATE TPS.
- TRD generation.
- Technical troubleshooting manuals. An IBM PC/AT/XT utility has been developed for this purpose.

STAMP may be used to develop the fault trees utilized by portable maintenance aids or the interactive fault isolation strategies that they employ (Ref[3]).

STAMP has demonstrated an ability to predict M-Demo results (Ref[1]).

STAMP is continually being improved.

One recent improvement is provision for IBM-PC assistance in the task of entering STAMP dependency input data. This utility helps make this task less tedious, reduces the amount of proofreading required, and reduces the chance of an erroneous input. This process entails transforming a list of elements into a picture which is easier to check and work with.

REFERENCES:

1. Simpson, W.R., "STAMP Testability And Fault Isolation Applications, 1981-84," IEEE AUTOTESTCON 1985.

2. Simpson, W.R., and J.R. Agre, "Experience Gained in Testability Design Trade-Offs," IEEE AUTOTESTCON 1984.

3. Simpson, W.R., "Active Testability Analysis and Interactive Fault Isolation Using STAMP," IEEE AUTOTESTCON 1987.

4. Naval Ocean Systems Center, "Testability Analysis Tools On A Military System," Technical Report, September, 1987.

5. ARINC Research Corporation
Attn: Dr. Randy Simpson
2551 Riva Road
Annapolis, MD 21401
(301) 266-4066

4.2.1.8 TESAP

NAME: **TESAP** - Test Strategy Assessment Program

YEAR:

FUNCTION: Allows the comparison of the costs of various testing strategies given varying fault spectra to make general assessments of what combination of tests are best.

CAPACITY:

CPU TIME:

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N

DESIGN ENV: Written in LOTUS 1-2-3

CIRCUIT DESCRIPTION: Not required

USE PREREQUISITE: One must enter parameters: times of test/repair, tester FOM, defect rate; also labor rates, # of boards, etc.

DEVELOPER: Hewlett Packard

COMMENTS: As a product matures, its quality improves and thus the likelihood of a fault decreases. This program advises when certain tests are no longer required.

REFERENCES:

1. Hamilton, S. A., "Optimizing Test Strategy Through Computer-Aided Test, ATE Instr. Conf East, 1987.

Please note that TESAP is not being marketed; however, inquiries can be directed to:

Hewlett Packard
3 Crossways Park West
Woodbury, New York 11797
(516) 682-7830/7800

c/o: Eileen N. Meenan
Sales Representative
Electronic Instruments
Eastern Sales Region

4.2.1.9 TIME

NAME: TIME - Testability Interfaced Maintainability Estimates

YEAR: 1989 (prototype)

FUNCTION: TIME is an automated maintainability prediction tool which takes into direct account the influence of testability/diagnostic design, and maintenance and repair philosophies on maintainbility. Testability characteristics and maintenance philosophies are directly incorporated into the prediction model. These include fraction of faults isolatable/detectable, levels of ambiguity, application of secondary fault isolation means and troubleshooting concepts pertinent to various levels of system indenture. Six maintenance philosophies are available from which to choose. Each philosophy has separate models for computing elemental maintenance task times. The task models relate to values for average time required to detect, isolate, acquire, disassemble, interchange, align, reassemble, checkout, and start-up.

CAPACITY:

CPU TIME:

APPLICATION: VLSI PCB SUBSYS SYSTEM
Useful for systems that are composed of ambiguity groups for the purpose of fault isolation and maintenance.

ACQUISITION PHASE:

PUBLIC DOMAIN: Y (GFE) (When complete)

DESIGN ENV: Written in Turbo Pascal for IBM PCs.

CIRCUIT DESCRIPTION: The system is described by its ambiguity group makeup.

USE PREREQUISITE: Required input parameters include various testability, maintainability, and reliability variables such as: failure rates, fraction of faults isolatable/detectable, times required to disassemble, remove and replace, reassemble, checkout, align, and startup.

DEVELOPER: RADC/RBET (Joe Caroli)

COMMENTS: The prediction technique is a modification of MIL-STD-472, Procedure 5.

REFERENCES:

RADC/RBET (Joe Caroli)
Griffiss AFB, NY 13441-5700
COMM: (316)330-4205
AV: 587-4205

4.2.2 Fault Simulation Tools

This section contains software tools available that are a subset of tools that aid in the prediction of the effectiveness of testability/diagnostic capabilities. In particular, they aid in predicting the TFOM, fraction of faults detected (FFD).

They are called fault simulation tools or fault simulators.

Automatic Test Generation (ATG) and fault simulation go hand-in-hand and sometimes are one in the same tool. However, for classification purposes, ATGs can be found in this appendix for design implementation tools under the subheading called Diagnostic Authoring (3.3).

Both tasks are essential during the design and evaluation of an effective diagnostic system.

Many systems that do not plan to include a particular diagnostic capability as part of the design process, deploy fault simulators simply because they make the production validation process economically feasible. However, these tool capabilities very much serve in the assistance of including diagnostic capabilities. For example, they may be useful in assuring ETE effectiveness or compatibility, for evaluating test vector sets deployed by BIT, or for fault-tolerant design and evaluation.

Tools that aid in fault simulation all have the following features in common:

- They would be deployed during the Full-Scale Development Phase for the purpose of deriving an effective and optimized set of test vectors necessary to perform the task of validating the functional design during production.

- They all apply to digital circuitry and primarily process faults at the gate level.

- Although each tool is capable of being run separately, ATGs and fault simulators are almost always run together. The ATG provides the test vector set. The fault simulator, either statistically or deterministically, evaluates what percentage of the total faults considered would be detected by such a test vector set.

- Speed enhancement is this family of tools prime competitive sport and each has a unique way of remaining in the arena. Refer to the chart with the column heading "INCREASED SPEED METHOD".

A summary description of tools that aid in fault simulation are provided in the following table.

ACRONYM	APPLICATION	INCREASED SPEED METHOD	COMMENTS
AIDA	VLSI/PCB/ SUBSYSTEM	A RISC CO SIMULATOR IS MOUNTED INTO THE SCAN WORKSTATION	PROVIDES EXCELLENT DESIGN ASSISTANCE
BITGRADE	VLSI/PCB	SPECIALIZES IN FAULT GRADING RANDOM PATTERN GENERATED TEST VECTORS	DETERMINES THE FAULT COVERAGE OF BILBOs, MISRs, LFSRs, ETC.
CADAT 6	VLSI/PCB	USES GATE ACCELERATOR	ALL LEVELS OF DEVICE MODELING FROM SWITCH LEVEL TO HARDWARE
HITS	VLSI/PCB	USES CONCURRENT SIMULATION	ASSISTS IN TPS DEVELOPMENT
IKOS 800	V[illegible]	SPECIAL PURPOSE HARDWARE LINKED TO HOST COMPUTER	FAST
LASAR VER 6: JUDGE 6 & PROSECUTOR	VLSI/PCB/	CONCURRENT SIMULATION	POST PROCESSING AVAILABLE TO MAKE STIMULI COMPATIBLE WITH TARGET TESTER
QUICKFAULT	VLSI/PCB/ SUBSYSTEM	LAN ACCELERATION MODEL	SUPPORTS MANY MODEL TYPES; SWITCHES
	SYSTEM		GATES, BEHAVIORAL, HARDWARE MODELS, QUICKPARTS
SOCRATES	VLSI	USES A FAST FAULT SIM AND IMPROVED ATG FAN	CARRIES INFO HEURISTICALLY FROM TESTABILITY ANAL TO ATG ALGORITHM PROCESS
STAFAN	VLSI	CALCULATES FAULT DETECTION PROBABILITY	CPU TIME INCREASES LINEARLY WITH INCREASE IN NUMBER OF GATES
STATGRADE	VLSI	SEE ABOVE	SEE ABOVE
TESTGRTADE	VLSI/PCB	CONCURRENT SIMULATION WITH LAN ACCELERATION OPTION	OPTIMIZE TEST VECTOR SET WITH STATGRADE, THEN USE TESTGRADE
THESEUS	VLSI/PCB	AFTER EACH TEST VECTOR, ALL DETECTED FAULTS ARE NO LONGER CONSIDERED	PROVIDES AN ALTERNATIVE TO SCAN DESIGN
ZYCAD	VLSI/PCB SUBSYSTEM	SIMULATION IMPLEMENTING IN HARDWARE EMPLOYING PARALLEL PIPELINE PROCESSING TECHNIQUES	ZYCAD'S NEXTGEN IS AN ACCELERATED ATG

4.2.2.1 AIDA

NAME: **AIDA Fault Simulation**

YEAR: 1987

FUNCTION: **The AIDA Fault Simulator performs accelerated full or partial fault grading at workstations for test set evaluations. It also provides a fault dictionary, including a list of undetected faults and locations.**

CAPACITY: **5,000 - 1,000,000 gates**

CPU TIME: **Negligible.**

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (PRTY)

DESIGN ENV: **Gate array and RISC co-processor mounted into an Apollo workstation; also on Sun. These tools are part of AIDA design system which includes a logic simulator, a timing verifier, and others. They also accepts designs translated from non-AIDA systems.**

CIRCUIT DESCRIPTION: **AIDA design language**

USE PREREQUISITE:

DEVELOPER: **AIDA CORP., Santa Clara, Ca.**

COMMENTS: **Testability can be achieved quite efficiently with this approach. Up to 100% testability if scan design methodology for targeted system is used. AIDA also supports Boundary Scan.**

- **The AIDA Fault Simulator can be used with the AIDA ATPG. When the ATPG creates a test vector, the Fault Simulator automatically checks what other fault classes can be detected by that vector.**

- Both the AIDA Fault and Logic Simulators are accelerated by the AIDA Co Simulator processor.

- AIDA recently acquired by Teradyne; refer to Lasar Version 6.

REFERENCES:

Pierre Wildman, Product Marketing Manager
c/o AIDA Corporation
5155 Old Ironsides Drive
Santa Clara, Ca. 95054
(408)980-5200

4.2.2.2 BITGRADE

NAME: **BITGRADE** - Built-In Test Grade

YEAR: 1986

FUNCTION: BITGRADE determines fault coverage for self-test designs, is interactive, and is not hindered by scan designs.

CAPACITY:

CPU TIME: (Example) 9,386 simulated faults, requiring 256 random test patterns, consumed 894 seconds of CPU time using a 68010 base workstation.

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (PRTY)

DESIGN ENV: There are versions for Apollo, SUN, VAX, IBM; used in conjunction with VERILOG, TESTGRADE, TESTSCAN, STATGRADE, and PCBLIB, other Gateway products.

CIRCUIT DESCRIPTION: BITGRADE'S HDL

USE PREREQUISITE: Enter circuit description from a netlist or a hardware description at a gate level.

DEVELOPER: Gateway Design Automation Corporation.

COMMENTS: Self-test random pattern generators can require hundreds of thousands of test patterns. BITGRADE is specifically designed to deterministically fault grade these tests.

- After LFSR and MISR descriptions are input, BITGRADE determines their capability.
- It supports all scan designs.

REFERENCES:

Gateway Design Automation Corporation
6 Liberty Way
PO Box 573
Westford, MA 01886
(617)692-9400

4.2.2.3 CADAT 6 Fault Simulator

NAME: **CADAT 6 Fault Simulator**

YEAR:

FUNCTION: The CADAT Fault Simulation option utilizes a functional Concurrent Fault Simulation algorithm to analyze the impact of potential manufacturing errors and circuit failures. The algorithm optimizes simulation throughput for all levels of device modeling, from switch level to hardware.

CAPACITY:

CPU TIME:

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (PRTY)

DESIGN ENV: CADAT 6 will run on the following hardware platforms:

Apollo/AEGIS with AUX
IBM PC-AT/PC-DOS (Personal Cadat)
IBM MVS
IBM VM/CMS
SUN Microsystems UNIX
VAX ULTRIX
VAX VMS

LANA (Local Area Network Acceleration) is also available.

CIRCUIT DESCRIPTION: Behavioral Design Language (BDL)

USE PREREQUISITE: Behavioral description or a circuit netlist must be input.

DEVELOPER: HHB Systems

COMMENTS: Refer to CADAT 6 in the Tools For Design Aid section.

REFERENCES:

HHB Systems
Attn: Mr. Kenneth Lipston
1000 Wyckoff Ave.
Mahwah, NJ 07430
(201) 848-8000

4.2.2.4 HITS

NAME: **HITS** - Hierarchical Integrated Test Simulator

YEAR: 1987 - HITS 14

FUNCTION: HITS is a Digital Automatic Test Program Generator (DATPG) software system. Its functions as a software tool are to assist in the development of digital Test Program Sets (TPS) and to serve as a means to evaluate/verify digital designs. The modules of particular interest are:

- PRIMARY MODEL PROCESSOR - Compiles and processes the user-defined network model and produces the initial circuit topology tables and data base.
- SWAPPER - Determines the circuit or network model fault universe, and on subsequent executions, at the user's option, the SWAPPER will produce fault segments or fault partitions for circuit elements identified by the user. The SWAPPER creates fault equivalence classes required by the TESTSIM module.
- TESTSIM - The function of this module is to generate and evaluate test patterns to detect the failures being considered for the current test segment.
- SIMULATE - Performs fault free and fault simulation using a concurrent methodology. Its function is to determine the quality of stimulus for the given circuit topology. It determines fault detection, fault isolation, and produces the fault dictionary.
- PROBE - The PROBE module is provided as a backup to the primary means of fault isolation, which is the fault dictionary. If the number of indicated replaceable packages is too high for an isolated failure, the data generated by the PROBE module, in conjunction with suitable hardware on the ATE, can be used to isolate to the failed node.

CAPACITY: The maximum HITS can process is:

- 150,000 nodes
- 75,000 nets
- 50,000 blocks
- 32,000 fault isolation sets

CPU TIME:

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (GFE)

DESIGN ENV: VAX 11/7XX; FORTRAN - 77 (O/S: VMS, UNIX)

CIRCUIT DESCRIPTION: HITS uses a Circuit Description Language (CDL), which is similar to a wire-list format, is simple to use, contains ATLAS-like statements, provides the capability to enter ROM/RAM & PLA data, and provides the capability to enter "black box" models using the Register Transfer Language (RTL) tool.

The RTL is Pascal-like and enables users to write behavioral descriptions of models/components which lack structural detailed information.

In addition, the user may define unique MACROS, access a system MACRO library, and/or utilize system primitives composed of combinational gates, sequential devices, and functional primitives..

USE PREREQUISITE: One must input description of UUT into model processor module, identifying components from HITS standard cell library, or use system primitives.

DEVELOPER: Naval Air Engineering Center

COMMENTS: HITS is very inexpensive and is always being updated and improved.

- In order to avoid excessive CPU time while using HITS, the UUT should be well designed for testability.
- Presently, HITS is predominantly used for modules integrated with MSI/LSI/VLSI circuits.
- TPS development on a firm fixed- price basis significantly increases management risks. Using HITS requires taking action to minimize cost and scheduling risks and improve product quality. See Ref[2] below.

REFERENCES:

1. HITS Users Guide
   Avionics Support Equipment Div.
   Naval Air Engineering Center
   Lakehurst, N.J. 08733
   TR-AIRTASK A552-5522/051D/3W08520000

2. Gorham, G.B., "Managing Risk in the HITS Environment," Test & Measurement World (magazine), Nov 1987, p 26.

4.2.2.5 IKOS 800

NAME: IKOS 800

YEAR: 1986

FUNCTION: A design verification system offering high-speed stimulus processing and accelerated logic simulation by means of special-purpose hardware linked to a host computer.

A unique IKOS Waveform Capture Stimulus Generator is provided that should prove to be a considerable improvement over traditional stimulus generation methods allowing the ASIC designer to quickly create millions of simulation vectors that emulate real system operation.

In addition to fault-free simulation for logic validation, the IKOS 800's Logic Simulation Hardware Accelerator supports high-speed stuck-at fault simulation in unit-delay simulation mode. The user may specify a table of faults to be simulated or may elect to simulate all possible stuck-at faults. The fault coverage report lists all faults which have not been detected by the proposed test program and those faults may be recycled back into the fault table for rapid re-simulation.

CAPACITY: For the Stimulus Processing Hardware Accelerator the capacity has the following linear relationship with the amount of Stimulus memory available:

- 4 MBYTE of Stimulus memory: 2.5 million I/O events
- 8 MBYTE of Stimulus memory: 5 million I/O events
- 16 MBYTE of Stimulus memory: 10 million I/O events

For the Logic Simulation Hardware Accelerator the capacity has the following linear relationship with the number of Evaluator boards available:

- 1 Evaluator board: 16 thousand primitives
- 2 Evaluator boards: 32 thousand primitives
- 3 Evaluator boards: 48 thousand primitives
- 4 Evaluator boards: 64 thousand primitives

CPU TIME: For the Stimulus Processing Hardware Accelerator, the rate at which the resident stimulus is presented is one million I/O events per second.

For the Logic Simulation Hardware Accelerator, the rate at which the events are processed in the Timing Mode is:

- 1 Evaluator board: .5 million I/O events per second.
- 2 Evaluator boards: 1 million I/O events per second.
- 3 Evaluator boards: 1.5 million I/O events per second.
- 4 Evaluator boards: 2 million I/O events per second.

The rates at which the events are processed in the Unit-Delay Mode are ten times faster than the rates of the Timing Mode.

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (PRTY)

DESIGN ENV:

Host Computers:

- IBM PC/AT
- IBM PC/RT (future support)
- Apollo DN3000 (future support)

Host Data Link:

- 5 MBit/second serial link with standard 25 pin RS232 connectors

Maximum cable length: 50 ft. (15.2m)

CIRCUIT DESCRIPTION: IKOS Systems provides both functional and timing libraries for a number of commercial semi-custom vendors. In addition, the IKOS 800 includes library development tools to allow users to input new semi-custom libraries or edit existing libraries. The basis of the IKOS 800 library support tools is the Delay Form and the user is provided with Delay Forms for a wide variety of common semi-custom macro-cell functions (e.g., two-input NAND gate, D flip-flop with preset, clear and scan test inputs, etc.).

USE PREREQUISITE: The IKOS 800 will accept semi-custom netlists in a variety of formats. The IKOS 800 netlist compiler will combine the user netlist with semi-custom library data and create the data base required for simulation. The netlist

compiler can link multiple netlists representing individual "pages" of a single design or complete netlists for several different circuits.

DEVELOPER: IKOS Systems, Inc.

COMMENTS: IKOS gives substantial improvement in simulation speed over software simulators - an 8-hour simulation on a Mentor/DN 3000 takes less than 30 seconds on the IKOS simulation system.

Much simulation is a good thing. The more the design engineer can simulate the more he can test and, therefore, the more likely it is that his circuit will work correctly. Being able to rapidly simulate is crucial to being able to abundantly simulate.

IKOS currently supports 32 ASIC libraries from 15 ASIC vendors.

Asynchronous system interfaces to the ASIC can be simulated.

In order to minimize netlist compile time, the IKOS 800 caches the appropriate semi-custom library data in high-speed RAM.

REFERENCES:

IKOS Systems, Inc.
145 N. Wolfe Road
Sunnyvale, CA 94086
(408) 245 1900

4.2.2.6 LASAR VERSION 6 - JUDGE

NAME: **LASAR VERSION 6 - JUDGE**

YEAR: 1982, origin 1978

FUNCTION: LASAR Version 6 provides a CAD simulation system for design verification and test program generation incorporating a testability subprogram called JUDGE. JUDGE provides:

- A fast, concurrent time-based simulation of stuck-at 0, 1, Z, and X faults, opens, and shorts.
- An accurate measure of test thoroughness: the user can decide when the desired level of fault coverage has been reached, or where additional test vectors are needed to meet fault coverage objectives.
- Identification of undetected faults, possible redundant circuitry, testability problems, or logic errors.
- Refer to Section 3.1.7 for more information on LASAR and to Section 3.3.2.3 for more information on the ATPG component of LASAR called PROSECUTOR. The JUDGE and PROSECUTOR subprograms, working together, provide an integrated environment for determining the test effectiveness of digital circuitry.

4.2.2.7 QUICKFAULT

NAME: **QUICKFAULT**

YEAR:

FUNCTION: An interactive deterministic fault simulator with the following features:

- Local Area Network (LAN) acceleration
- 12 simulation states; (1,0,X) (strong, resistive, HI Z, indeterminate)
- Supports all Mentor Graphics model types (switches, gates, behavioral, hardware models, Quickparts)
- Statistical projection
- Faults displayed on schematic
- Reports actual and percent detected, possible detected, oscillatory, and undetected faults
- Fault detection charts
- Fault dictionary
- "Stuck-at" fault model (input and output pins)
- Interactive fault selection
- Hierarchical selection
- Graphical fault selection
- Pause/Restart and Save/Restore capability

CAPACITY:

CPU TIME: Extensive jobs require overnight runs

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (PRTY)

DESIGN ENV: Mentor Graphics engineering workstations

CIRCUIT DESCRIPTION: See above for all Mentor Graphics model types

USE PREREQUISITE: Schematic must be captured using a Mentor Graphics model type; use with QUICKSIM.

DEVELOPER: Mentor Graphics Corporation

COMMENTS: Listed below are some further advantages of QUICKFAULT:

- Displaying results graphically on the schematic can save tens to hundreds of hours of analysis time.
- Having test vectors developed and evaluated using the same design engineering data base saves time and money.
- Provides friendly user prompts.
- The performance increase with LAN acceleration is typically .9N, where N is the number of workstations used in the analysis, and 1 is the analysis time on one workstation. For example, a benchmark on a large gate array design took 15 hours, 40 minutes on one DN 3000. Using three DN 3000s, the run time was reduced to 5 hours, 20 minutes.
- Behavioral Logic Models (BLMs) are an effective modeling method for use with QUICKFAULT. BLMs have proven to be an effective method of addressing the increasing complexity of fault simulation for board and system level designs.

REFERENCES:

Frank Binnenkyk
Product Manager
Design and Analysis Division
Mentor Graphics Corporation
8500 S.W. Creekside Place
Beaverton, OR 97005-7191
(503) 626-7000

4.2.2.8 STAFAN

NAME: **STAFAN** - Statistical Fault Analysis

YEAR: 1984

FUNCTION: STAFAN performs a fault-free logic simulation and the data collected is used to calculate the fault detection probability for stuck-at-one and stuck-at-zero faults.

CAPACITY: 3,000,000 gates.

CPU TIME: Will increase linearly as the number of gates increase. Please note that CPU time will increase **exponentially** for traditional deterministic fault evaluation.

APPLICATION: VLSI (Planned extension to PCB)

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (PRTY)

DESIGN ENV: The development program was done on a CMOS VLSI fault simulator; it is presently part of MIDAS - Modular Integrated Design Automation System, Control Data's design support environment.

CIRCUIT DESCRIPTION: MIDAS' Logic Interconnect Language netlist.

USE PREREQUISITE: Fault-free simulation of the VLSI circuit.

DEVELOPER: Initially AT&T BELL Labs; currently Control Data Corp.

COMMENTS: STAFAN adds only a small overhead to the fault-free simulation task, requiring two operations. The first involves updating the zero and one counters for every test vector, and the other involves computing statistical controllabilities, observabilities, detection, and fault coverage, which is only performed once every N vectors. Thus STAFAN's main overhead is due to updating the counters.

Application of STAFAN is limited to combinational circuits.

REFERENCES:

1. Jain, S.K., Agrawal, V. D., "STAFAN: An Alternative To Fault Simulation," Design Automation Conference, 1984.

2. Control Data Corporation
Attn: Robert Bigge HQM274
ADAM Marketing
Minneapolis, MN
(612) 853-3117.

4.2.2.9 STATGRADE

NAME: **STATGRADE**

YEAR: 1987

FUNCTION: It estimates total fault coverage of test vectors through statistical fault analysis. After measuring CY & OY values of simulated circuit nodes, the program will output the fault coverage of a given test vector. It also lists statistically undetected faults to promote interactive test generation for specific areas.

CAPACITY:

CPU TIME: 20 to 50 times faster than concurrent fault simulators

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (PRTY)

DESIGN ENV: Workstations to mainframes, including APOLLO & SUN w/s; DEC VAX & MICRO VAX computers; IBM mainframe. It is written in FORTRAN-77.

CIRCUIT DESCRIPTION: STATGRADE's HDL; User defined MACROs are possible and extensive primitives are available.

USE PREREQUISITE: Build s/w model of the circuit with STATGRADE's HDL; use in conjunction with VERILOG, TESTSCAN, TESTGRADE, and BITGRADE, other Gateway software products.

DEVELOPER: Gateway Design Automation Corp.

COMMENTS: STATGRADE is used to first establish a set of test vectors and provide a reasonable confidence in how effective this set of test vectors is. Afterward a more accurate fault coverage determination can be made with a fault simulator (TESTGRADE).

STATGRADE can interface with other CAE system test pattern sets.

REFERENCES:

Gateway Design Automation Corporation
Six Liberty Way
PO Box 573
Westford, Ma. 01886
(617)692-9400

4.2.2.10 TESTGRADE

NAME: **TESTGRADE**

YEAR: 1987

FUNCTION: Concurrent processing, comparing a faulted machine with a good machine model. Once a set of test vectors is developed, TESTGRADE can create a fault dictionary, a listing of specific faults with associated responses to given test vector inputs. Test pattern grading determines the effectiveness of a test set.

CAPACITY: Random fault sampling and incremental test grading make processing large jobs more feasible.

CPU TIME: Orders of magnitude faster than earlier generation fault simulators

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (PRTY)

DESIGN ENV: Workstations to mainframes, including APOLLO & SUN w/s; DEC VAX & MICRO VAX computers; IBM mainframe; ELXSI multi - computers. It is written in FORTRAN-77.

CIRCUIT DESCRIPTION: STATGRADE's HDL; User defined MACROs are possible and extensive primitives are available.

USE PREREQUISITE: Build s/w model of the circuit with TESTGRADE's HDL; use in conjunction with VERILOG, TESTSCAN, STATGRADE, and BITGRADE, other Gateway software products.

DEVELOPER: Gateway Design Automation Corporation

COMMENTS: TESTGRADE provides a facility to generate test patterns automatically.

It accepts test and response patterns from other simulators.

There is a parallel processing version using multiple workstations for speed improvement.

It uses proprietary techniques to reduce the memory required for storing faulty machine models.

REFERENCES:

Gateway Design Automation Corporation
Six Liberty Way
PO Box 573
Westford, MA 01886
(617)692-9400

4.2.2.11 ZYCAD

NAME: **ZYCAD** Fault Evaluator

YEAR: 1985

FUNCTION: Perform fault simulation tasks. It implements the simulation in hardware employing parallel pipeline processing techniques. This affords greater speed than possible with software routines which must fetch instructions from external memory and execute them sequentially.

CAPACITY: up to 512K gates

CPU TIME: Claimed speed up to 200 times faster than software based simulators on mainframe computers.

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (PRTY)

DESIGN ENV:

Hardware:	Operating System:
DEC VAX	VMS 3.7 / VMS 4.0
DEC VAX	ULTRIX (Berkeley 4.2 UNIX)
IBM	MVS/2.0 / CMS
Apollo	AEGIS 9.X
Sun	UNIX 4.X

CIRCUIT DESCRIPTION: Host converts user's netlist to Zycad Intermediate Format (ZIF). ZIF can be used by other ZYCAD simulators. ZIF is not a simulation netlist language.

USE PREREQUISITE: Conversion to ZIF. ZILOS and NEXTGEN ATG interfacing available.

DEVELOPER: ZYCAD

COMMENTS: The concurrent fault simulation algorithm is based on the event-driven logic simulation algorithm. The concurrent algorithm only processes modeling elements which are active (switching). Any modeling element or group of modeling elements which is not active is ignored.

Users determine the number of times a fault is potentially detected until it is considered a hard detected fault.

Use of ZILOS, a friendly simulation environment, promotes the use of the same data base files for logic simulation, fault simulation, test analysis, and test generation.

A number of optional translators are available that enable engineers to convert their design descriptions to ZILOS format and immediately run simulations on the Fault Evaluator without disrupting existing tools:

- Daisy to ZILOS
- Mentor to ZILOS
- TEGAS to ZILOS
- HILO to ZILOS

The following interface to the Fault Evaluator within the existing simulation environment:

- LSI Logic MDE
- CDC MIDAS
- CAE/TEK

REFERENCES:

ZYCAD
3900 Northwoods Drive, Suite 200
St. Paul, MN 55112
(612) 490-2500
(800)631-5040 (within MN)

5.0 DEMONSTRATION TOOLS

This section contains tools available to aid in the task of demonstrating the effectiveness of system diagnostics.

Only one tool is included at this time.

5.1 MIL-STD-471A Maintainability Verification, Demonstration, Evaluation

YEAR: 1978

FUNCTION: Provides standard procedures for evaluation and demonstration of equipment/system built-in test and external test subsystem fault isolation and testability attributes which relate to maintainability and various logistic support factors which are impacted by maintainability.

CAPACITY: N/A

CPU TIME: N/A

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N

DESIGN ENV: Pencil, paper, and testing facilities.

CIRCUIT DESCRIPTION:

USE PREREQUISITE: Ample development or release of the system requiring diagnostic capability.

DEVELOPER: Rome Air Development Center/RBE

COMMENTS: Considering the mammoth task of either establishing contractual requirements or performing the actual task of demonstrating and validating a systems diagnostic capability, this standard is quite good.

After reviewing a series of tests, as described in this standard, sound engineering judgment plays an important role in this task because the ruling as to how well a system is diagnosed will never be black or white.

REFERENCES:

Naval Publications & Forms Center
Attn: PFC 1032
5801 Tabor Ave.
Philadelphia, Pa. 19120

6.0 MATURATION TOOLS

This section contains software tools available to aid in the task of diagnostic capability maturation.

Only two tools are listed in this section at this time.

6.1 CITS/CEPS

NAME: **CITS/CEPS** - Central Integrated Test System/CITS Expert Parameter System

YEAR:

PH I - CONCEPT (5 1/2 MO) 1985
PH II - DEM/VAL (17 1/2 MO) 1986 - 1987
PH III - PRODUCTION/DEPLOYMENT (expected start 9/88)

FUNCTION: CEPS is a rule-based expert system, initially targeted to increase fault isolation through the use of expert system technology. As a ground-based maintenance aid, it is intended to reduce maintenance man hours expended resolving ambiguous failures, false alarms, repeat/recurring write-ups, cannot duplicates, and retest okays.

To accomplish this, CEPS accesses a significant amount of on-board recorded parametric data, and ground-based maintenance historical data. The on-board recorded data is provided by the CITS system, and is augmented by design, and maintenance expertise, and combined with the historical tracking mechanism that is part of the CEPS system. The provision of this type of tracking system, provides a natural source of feedback from field experience, which can be readily available for future design and development of weapon systems.

CAPACITY: N/A

CPU TIME: N/A

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (GFE) See first comment

DESIGN ENV: Expert system demonstrated on a Symbolics computer and converted to run-time system. It is fielded on a Microvax II, which also contained a Data Base Management System (DBMS).

CIRCUIT DESCRIPTION: N/A

USE PREREQUISITE: Access to existing maintenance AF Management Information System (MIS) called CAMS (Core Automated Maintenance System), used for tracking maintenance actions.

DEVELOPER: Rockwell International under contract from the US Air Force.

COMMENTS: It is at this time premature to tell to what extent the government will allow public access to CEPS material. The software itself basically only applies to the B-1B aircraft. What is 're-usable,' however, are the lessons learned in the overall process of feeding back information to improve the diagnostic capability.

CEPS data feedback will conceptually make improvement possible to both avionic system testability and CEPS performance.

Fault isolation improvements discovered will be integrated into CITS software, T.O. fault isolation procedures, and I-Level TPS T.O.s.

CEPS usefulness is directly dependent upon a user-friendly system that is accepted by the maintenance technicians.

REFERENCES:

1. Anne M. Stanley, "B-1B Integrated Diagnostics," NSIA Conference at Alexandria, VA, Feb. 1986.

2. Ken Derbyshire, "B-1B On-Board Fault Detection/Fault Isolation System," IEEE AUTOTESTCON 1985.

3. Rockwell International, Mr. Britt, Autonetics Program Manager; Anne Stanley, lead engineer (714) 779-3379.

6.2 IDSS - FA

NAME: **IDSS - FA** - Feedback Analysis

YEAR:

FUNCTION: Collects the following type of field failure and diagnostic data from all ADS (3.2.1) sites:

- Data for each recorded site such as identification of the equipment configuration and the environment in which it has been operating.
- The symptoms observed and the actual faults isolated.
- Information regarding test performance at each site including relevant performance statistics.

Once this collected data has been consolidated, statistical analysis is performed on the data. Summary reports are prepared which provide site-to-site performance differences and the factors such as environment and skill levels which account for these differences.

CAPACITY:

CPU TIME:

APPLICATION: VLSI PCB SUBSYS SYSTEM

ACQUISITION PHASE: CONCEPT DEM/VAL FSD PRDCTN DPLYMNT

PUBLIC DOMAIN?: Y/N (GFE)

DESIGN ENV:

CIRCUIT DESCRIPTION: N/A

USE PREREQUISITE: Provision for collecting ADS data from all sites of deployment.

DEVELOPER: Harris Corporation under contract from the US Navy.

COMMENTS: The FA conceptually makes improvement possible to both weapon system testability and IDSS performance.

The global FA "learning" loop is to be distinguished from the local "learning" loop resident at each site which updates system parameters based on information derived from the results of each diagnostic session. This local loop is a quicker but more elementary "learner" and is subject to more rigid update time requirements.

REFERENCES:

1. Dr. Bruce J. Rosenburg, "The Navy Integrated Diagnostic Support System - System Overview, Architecture and Interfaces," IEEE AUTOTESTCON 1987.

2. Navy Point of Contact
NAVSEA - Code CEL-DS
Washington, DC 20362-5101
(202) 692-2035/2036

Table of Contents

1.0	SCOPE	2
1.1	Purpose	2
1.2	Application	2
1.3	Definitions	2
2.0	OVERVIEW	2
2.1	Testing Process	2
2.2	Cone of Tolerance Overview	5
3.0	VERTICAL TEST METHODS AND CRITERIA	7
3.1	Factory/Design Environment	7
3.1.1	Overview	7
3.1.2	Typical Design Analysis Procedure	8
3.1.2.1	Worst-Case Analysis	9
3.1.2.2	Root Sum of Squares (RSS) Statistical Tolerancing	9
3.1.3	Factory ATE	11
3.2	Factory and O-(BIT)/I-/D-Level Interface	12
3.3	Generic Data Base Requirements	16
3.4	Vertical Testability Method Alternatives	16
3.4.1	Common ATE	17
3.4.2	ATE Emulation	19
3.5	Criteria	21
3.5.1	Tolerance Cone Budgeting (TCB)	22
3.5.2	Modular Emulation ATE	22
3.5.3	Common ATE	22
4.0	DESIGN PROCEDURES AND DOCUMENTATION	23

1.0 SCOPE

1.1 Purpose

The purpose of this appendix is to provide guidance on the implementation of vertical test methods as part of the diagnostic design process.

1.2 Application

This appendix is composed of the following sections:

2.0 Overview

3.0 Vertical Test Methods and Criteria

4.0 Design Procedures and Documentation

Application of this guidance will ensure that vertical testability goals are met.

1.3 Definitions

Vertical Test Methods:

A system engineering approach for establishing and maintaining compatible test methods and data correlation (i. e., test tolerances) through the various echelons of weapon system development and support (i. e., development, production (factory), Intermediate Level, Depot Level, Organization Level).

Vertical Commonality:

Vertical commonality is the utilization of common testing resources between levels of maintenance. Implementation of the vertical commonality concept manifests itself in "shrinking" of the cone of tolerance phenomena and the enhancement of testing integrity between levels of maintenance.

2.0 OVERVIEW

2.1 Testing Process

Testing is necessary to successfully design, develop, produce, and maintain an operational system. Throughout all phases of a program, tests are performed to assure that the product, as designed and manufactured, meets the customer-prescribed requirements. Tests to verify design concepts, interface

capabilities, and performance capabilities are conducted during the Validation Phase of a program on engineering prototype models or with simulations in a computer-aided engineering environment. Qualification tests are then performed on full-scale engineering development models to prove that the unit, as designed and fabricated, meets the system requirements through all operating conditions. This normally includes environmental tests, flight tests, reliability tests, and maintainability demonstrations on the design configuration. Once the design is proven and the unit qualified to meet its operational requirements, the system is ready for production. There again, testing is an important criterion to assure delivery of failure-free operational systems. Factory test normally includes receiving inspection on incoming components and subassemblies, test on circuit card assemblies and modules, and assembly and/or acceptance test on deliverable units. After the unit is delivered and in operation, testing is again necessary to maintain the system free from operational failures. In military systems, this maintenance support is typically implemented at three operational levels: at Organizational Level, on the operational vehicle; in an Intermediate Level shop, at the operating site; at a permanently located Depot Level shop; or, in some instances, at the factory.

Throughout this testing process, it is imperative that an integrated approach to test and maintenance is effected, in order to ensure that CND and RTOK instances are minimized and that the integrity of the testing process is maintained.

Design verification typically includes bench tests on functional prototype models conducted by engineers or skilled technicians using versatile, highly interactive test equipment which is easily programmed and readily changed. These tests are performed a few times, the results are recorded, and the equipment reconfigured to obtain additional information. Since the objective is to determine the suitability of the design for the operational application, much of this test and evaluation involves simulation of the operational interfaces and environment. Test equipment for this phase normally involves a combination of "off-the-shelf" commercial instrumentation, emulation systems, and specially designed simulation and monitoring equipment.

Factory test requirements begin with components and progress to completed assemblies. The objective of manufacturing test is to eliminate faulty components and manufacturing defects at the lowest level possible.

In-circuit tests are typically performed on circuit card assemblies because of the capability of this type of testing to detect manufacturing defects without application of power and loads on the circuit board. This is especially useful for eliminating incorrect components and soldering defects, without unduly stressing components on the circuit card assembly.

In-circuit test, however, is inadequate for detecting all manufacturing defects. Functional board test must, therefore, also be performed with the unit operating at, or near, its performance characteristics. Functional board tests also provide acceptance criteria for production spares. This operation involves application of stimulus and measurement of response at the board interface connector(s) under conditions of power and load, which should "emulate" as closely as possible the environment the subject UUT will experience in the next higher assembly.

Tests on the next higher assembly normally include manufacturing alignment and verification, followed by a burn-in and/or vibration cycle and, finally, an acceptance test, in preparation for delivery.

Factory test priorities are primarily time-related. They must be available in time to test initial deliveries; they must be performed in minimal time to support throughput and production rates; and they must be time efficient to minimize labor cost and expense of test equipment.

Maintenance support begins with a malfunction in a one-time operational assembly, confirms or detects that malfunction, and supports isolation of the malfunction to a replaceable, failed component. The standard three-level maintenance system begins with operational maintenance implemented on the flight line in an operational vehicle, utilizing built-in test (BIT) to detect and perform fault isolation for black boxes, line replaceable units (LRU) or weapon replaceable assemblies (WRA). Repair action at this stage consists of removal and replacement of the malfunctioning assembly and successful operation of BIT, to verify that the repair rendered the vehicle ready for service.

The malfunctioning unit, or assembly, is then sent to the Intermediate Level shop, where it is tested to determine the cause of the malfunction and isolate the failure to a shop replaceable unit (SRU). Repair is effected by removal and replacement of the faulty SRU and successful performance of the LRU to verify that it is ready for service. The faulty SRU is sent to a Depot Level repair facility, where it, in turn, is tested to determine the cause of the malfunction. Faulty components are removed and replaced, and the SRU is verified ready for service by successful performance test at the Depot or sent to the factory for test and repair.

Maintenance support priorities are typically efficiency related, with the criteria placed on fault isolation and elimination of unnecessary testing caused by RTOKs, CNDs, and fault isolation ambiguity groups. Another problem experienced by maintenance support facilities is inadequate configuration management or delays between development of equipment design changes and updated test capability.

CND and RTOK problem areas can exist between all levels of maintenance from lowest to highest.

- o Organizational and Intermediate Level
- o Intermediate and Depot Level
- o Depot and Factory Level.

Two of the primary contributors to RTOK problems are test tolerance problems (i. e., limit selection) and test bed incompatibilities (i. e., environment and performance capabilities) between levels of maintenance.

2.2 Cone of Tolerance Overview

All electronic circuits can be regarded as an approximation of some idealized mathematical model. The model for a linear circuit is most often a transfer function. The model for a digital circuit is a Boolean equation. In theory, a circuit could be specified in terms of the mathematical model by stating the equation and the allowable deviation over a specific dynamic range of amplitudes and frequencies. Consider, for example, a linear circuit designed to provide some given transfer function. A test of this circuit might consist of selecting a number of discrete frequency signals, measuring the gain and phase shift, calculating actual pole and zero locations, and comparing these to the mathematical model. In practice, this idealized approach to specifying and testing circuit performance is not often used. The reasons are quite pragmatic: real circuits always exhibit nonlinearities and random noise; power sources are never pure DC; and depending on the available instrumentation, some characteristics are more easily measured than others. Consequently, circuit performance requirements must often be specified in terms of both circuit component tolerances and very specific test conditions. Tight restrictions must be placed on power supply accuracy and regulation and precise stimulus, and measurement values must also be specified.

In the design of a weapon system, a great deal of time is usually spent in system integration and checkout. Typically, this is done by setting up a "hot mockup" of the system and interconnecting the individual assemblies. With this situation, it is only natural that much of the engineering effort will be devoted to getting the weapon system to pass system-level tests. As a result, subsystem (I. e., LRU/SRU and lower assembly) test specifications and test procedures are often neglected. Generally, they will be incomplete and inaccurate. Tolerances on the test specifications may be unrealistic. When there is not sufficient time to make accurate tolerance calculations, the tendency is to err in the direction of tighter tolerances to ensure that an assembly that meets these tolerances will work in the next higher assembly. Too often these excessively tight tolerances are propagated

into the depot and field test procedures, giving rise to unnecessary ATE compatibility problems.

In some cases, tight tolerances on power supply and instrument accuracies are an attempt to ensure that the test instruments are as good as, or better than, the ones used by the circuit designer in engineering tests on breadboard or preproduction circuits back at the factory. Tolerances should always be expressed in absolute terms and not relative to a particular test instrument.

In specifying circuit tolerances, there is no substitute for good analysis which is supported by sufficient laboratory testing. The designer should know precisely how variation in any component will affect the operation of the circuit. The system tolerance should be the basis for assigning an error budget to each system and, hence, to each replaceable assembly in the subsystem.

Many engineers are familiar with the systematic approach to allocation of reliability requirements. A corresponding approach to allocation of error budget may be of some value. Of course, the calculations and derivation of mathematical models need not be as well documented or as formalized as the reliability allocations. Formalized documentation requirements are only necessary when there is an interface between technical disciplines--as there often is between the reliability, maintainability, and design disciplines. What is needed is a system approach, without the system approach's paperwork.

In the areas of tolerance calculation and sensitivity analysis, one should consider, as an aid, the use of computer-aided techniques. An advantage of these techniques is the number of calculations that can be completed in a relatively short time. These calculations, however, are only as good as the mathematical model used in the analysis program. The user must be fully aware of limitations in simulation programs used for this purpose.

Thus equipment designers must establish test tolerance values at all levels of test, with tighter tolerances at the Factory Level, increasing as shown in the Cone of Tolerance in Figure 1. This will preclude "bouncing" the UUT back and forth between levels of repair (i. e., RTOK problem). If the designer does not consider the tolerance cone in development, tighter test requirements will result in Organizational and Intermediate Level overdesign and increased acquisition costs for the UUT.

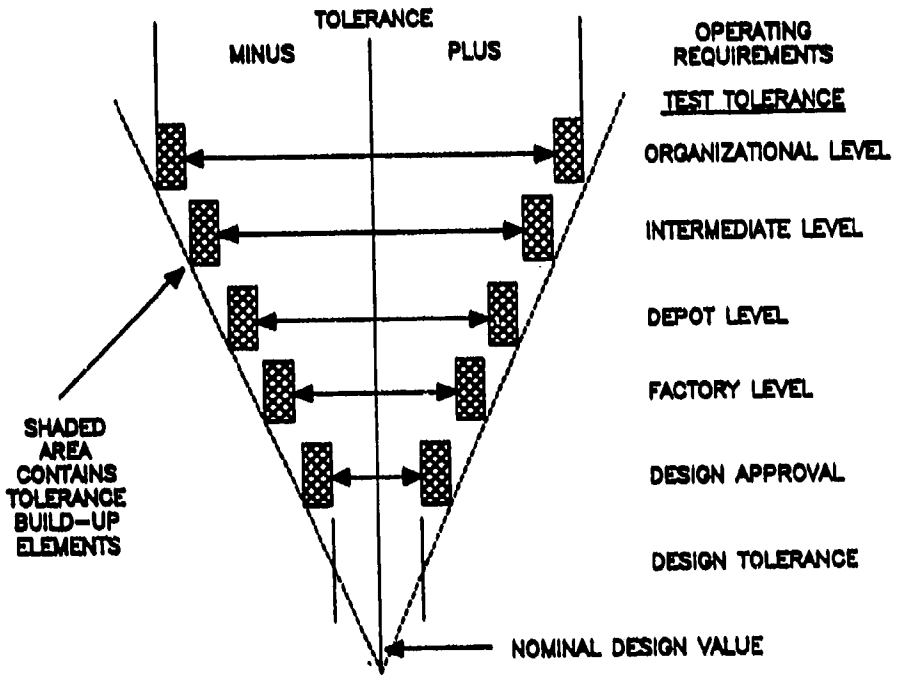


FIGURE 1. CONE OF TOLERANCE

3.0 VERTICAL TEST METHODS AND CRITERIA

3.1 Factory/Design Environment

3.1.1 Overview

The mission of the factory is to design, develop, and performance verify UUT for production applications. The primary testing mission is to ensure that a "good" product leaves the factory. A "good" product is defined as an end item which meets its performance verification goals.

In order to achieve this goal, it is important to establish proper test limits for static and dynamic tests. If the tolerance bands are made too loose, it is possible to pass a defective UUT, and if the tolerance bands are made too tight, it is possible to fail a good one.

Typically, test program set software implemented on factory test equipment is the primary vehicle for performance verifying and fault isolating production UUTs. The following provides an overview of typical design analysis procedures used to derive UUT performance specification limits in a factory environment.

3.1.2 Typical Design Analysis Procedure

There are many priorities considered by the electronic design engineer during the design phase of a product. Many product features are considered: performance, development and product cost, reliability, testability, producibility, maintainability, size, weight, power, and efficiency. The pertinent priority considered here is performance--specifically, the limits that are used to guarantee product performance through test.

During the development of an electronic product, the design engineer typically uses a worst-case design analysis procedure because it is a safe and reliable approach and is the easiest analysis to perform. This analysis virtually guarantees a 100% yield during any type of performance test. If worst-case techniques do not fulfill all of the product requirements, then compromises are usually made that result in a change to the design or a more realistic analysis approach is used to determine output performance limits.

A more realistic design approach can usually be implemented by using a statistical approach, which is a more difficult analysis than worst-case. The most common statistical analysis is termed Root Sum of Squares (RSS). There are other statistical analyses, such as Monte Carlo, that will produce virtually the same results as RSS, when the number of trials becomes larger. Statistical limits place tighter tolerances on the outputs and result in less safety margin and less yield in production.

If worst-case and RSS analyses do not meet the design specifications and requirements, then a systematic analysis is normally invoked. This analysis uses predictable characteristics, such as component or power supply tracking. Usually, a systematic analysis approach is a subset of worst-case or RSS.

In virtually all designs, a combination of these analysis techniques is used. Naturally, worst-case is attempted first because of its preferred features, followed by RSS, and then, as a last resort, by systematic.

A brief overview of typical test limit calculations in the factory, utilizing worst-case and statistical tolerance techniques, is discussed in the following paragraphs.

3.1.2.1 Worst-Case Analysis

If worst-case analyses are used to establish pass/fail test limits for a UUT, then it is possible to have a failed or defective component and still pass test. This is true because it is extremely unlikely that a worst-case limit can be experienced in normal situations. Although this may not be detrimental in the immediate test environment, it could render the UUT inoperative, or significantly degraded, in actual operation.

For worst-case analysis, the maximum (+) worst-case tolerance can be found by analyzing a circuit's transfer function and ascertaining which components in the numerator should be at their maximum values and which components in the numerator should be at their minimum values. A similar technique can be utilized to calculate the minimum (-) worst-case tolerance.

For analyses involving timing analysis, maximum and minimum worst-case tolerances are calculated utilizing summing techniques for each component in the subject timing chain.

3.1.2.2 Root Sum of Squares (RSS) Statistical Toleranoing

Although RSS analysis predicts a yield of 99.7% and a Gaussian output distribution, it is based on the condition that all contributing components possess Gaussian (normal) distributions over their complete specified range. As an example, a + 10%, 100 ohm resistor is expected to possess a distribution as shown in Figure 2. This is hardly ever the case in actual practice, however. Most often, the distributions are quasi-Gaussian and "skewed." What causes this condition is that the manufacturer of the components desires a high yield, and the mean of the distribution will vary from lot to lot. Consequently, "skewed" distributions are more realistic distributions to expect in a manufacturing environment.

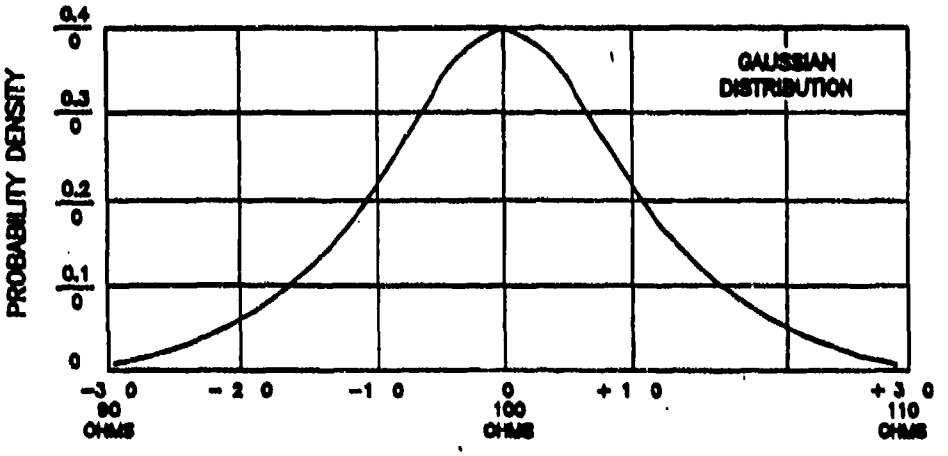


FIGURE 2. IDEAL DISTRIBUTION FOR A 100 ± 10 OHM RESISTOR

If "skewed" distributions are indeed realistic, how does the factory justify RSS analysis?

In the world of statistics, the Central Limit Theorem predicts that no matter what the individual distribution of the contributing components, the resulting distribution of the subject performance parameter approaches Gaussian characteristics as the number of samples becomes large. However, real-world acquisition scenarios sometimes play havoc with the Central Limit Theorem's generalization. Quite often, military systems are subject to short production runs and "on-again, off-again" procurements, which are spread out over many years and utilize many different second-source suppliers Unless qualified, screened components are utilized, it is possible that the performance characteristics of the resulting end item will deviate from the "nominal" performance characteristic specifications.

That is, the "real life" nominal value, which is manifested in the subject end item, will be "skewed" significantly to the right or left of the "normal" nominal value.

Where component tolerances are controlled and maintained, RSS techniques at the factory can be utilized to derive statistically sound limits for a UUT, when applied in concert with ATE instrumentation and switching characteristic data. For RSS analysis, the maximum output for a subject performance parameter can be found by calculating the change to the performance parameter (P_X) due to a change in each component value (C_X) of the circuit, squaring each term, and taking the square root of the resulting expression.

Mathematically speaking, RSS can be calculated as follows:

$$\Delta P_X = \sqrt{\left(\frac{\Delta P_X}{\Delta C_a}\right)^2 + \left(\frac{\Delta P_X}{\Delta C_b}\right)^2 + \cdots \left(\frac{\Delta P_X}{\Delta C_z}\right)^2}$$

There are other, less laborious methods, that can be used for RSS analysis, such as a method involving partial derivatives. This method is very easily performed with available computer-aided design programs, such as SPICE. Utilizing the circuit design data base, each component circuit value is edited to its worst-case maximum value, a circuit simulation run of the "adapted" circuit, and the resulting output changes recorded. This technique is done for each circuit component which has an affect on the subject output parameter in question. Then utilizing a "canned" software routine, each change is squared, summed with the other changes, and the square root calculated to yield the resulting RSS change.

When utilizing this procedure, all terms except the component in question assume their nominal value, while the one in question assumes its worst-case value. Nominal circuit parameter tolerances are derived for a UUT by running a good circuit simulation with all components set to their "nominal" values. The results of an RSS analysis yields a distribution that is Gaussian in form and predicts that 99.7% of the performance parameter values for a "good" UUT will fall within the Gaussian + 3 limits. A similar type of analysis can also be performed for the negative extreme.

3.1.3 Factory ATE

It is important to note that the factory test environment differs drastically from the field environment. Testing in the factory is primarily addressed from a "bottom up" point of view. The spectrum of factory ATE varies from: component test, bare board test, loaded board, LRU/assembly test, and in-circuit test to functional board test, "hot mockup" testing, or certification testing in the next higher assembly.

The factory test environment is exacting and comprehensive. Factory testing is orchestrated to fault detect and fault isolate a broad cross section of failures, such as shorts, opens, solder splashes, components out of tolerance, components inserted incorrectly, wrong value of components utilized, etc. Typically, once the subject end item is tested utilizing factory ATE, it is integrated into the

deliverable product (i. e., computer, radar, etc.) for final functional testing. This integration process assures that the whole is equal to the sum of the parts and that the system functions to performance specifications. If the system does not meet performance requirements, a combination of manual, semiautomatic, and automatic testing is utilized to ascertain which system component is the cause of system failure. The subject system component (i. e., SRU, assembly, LRU) is then sent back to the appropriate level of test in the factory to ascertain failure resolution within the subject end item.

If it is determined that the subject end item passes its ATE functional test and fails system integration testing, an analysis is typically performed to ascertain what changes are required to the factory ATE and/or TPS to eliminate, or minimize, the RTOK problem at the factory. Typically, primary "fixes" involve:

- Modification to test limits, to reflect more accurately performance limits required in the next higher assembly
- Interface device loads and circuitry, to "emulate" more accurately the next higher assembly in the ATE functional test test bed.

This data then is fed back to the manufacturing test group to ensure that the factory ATE/TPS are updated to ensure increased yield in the manufacturing test process. The repository for this information is embodied within factory ATE and TPS configuration base lines and the associated factory system acceptance test procedures. Typically, this data is not formally configuration managed, per MIL-STD-480 procedures, but maintained as engineering information released by test engineering to manufacturing test.

This data is typically not a contractual deliverable.

3.2 Factory and O-(BIT)/I-/D-Level Interface

Parametric testing, utilizing the ATE, is the prevalent method used to evaluate performance of electronic assemblies during their manufacture at the factory and during the operations phase of their lifetime at the Depot and Intermediate Levels of maintenance.

The most common method of parametric testing is to construct an emulation of the next higher assembly, consisting of the UUT, the ATE, and an Interconnection Device (ID). Emulation of prime system signals and basic measurement capability are provided by the ATE. The ID is generally passive, but occasionally is used to modify, or buffer, signal paths in some manner. Assuming that the test designer knows the level of performance which is required of the UUT to allow the prime to meet its performance requirements, a set of test limits is

per the discussion in the previous section. These limits must also take into consideration the uncertainties present in the stimulus/measurement process, as illustrated in Figure 3, if we are to guarantee correlation between the ATE test and operation in the prime system.

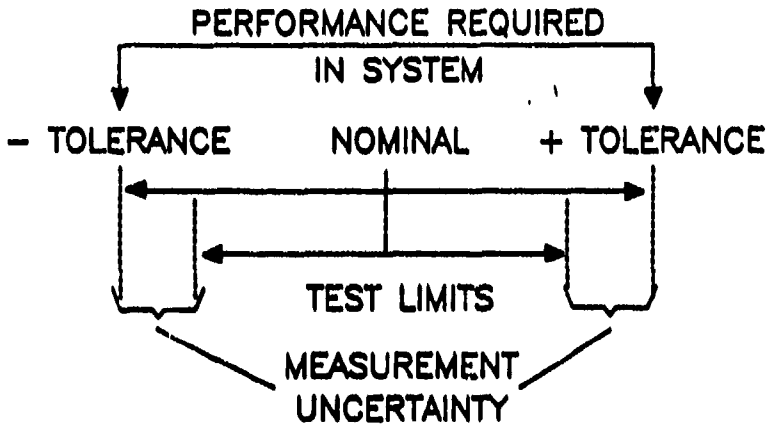


FIGURE 3. TEST LIMITS MUST BE SET TO COMPENSATE FOR MEASUREMENT UNCERTAINTIES

Tolerances for testing are commonly calculated as a Root Sum of Squares (RSS), when the error sources are statistically independent and normally distributed. For example:

$$T = [e_u^2 + e_m^2 + e_s^2]^{1/2}$$

where

e_u = UUT output tolerance as calculated utilizing sensitivity analysis, modulated by real-life experience, and as documented in the appropriate UUT performance specification

e_m = Test system error, including measurement error and the effect of test system noise sources

e_s = Error reflected at UUT output due to stimulus uncertainty

The term e_s can be reduced by programming techniques which eliminate the long-term effects of component aging, thermal drift, and other factors which remain constant, or essentially constant, over the relatively short time that it takes to execute a test on the support system. Similarly, the measurement error e_m can be reduced. If the test system includes a calibration reference for the measurement instrument, long-term drift in measurement accuracy can be eliminated by appropriate calculations at the time of test execution. Transmission line effects can also be reduced, by taking into account the insertion of losses and impedance mismatches or reflection coefficients along the transmission line between the UUT and ATE stimulus and measurement devices.

These stimulus and measurement parameters, for the most part, are not known until the Depot Level ATE to be used has been defined. This definition does not always occur in the proper time frame. In addition, the parameters for the factory ATE are not always characterized and documented for depot and field uses. The result is often evident in the form of nonoptimized test limit assignments and as RTOK problems between the factory and the field.

A potential problem exists of nominal value skew when circuit component tolerances are not controlled in the factory design environment. This anomalous situation can potentially cause RTOK problems in the factory for end items which have tested "bad" in the field due to a combination of factors: "nominal circuit tolerance skew" and the "measurement uncertainty" of Depot Level ATE (see Figure 4). A similar condition can also exist between the Depot- and Intermediate-Levels of Test.

Thus appropriate attention to detail with respect to the setting of circuit tolerances and the control of these tolerances, via the acquisition and spare parts procurement process, is essential, in order to assure that a "nominal value skewing" problem does not manifest itself as a RTOK problem in the UUT maintenance chain.

If all maintenance activities associated with the UUT are included, a conic test tolerance envelope results, as previously depicted in Figure 1. The base dimension of the cone represents the level of performance required in the system in its operational environment. This tolerance value is typically the accuracy required of Organizational Level BIT stimulus/measurement devices to certify UUTs Ready for Issue (RFI) at the platform level. The slope of the cone is a composite of the test uncertainties at all levels of test. The height of the cone is defined by the number of different maintenance levels at which that parameter is tested. Note that these factors act to define the level of accuracy required of the ATE at each level of maintenance.

"Tolerance cone budgeting" is an accepted method of distributing test limits across the various uncertainty factors associated with the manufacture and maintenance of electronic assemblies. The intent is to define test limits at each test level, so that adequate margins for test equipment performance and prime product performance are allowed. Properly applied, tolerance cone budgeting results in good correlation between test results at each maintenance level and, therefore, minimizes nonverified failures with their attendant cost and lost time implications.

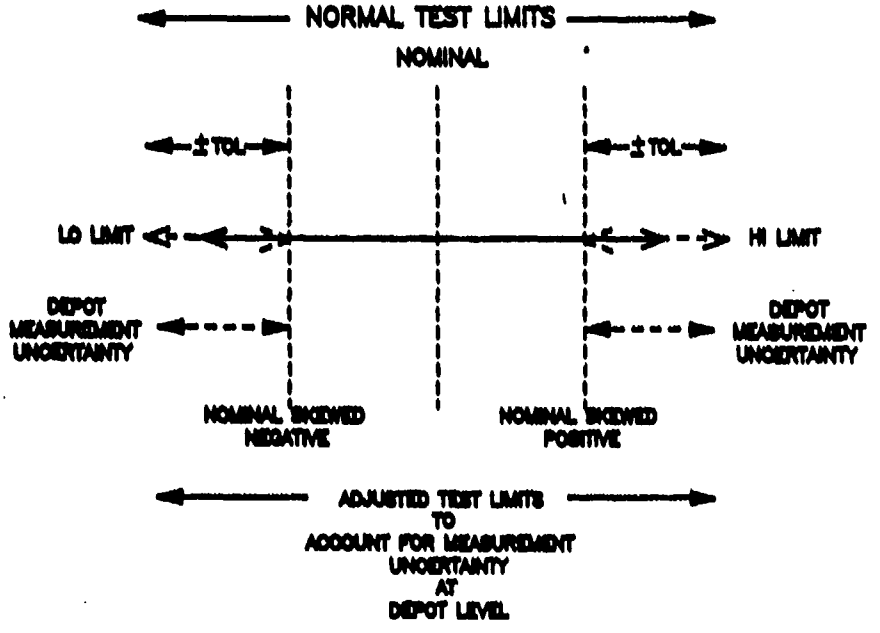


FIGURE 4. EFFECT OF SKEWED NOMINAL TEST LIMITS AT THE FACTORY AND MEASUREMENT UNCERTAINTY OF DEPOT ATE — RTOK CONDITION AT FACTORY

Failure to perform a rigorous analysis of the affects of test tolerance build-up often results in test limits being arbitrarily assigned on the basis of ATE/diagnostic element capabilities, rather than on the basis of correlation at the next level of test. This will invariably result in one of three problems:

1. A failure of the prime system to perform to specification, even when all individual devices test "good"

2. An excessive number of devices in the pipeline, due to UUT which test "good" at one level, but "bad" at the next maintenance level (RTOK)

3. Unnecessary repair actions caused by test limits which are excessively stringent.

Occurrence of any one of the above situations will result in excessive maintenance activity and probably will impact prime system availability or performance, as well.

If tolerance cone budgeting is to succeed as a testing strategy, a data base of controlled parameters must be defined and characterized at each level of maintenance for use at other maintenance levels in setting their test limits.

3.3 Generic Data Base Requirements

Based on the discussions in the previous sections, tabulated below are generic data base requirements for diagnostic ATE, which must be data based to resolve vertical test and RTOK problems between the Factory and Intermediate/Depot Levels of test and which, when controlled, ensure traceability of test requirements from the factory:

- Component tolerances (%) for all components which contribute to or affect the subject performance parameter tolerance
- ATE stimulus stability and accuracy for each performance parameter test
- ATE response stability and accuracy for each performance parameter test
- Stimulus switching uncertainty (path resistance and distributed shunt capacitance) for each performance parameter test
- Test accuracy ratio achieved for each performance test, based on the factors identified.

As a rule of thumb, a test accuracy ratio of greater or equal to three-to-one is typically maintained between the test measurement accuracy and the test requirement to ensure test measurement integrity.

3.4 Vertical Testability Method Alternatives

Besides tolerance cone budgeting and the maintenance and traceability of controlled parameters, other criteria and methods may be employed to ensure vertical testability across maintenance levels. Two such methods which have, or are starting to have, wide-spread application are:

- Employment of common ATE between maintenance levels

- o ATE emulation.

A brief discussion of each of these vertical test methods is given in the following paragraphs.

3.4.1 Common ATE

One approach to reducing the tolerance cone "build-up" problem is the utilization of common functional ATE and test program sets in both the factory and the field (vertical commonality). The approach is predicated on the assumption that the same functional ATE that is used in tandem with other factory ATE (i. e., in-circuit, component, etc.) can also be utilized to support the Depot and Intermediate Level testing functions. This testing concept reduces the "uncertainty" of test to include only the calibration variation of the ATE, thus minimizing the need to budget test equipment performance at each maintenance level--in effect, "shrinking the tolerance cone."

This approach is not without its drawbacks, however. The manufacture test, field test, and depot repair environments are considerably different, and an efficient ATE system for one may not prove to be efficient for others, unless properly managed and/or adapted. Some differences between these two environments are:

- o Manufacturing final acceptance tests using ATE are typically designed to test one type of product, to test it rapidly, and to diagnose to a parameter level only. Fault isolation and repair are typically done elsewhere.

- o Field "black box" level tests using ATE are typically required to support a quantity of different types of UUT which arrive at irregular intervals. Fault isolation to the replaceable assembly level and repair verification/alignment are performed on the ATE by relatively low-skill-level operators. Support of the ATE itself is limited by local shop resources and training.

- o Depot ATE is required to handle many different types of assemblies arriving in small quantities. Repair is to the defective component level. Repaired UUT must be verified to a high degree of confidence, since the prime system is not available to validate the repair action. Normally, the repaired UUT becomes a spare.

These differences, coupled with the thrust to perform more maintenance using ATE, make it important to develop ATE hardware and software which is appropriate for use at all levels of maintenance. Merely specifying "modular"

systems is not adequate. Thought must be given to how the equipment is to be used and by whom. Only then will benefits accrue from modular systems architecture.

While factory and maintenance support have different operating environments, they have many common requirements and objectives. Final acceptance test in the factory is an equivalent requirement to operational performance verification in the field maintenance support environment. Manufacturing test at the factory must include capability for fault detection and fault isolation, which is the primary objective of field testing.

It is reasonable, therefore, to consider common test capabilities (i. e., vertical commonality) to support both factory and maintenance support requirements at assembly and subassembly levels.

The concept of vertical commonality is (the implementation of) common requirements in the factory, field, and depot predicated on a common automatic test system architecture, consisting of common bus structures, stimulus generators, response monitors, switching devices, interface devices, software operating system (SOS), and test program sets (TPS).

The primary advantage of this approach is the inherent built-in vertical and horizontal compatibility. The use of the same instruments to perform the tests, the same software to control the tests, the same interface pins, the same bus structures to tie the modular components together, the same mechanical fixtures to implement the tests, and the same criteria to evaluate the test at the Factory, Field, and Depot Levels eliminates many of the unnecessary retests and ambiguous determinations which impede throughput in the factory and maintenance support facilities.

Common test systems for Factory, Field, and Depot Levels of maintenance provide many benefits for both the supplier and the customer. The overall program cost is reduced by one-time development of a TPS to serve both requirements. Common tests at each level produce consistent results by reducing unnecessary testing (i. e., RTOK) of operational units. Use of common test systems in the factory provides the supplier with a local proving ground for TPS effectiveness, resulting in a proven TPS which can be delivered to the customer in time to support initial system deliveries and negating the requirement for contractor maintenance. Another inherent advantage is configuration management, based on the fact that units delivered from the factory must be tested on the same equipment and with the same test programs used for maintenance support. Changes in the prime equipment, therefore, must have corresponding changes incorporated into the test program sets before the units can be delivered, thus making test program set updates concurrent with prime equipment configuration changes.

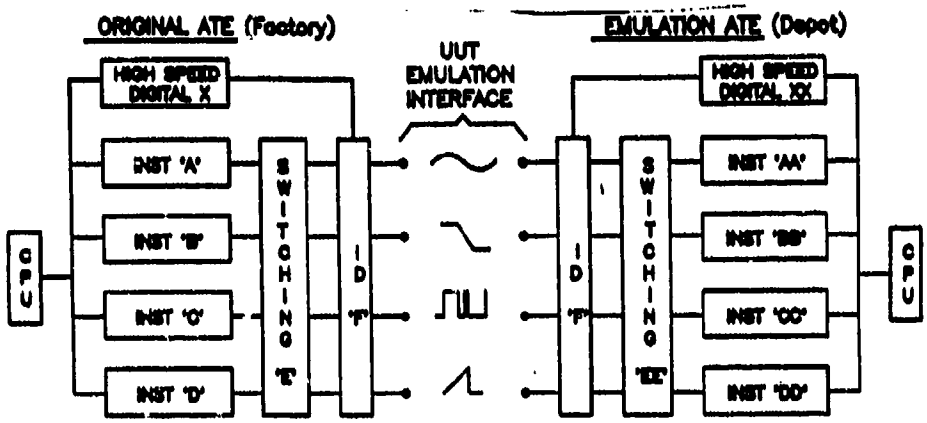
The net result is a product delivered with a proven support capability, available concurrently with the program needs, and maintained current with configuration changes by factory acceptance test requirements. The customer is benefited by a better supported product, while the supplier gains a reputation for product support, which enhances future business opportunities. Vertical commonality provides the capability to test during design to prove that the product works; to test during qualification testing, to prove that it works under all conditions; to test during production, to prove that it works with all combinations; and to test during support, to prove that the product works throughout its entire service life.

3.4.2 ATE Emulation

Another approach to vertical testability has been to adopt the approach of "ATE emulation." Emulation is an approach by which the I/O interface of an ATE system is functionally replicated, on a test-by-test basis, by a substitute system which utilizes a combination of hardware and software. This approach is typically utilized when the sunk cost investment in test program set software is extensive on an obsolete, or soon to be obsolete, piece of equipment and new, replacement ATE is to be procured, or when a Depot- or Intermediate-Level ATE TPS is developed, utilizing factory "source" programs.

Figure 5 depicts the classical approach to achieving ATE emulation, utilizing the concept of hardware reconfiguration. Vertical transportability between factory ATE and depot ATE is dependent on a number of factors, some of which are:

- Performance capability of "functionally equivalent" instrumentation (i. e., ranges, accuracy, stability, granularity, etc.)
- Performance compatibility of switching systems (i. e., path resistance, shunt capacitance, off-set voltage, noise, etc.)
- Calibration accuracy differences between instruments.



- REPLACEMENT ATE EQUALS ORIGINAL ATE?
- TPS TRANSPORTABILITY ONLY AS GOOD AS HARDWARE/SOFTWARE SELECTION AND "TUNING" WILL ALLOW.
- HARDWARE RECONFIGURATION CONCEPT REQUIRES RECURRING SOLUTION TO ATE REPLACEMENT PROBLEM ON AN APPLICATION BY APPLICATION BASIS.

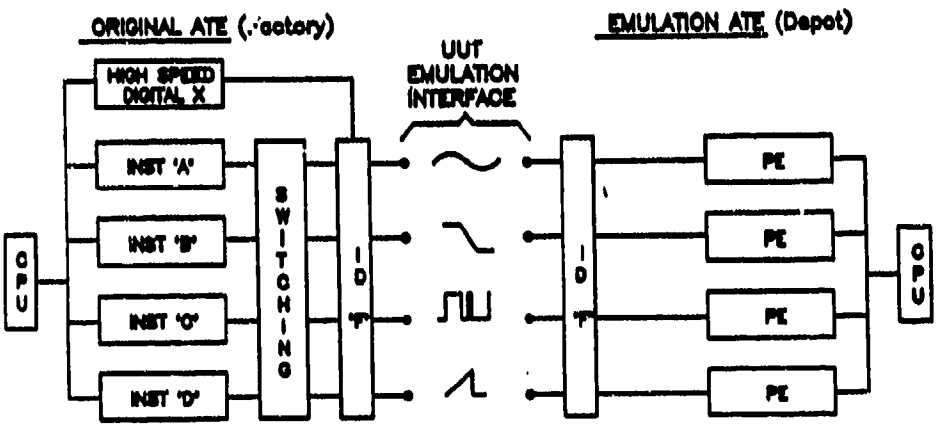
FIGURE 5. ATE EMULATION — THE CLASSICAL APPROACH — HARDWARE RECONFIGURATION

Thus the classical ATE emulation concept minimizes, but does not totally eliminate, "measurement uncertainty" between the factory ATE and the emulation ATE.

An alternative approach to hardware reconfiguration is software reconfiguration, utilizing concepts such as pin electronics (PE) as the emulation vehicle (see Figure 6). Pin electronics employs high-speed D/A and A/D technology behind each UUT I/O pin. No electromechanical switching is employed. Each PE channel is a "virtual instrument" in disguise which, via software control, can emulate, up to approximately 25 MHz, any electrical interface signal and propagation delay. Any ATE anomalies/characteristics of the factory ATE can be emulated by the PE ATE. This emulation approach can be utilized over a broad spectrum of applications. The ramifications, from a vertical testing compatibility point of view, are encouraging:

- o Because of the employment of high-speed/highly granular D/A and A/D technology, measurement and stimulus uncertainty is minimized
- o The absence of a switching function from the PE system architecture eliminates stimulus/measurement switching uncertainty from testing tolerance limit calculation

- Calibration accuracy differences in the limit become the primary drivers of measurement uncertainty between the original ATE (factory) and emulation ATE (depot).
- Emulation ATE (EATE) provides same input/output functional capability as the old, or original ATE being replaced by EATE
- TPS transportability only as good as hardware/software selection and "tuning" will allow
- Hardware reconfiguration concept requires recurring solution to ATE replacement problem on an application-by-application basis



- PE BASED REPLACEMENT ATE IS SOFTWARE RECONFIGURABLE & CONTROLLABLE
- ORIGINAL ATE "ANOMALIES" CAN BE CLEANLY EMULATED VIA SOFTWARE
- GENERIC EMULATION APPROACH CAN BE UTILIZED OVER A SPECTRUM OF APPLICATIONS

FIGURE 6. ATE EMULATION — PE APPROACH — SOFTWARE RECONFIGURATION

3.5 Criteria

Criteria to select potential solutions to resolve vertical test problems must be formulated. At the present time, primarily three potential solutions are available to resolve vertical test problems. As other solutions are identified, criteria for the selection of each proposed solution must be formulated.

It should be noted that a combination of these diagnostic methods may be used to effect vertical test solutions. For example, common ATE may be utilized at the Factory, Depot, and Intermediate Levels of maintenance and tolerance cone budgeting may be utilized to derive test tolerances at the Organizational or Platform Levels of maintenance. Criteria for the selection of each of the identified diagnostic solutions is provided in the paragraphs that follow.

3.5.1 Tolerance Cone Budgeting (TCB)

Tolerance Cone Budgeting should be utilized when life cycle cost studies and/or program constraints dictate the use of existing test vehicles in the factory and in the field. In this severe type of environment, tolerance cone budgeting is the only vertical test method (VTM) alternative available to the support system manager.

3.5.2 Modular Emulation ATE

Modular Emulation ATE should be utilized when:

1. Old, outmoded ATE is to be replaced in the field and sunk cost investment in test software is to be preserved, and

2. Existing factory ATE is not cost effective or technically feasible to upgrade. Employing common ATE in both the factory and the field is not a feasible alternative.

It should be noted that the utilization of modular emulation ATE implies the preservation of tolerance cone budgeting and associated test limits employed within existing test program sets. In essence, the modular emulation ATE replacement concept goal is to "preserve," within the maintenance hierarchy, the tolerance cone budgeting contribution of the "replaced" ATE.

3.5.3 Common ATE

Common ATE is the preferred VTM approach and should be utilized as a VTM for the following scenarios:

1. New program starts. Situations in which test equipment and TPs have yet to be specified in an RFP. Promulgation of the concept of common modular ATE in both factory and field should be the VTM strategy employed in this situation.

2. Situations in which critical UUT performance parameters have small, but measurable, day-to-day drift variation (i. e., "nominal value drift"),

intrinsic within themselves. Use of common ATE will minimize measurement uncertainty in these particular situations.

3. Situations in which critical UUT performance parameters are "pushing" the state-of-the-art instrumentation (I. e., TAR < 3:1).

In these types of situations, tolerance cone budgeting becomes impractical. These situations will cause failure of the prime systems to perform to specification, even when all critical performance parameters test "good."

Use of common ATE will minimize measurement uncertainty in these particular situations by virtue of the fact that the same critical parameters are measured at other levels of maintenance with the same common ATE and TPS.

4.0 DESIGN PROCEDURES AND DOCUMENTATION

Traceability of the testing functions and parameters throughout all levels of maintenance is important to both the Government Program Office and the contractor. To achieve satisfactory traceability for both government and industry use, documentation of the testing functions and parameters, together with the environment in which these are measured, is required. This documentation must take two forms. The first type deals with documenting the approach utilized by the contractor to establish testing tolerances. MIL-STD-2165 is the governing standard for documenting the method used. Task 202.2.1 of this standard establishes the approach used to achieve vertical testability. This should be accomplished during Dem/Val. Task 203.2.1 of MIL-STD-2165 documents the procedures utilized to achieve vertical testability. This is accomplished during Full-Scale Development.

Documenting the results of the vertical testability analyses is accomplished by invoking MIL-STD-1519 or MIL-STD-1345, Test Requirements Documents. Caution must be used in implementing these standards. Historically, the test requirements documentation was often prepared after-the-fact and served no useful purpose, except to document that these parameters, tolerances, etc., had been established. However, this is not the main purpose of these standards. Rather, the documentation required must be utilized to define the testing requirements at all levels of maintenance (Organizational, Intermediate, and Depot) and for all types of testing (BIT, ATE, etc.). Thus this documentation plays a key part in the diagnostic design process, which includes vertical testability analyses. When requiring Data Item Deliverables under MIL-STD-2165, MIL-STD-1345, and MIL-STD-1419, the CDRL must reflect this need. The timing of generating these test requirements is not only essential in the design of the testing function, but must be a major input in the development of the technical information which supports the testing function (e. g. technical publications). Thus the preparation of a technical publication or software, in the case of electronic delivery of this technical

in the case of electronic delivery of this technical information, must depend on the timely preparation of the Technical Requirements Document.

Figure 7 depicts vertical testability design procedures and the associated documentation. The vertical testability activities are tied to the diagnostic activities found in the the Diagnostic Activity Roadmap contained near the beginning of this guide. Documentation of both the approach to vertical testability analysis and the performance of this analysis are documented in the Testability Analysis Report (DI-T-7199), which is an overall testability report required during Dem/Val and FSD. Care must be taken by both the Government Program Office and the contractor to assure that vertical testability is addressed in these reports. If additional instruction on the Data Item Deliverables is required, it can be accomplished through the use of the CDRL, which can tailor the DID.

The documentation of the vertical testability analysis is contained in the Test Requirements Documents. MIL-STD-1519 or MIL-STD-1345, Test Requirements Documents, are the governing standards. Either standard can be applied. The testing parameters and the test environment information inputs to the test programs which are generated for BIT, ATE, etc., and the test procedures can be utilized as a direct input in the preparation of technical publications or other electronic delivery methods.

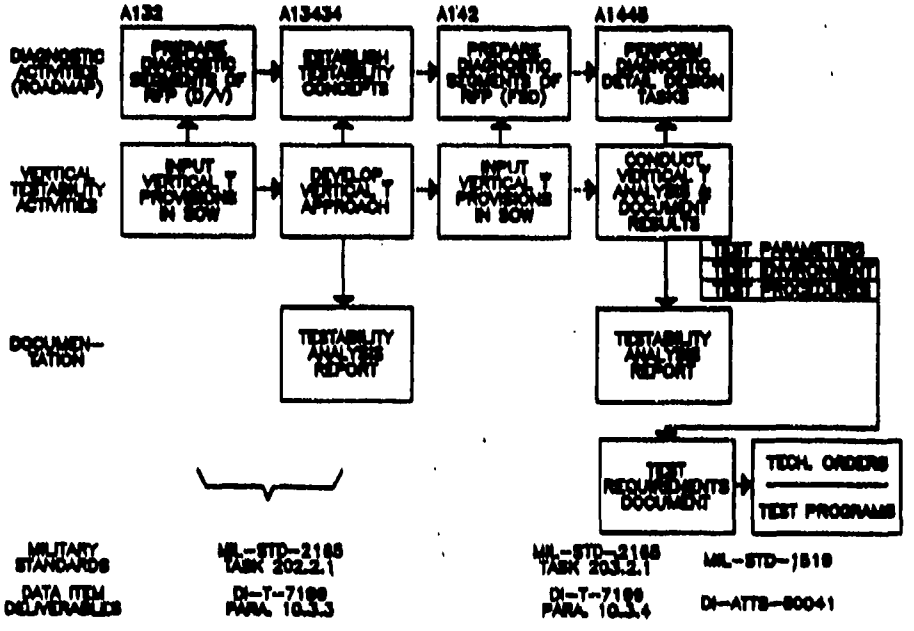


FIGURE 7. VERTICAL TESTABILITY DESIGN PROCEDURES & DOCUMENTATION

Table of Contents

1.0	OVERVIEW	3
2.0	DESIGN LEVELS	4
3.0	GENERAL CONSIDERATIONS	5
3.1	On-Line/Off-Line Testing	5
3.2	Digital Guidelines	6
3.3	Analog Guidelines	9
3.4	LSI/VLSI/Microprocessor Guidelines	10
3.5	Test and Maintenance (TM) Bus	11
3.6	Applicability To Design Levels	12
4.0	STANDARD TESTABILITY APPROACHES	12
4.1	Scan Techniques	12
4.1.1	Scan Path	12
4.1.2	Level-Sensitive Scan Design (LSSD)	18
4.1.3	Scan-Set	19
4.1.4	Random Access Scan	21
4.1.5	Scan and the TM Bus	25
4.1.6	Boundary Scan	25
4.1.7	Applicability to Design Levels	25
4.2	Signature Analysis	26
4.2.1	Stimulus Generation	27
4.2.2	Signature Development	32
4.2.3	Signature Analysis and the TM Bus	35
4.2.4	Applicability to Design Levels	37
4.3	Wraparound Techniques	38
4.3.1	General	38
4.3.2	Core, ROM, RAM Testing	38
4.3.3	Processor Controlled Gating	40
4.3.4	Applicability to Design Levels	41
4.4	Analog Techniques	41
4.4.1	General	41
4.4.2	Active Versus Passive	42
4.4.3	Conversion to Digital Format	42
4.4.4	Applicability to Design Levels	43
4.5	Concurrent Techniques	43
4.5.1	General	43

4.5.2	Fault-Tolerant Design	46
4.5.2.1	General	46
4.5.2.2	Use of the TM Bus with Fault-Tolerant Circuits	50
4.5.3	Applicability to Design Levels	50
5.0	FAULT-ISOLATION TECHNIQUES	51
5.1	Test Points	52
5.2	Test Header	66
5.3	Increasing I/O Visibility	67
5.4	Fault Signature/Fault Dictionary	69
5.5	Guided Probe/Clip	69
5.6	Use of TM Bus	74
6.0	PHYSICAL PACKAGING	74
6.1	Surface Mounted Devices	74
6.2	Chip Carriers	75
6.3	Small Outline (SO) Packages	77
6.4	Pin Grid Arrays	77
6.5	Packageless Configurations	80
6.6	Testing Surface Mounted Devices	80
7.0	TEST AND MAINTENANCE BUS	81
7.1	Overview	81
7.2	VHSIC TM BUS	81
7.2.1	Physical Requirements	82
7.2.2	Electrical Requirements	83
7.2.3	Data Link Requirements	83
7.2.4	Element TM Bus	84
7.3	Other Initiatives	84

TESTABILITY/DIAGNOSTIC DESIGN TECHNIQUES

1.0 OVERVIEW

Weapon systems are becoming increasingly complex and difficult to support. To a large extent, their capability to be available to successfully complete a mission is dependent on the testability and diagnostic characteristics designed into their associated support systems. These support systems consist of both embedded (i.e., on-line BIT) and off-line (i.e., external test equipment/ATE) test capabilities.

Innovative new approaches in test system architecture, such as the virtual instrument concept and expanded pin electronics techniques combining analog and digital test capability on each pin, are coming together to enhance system readiness. These advances will yield the highest probability of mission success only when a design for testability philosophy is incorporated in the earliest stages of system acquisition.

Life cycle cost trade-offs between the varied system elements automatically preclude the establishment of a set of mandatory testability features for all procurement evaluations. The techniques and applications identified herein provide a comprehensive outline of those processes that are real, available, and acceptable in accordance with established specifications and system goals. They provide a base from which greater systems operational availability will evolve as technology also evolves.

The sections to follow will give detailed information on testability and diagnostic design techniques. Section 2 is devoted to the various levels of design according to the hierarchy of Integrated Diagnostics. The maintenance spectrum must deal with testing at the system level, testing of black boxes or subsystems that are parts of that system, and testing of boards or modules which are part of the subsystem or black box. The components on those boards must then also be dealt with as part of the overall testing spectrum.

Section 3 deals with general considerations for the designer. Included is on-line and off-line testing, digital guidelines, analog guidelines, guidelines for LSI and VLSI, guidelines for microprocessors, and a section on the use of a test and maintenance (TM) bus.

Section 4 deals with standard testability approaches including scan techniques. Covered in this general category of scan techniques are scan path, level-sensitive scan design, scan set, random-access scan, and scan as it applies to the test and maintenance bus. Also included is the new technique of boundary scan. In addition, signature analysis is covered in detail. Included are stimulus

generation, the development of signatures, signature analysis and its applicability to the TM bus, and the various design levels in which signature analysis can be used. Also covered is the standard microprocessor wrap-around technique including the defining of core operability, the testing of Read Only Memories (ROM), the testing of Random-Access Memories (RAM), processor controlled gating, the generation and the capture of the test vectors, and the applicability to various design levels. Also discussed are various analog techniques including active versus passive testing, analog stimulus generation and measurement, conversion of analog signals to the digital format, and, finally, applicability to various design levels. Concurrent test techniques are covered in Section 4, including concurrent versus non-real time, the new pin electronics architecture, fault-tolerant design, and the applicability to various design levels.

Section 5 deals with fault isolation techniques including the use of test points, the use of a test header to increase I/O visibility, fault signature/fault dictionary as a fault isolation technique, and the use of a guided probe or clip for fault isolation. Also covered is the use of the TM bus.

Section 6 is devoted to physical packaging including surface mount devices, chip carriers, small outline packages, pin grid arrays, packageless configurations, and the testing of surface mount devices.

Section 7 covers the use of a test and maintenance bus. Included are the VHSIC TM Bus, the element test and measurement bus, plus current efforts on the part of the JTAG Committee and the IEEE.

2.0 DESIGN LEVELS

Integrated diagnostics is an all-encompassing discipline that is meant to deal with diagnostics at all levels in a system. For many years the maintenance spectrum in an ideal situation operated as follows: the prime system (e.g. an aircraft), would have sufficient built-in test in order to ascertain whether or not the system was working correctly or, if it was not working correctly, to be able to fault isolate to a malfunctioning black box onboard. The suspect black box would be removed from that operational scenario and returned to an intermediate-level shop for testing. A test system in the I-level shop would check the suspect black box to ascertain whether it was operating correctly or if it did indeed have faults. If a fault or faults were uncovered, the procedure would be to isolate to one or more malfunctioning printed circuit boards or modules. The faulty board(s) or module(s) would be replaced, the black box retested and, if operational, returned to the spares inventory. The faulty boards or modules would be returned to a depot for testing on a different piece of test equipment which would be used to determine whether or not they were operational or whether they were malfunctioning. If they were faulty, the approach would be to isolate to a malfunctioning component or components. After replacement of these faulty component(s), the now operational board would be

returned to this spares inventory. This overall approach required at least three levels of maintenance.

An attempt is now underway to create a two-level maintenance scenario. This newer approach is an attempt to eliminate the I-level shop. The built-in test on board the operational system is intended to fault isolate without ambiguity to a malfunctioning line replaceable module. That suspect module would then be returned to the depot for repair. This procedure would eliminate the I-level shop.

Regardless of the maintenance scenario, when diagnostics are built into a system they can be built in at the chip level, the board or module level, the black box or subsystem level and the system level. Since integrated diagnostics is an all encompassing discipline, every attempt will be made in the following sections to point out at what level or levels the various techniques or approaches are applicable. For example, when scan techniques are discussed in Section 4.1, every effort will be made to recommend at what level of the design process scan techniques can be utilized effectively.

3.0 GENERAL CONSIDERATIONS

3.1 On-Line/Off-Line Testing

The development of an effective system support strategy that capitalizes on the current trends in enhanced alternate mission identification capability and fault-tolerant design is dependent on unambiguous error detection. Current weapons system operational software can be relied upon in most cases to fault isolate the system to a defective black box. In a properly designed system, utilizing a TM bus and supporting BIT/BITE circuitry, maintenance diagnostic software will be able to fault isolate to a defective card or module. The percentage of fault coverage is a function of the design for testability investment in cost, circuit real estate, program requirements and technology capture.

The increasing population density of VLSI devices is skewing circuit BIT design momentum toward a distributed BIT architecture where critical circuit monitoring components are included on each card. This can be particularly valuable when a dedicated maintenance diagnostic bus allows active injection of stimuli, derived from the BIT itself, to alter the circuit state. Active BIT characteristics can also enhance the capability of defining system operational capacity and identifying degraded mode mission performance.

Many of the design for testability enhancements that are required to yield a significant off-line test capability also affect on-line testability. Off-line electronic module and board screening and defective component identification will normally be accomplished through the use of ATE. For the purposes of this discussion it is assumed that the regiment will be implemented in hardware, software and firmware.

The allocation of test function responsibility between these three elements is dictated by the system design goals and technology level. Following segments will address the off-line testing of boards with:

- o BIT/BITE
- o Embedded Scan Design or Signature Analysis
- o Fault-Tolerant Design
- o Some portion of Analog Circuitry.

The architecture of the supporting ATE has, in the past, been determined to a large extent by the system to be tested. Now the designer of a circuit who is aware of the newest and most powerful test techniques can alter the testability profile significantly at an early stage in the acquisition process. For this reason, the use of regular input/output pins and incorporation of a test header becomes an inexpensive adjunct to facilitating testability when the ATE has the capability of controlling the state of each pin. This enhanced tester capability supports the off-line testing philosophy.

Other considerations in maximizing fault isolation efficiency during the off-line testing of circuit boards are greatly impacted by the physical layout of the board. The effects of pin positioning and surface mount device conformation forces alternate probing techniques. In many cases, unless test accessibility is designed into the board (test header/test points), the board may not be economically tested at the intermediate level and will be coded for depot repair or scrap. The ability of the circuit card designer to capitalize on the advantage of the newer packaging techniques will be enhanced if consideration is given to including a built-in access route to the critical nodal points.

3.2 Digital Guidelines

All circuit design must support the overall maintenance concept. It is of particular importance that the maintenance concept and its implementation approach be finalized before any design efforts begin. Table 1, Diagnostic Support Activities, provides a listing of the seven specific actions that need to be accomplished to provide adequate condition monitoring and evaluation of a circuit's performance capability. The acquisition program goals of operational availability and life cycle cost will determine the actual depth of application of each of these activities. For example, off-line fault isolation capability will be determined by the size of the ambiguity group for failed components, the ATE characteristics required, the extent of manual intervention and the system software characteristics.

TABLE 1 DIAGNOSTIC SUPPORT ACTIVITIES

Activity	Result
Performance Monitoring	Continuous system status
Confidence self test	Test on demand, report status
Diagnostic self test	Test on demand, identify failed item
On-line screening test	Accepts signal manipulation to determine system status
On-line diagnostic test	Accepts signal manipulation to determine faulty module
Off-line screening test	Test sequence that identifies maximum number of failures
Off-line fault isolation	Test sequence that identifies component failure

The design techniques to be used that will provide the desired level of testability are techniques that support one or more of the following strategies:

- o Making circuits initializable
- o Providing measurement test points
- o Providing stimulus test points
- o Partitioning the circuit for testability
- o Providing test data collection and distribution circuitry
- o Embedding self-test in the circuit using microprocessors
- o Monitoring performance of the circuit.

Specific circuit arrangement is required to accomplish each of the items cited above. For instance, in order to evaluate a circuit's condition, it must be in a specific state (usually initialized). For example, in the case of a memory circuit, an input identified as clear (CLR) is brought to some edge connector and, on demand, a pre-determined pin state is established.

The following items provide a comprehensive list of general digital module testability guidelines:

- o Circuitry shall be initializable to a well-defined state to commence test/pattern development.

- Clocks and data should be independent.

- All memory elements must be able to be switched to both logic states (i.e., logic "0/1") and the output states for a given set of specified conditions must be predictable. In some cases, direct data input (i.e., pre-set input) to the memory circuit must be provided to effect efficient loading of memory elements with initialization of test data.

- Test coverage loss in a counter is directly proportional to the degree of the constraints that are imposed. Testability can be at least partially restored in these cases by insuring that the high-order bit of the counter be an observable output.

- A mode control should not be removed from the counter or shift register.

- The Load or Clock lines of a counter should not be driven from the memory outputs of the same counter.

- All ROM and RAM outputs must be observable at the module I/O connector. The Chip Select line of all ROMS and RAMS must not be fixed at the logic polarity which allows active operation. RAMS shall allow sufficient control by the tester to perform standard memory tests such as galloping patterns.

- A Single Shot can be used to drive the Clock line of a memory block without loss of testability. If the Single Shot drives combinational circuitry, there can be significant loss of testability.

- Long strings of sequential logic should be broken and reconnected via gate control.

- Large feedback loops should be broken and reconnected via gate control.

- Multiple reset lines should be provided instead of one common reset line for a large number of memory blocks.

- All parity checkers and generators must be switchable to both output logic states.

- All analog signals and grounds must be separated from the digital logic.

- Any device that does not have a predictable output must be separated from all digital lines.
- Wired-OR signals that originate from five or more different physical locations must be separated into smaller groups.
- The number of module designs and IC types should be minimized.
- The module characteristics (function, pin count, clock rate, etc.) shall be compatible with planned ATE resources.
- Error correction functions must have the ability to be disabled so that the primary circuit can be tested independently for faults.

3.3 Analog Guidelines

Analog module testability guidelines include the following items:

- Input and output pins should be physically separated.
- All outputs exceeding one amp should have multiple output pins if voltage level is critical. This allows a Kelvin-type connection of the analog outputs and enables voltage sensing and feedback to the current control circuitry in the UUT. Consequently, Kelvin connection capability permits a prescribed voltage to be maintained at the UUT output terminals.
- Intermediate stages of a circuit should be independently testable by breaking signals through the I/O connector.
- The output of all stages of an analog circuit should be available, through isolation resistors, to a module pin.
- Modules with complex feedback circuits should have the capability to disconnect the feedback to allow independent test of the feedback and/or devices.
- All internally generated reference voltage levels should be brought out to module pins.
- All digital control functions should be independently testable.

3.4 LSI/VLSI/Microprocessor Guidelines

Digital LSI/VLSI/microprocessor module testability improvement is accomplished by applying the concepts listed below:

- Direct parallel access to LSI/VLSI/microprocessor devices shall be provided to the maximum extent possible. Support circuits driving the inputs of the LSI/VLSI/microprocessor should be tri-statable, allowing the tester to drive the inputs directly.

- Provisions shall be made for allowing tester control of tri-state enable lines and the outputs of tri-state devices.

- If bidirectional bus drivers are used in a microprocessor module design, they shall be located between the processor/controller and any of its support chips. The controls for bidirectional buffers on microprocessor I/O pins should be easily controllable, and preferably automatically controlled by the microprocessor without the tedious task of deciphering whether the microprocessor pins are inputs or outputs for each pattern applied.

- Signal breaks should be used to provide access to various data buses and control lines. If, due to I/O pin limitations, this cannot be done then scan in/scan out and multiplexing circuitry should be considered.

- Select components with known properties (internal structure, device function, failure modes, controllability/observability, etc.) and preferably with functional patterns already in existence (MIL-M-38510 or other high-quality test patterns).

- Make buses available to the tester. The data bus is the highest priority. Tester control of a bus is the most desirable feature, although monitoring capability alone will help fault resolution.

- A microprocessor on a module with other complex logic devices should not be ignored as a test resource. Once recognized, it is essential that the necessary features be incorporated in the design to use this resource.

- Provide ATE control of clocks through inhibit/over-ride techniques or by direct independent pinout.

- Provide for "single-stepping" of dynamic microprocessors/devices, if possible.

- o Partitioning can be enhanced by the use of tri-state buses, thereby reducing module testing to a series of device functional block tests.

- o Tri-state devices should utilize pull-up resistors to control the float level. This helps simulators avoid the introduction of unknown states into the circuit during automatic test vector generation.

- o Free-running clocks and power-up-reset functions should not be directly connected to the LSI/VLSI/microprocessor in a manner such that they cannot be disabled and tested independently.

- o Controllability and observability of all BITE designed into LSI, VLSI/hybrid, or microprocessor devices shall be provided to the module I/O connector.

3.5 Test and Maintenance (TM) Bus

The use of a standard bus for testability provides a most significant opportunity to enhance the effectiveness of module and system testing. The incorporation of a TM bus need not be overly restrictive on its users since it need not dictate the form of testability implementation on the unit under test (UUT) or the actual codes that transverse the bus. More importantly, it gives the government the mechanism to require each user to build testability into his designs and gives the means to negotiate specific testability requirements with each equipment supplier.

The functions which a standard TM bus should implement include the following:

- o Input pattern information to UUT
- o Output pattern information from UUT
- o Clock Signal
- o Addressing
- o Enable/disable control
- o Interrupt
- o Reset.

Implementation of these functions in a parallel fashion maximizes the data transfer rate but requires an increased number of input/output pins. A serial approach reduces the input/output pin requirement but functions at a slower speed. The recommended approach uses a serial path for test and maintenance control and data information. Section 7.0 describes this subject in more detail.

3.6 Applicability To Design Levels

The objective of Section 3 is to identify established testability/diagnostic techniques that should be considered for incorporation in the design of electronic cards/boards/modules identified for use in modern weapon systems. The ultimate goal is to ensure that the testability features contribute significantly to an improved operational availability. Although general in nature, their applicability is more germane to board/module level considerations.

4.0 STANDARD TESTABILITY APPROACHES

4.1 Scan Techniques

As digital circuits have become more complicated,so the test patterns for comprehensive fault coverage have become more complicated and longer. This trend continued to the point that it often became impractical for a test engineer to develop a comprehensive pattern set manually. Automatic test program generators and simulators were developed to help the test engineer. The size of digital circuits has continued to grow, however, to the point that generating test patterns and simulating fault coverage on the most powerful automatic test program generators can be very expensive and time consuming.

Scan techniques are a group of design methodologies that separate combinational logic and sequential logic into separate, easily testable groupings. Scan design depends upon the fact that digital logic may be partitioned into groups of combinational logic separated by sequential logic. (See Figure 1.)

4.1.1 Scan Path

In the scan-path technique the circuit is designed so that it has two modes of operation: one that is the normal functional mode and another that is a test mode in which the circuit flip-flops are interconnected into a shift register. With the circuit in test mode, it is possible to shift an arbitrary test pattern into the flip-flops. By returning the circuit to normal mode for one clock period, the combinational circuitry can act upon the flip-flop contents and primary input signals and then store the results in the flip-flops. If the circuit is then placed into test mode, it is possible to shift out the contents of the flip-flops and compare these contents with the correct response.

In the following discussion of scan-path techniques, it is assumed that the circuit is constructed of flip-flops interconnected by combinational networks. It will also be assumed initially that all of the flip-flops are clocked by a single common clock signal. These assumptions mean that the circuit can be considered to have the general structure shown in Figure 2. Drawing the circuit in this form is done to

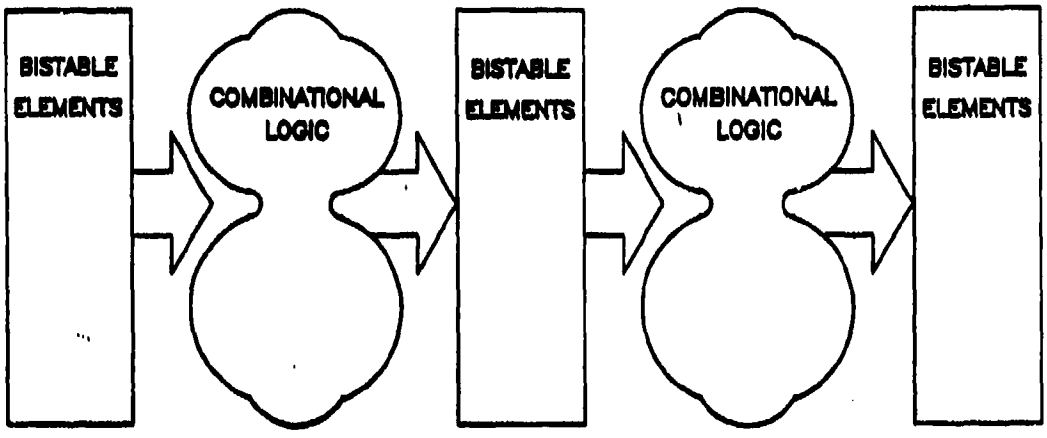


FIGURE 1. PARTITIONING OF CIRCUITS FOR SCAN DESIGN

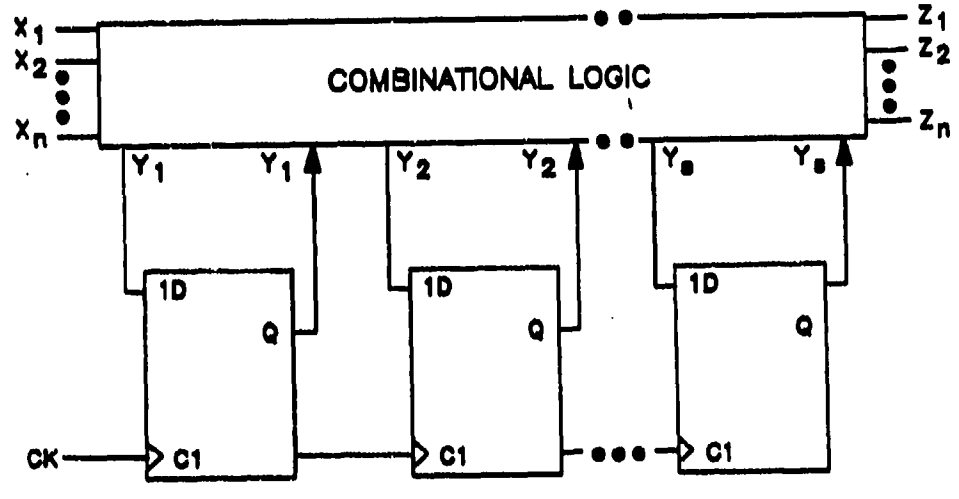


FIGURE 2. GENERAL STRUCTURE OF CIRCUITS FOR SCAN-PATH DISCUSSION

simplify the following di, but is not meant to imply any restrictions on the designer, other than the above.

The structured here is the most common type of digital circuit and is the easiest to m the scan-path technique. D flip-flops are shown in Figure 2 and will be usedollowing discussion. The same general results hold true if other types of flipS, JK, T, etc. - are used. Because the extension to other flip-flops is straightthe details will not be presented here.

It is importatinguish between latches and flip-flops. This distinction is illustrated 3. A latch has the transparency property; if the data input changes whock is active, the latch output will follow the data input change. This is illn Figure 3 by the changes in the Q latch waveforms at times 21, 28, and 3:flop changes only when the clock input makes a specific transition - thransition. For the flip-flop in Figure 3, the active transition of the clock inrs when the clock changes from zero to one. The flip-flop output takes alue present at the data input when this active transition occurs. Subchanges of the data input have no effect until the next active transition of t

An importanteristic of flip-flops is that a shift register can be constructed by connectitput of one flip-flop directly to the data input of the next flip-flop. The convta latch register to a shift register requires an extra latch between each regis.

In one technio of the circuit flip-flops is replaced by the flip-flop structure shown in Figumultiplexer is placed at the data input to permit a selection of two differenputs - d0, or normal system operation; and d1, or test mode. The choice nput is based on the value of the control input, T. When T equals zero, datd from the d0 input upon an active clock transition. Data is taken from d1 if Tone. A type D flip-flop with multiplexed data inputs, such as in Figure 4(a), wled a multiplexed data flip-flop or MD flip-flop.

It should be nothe design of Figure 4 has the undesirable feature of increasing the propagay of the flip-flop. This is not inherent in an MD flip-flop. The additional debe eliminated, except possibly for the effect of additional gate fan-in, bying the flip-flop to incorporate the multiplexer into the flip-flop circuitry.

The modificatiobasic circuit structure of Figure 2 to obtain a scan-path architecture using Mps is shown in Figure 5. One additional input, the T input, has been added.mal operation, T is equal to zero and the circuit is connected as in Figure 2per data inputs ($Y_1 \ldots Y_s$) originate from internal nodes of combinational ciling observed and act as the flip-flop D inputs. In

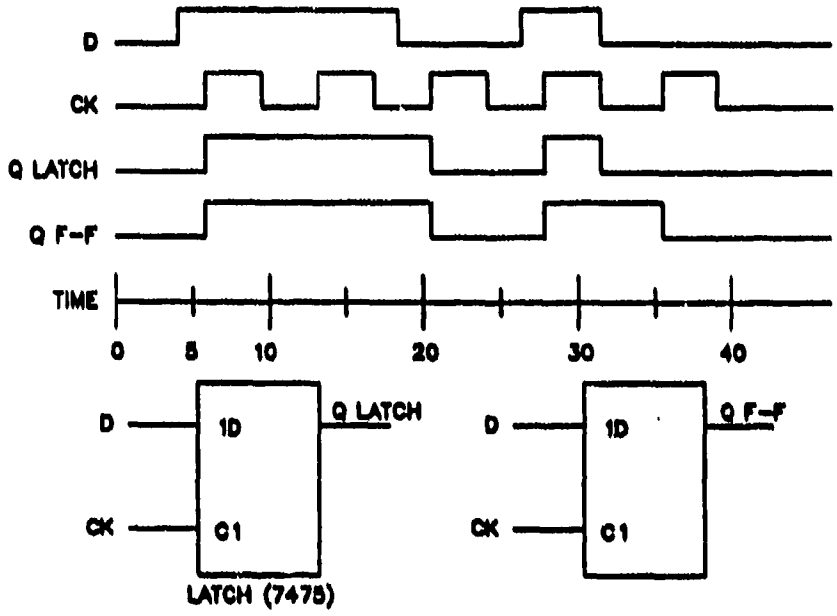


FIGURE 3. LATCHES & FLIP-FLOPS: SYMBOLS & WAVEFORMS

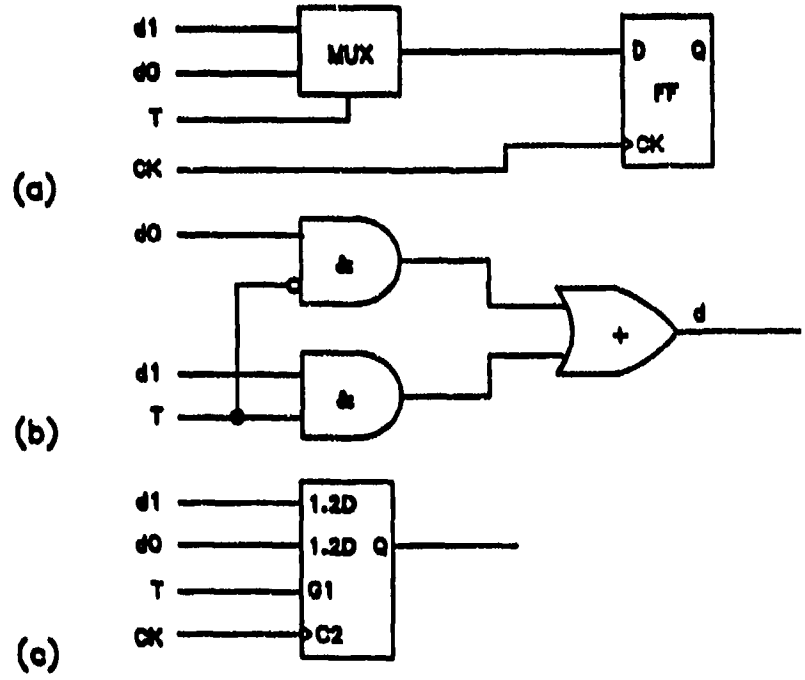


FIGURE 4. FLIP-FLOP MULTIPLEXER (a), MULTIPLEXER CIRCUIT DIAGRAM (b), AND SYMBOL (c) FOR MUTIPLEXED DATA FLIP-FLOP (MD FLIP-FLOP)

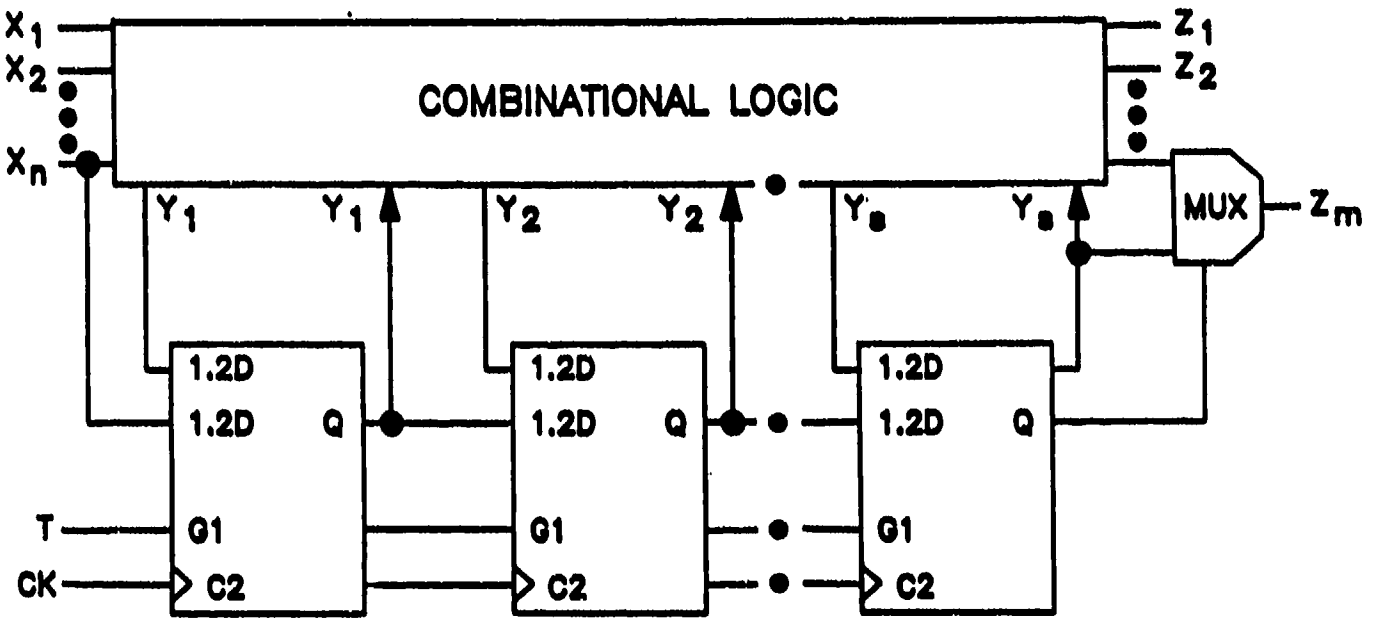


FIGURE 5. STANFORD SCAN--PATH ARCHITECTURE

order to test the circuit, is set equal to one. The lower data inputs become the flip-flop D inputs. Thus, D_i equals Q_i -1 for i from 2 to s, and a shift register is formed.

The primary input X_n is connected to D1 and becomes the shift register input. The shift register output Q_s appears at the primary output Z_m.

Testing of the combinational logic is accomplished by:

- Setting T equal to one, or scan mode;
- Shifting the test pattern y_j values into the flip-flops;
- Setting the corresponding test values on the X_i inputs;
- Setting T equal to zero and after a sufficient time for the combinational logic to settle, checking the output Z_k values;
- Applying a clock signal to CK;
- Setting T equal to one and shifting out the flip-flop contents via Z_m. The next y_j test pattern can be shifted in at the same time. The y_j values shifted out are compared with the good response values for y_j.

The flip-flop must also be tested. This is accomplished by shifting a string of ones and then a string of zeros - or a string of alternating ones and zeros - through the shift register to verify the possibility of shifting both a one and a zero into each flip-flop.

A number of manufacturers have adopted DFT methods that are very similar to this scan-path scheme. The major differences are in the basic scan-path bistable element design and in the way in which the scan path is interfaced with the functional circuitry.

A basic requirement of the scan-path technique is that it is possible to gate data into the system flip-flops from two different sources. One method of doing this is to add multiplexers to the system flip-flops as shown in Figure 5. Another possibility is to replace each system flip-flop with a two-port flip-flop, which is a flip-flop having two control inputs and its data source determined by which control is pulsed.

A circuit for a two-port flip-flop is shown in Figure 6. When a pulse is applied to CK1, data is entered from D1; and when a pulse occurs at CK2, data is entered from D2. The two-port flip-flops are preferred over MD flip-flops because using them make it easier to implement the control for a structure.

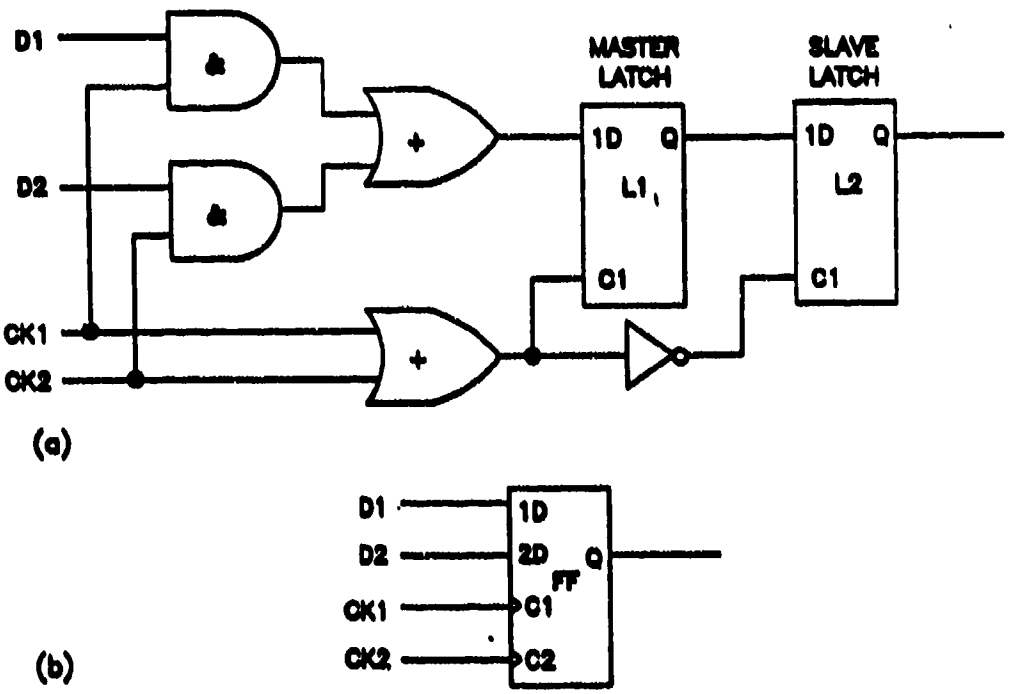


FIGURE 6. TWO-PORT D FLIP-FLOP

Figure 7 shows the structure of a network with two-port flip-flops used to provide the scan path. The testing procedure is basically the same as that described in connection with Figure 5. In the circuit of Figure 7, changing between test mode and normal mode is accomplished by changing the clocking rather than by changing the mode signal (System Clock is SCK . . . Test Clock is TCK).

4.1.2 Level-Sensitive Scan Design (LSSD)

Some systems are designed to use latches, rather than flip-flops, as the bistable elements. For latch-based systems it is not possible to reconfigure the system bistable elements directly into a shift register for test purposes. Several different approaches have been developed to permit control and observation of the system latches through a small number of I/O pins.

One of the most popular techniques for introducing scan-path testability into latch-based systems is LSSD. LSSD is a scan-path design method for latch-based systems. In this method, each system latch is replaced by a two-port latch, and L1 latch. A second single-port latch, an L2 latch, is added to permit reconfiguration of the system's latches into a shift register for test purposes. The L1 latch is a two-port latch that is directly analogous to a two-port flip-flop. It is a latch with two data inputs, each of which is controlled by a separate clock. This method is

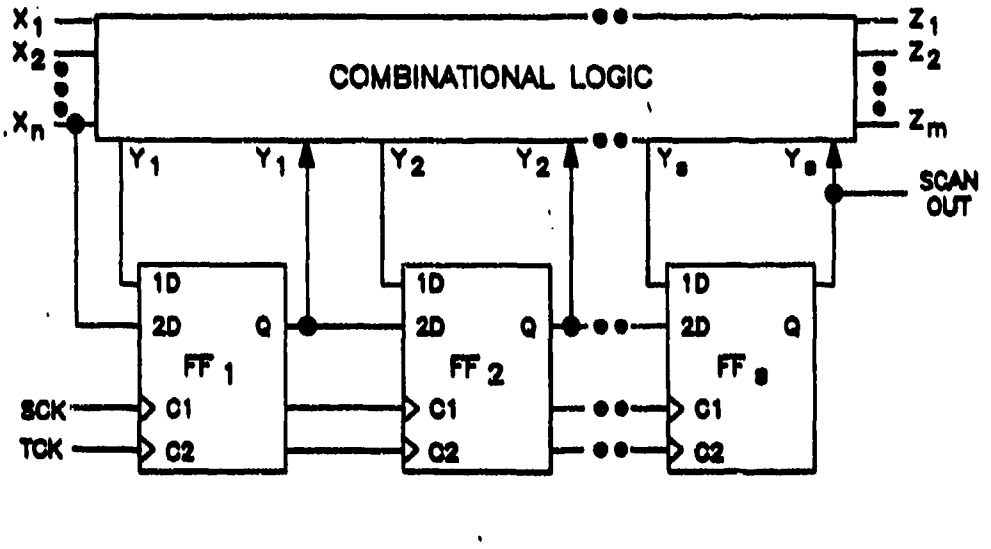


FIGURE 7. GENERAL STRUCTURE OF CIRCUIT USING TWO-PORT FLIP-FLOPS TO PROVIDE SCAN PATH

called level-sensitive because the latches are hazard free: that is, they are not dependent on the clock rise and fall times for correct operation.

A general structure for a system designed using the LSSD technique is shown in Figure 8. During normal operation, the system is clocked with two interleaved, non-overlapping pulse trains applied to the CK1 and CK2 inputs.

4.1.3 Scan-Set

All of the previous methods use the functional system flip-flops or latches to scan test data into and out of the circuit. It is also possible to add to the functional circuitry a shift register whose sole purpose is the shifting in and out of test data. Note that in this technique the term "flip-flop" means either a latch or flip-flop. When it is necessary to distinguish, a latch is called a "latch flip-flop" and a flip-flop is called an "edge-triggered flip-flop." The resulting structure is shown in Figure 9.

Test data is shifted into the flip-flop register (FF1 - FFs) from the SDI connection by clocking TCK. The test data is transferred in parallel to the system latches through their 2D inputs by applying a pulse to UCK. Scanning out the latch data is the reverse process: the latch contents are loaded in parallel into the shift register by pulsing DCK. Shifting out other register contents is accomplished by clocking TCK. The data is shifted to the Shift Data Out (SDO) terminal.

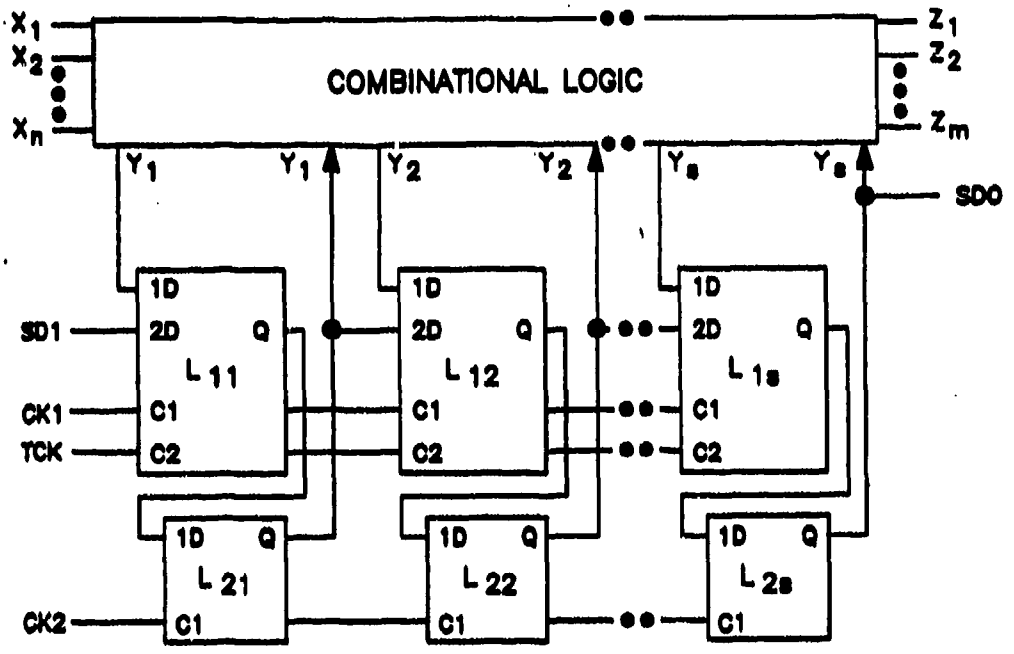


FIGURE 8. GENERAL STRUCTURE OF CIRCUIT USING TWO-PORT LATCHES TO PROVIDE SCAN PATH-LSSD DOUBLE-LATCH DESIGN

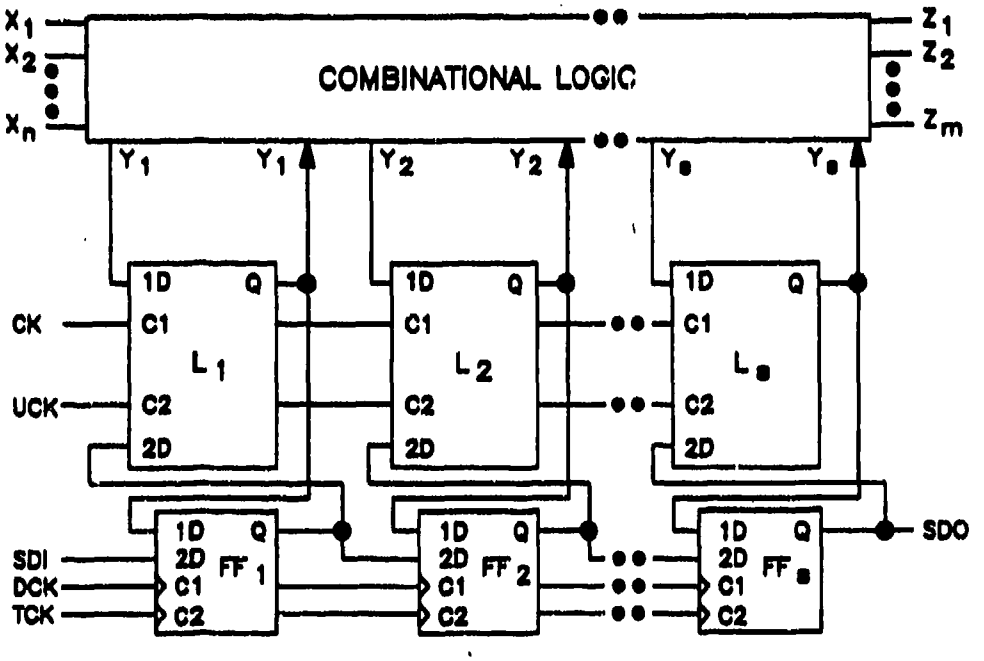


FIGURE 9. GENERAL STRUCTURE OF CIRCUIT USING SCAN-SET TECHNIQUES

To implement the structure of Figure 9, the system latches must be converted into two-port latches and a shift register stage must be added for each such latch. There is more hardware overhead in this technique than in LSSD: two latches per system latch for scan-set versus one latch per system latch for LSSD. Both techniques require the conversion of the system latches into two-port latches. Scan-set requires one shift register stage per system latch and each such stage requires the equivalent of two latches.

Scan-set does have an important advantage compared with the techniques described above: with scan-set it is possible to gate the latch contents into the test shift register during normal system operation. This provides a means for getting a "snap-shot" of system status. Another important feature of scan-set is the ability to scan circuit nodes other than latch outputs into the test shift register. Thus, it has the ability to introduce observation test points at non-latch nodes.

All of the previously discussed scan-path techniques use a shift register to convert between serial and parallel data. Serialization of parallel data can also be done with a multiplexer. A circuit structure with a multiplexer, which is used to scan out the system latches, is shown in Figure 10. Use of more than one scan-out point increases the speed of scanning and also increases the number of I/O connections required. One possibility for avoiding this increase is to place multiplexers on output pins to permit some of the output pins to be used both for system output and for scanning out test data.

With a multiplexer scan structure, nodes other than latch outputs can be accessed. The scanning operation can take place while the system is operating. Complete scan out of all scan points is simplest if the scan data address register can be configured as a counter that steps through all addresses when clocked.

This multiplexer structure improves the observability of a design, but does nothing for the controllability. Setting of the system latches can be accomplished with a demultiplexer. The use of a demultiplexer for setting the system latches and a multiplexer for scan out forms the basis for the random access structure.

4.1.4 Random Access Scan

The principles of multiplexing and demultiplexing can be used to implement a scan technique for latch-based systems. A simplified version of the latch design used is shown in Figure 11. This is called an addressable latch. Inputs 1D(Y1) and C1(CK) are used during normal system operation.

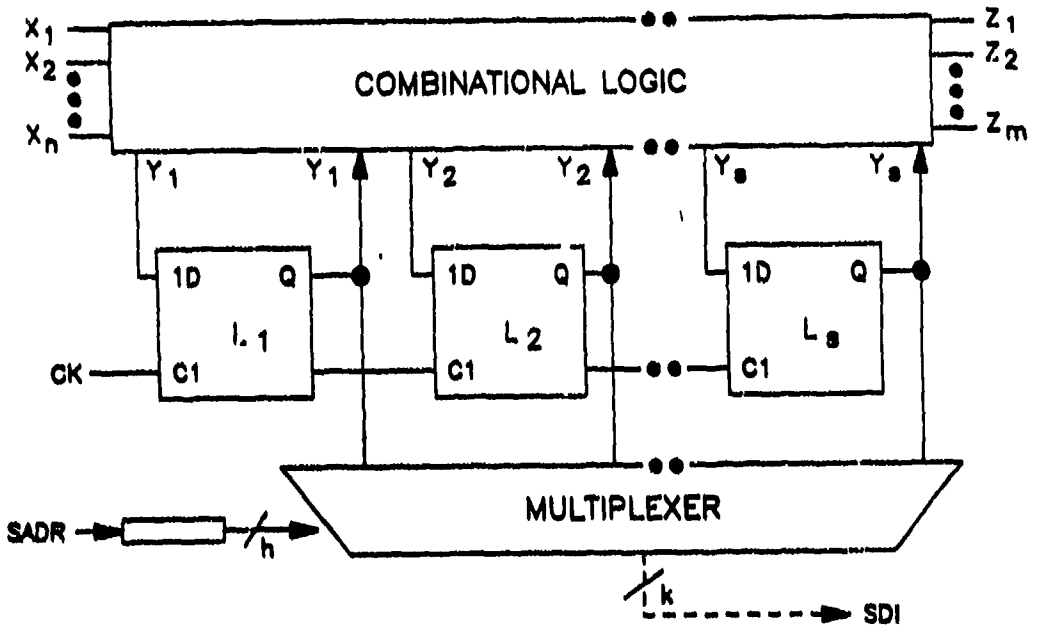


FIGURE 10. GENERAL STRUCTURE OF CIRCUIT USING MULTIPLEXER TO SCAN-OUT LATCH CONTENTS

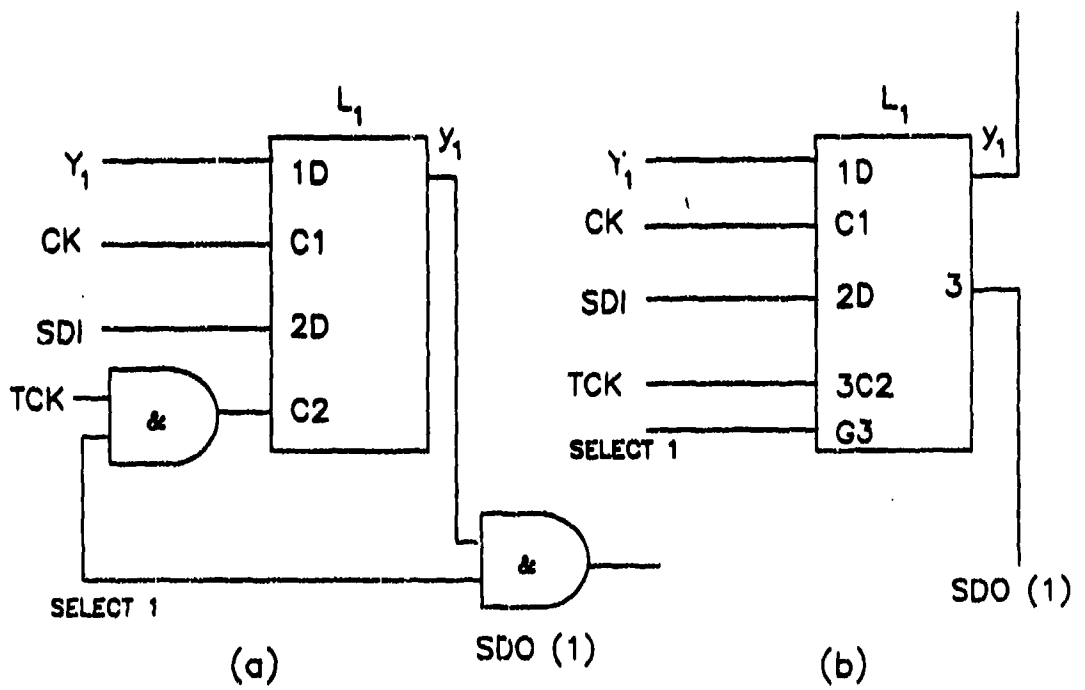


FIGURE 11. ADDRESSABLE LATCH CIRCUIT (a) & SYMBOL (b)

In order to access Latch I for test purposes, the signal "select I" must be set to one. With "Select I" equals one, the latch content is placed on SDO (I) and SDI is clocked into the latch if TCK is pulsed. The structure of a system using these latches is shown in Figure 12.

There is associated with the circuit an address register whose contents are decoded to produce the "Select I" signals. At most, one of these signals is equal to one at a given time. Data are scanned into the latches by placing the latch I data value on SDI, the I address in the address register, and then pulsing TCK. The address register is implemented as a counter. Thus, a sequence of data can be scanned into the latches by placing the sequence on SDI and pulsing the address register counter and TCK in the proper time relationship.

The latch contents are scanned out via SDO by pulsing the address register to select the latches in turn. An important feature of this structure is the ability to scan out the latches during normal system operation.

Actual implementations of this technique using addressable latches have two or three select signals per latch. These signals are decoded at each latch using the circuit shown in Figure 13 for the case of two select signals. Somewhat more complex latches are used in the actual systems in order to take advantage of the Emitter-Coupled Logic (ECL) technology and minimize the penalties due to addressability.

A number of different methods for providing circuitry that allow access to internal nodes through a very small number of test pins have been presented.

Some of the different characteristics, such as the use of latches or flip-flops, occur because of features of the system being designed. Other characteristics, such as the use of a test clock or a level test signal, are design decisions that depend on many factors, including the chip technology, the experience of the designer, the relative cost of interconnect and logic, the level of fault coverage and diagnosis desired, etc. There does not exist a single scan-path technique that is best for all applications. This section introduced the various approaches that are used to help the designer make an informed choice of which scan-path technique, if any, is best for his application.

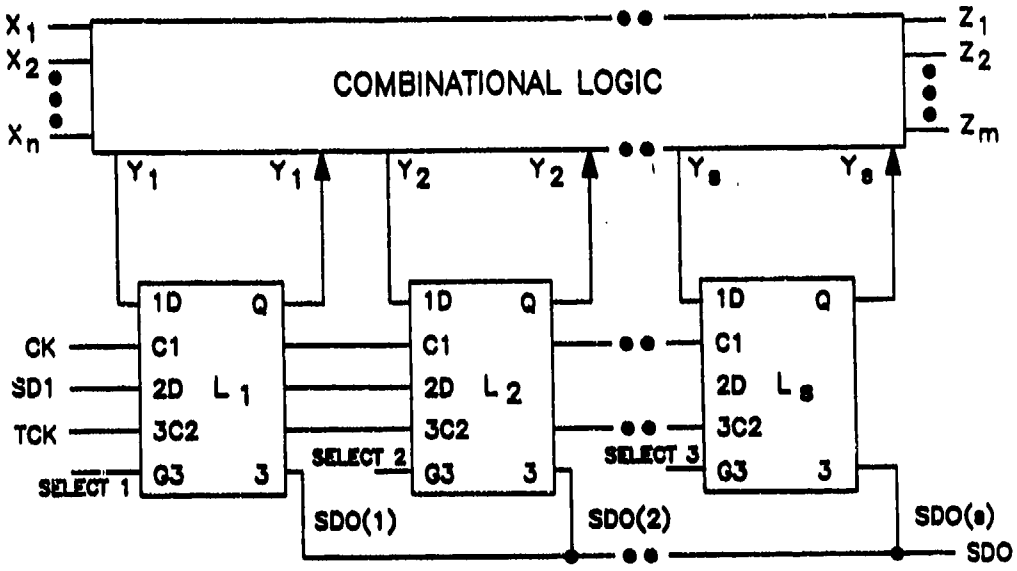


FIGURE 12. GENERAL STRUCTURE OF CIRCUIT USING RANDOM-ACCESS SCAN

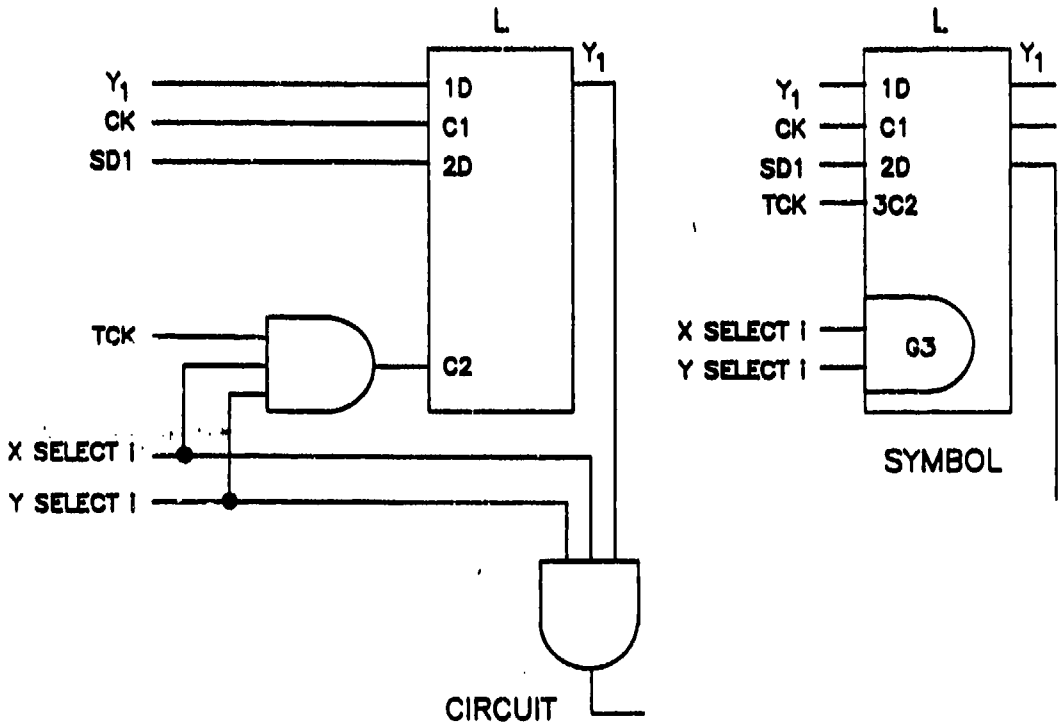


FIGURE 13. ADDRESSABLE LATCH WITH COINCIDENT SELECTION

4.1.5 Scan and the TM Bus

As can be seen from an examination of the TM bus specification in Section 7, this bus has been carefully selected by the VHSIC Phase 2 Contractors to be compatible with both the scan design approach and the signature analysis approach of implementing testability. For those contractors preferring scan design, there is a virtual one to one correspondence between the functions of the four bus lines and the four lines (maximum) generally required by scan design. Thus, it is recommended that any producer desiring to utilize scan design follow closely the specification of Section 7.

4.1.6 Boundary Scan

The scan-path techniques described in the previous sections improve testability by increasing controllability and observability through better internal access and by eliminating the necessity of sequential circuit test pattern generation. The technique described in this section improves testability by reducing the requirements placed on the physical test equipment.

The general I/O scan-path structure is shown in Figure 14. The system latches are implemented in an LSSD-type design so that they form a scan path, called internal scan path, or ring, for test purposes. In addition, a pair of scan-path latches are introduced for each I/O bounding pad. These I/O latches are configured as another LSSD-type scan path, called external scan path, or ring.

The test procedure is very similar to that described in the LSSD structures section. The necessary modifications are that the X_i values are scanned in via the SDI pin. The DMUX control must be set to direct its inputs to the external ring.

The Z outputs from the combinational logic must be clocked into the external ring latches that are then shifted out via the SDO pin.

The presence of the external scan path allows a chip to be tested through a small number of probe pins: seven control pins plus two pins for power and ground. Another feature is that this structure can easily be modified for use in a built-in self-test configuration.

4.1.7 Applicability to Design Levels

Scan techniques are appropriate for use at the component level. Both VHSIC and non-VHSIC chips are currently available with this powerful feature. Scan

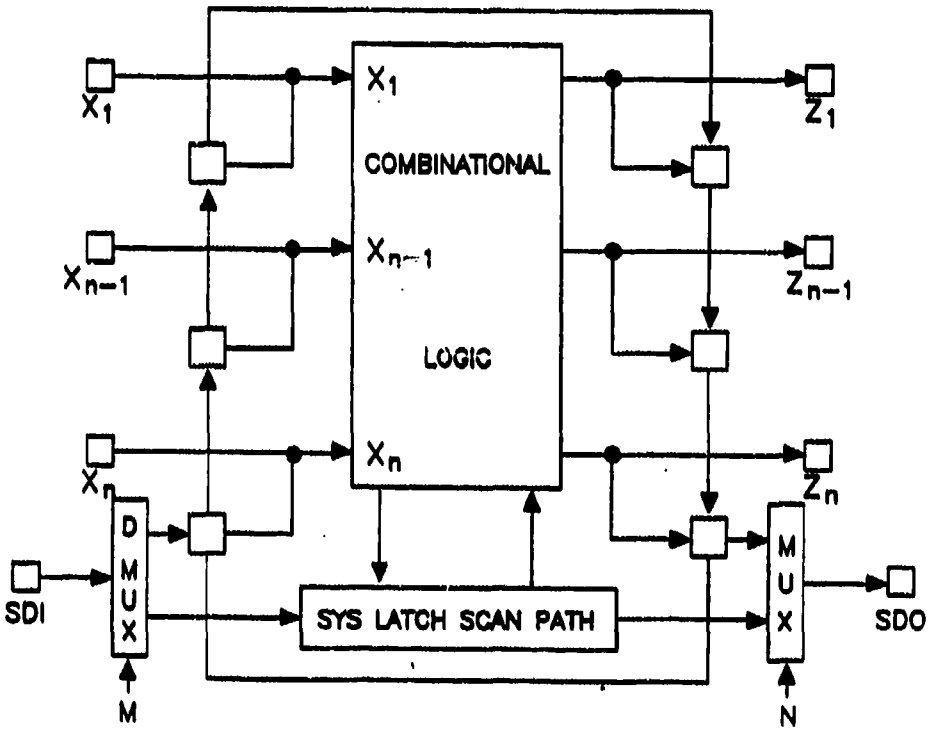


FIGURE 14. GENERAL STRUCTURE CIRCUIT USING SCAN LATCHES ON INPUT/OUTPUT PINS

is also a very powerful tool for use at the board or module level. Boundary scan has the promise of being applicable at the box or subsystem level and at the system level itself. Although its use at these higher levels is in a formative state, boundary scan could play a key role in the increased BIT capability necessary for a successful implementation of two-level maintenance.

4.2 Signature Analysis

Normally speaking, signature analysis is a technique of compressing the sequential digital values of a signal to be evaluated into a smaller number of bits called a signature. In this section, signature analysis is defined broadly as a technique which utilizes one or more Linear Feedback Shift Registers (LFSR) to generate pseudo-random stimuli and one or more LFSR to compress resultant patterns and compare same against stored known-good results. Positive aspects are the small amount of circuitry required, the advantages of an at-speed test, and

the elimination of the process of obtaining logically derived test vectors. Negative aspects are possible aliasing problems and reduced fault coverage in certain test situations.

Techniques for data compression and for embedding the signature analysis technique in a module are described in the paragraphs to follow. For purposes of addressing the complete problem associated with testing, this discussion is aimed at the task of incorporating self test into the module. Depending on the maintenance concept and its implementation strategy, some or all of the components described may be designed into other modules or embedded in external automatic test equipment. The available techniques, if not their relative advantages, will be the same.

4.2.1 Stimulus Generation

Designing a module to provide full self-test capability requires that both stimulus and response evaluation be provided on the module. Each function must be implemented by selecting and adapting elements from among the available techniques. These techniques are described below in separate paragraphs.

The underlying concept of signature analysis is that of the "toggling line" (a logic node that changes from one logic state to another). Only a limited amount of information can be derived from a logic node that never changes state during a particular test sequence. It doesn't get "exercised." This applies to nodes on modules as well as the logic nodes internal to a package (IC). On the other hand, if a node actively changes state in a proper manner at the correct time for that circuit, valid information and confidence result. In fact, it often statistically doesn't matter very much "what" caused it to toggle or wiggle for it to provide useful diagnostic information. However, a "truth table" exercise on the components is generally more exhaustive and will catch a greater number of failures.

The signal that causes the node to toggle is the "stimulus." In self test, the stimulus is supplied by the product itself. By doing this, a controlled environment can be created wherein selected circuit portions can be tested independently of others. Additionally, synchronization and measurement intervals for the signature generator must be controlled. In microprocessor systems, the stimulus needs to be nothing more than a program (generally in ROM) that exercises the rest of the system. Taking advantage of the data manipulative capabilities of microprocessors, generating good stimulus patterns that exercise individual devices in the module is sometimes not difficult. It is often true that the more complex the system, the greater the benefit derived from using signature analysis.

For modules which do not contain microprocessors, it is still possible to store test patterns on a ROM and to apply them as stimulus to the remainder of the circuitry. The outputs of the ROM could be multiplexed with the inputs and stimulus

test points in a manner illustrated in Figure 15. This arrangement permits any specific inputs to be accommodated on a module and to be used to generate self-test stimuli.

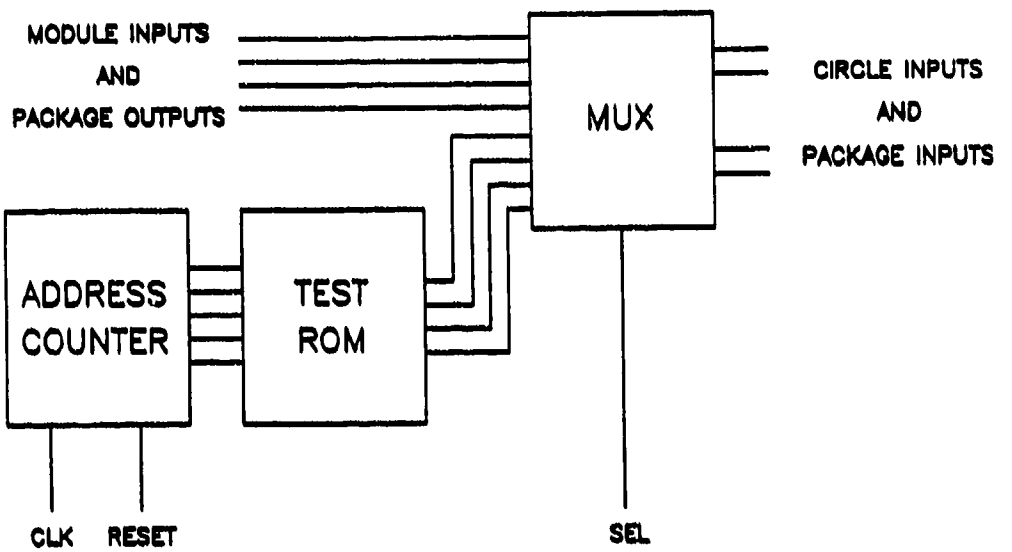


FIGURE 15. TEST ROM STIMULUS

A major disadvantage of saving self-test patterns on ROM is that the number of packages (ICs) which must be devoted to test stimulus is large. This number is as sensitive to the number of points to which stimuli must be sent as to anything else. Typical ROMs provide 8-pin or 16-pin outputs. Thus, as more points are required, more packages are needed. One alternative is to distribute the ROM outputs to multiple points. Multiplexing or shifting of the ROM outputs can be accomplished by various methods. Each such approach limits the stimulus either as to the number of stimulus test points which can change status in a test sequence or as to the rate at which the test can be carried out.

Another method of generating stimuli is to use a binary counter to apply all 2 to the nth input combinations to the (combinational) circuit being tested. This is called exhaustive testing.

Exhaustive testing provides a thorough test, but can require a prohibitively long test time for circuits with 20 or more inputs. It is possible to reduce the test time to a practical value while retaining many of the advantages of exhaustive testing. The method is to apply all possible inputs to portions of the circuit under test rather than to the entire circuit. The simplest approach is applicable to circuits in which none of the outputs depends on all of the inputs.

Most combinational networks have more than one output. In many cases each of the outputs depends on only a subset of the inputs. For example, the parity generator network of the TI SN54/74LS630 has 23 inputs and six output functions, but each output depends on only 10 of the inputs. It may not be practical to exhaustively test the outputs by applying all combinations of the network inputs. However, it may be possible to exhaustively test each output by applying all combinations of only those inputs on which the output depends. For the SN74LS630, each output can be exhaustively tested with 2 to the tenth = 1024 input patterns, and all six outputs can be tested one after another with (6)(1024) = 6144 patterns. In fact, for this circuit it is possible, by an appropriate choice of input patterns, to apply all possible input combinations to each output concurrently rather than serially. Thus, with only 1024 rather than (6)(1024) test patterns, each output can be tested exhaustively.

This method of reducing the number of stimulus test steps required for exhaustive testing may not be applicable or may still result in a test too long to be practical. Another method is required for these cases. That method is to use stimulus test points to partition the circuit in several ways. In each partitioning, each output point depends on only some input points. Thus, for each partitioning the total number of steps required to test the circuit is significantly reduced. More than one partitioning may be required to test all the gates and gate inputs.

Most modern circuitry is not combinational. Thus, the exhaustive method of testing is not applicable. In these cases, stimulus test points can be divided between ROMS for driving sequential circuits and counters for exhaustively testing the combinatorial portions. This approach may also require the addition of gates to separate and decouple sequential circuits from combinationalcircuits.

Stimuli may be generated randomly rather exhaustively. For methods which use this approach, the word random is really a misnomer. The stimuli sequence is always the same and can be described by some generating function. It is called random because the generating function is so complex that no pattern is apparent. Such patterns are often called pseudorandom.

The most common hardware approach for generating pseudorandom patterns is based on a simple circuit called an autonomous LFSR. An LFSR is a series connection of delay elements (D flip-flops) with no external inputs and with all feedback provided by means of exclusive-or gates (XORs). A four-stage LFSR is

shown in the top portion of Figure 16 and the general standard form of LFSR is shown in the bottom portion of Figure 16. The symbol hi in Figure 16 indicates the possible presence of a feedback connection from the output of each stage. If h_i = 1, there is feedback from stage i; and if h_i = 0, the stage i output is not connected to the XOR feedback network. The LFSR can be specified by just listing the values of the h_i or by specifying the generating function as shown in Figure 16.

Another possible realization of an LFSR, called a "modular realization," is shown in Figure 17. There are as many XOR gates in the modular realization as there are feedback taps in the standard circuit. The gates are placed in the "reverse" positions from the locations of the feedback taps. If in the standard LFSR there are m "taps" (inputs to the XOR network generating the feedback signal), m - 1 two-input XOR gates are required if an iterative structure is used to realize the XOR network. This is the minimum gate realization. It is slower than a tree network, which also requires m - 1 gates, but has a delay of log m gate propagations rather than m - 1 gate delays. The modular circuit also requires m - 1 XOR gates. It has a delay of only one gate propagation. For circuits with more than two feedback signals, faster operation always results with the modular rather than the standard LFSR.

The sequence of states for the LFSR of Figure 16 is shown in Table 2. Note that the sequence repeats after 15 (2 to the 4th - 1) clocks. This is the maximum period for a four-stage LFSR; the all-zero state of the register cannot occur in the maximum-length cycle since an all-zero state always has a next state that is also all zeros due to the use of XORs to form the feedback signal. In general, the maximum period for an n-stage LFSR is 2 to the nth - 1. There are maximum-length realizations for all values of n. The generating function corresponding to a maximum-length LFSR is called a primitive polynomial. Tables of primitive polynomials can be found. They have fixed lengths or periods.

One period of the output sequence produced by the LFSR of Figure 16 is:

(0 0 0 1 1 1 1 0 1 0 1 1 0 0 1)

The five-stage LFSR with feedback connections given by H = (100101) has the following output sequence:

(1 1 1 1 1 0 0 0 1 1 0 1 1 1 0 1 0 1 0 0 0 0 1 0 0 1 0 1 1 0 0)

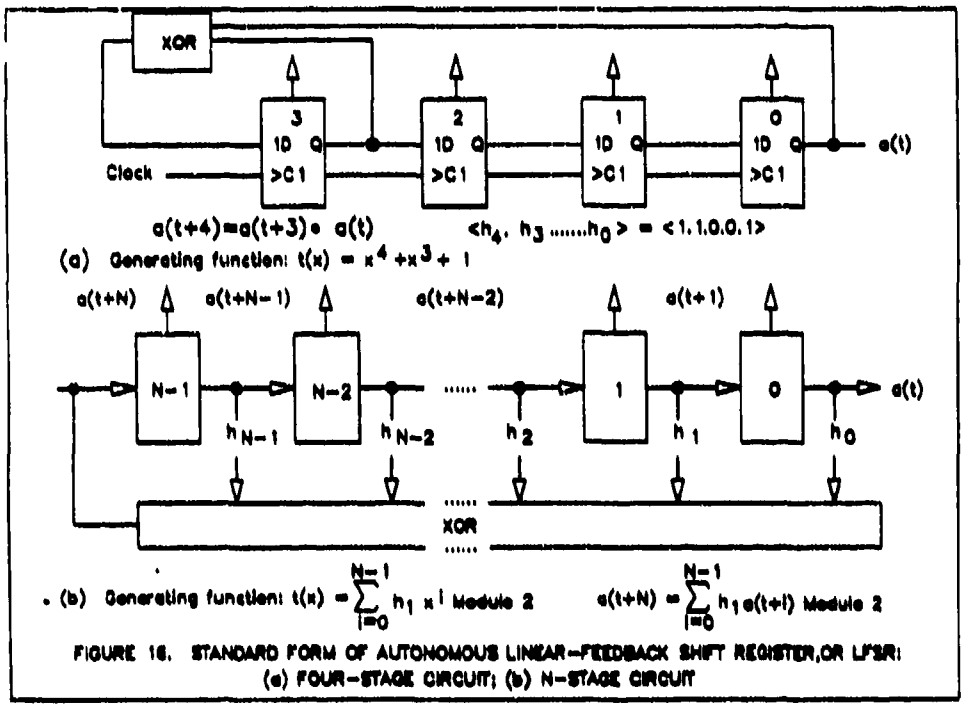


FIGURE 16. STANDARD FORM OF AUTONOMOUS LINEAR-FEEDBACK SHIFT REGISTER,OR LFSR: (a) FOUR-STAGE CIRCUIT; (b) N-STAGE CIRCUIT

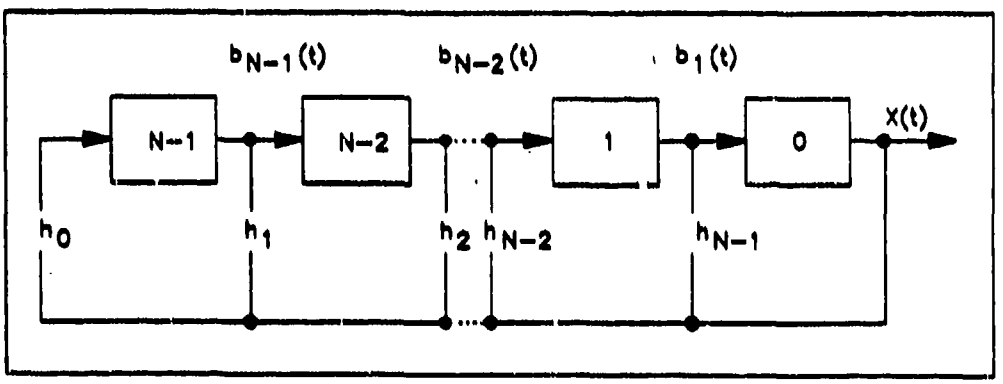


FIGURE 17. GENERAL MODULAR REALIZATION OF AN LFSR

TABLE 2

STATE SEQUENCE FOR FIGURE 16

State	Q1	Q2	Q3	Q4
0	1	0	0	0
1	1	1	0	0
2	1	1	1	0
3	1	1	1	1
4	0	1	1	1
5	1	0	1	1
6	0	1	0	1
7	1	0	1	0
8	1	1	0	1
9	0	1	1	0
10	0	0	1	1
11	1	0	0	1
12	0	1	0	0
13	0	0	1	0
14	0	0	0	1
15 = Q	1	0	0	0

4.2.2 Signature Development

A technique for generating signatures is based on an LFSR in which exclusive-or gates are used to control the feedback. To many people, this technique is synonymous with signature analysis. Accordingly, the compression code generated by this technique is simply called the "signature." An example of a four-stage LFSR circuit is shown in Figure 18. In that figure, outputs of some of the flip-flops provide feedback. One such flip-flop is the first one (FF1) and one is the last one (FF4). Other configurations of feedback can be selected as well as other flip-flop counts.

Two well known versions are 16 stages long. These are the 16-bit Cyclic Redundancy Check (CRC-16) in which the outputs of flip-flops 2, 15, and 16 are fed back to the input, and the Synchronous Data Link Control (SDLC) code in which the outputs of flip-flops 5, 12, and 16 are fed back to the input. Another version, which is used by Hewlett-Packard, feeds the outputs of flip-flops 7, 9, 12, and 16 back to the input.

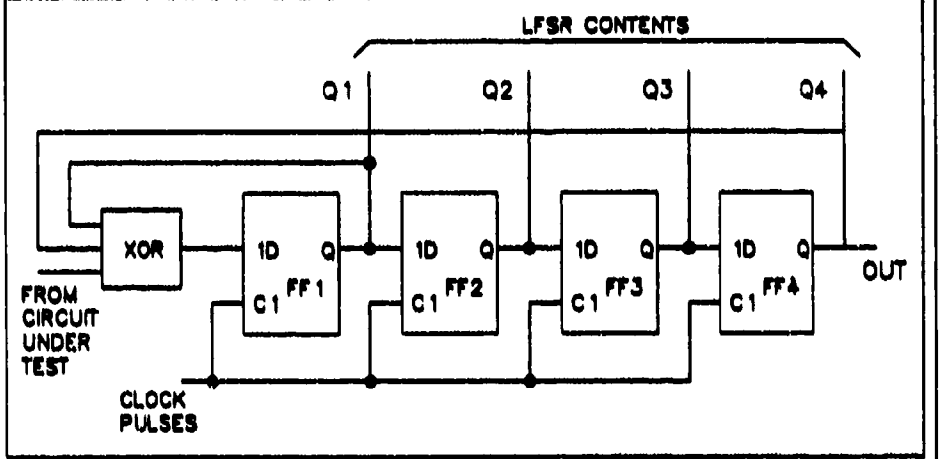


FIGURE 18. FOUR-STAGE LINEAR-FEED-BACK SHIFT REGISTER

Like signatures selected for the parity techniques, the LFSR technique divides the possible input data value sequences into different sets. The number of sets depends on how many stages of flip-flops there are. Signatures can be used to distinguish between any two data value sequences which are in different sets, but cannot be used to distinguish between data value sequences in the same set. Different feedbacks will result in sets with different related characteristics. Both the CRC-16 and SDLC codes have sets with the property that every data value sequence in a given set will have the same parity. The code selected by Hewlett-Packard does not.

The LFSR techniques can be adapted for use with multiple input data streams. Such an approach could be used to reduce the data from several test points to one value which can be used for module screening. It can also support fault isolation to the component level by reducing the data value sequences from all the outputs of a package to one signature for each module. In effect, one signature is provided to check each package. The basic circuit, called a multiple-input LFSR, is shown in Figure 19.

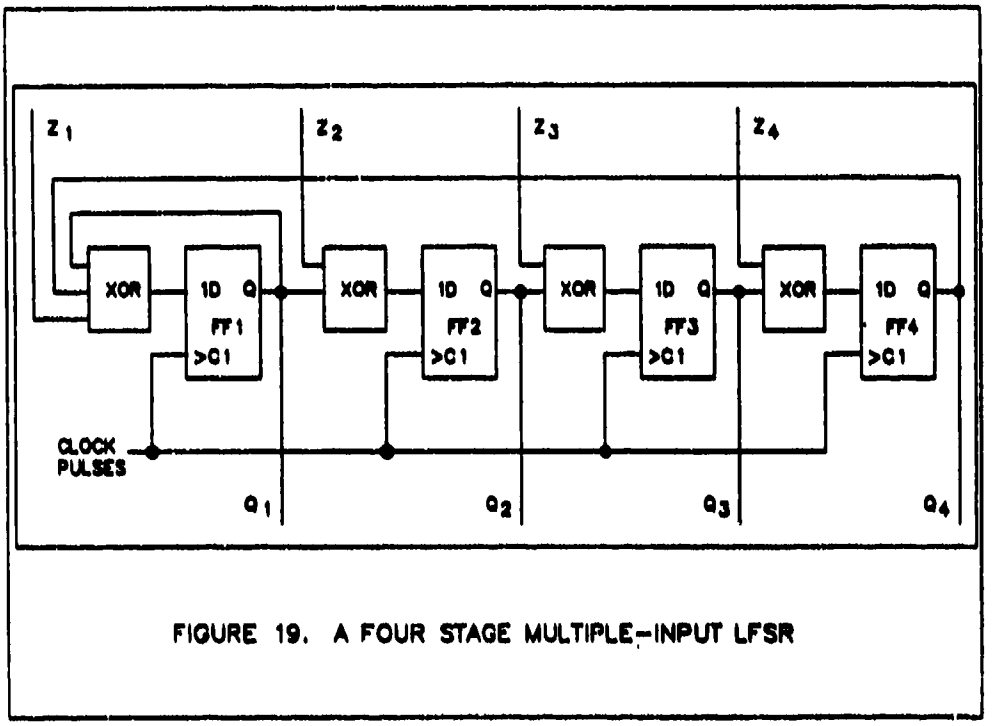


FIGURE 19. A FOUR STAGE MULTIPLE-INPUT LFSR

Care should be taken in how errors will propagate onto the input lines of multiple-input LFSRs because a failure on one input followed by a failure at the next clock on the input entering the next stage flip-flop will exactly cancel out and result in an erroneous indication that the circuit is correct. This type of flaw can be ameliorated by providing inputs only to non-adjacent flip-flops or eliminated by separating inputs with flip-flops that output to feedback taps.

Use of signature analysis to support built-in self-test of modules can only be accomplished if the circuit to which it is to be applied has several testability properties. Among those properties are:

- o The data changes at all measurement points must be synchronous with one Clock line.
- o There must be no feedback loops or there must be methods of breaking feedback loops for testing.

- o It must be possible to isolate the circuit from all input circuits for purposes of testing.

- o There must be a method of forcing the circuit to a known state at the beginning of a test stimulus sequence.

- o There must be a known value at every measurement point for each clock time during the test stimulus sequence. Particular concern must be shown for floating bidirectional lines.

- o There must be methods of disconnecting buses onto which the outputs of several packages are put. The disconnections must be at enough points so that the fault isolation ambiguity group size requirements are met. If this bus also serves for data or control input to the packages, each separated segment must be provided with stimulus data.

4.2.3 Signature Analysis and the TM Bus

Signature analysis can be used to provide self test for a module and to permit fault isolation to the component level under control of an on-line microprocessor or an external ATE. This capability can only be provided by making a large amount of information available at an edge connector and providing a significant amount of control for developing that information. The TM bus is an effective and natural way of meeting these needs without using a large number of input/output lines.

Depending on the system requirements and the module capabilities, evaluation of signatures may take place on the module or it may be done outside of the module. For purposes of simplicity, the case where signature evaluation is done outside the module (either by a system microprocessor or by an ATE) is described. The circuit components needed for this case are shown in Figure 20.

In Figure 20 there is no requirement for an embedded microprocessor controller. Such a component is not precluded and if it is available, it can be put between the TM BUS CONTROL circuitry and the other self-test circuitry to expand and automate the testing services provided.

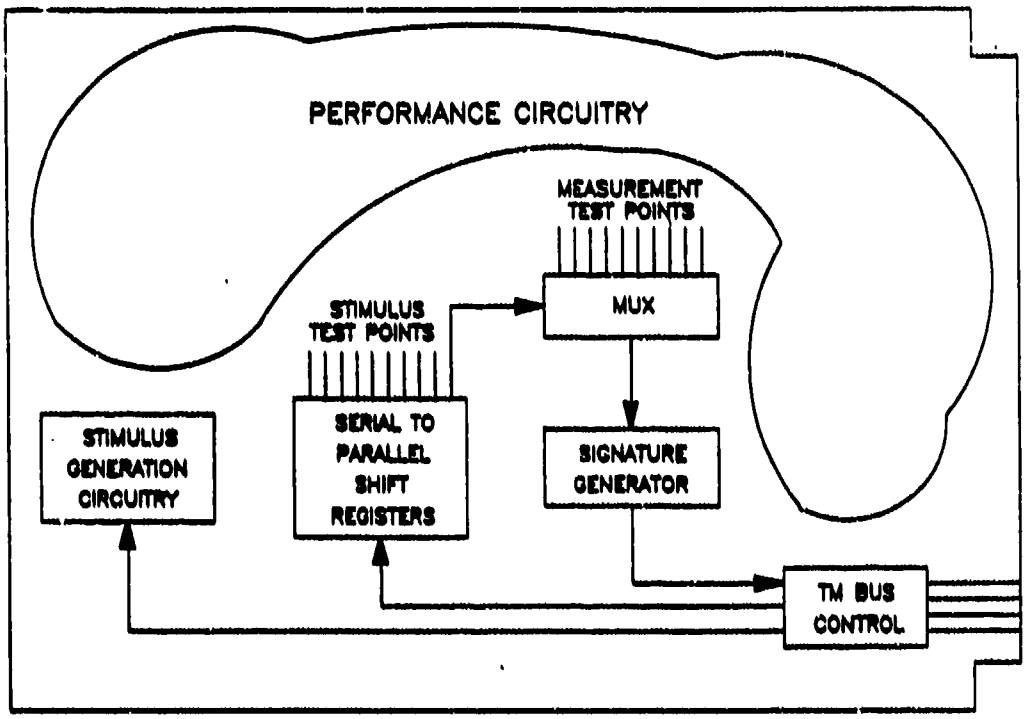


FIGURE 20. EXAMPLE USE OF SIGNATURE ANALYSIS WITH TM BUS

The TM BUS CONTROL circuitry is the central control element for self-test capabilities when there is no embedded microprocessor. In this role, its function is to accept test commands from the TM bus and implement them through the self-test circuitry. There are only three types of commands which are necessary:

- o Transfer data to the stimulus test points
- o Run the self-test stimulus
- o Collect and return the generated signatures over the TM bus

In Figure 20, the stimulus test point control circuitry and stimulus generation circuitry are shown as separate functions. One of them, the stimulus test point control, reconfigures the circuit for testing and selects the measurement test point whose signature is to be made available for evaluation. The other stimulus function, the stimulus generation circuitry, consists of ROMs, Binary Counters, and LFSRs as required to generate a stimulus signal.

The stimulus control block is described in Figure 20 as being serial to parallel shift registers. It is not mandatory that this technique be used to accomplish the function of setting the values of stimulus test points. But the technique does offer advantages in this situation. One advantage is that the data to be entered can be sent directly (through a buffer) from the TM BUS MASTER DATA line into the shift register. This reduces the amount of processing required of the TM BUS CONTROL circuitry. Another reason is that this method requires fewer interconnections within the module than would be required by demultiplexing and decoding techniques. This may reduce the board area requirements of the test circuitry. A final advantage is that some of the serial to parallel shift registers may be incorporated within semi-custom or custom packages (ICs) without requiring a large number of package input/output pins.

The stimulus generation circuitry needs to be treated differently. It must generate a sequence of test steps which will expose any faults that exist at one of the test measurement points. One or more of the approaches described in paragraph 4.2.1 would be used (i.e. ROM sequences, binary counters or LFSRs). It must operate dynamically along with a synchronizing clock and provide start and stop signals for the signature generation circuit.

Only one signature generator is shown in Figure 20. More signal generators could be used or a multiple input signal generator could be used. The approach depicted utilizes a multiplexer and requires the minimum amount of circuitry for testing. It will support as many test points as may be needed. In any case, only one or possibly a few signatures can be developed for each stimulus test sequence. This data is made available to the TM-BUS CONTROL circuitry to transmit to the test control microprocessor or ATE.

External to Figure 20, but important to the overall approach, is a microprocessor or ATE which controls the utilization of the test circuitry embedded in the module. It must control the module resident test circuitry to set it for testing, run a stimulus test, and collect the resultant signatures. It must also determine which signatures to collect and must evaluate those signatures. Finally, it must communicate the overall results to whoever is responsible for test and maintenance.

4.2.4 Applicability to Design Levels

Signature analysis techniques are currently available in various VHSIC chips. This technique is also appropriate at the board or module level. It has been successfully employed in this role for many years. Applications at higher levels are theoretically possible but a review of current literature has uncovered little.

4.3 Wraparound Techniques

4.3.1 General

The wraparound technique has become a standard in the testing of microprocessor based systems. The system to be tested can consist of a single microprocessor based board or a system made up of many such boards. The start of the test consists of determining whether or not the core of the microprocessor based system is operational. This might be done in some instances through the use of hardware redundancy. As the tested core is found to be operational, the process expands further out through the logic, continually building upon the operational core. A failure at any point must provide sufficient data for diagnosis. As the tested core expands, a transition to other test resources occurs.

At the very heart of the core, hardware techniques are used. As the operational core builds, firmware becomes the medium. As the growth builds further, software in the semiconductor memory of the system would become the medium. After it is determined to be operational, the move would be out into some type of bulk storage medium. The idea is to transition as quickly as possible along this route because as the move transitions from hardware to firmware to software, the cost of the test function is reduced.

A typical microprocessor based system is depicted in Figure 21. The system consists of a microprocessor, an address bus, a data bus, Read Only Memory (ROM) associated with operational software, and Random-Access Memory (RAM) which is provided for user software. There is normally a programmable interface assembly associated with data coming out the microprocessor system. Signals coming out of the microprocessor pass through this interface assembly, through circuitry associated with these outputs, and out to the operational circuitry. The signals coming back from the operational circuitry come through input lines and through other Programmable Interface Assemblies (PIA) back into the microprocessor system. As mentioned, the operational circuitry might be self contained on the board housing the microprocessor itself, or it may be on one or more additional boards in the system.

4.3.2 Core, ROM, RAM Testing

The core test would begin by performing some very rudimentary routines with the microprocessor. For example, data may be moved from one register to another. A simple "ADD" routine might be performed where two numbers are summed and compared against known-good results stored in ROM. After this core capability is validated, the next step would be to move onto the testing of read only memories. A simple test which can be used to ascertain whether or not the ROM is operating correctly would be a checksum. This procedure can be done by the microprocessor itself and requires little, if any, additional circuitry for the test

purpose. However, there is the possibility of compensating errors with the checksum technique. A more elaborate test might be a Cyclic Redundancy Check (CRC). This type of check is time-dependent and will weed out that class of errors. However, it takes additional hardware to perform this particular function. Therefore, a trade-off must be performed to ascertain whether or not the additional hardware is worthwhile in order to catch time-dependent type faults.

After the microprocessor core is deemed operational and the ROM checks have been done, checks on the system RAM can be initiated. There are standard patterns that have become available throughout industry known as GalPats or galloping patterns, walking ones and walking zeros, checkerboards, inverted checkerboards, and other standard tests of this type. The process is to write to memory and then to read this data back to determine whether or not the memory functioned correctly. Again, this can be done on a software basis or it can be done with hardware. A software-based test is necessarily slower because the microprocessor is in the loop. A hardware-based test can be much faster because it can run at the speed at which the memory can cycle, without being constrained by the instruction time of the microprocessor. Again, a trade-off is necessary. If speed is of the essence, then one must pay for the hardware to do the faster tests. If the hardware penalty is too much to pay, then the software-based test would be a good alternative, although it will take longer to perform.

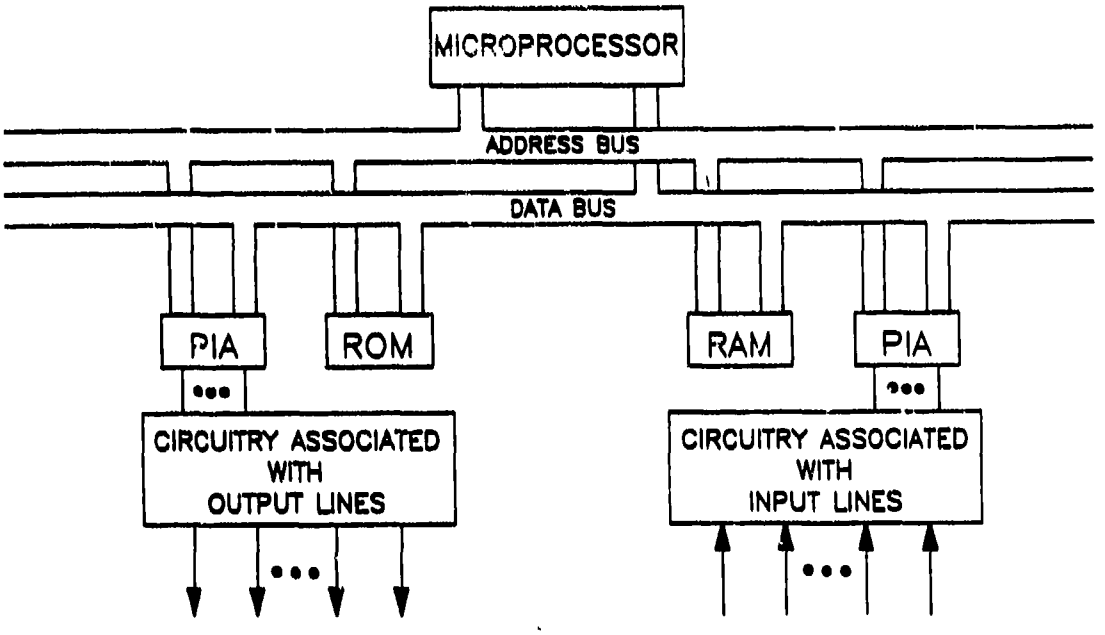


FIGURE 21. TYPICAL MICROPROCESSOR–BASED SYSTEM

4.3.3 Processor Controlled Gating

At this point in time, if the microprocessor core is deemed to be operational and the ROM and RAM are operational, a very powerful core of test capability exists. The next step is to check the remainder of the circuitry, which might be only a relatively small amount on a single board on which the microprocessor is contained, or it might be a larger system consisting of many boards. To do this, a closed-loop system must be created so that stimulus test vectors stored in an area of ROM can be used by the system microprocessor to drive out through the interface assembly and the circuitry associated with output lines to stimulate the rest of the operational circuitry. Processor controlled gating must be included in the system in order to take that data and wrap it around so that a closed loop can be created (refer to Figure 22). This allows the resultant test vectors to come back through the circuitry associated with output lines so that the microprocessor can effect a comparison against known-good test vectors also stored in ROM. The basic idea of this particular test philosophy would be to drive out the stimulus test vectors, toggle the various nodes in the operational circuitry looking for the classical stuck at one and stuck at zero faults, and to wrap these vectors around so that they find their way back to be compared by the microprocessor against stored known-good resultant test vectors. If each one of those closed loops is indeed operational, one and only one known-good vector can be the result in each instance.

This type of built-in test or self test is a very, very powerful technique and has become a standard in the industry. A relatively small amount of test-oriented ROM need be included in a system for test purposes. Estimates on the processor control gating range from 5 to 15 percent of the available real estate in order to effect the wrap around. Of course, it all depends on an operational core. In summary, the technique is to first ascertain whether or not the operational core is working, then to check key items such as system ROM and RAM, and then after having ascertained whether or not that operational core is functioning correctly, to use that operational core as essentially a tester to check the remainder of the circuitry in the system.

This, of course, is a non-real-time type of test. When it is being conducted, the system is not performing its system purpose. It is not a concurrent or real-time test, nor does it deal with the analog testing problem. It is essentially a digital technique. There is always the necessity of a trade-off between how much of the testing is done with hardware dedicated to that purpose (e.g., the CRC or the GalPat generators and monitors) and how much is software-based. In a more software intensive test, the duration is longer. If it is software-based, however, there is much less of a penalty from a hardware standpoint. The generation of the test vectors for the checking of the operational hardware must be derived via standard techniques. Such techniques include digital logic simulators, automatic test pattern generators, and the like. However, this is a one-time development and much of the

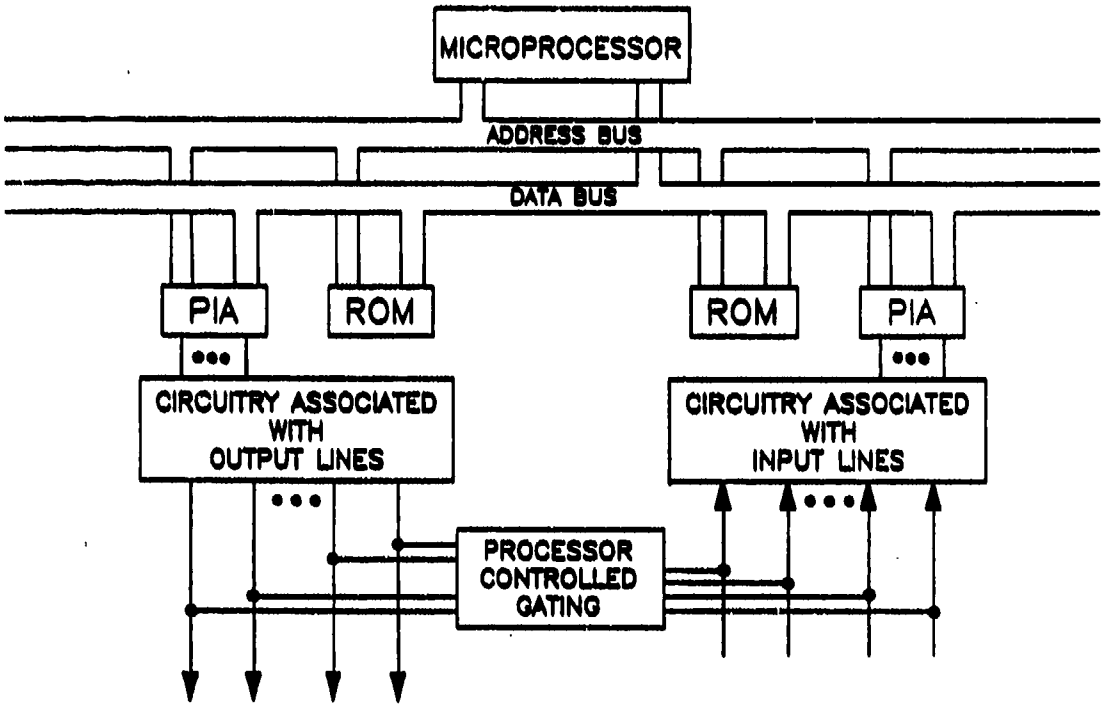


FIGURE 22. TEST DATA WRAPAROUND

software may already be available via the operational system needs. Therefore, the cost associated with it should not be overly severe.

4.3.4 Applicability to Design Levels

The wrap-around technique is appropriate at the board/module or subsystem level. As long as a microprocessor or microprocessors affect a given amount of circuitry, the technique is valid whether the said circuitry is physically on a single board/module or on many boards/modules (i.e., a subsystem).

4.4 Analog Techniques

4.4.1 General

Many electronic systems combine analog and digital circuitry. The analog portion of these systems may range from the DC to Radio Frequency (RF) and microwave mixing and signal down conversion. Mechanical or environmental

signals (e.g., temperature monitors, infrared sensors, etc.) may also be included. In general, analog fault types and symptoms may be more complex than their digital counterparts. Although faults can be modeled as simple shorts or opens, the impact of some analog shorts or opens may be marginal and may not cause a noticeable degradation of operational performance under normal conditions. This contrasts with digital circuitry where, with enough patterns, a fault will eventually propagate to cause a clearly incorrect output. Analog signals may have an infinite number of values within a given range, while digital signals have a finite set of values for each given node at each point in time.

The major difficulty in testing analog circuits is not the number of variables that need to be tested, but the variation in signal parameters. Digital signals have a set clock frequency and set voltage levels where analog signals may constantly vary in frequency levels and may consist of multiple complex signals modulated onto a carrier. Failure modes causing marginal out-of-tolerance conditions may not be detected because subsequent stages may filter, compensate or mask this condition. With a different input signal (e.g., weaker but still within specification), the fault may have a much greater impact on the output signal quality. The impact of a failing component may not be detected until one or more stages after the actual fault. This leads to difficult isolation with the risk of the fault being perceived as a false alarm.

As BIT detection requirements approach 100 percent of all possible faults, BIT test thresholds must be set closer to the operational limits of a parameter in order to observe subtle, second-order faults. This increases the probability that the BIT may indicate a fault due to transient noise or to measurement error when, in fact, no actual fault exists. Thus, increased BIT coverage can lead to increased false alarm rates. Current applications indicate the future of analog testability as a merging of digital overhead with well-partitioned analog functions.

Virtually all of the techniques previously discussed (such as scan techniques, signature analysis, wrap around, etc.,) are inherently digital in nature. None has any real applicability in the analog world. As much as the use of digital circuitry has continued to expand, there is probably some 15 to 25 percent of the circuitry that is still in the analog domain.

4.4.2 Active Versus Passive

Analog techniques can be either active or passive in nature. Active stimulus injection is a more thorough test, in general, because a higher fault coverage can generally be achieved with the injection of active stimuli. However, passive monitoring has the advantages of being less complex in nature and, therefore, less costly and less interfering. The interference factor refers to the situation that if the test circuitry has the ability to actively inject stimuli during a test sequence, there is always the chance that the test stimuli can be inadvertently injected during system operation. This would, of course, be very interfering in

nature. The trade-off between active stimulus injection and passive monitoring is essentially a trade-off that depends upon how much test hardware can be accommodated and what degree of fault coverage is necessary for that particular system.

4.4.3 Conversion to Digital Format

Today most analog circuitry is not found in a stand-alone configuration. Much of it is found in so called "hybrid" circuits where there are both analog and digital circuitry. In this case, it is very helpful in most instances to convert the analog signals to a digital form as quickly as possible. This is because there is a very large probability that there is a microprocessor in the digital part of the circuitry which is being utilized to effectively handle the digital test requirements. By taking the analog circuitry and converting its information into a digital format, all signal processing can be done in a digital mode. Therefore, the same microprocessor can be used to set limits, modify limits, and therefore cause the analog circuitry to be accommodated in much the same flexible fashion as the digital. It also contributes to the intelligent BITconcept where, if a failure occurs, the microprocessor can ascertain that fact and return to test the suspected part of the circuitry a number of times. This cyclic concept is used to ascertain whether the failure was merely a transient or whether it was indeed a hard failure. In short, the analog approach most often recommended is to convert such signals to the digital format as quickly as possible. This allows limits to be set and modified under control of the digital self-test or built-in test circuitry. It also allows all of the data, whether it be analog or digital, to be processed in the same fashion. This approach supports the overall concept of intelligent BIT.

4.4.4 Applicability to Design Levels

The analog techniques discussed are appropriate at various levels from the board/module upward. Little analog capability exists at the component level. Most of the industry effort to date has been in the digital arena; much remains to be accomplished in the analog.

4.5 Concurrent Techniques

4.5.1 General

All of the techniques discussed heretofore cannot be considered as concurrent in nature. For example, when scan design, signature analysis, or wrap-around techniques are being employed, the prime system function cannot be simultaneously executed. These types of tests can only be done in non-real time when the system is not doing its prime purpose. These techniques are also digital in nature and do not deal effectively with analog circuitry. Another common built-in test

technique injects a known signal into the operational circuitry and compares the eventual result to a known-good result.

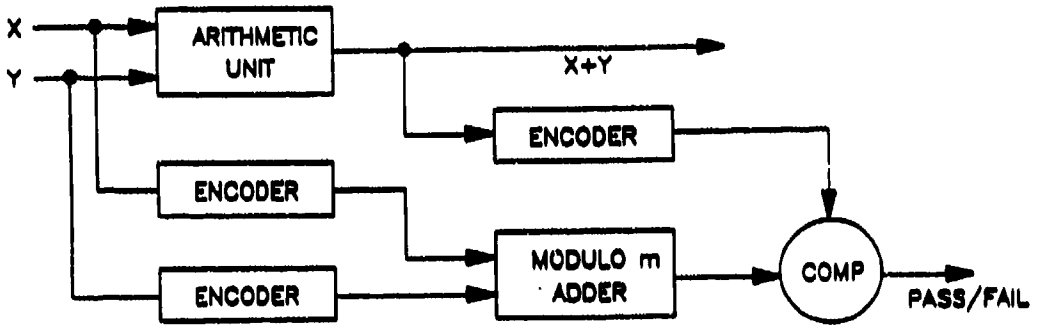
In summary, when dealing with concurrent testing or with analog circuitry, most of the previous techniques are not applicable. Concurrent testing can be either passive or active in nature. A passive approach would merely use monitoring circuits to ascertain whether or not the correct signal was present at various stages. An active circuit would require an injection of a stimulus and then a subsequent measurement. The latter would be done on a time-shared basis, so as not to interfere with concurrent system operation.

There are various techniques which are concurrent in nature and have been used in industry for many years. Parity is such a technique. Parity can be used to ascertain whether or not a bit has been dropped in the transmission of a code or whether a bit has been injected due perhaps to noise. Parity is indeed a concurrent type test. Residue codes are another example of on-line or concurrent testing. Residue coding uses duplicate circuitry of a much simpler design to do a concurrent operation so that a check can be made on the residue of the signals in question (Refer to Figure 23). This allows a concurrent check with a high degree of confidence. If the secondary circuit determines something is wrong, there is a good chance the primary circuit has indeed malfunctioned.

Other examples of concurrent testing include the M out of 2M code where each 2M bit code must have exactly M logic one bits. This technique requires a specific number of ones to be used in each code word. This results in some of the codes being valid and some invalid, so that dropped bits or extra bits injected by noise can be detected. One negative aspect is that many normally valid codes are no longer usable. For example, in a 3 out of 6 code, you would normally have 2 to the 6th power or 64 legal combinations. With this particular approach there are only 20 legal combinations.

Other concurrent techniques include the use of a watchdog timer which closes the loop in signals that are transmitted between a control system and any of its peripheral devices (Refer to Figure 24). In other words, when information is sent from the CPU to a peripheral or visa versa, a free running counter is initiated. If the receipt of signal is not acknowledged at the receiving end within a certain amount of time, an interrupt is generated which informs the operator that the closed loop has been broken.

There are many additional codes that are of the fault-tolerant category. There are many error correcting and error detecting codes that are used within industry. The Hamming code is a good example of one which is covered extensively in the literature. Such codes allow continued correct operation in the presence of solid faults and during transient disturbances. The penalty is that



MODULUS (m)	FRACTION OF WORK CHECKED
3	.66
5	.80
11	.91

FIGURE 23. RESIDUE CODES

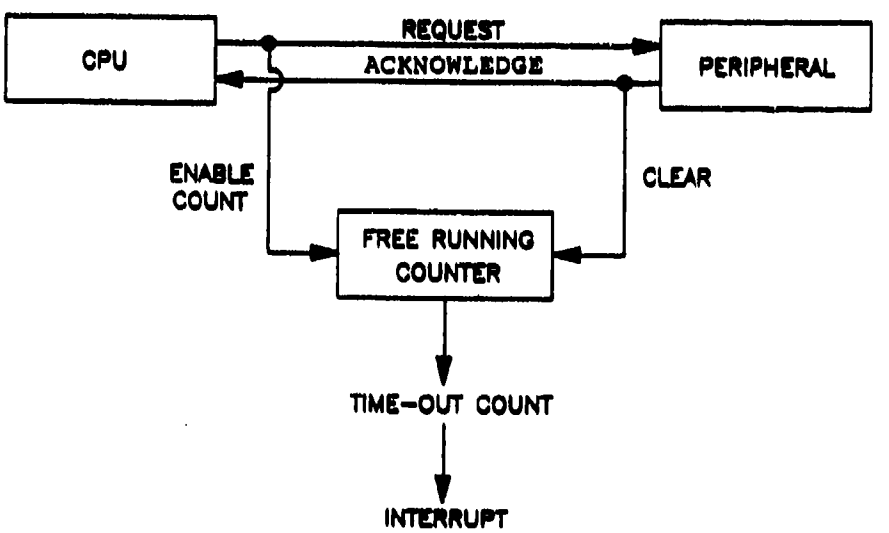


FIGURE 24. WATCHDOG TIMER

additional hardware is necessary in order to provide this error detecting and correcting capability.

It is very difficult to do concurrent testing without the presence of test circuitry in the operational system. If it is an airborne system, for example, the test circuitry must be small and lightweight and yet very powerful. To date, having test capability that is small enough, light enough, and yet powerful enough to do concurrent testing has been extremely difficult. Today there are new techniques which can make this possible.

Standard architectures have been developed that can then be made much smaller and much lighter via the use of ASIC technology. The key is to come up with standard channels of test electronics so that once this reduction in size is effected, it can merely be replicated for however much capability is necessary. One

4.5.2 Fault-Tolerant Design

4.5.2.1 General

For a circuit to be fault tolerant, it must implement some technique for identifying faults and provide some method of generating the correct results despite any faults which have been identified. These design elements are part of fault-tolerant designs at every level of a system including circuits within custom or semi-custom packages (ICs), circuits contained within a module, and circuits implemented on several modules within a system or subsystem.

(FIGURE 25 HAS BEEN WITHDRAWN)

Because fault tolerance utilizes techniques for identifying faults, it is a proper concern of the testability discipline. However, there is a difference in emphasis - a majority of the techniques implemented for the testability discipline have traditionally been for off-line test and on-line test during periods when the system is not being used, while the majority of the techniques implemented for fault-tolerant design are for use during the mission itself. These techniques must be active and effective during the mission.

In addition to the identification of faults, fault-tolerant designs must provide some method of generating the correct result despite the fault. There are three basic methods:

- Provide redundant circuitry
- Provide partially redundant circuitry
- Provide a method of retrying the affected task (for non permanent faults).

When complete redundancy is provided, the circuitry will take the form illustrated in Figure 26 or if the fault identification process is based on a "voting" arrangement, it will take the form illustrated in Figure 27.

In Figure 26, the selection of which output to use is based on the output of the fault identification circuit. This circuit could be placed between the primary circuit and the multiplexer. In that case, it could return processing to the primary circuit if the failure proves transitory. In most cases, the initial failure would make the primary circuit suspect and using the secondary circuit would be preferred. The advantage of the placement of the fault identification circuit shown in Figure 26 is that it can be used to identify the existence of failures in both the primary and secondary circuits. To do this, it must use the fact that a primary circuit failure is remembered for controlling the multiplexer. Thus, all the information is available to determine whether a new failure invalidates the last redundant circuit.

In Figure 27, the fault redundancy is controlled by voting circuits. These circuits in effect compare signals and select the ones which match. The comparison process therefore must yield status information necessary for later repair.

One form of partial redundancy of a circuit is to provide complete redundancy to subsets of that circuit. This form of partial redundancy may be selected to make use of the fact that identification of failures is easier for the subset.

Partial redundancy could also be used to make the circuit more fault tolerant. The approach for doing this is shown in Figure 28. The top diagram of that figure shows a circuit not using partial redundancy. Any of the pairs of failures

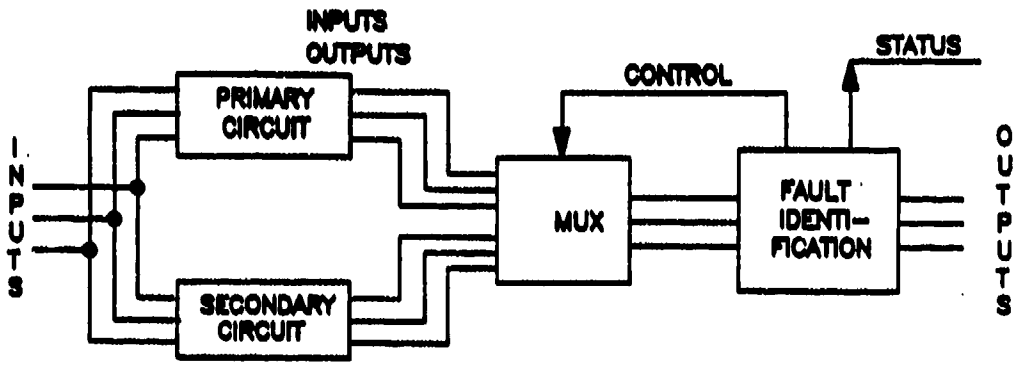


FIGURE 26. EXAMPLE ARCHITECTURE FOR FAULT TOLERANT CIRCUIT

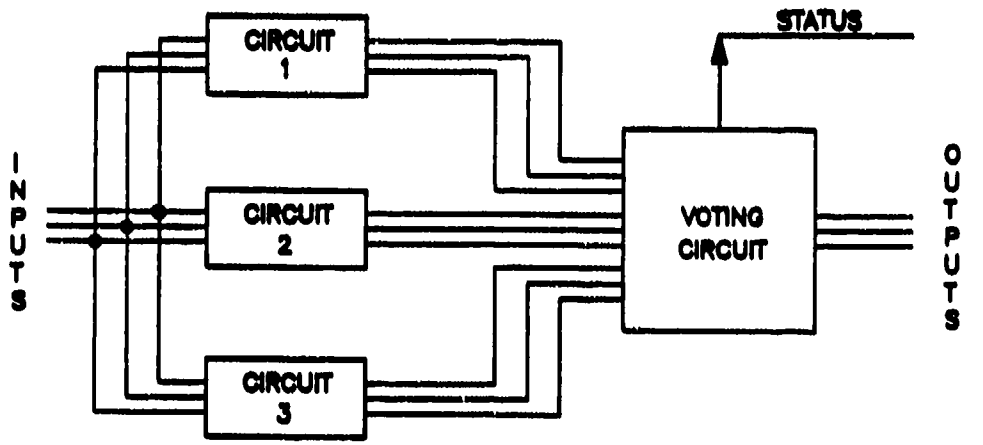


FIGURE 27. EXAMPLE ARCHITECTURE FOR FAULT-TOLERANT CIRCUIT USING VOTING ARRANGEMENT

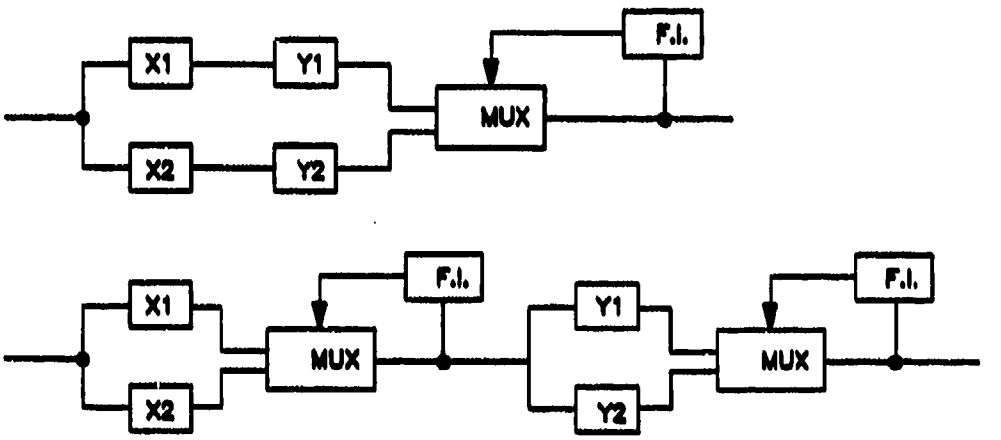


FIGURE 28. EXAMPLE OF TWO FAULT-TOLERANT CIRCUIT ARCHITECTURES

X1-X2, X1-Y2, Y1-X2, Y1-Y2 will cause the complete circuit to fail. In the bottom diagram, only the pairs of failures X1-X2 or Y1-Y2 will cause a complete circuit failure. The result is that more faults are tolerated.

A disadvantage of the partial redundancy approach is that an additional multiplexer (MUX) and Fault Isolation (FI) circuit is required. The reliability of these items must be taken into account in selecting the overall design approach.

Another form of partial redundancy is the circuitry used to support Error Correcting Code (ECC) strategies. This type of partial redundancy is primarily restricted to data transmission and data storage/retrieval circuits. In most such cases, the failures being addressed are "soft." That is, they are produced by random conditions such as noise, radiation, etc., on the circuits or transmission lines involved. Thus, a failure is not in and of itself a reason for circuit repair. However, it is possible for the data from one memory bit to be permanently low. The circuitry

sensing and correcting the errors may compensate for the failure during a mission (circuit is fault tolerant), but it will be necessary to identify the condition for future repair. The simplest solution is to use a binary counter to record the number of failures. The count of failures detected divided by time will give a failure detection rate. A failure detection rate threshold can be used to identify the existence of system faults.

Another method of compensating for "soft" failures in data transmission and data storage retrieval circuitry is to use checksums, parity codes, or residue codes to identify failures and then to retry the operation encompassing the failure. These approaches were discussed in earlier sections. These methods will not provide tolerance for "hard" (permanent) failures and will result in performance degradation for "soft" failures. Thus, they provide only a limited form of fault tolerance.

4.5.2.2 Use of the TM Bus with Fault-Tolerant Circuits

A part of each fault-tolerant circuit is a circuit which recognizes that there is a fault and provides a signal which is used to take corrective action. Because the purpose of fault tolerance is to permit continuation of a mission in the presence of a fault, it is not critical, or even of high priority, that the failure be made known immediately. It is, however, necessary that the fact be recorded so that the faulty item can be replaced or repaired for the next mission. The TM bus is an excellent method to meet this need.

Figure 29 provides an example of how these circuits would fit together. This diagram shows two redundant microprocessors monitored by a watchdog circuit and the use of ECC for both memory and external data transmission. Thus, there are three elements of redundancy and fault tolerance. These elements of fault tolerance are monitored and the results stored in the TM bus controller circuitry for transmission to on-line or off-line test equipment at the convenience of that equipment.

The test information maintained in the TM BUS CONTROL circuitry need not be single bits of data. For example, the watchdog timer could time-stamp a failure event, and separate counts could be generated of errors correctable by the ECC circuitry and errors which were uncorrectable by the ECC circuitry for both the memory and the communications channel. The TM bus could then accept commands for transmitting each piece of data as requested to the testing circuitry.

4.5.3 Applicability to Design Levels

Most of the concurrent techniques discussed are appropriate at various levels from the board/module upward. This is certainly true of parity, residue codes and the watchdog timer. Pin electronics is also appropriate for use with both analog

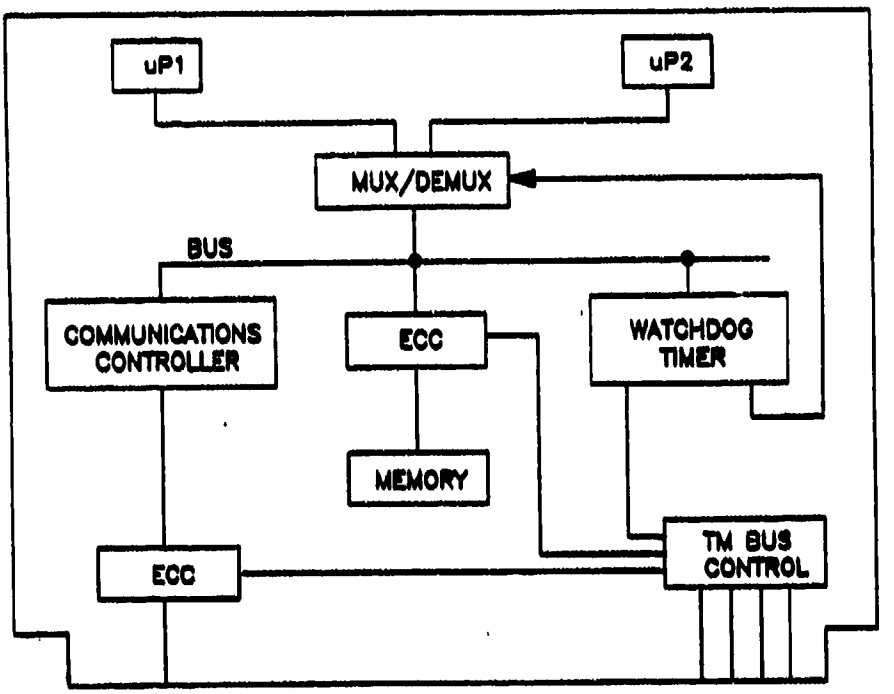


FIGURE 29. EXAMPLE OF TM BUS WITH FAULT-TOLERANT MICROPROCESSOR SYSTEM

and/or digital circuitry at all such levels. Fault-tolerant design is an emerging technique with applicability at all levels from the individual component to an entire system.

5.0 FAULT-ISOLATION TECHNIQUES

Early in the Conceptual Phase of the acquisition process of a weapon system, a maintenance concept is formulated that will sustain the operational availability requirements. This maintenance concept requires a specifically targeted design effort that will advance those diagnostic elements that are outlined in Table 1 - Diagnostic Support Activities. Within the realm of fault isolation to a circuit component or set of components the task at hand is to gain visibility into particular state(s) existing at some predetermined point or node. The physical packaging of the circuit, the type circuit and the maintenance philosophy will impact the techniques selected to gain this visibility. The following paragraphs provide an overview of several techniques that may be employed. All the techniques addressed depend, to some extent, on the ability to force the circuit into a specific state and then provide a test sequence of stimulus and anticipated response. This process, called initialization, must be designed into the circuitry to be examined.

5.1 Test Points

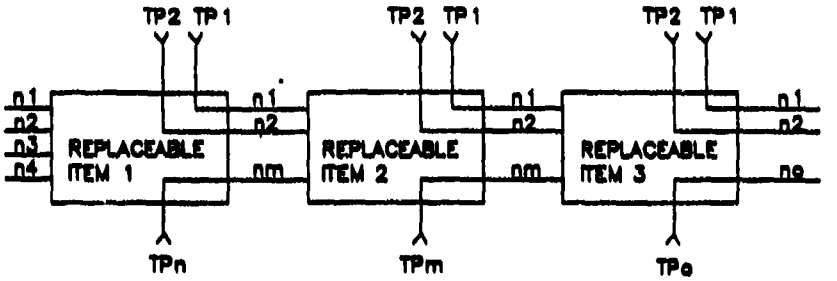
The discussion of test points will focus on the incorporation of these points for the injection of system stimulus and the performing of associated measurement to validate the effect on the circuit of that stimulus, or to verify that the circuit is in a specific quiescent state.

The placement of measurement test points on a module provides information about the internal status of that module. This information is used by either on-line or off-line external test equipment. It is particularly useful for fault isolation in situations where there may otherwise be no way of determining which component caused the failure. Placement of measurement test points may also be a part of the test strategy which is used to screen modules. This strategy is based on partitioning. The circuit is broken into separately tested partitions by stimulus test points and the results read out of the measurement test points.

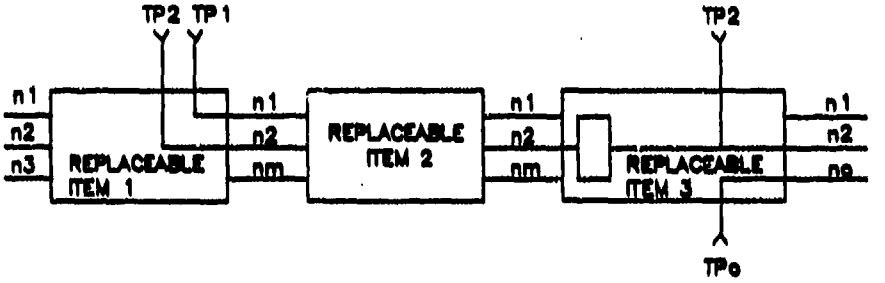
The key issues associated with measurement test points are how to place them to obtain the optimum amount of information and how to protect the circuit from loads and capacitances added by the test points. These issues become particularly critical when there is both a limited number of edge connectors which can be used for test points and when the board area or weight restrictions limit the amount of circuitry which can be used to multiplex test points to the edge connector. The following principles have been found to best address these key issues.

Whenever possible, place a test point at the output of each replaceable item. When this cannot be done, it may not be possible to fault isolate to one replaceable component because an observed failure can be caused by either of two components. A simple case where this is true is shown in Figure 30.

Providing visibility through this approach is particularly important when several packages make up a shift register, comparator, or parity generator/checker. The proper placement of test points for these cases are shown in Figure 31. Another case where test points may be particularly necessary for fault isolation are embedded RAMs and ROMs. The placement of test points in these cases is shown in Figure 32.



A FAILURE CAN BE ISOLATED TO FAILED ITEM DIRECTLY



A FAILURE DETECTED AT TP2 OF REPLACEABLE ITEM 3 CAUSES AN ISOLATION AMBIGUITY BETWEEN REPLACEABLE ITEM 2 AND 3

FIGURE 30. IMPACT OF TEST POINTS LIMITED TO CERTAIN OUTPUTS

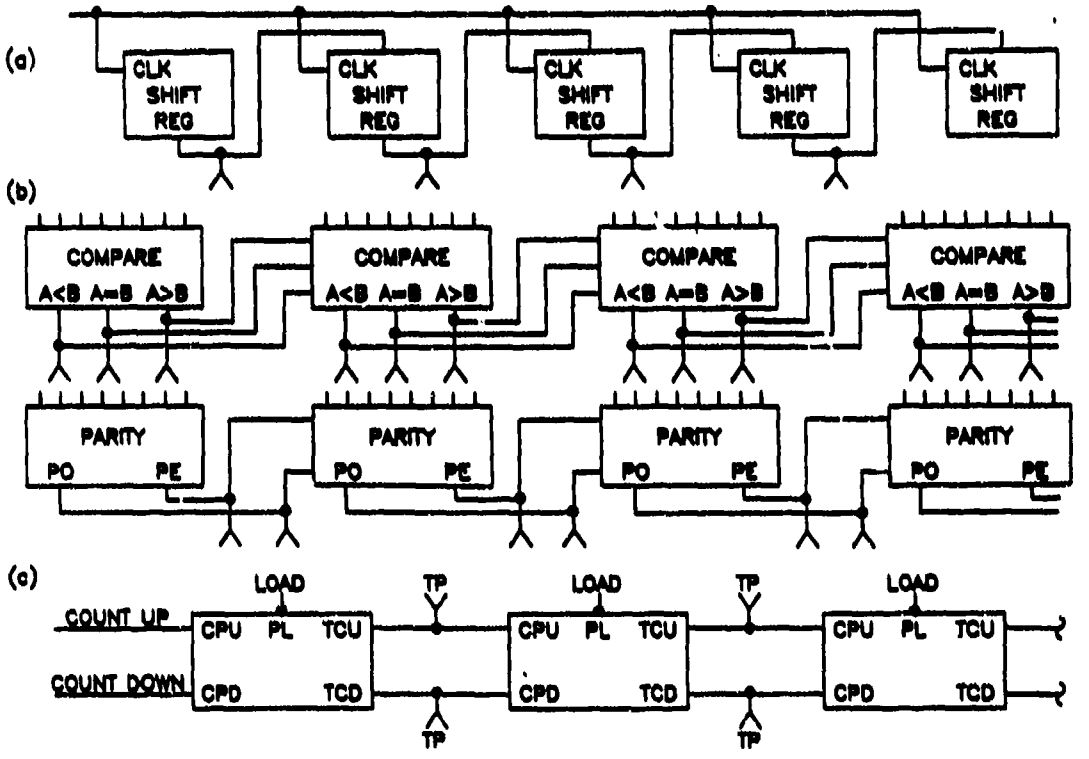


FIGURE 31. EXAMPLES OF APPROACHES TO ACHIEVE TESTABILITY IN MULTIPACKAGE SHIFT-REGISTER, COMPARITOR, PARITY GENERATOR/CHECKER CHAINS

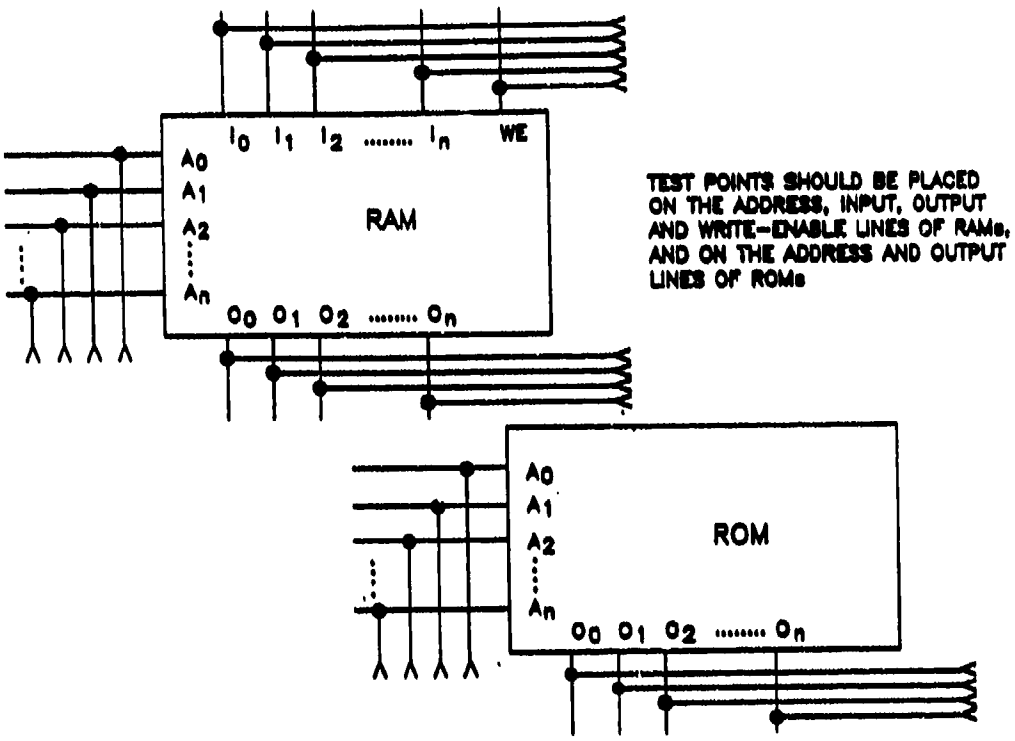


FIGURE 32. ROM/RAM TEST POINTS

Place test points so as to meet fault isolation ambiguity group size requirements. If the ambiguity group size is three components or less, then test points should be placed at least between every third replaceable component of a sequential circuit. An example of this type of placement is shown in Figure 33. The fault-isolation ambiguity group size requirement places a minimum test point count burden on the circuit designer.

Place test points on the controlled side of LED or lamp display circuits. Placing test points on the uncontrolled side of an LED or lamp provides no information because it is tied directly to VCC or ground. The use of this principle in the selection of the proper place to insert a test point is demonstrated in Figure 34. As part of this demonstration, Figure 34 also shows that a test point, TP1, on the base of a transistor, driven by a load to VCC, provides little additional information and should be removed. That particular test point can not be used to determine whether the transistor is operative or not because its value normally reflects whether the transistor is on or off. For the same reason, that test point would not show whether the inputs on the other side of the capacitor were faulty or not unless the capacitor itself failed - an unlikely event.

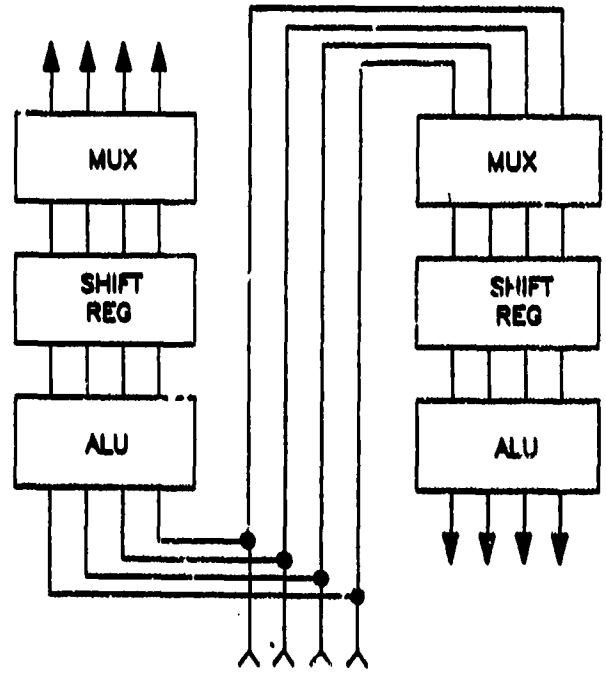
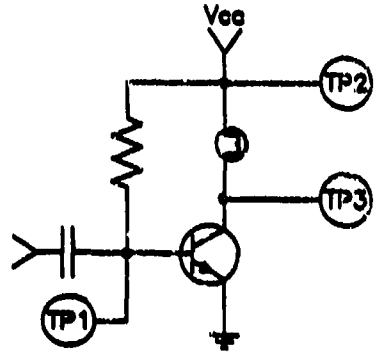
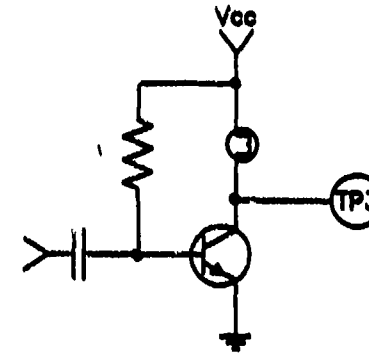


FIGURE 33. LOGIC TEST POINTS FOR MULTIPACKAGE SEQUENTIAL CIRCUITS

NON--PREFERRED METHOD



PREFERRED METHOD



TP3 COULD BE USED TO DETERMINE THE LAMPS STATUS & OPERABILITY OF THE TRANSISTOR

CAPACITOR/RESISTOR HAS LOW PROBABILITY OF FAILURE WHEN COMPARED WITH TRANSISTOR

FIGURE 34. OPTIMUM TEST POINT SELECTION FOR LAMP OR LED CIRCUITS

Provide test points to show the status of redundant circuits. Redundant circuits provide a particular problem for fault isolation since a failure in them may not result in degraded or modified performance. This characteristic makes them attractive for increasing the system or subsystem reliability. Such an improvement, however, is lost if the failure of the redundant part is not observed and corrected. Figure 35 shows an example of how test points could serve this purpose.

Where signals may be sensitive to load or noise, isolate the test point from the external edge connector. Test points should not modify the performance of the circuits to which they are attached. To assure this, heavily loaded or sensitive circuits should be isolated from the test point by resistors or buffers. Examples of these types of circuits appear in Figures 36 and 37.

Where one-shots must be used, provide test points at their output. The timing of one-shots is determined by analog components which do not have the accuracy of synchronous digital circuitry. Such circuitry is usually driven by crystal-controlled clocks. In addition, one-shot timing is fixed and cannot be altered under the control of an external test signal as it has only one stable state, rather than two. This creates a variety of problems in testing logic circuitry that contains one-shots. The larger the number of one-shots, the more difficult the test problem. Therefore, one-shots should be avoided wherever possible. If they are used, test points should be provided at their outputs to allow monitoring of the output pulse duration so that it can be tested for minimum and maximum duration.

Stimulus test points are primarily used to isolate circuit partitions from each other for test purposes, to break feed back loops, and to select test data to be observable at edge points. In effect, they are used to reconfigure inherently untestable circuit elements into forms which can be tested. Thus, they are often the key technique used in making a circuit testable.

As with measurement test points, there frequently are limitations imposed on the number of stimulus test points. Such limitations are imposed by the limited number of edge connector pins available and system requirements which restrict the total board area or weight. Thus the key stimulus test point issues are how to provide the optimum circuit testability with a limited number of test points and how to do it in a way which does not impact circuit performance. The following paragraph addresses the key issues:

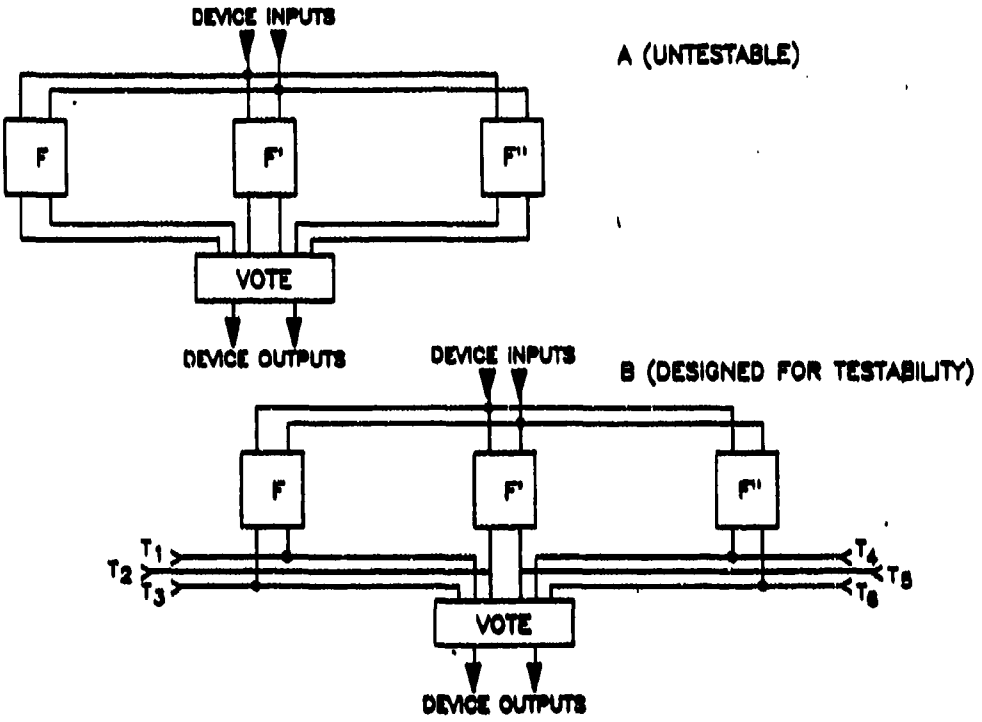
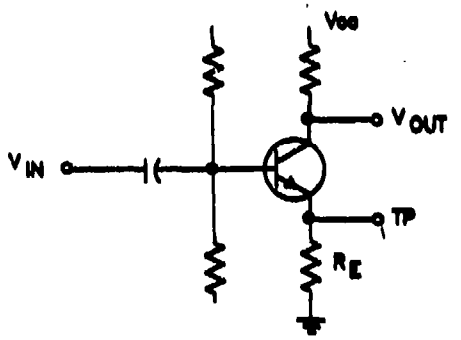


FIGURE 35. TEST POINTS FOR TRIPLE MODULAR REDUNDANT CIRCUITS



COMMON EMITTER AMPLIFIER WITH UNBYPASSED EMITTER RESISTOR

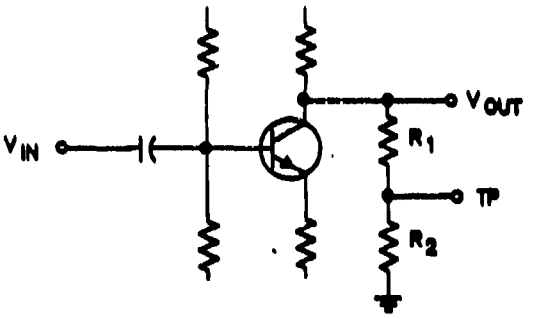


FIGURE 36. ISOLATING THE CIRCUIT FROM A TESTPOINT WITH A RESISTOR

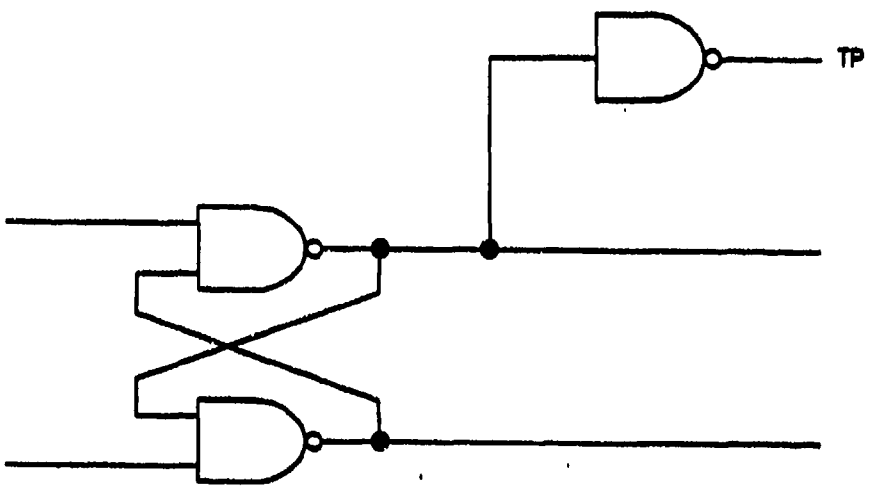


FIGURE 37. ISOLATING THE CIRCUIT FROM A TESTPOINT WITH A BUFFER

Use stimulus test points to isolate free-running oscillators from other circuits. Circuits which cannot be disconnected from free-running oscillators are almost always difficult to test with off-line ATE. This can be caused by several factors, including:

- The test rate required to synchronize the ATE with the oscillator may be higher than the ATE capability.
- The beginning of a test sequence may not be synchronizable with a known circuit state.
- The ATE may require an indeterminate amount of set-up time between segments of a test sequence with the result that the circuit state at the start of the next segment cannot be determined.

For all these reasons, it is important to disconnect the free-running oscillator from the remaining circuits and to provide a method of controlling the actual clock signals with stimulus test points. Figures 38 through 41 show techniques for accomplishing this task.

Provide stimulus test points to force input values which would otherwise require many steps to set. Especially with counter circuits, test program execution may become inordinately long if the entire circuit chain can only be stimulated from one end. In these cases, as illustrated in Figure 42, internal points should be made accessible at locations which the performance of the individual packages (ICs) may be both monitored and have test stimuli injected. As an example, a 3-stage 12-bit counter chain would take 4,096 steps to test from one end, whereas only 16 steps are required for test if the inputs and outputs of the individual packages are accessible.

Provide stimulus test points to open feed-back loops. Feed-back loops in logic circuitry present two testability problems. First, they make it more difficult to generate the test patterns because the feed-back modifies the effect of the injected test patterns. Secondly, they increase the size of the fault-isolation ambiguity group because any fault in the loop propagates to all points in the loop. The solution to the feed-back problem is to break the feed-back loop either through jumper wires on the board edge or by the insertion of additional logic components which allow one to block the feed-back and possibly also to inject additional test signals. Examples of these techniques are shown in Figures 43 through 47.

The additional 2-input AND gate inserted in the circuit of Figure 43 allows one to block the feed-back by placing a logic low on the testpoint, thereby producing a logic low at the input to the left-hand logic block.

In Figure 44, the use of an external jumper to break the feed-back path is shown as one solution which, however, may not be usable at high frequencies because of the added path length. It requires two connector points. An alternate approach which allows one to put a logic high on one of the inputs to the right-hand AND gate is shown in the lower portion of Figure 44. Placing a logic low on the input of the inverter will disable the feed-back path and allow input C to reach the D-type flip-flop. Figures 45 and 46 show similar examples of the use of additional logic to control the feed-back path.

Sometimes the feed-back is not implemented by a single logic line but by (potentially a large number of) parallel lines. In this case, as illustrated in Figure 47, a multiplexer can be used effectively to disable the feed-back path and allow the injection of test signals via pins 2, 3, 4, and 5 of the multiplexer.

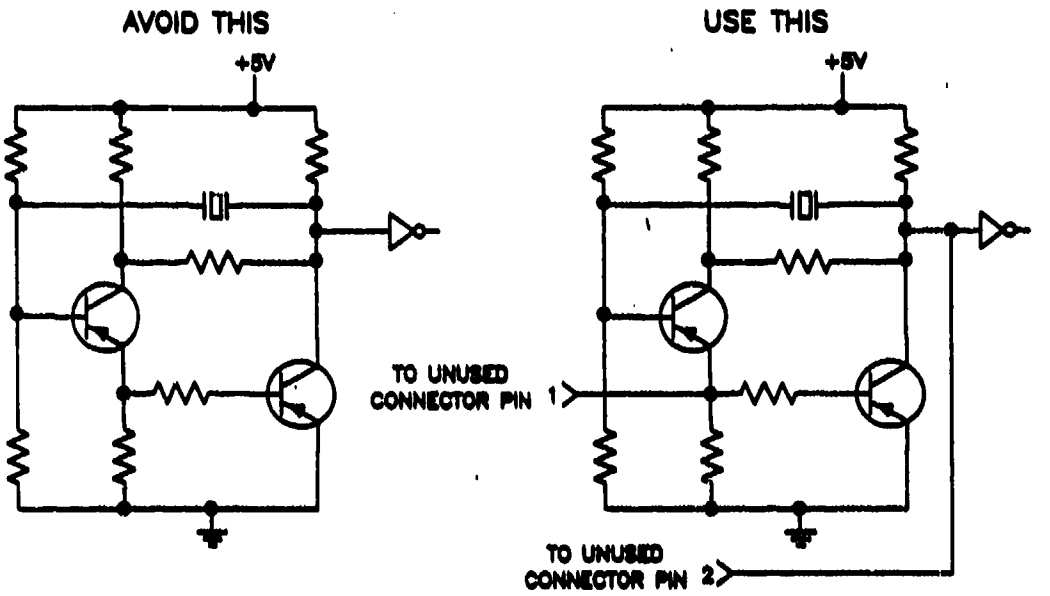


FIGURE 38. CIRCUIT DESIGN APPROACH TO ISOLATE FREE-RUNNING OSCILLATORS

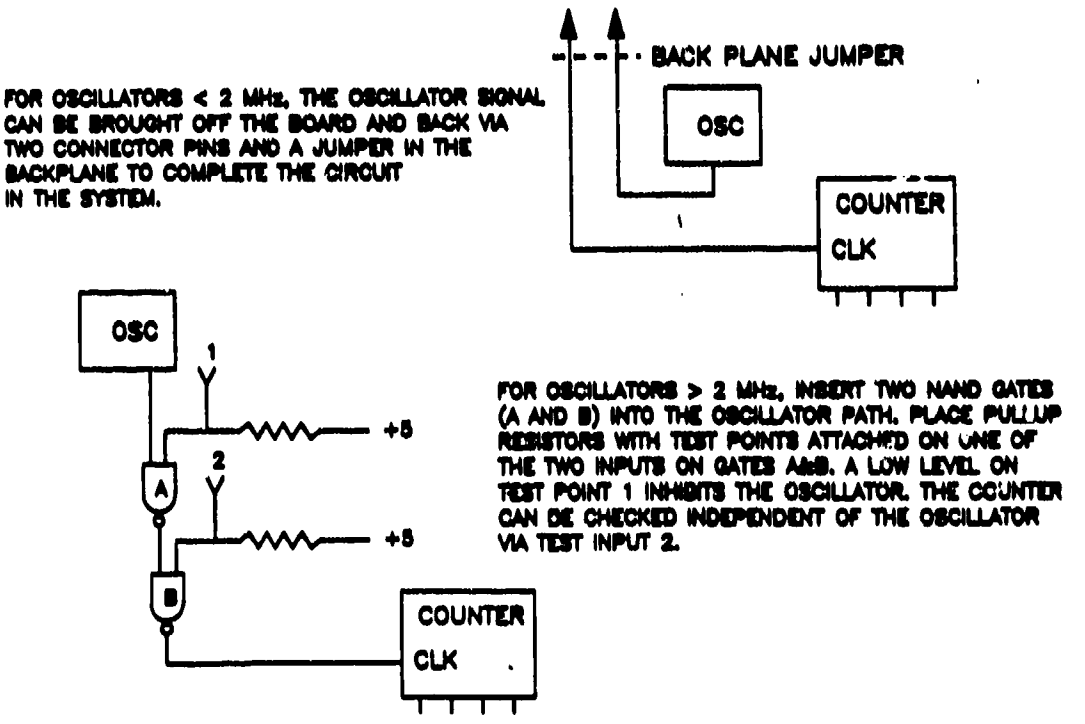


FIGURE 39. TECHNIQUES FOR ISOLATING FREE-RUNNING OSCILLATOR

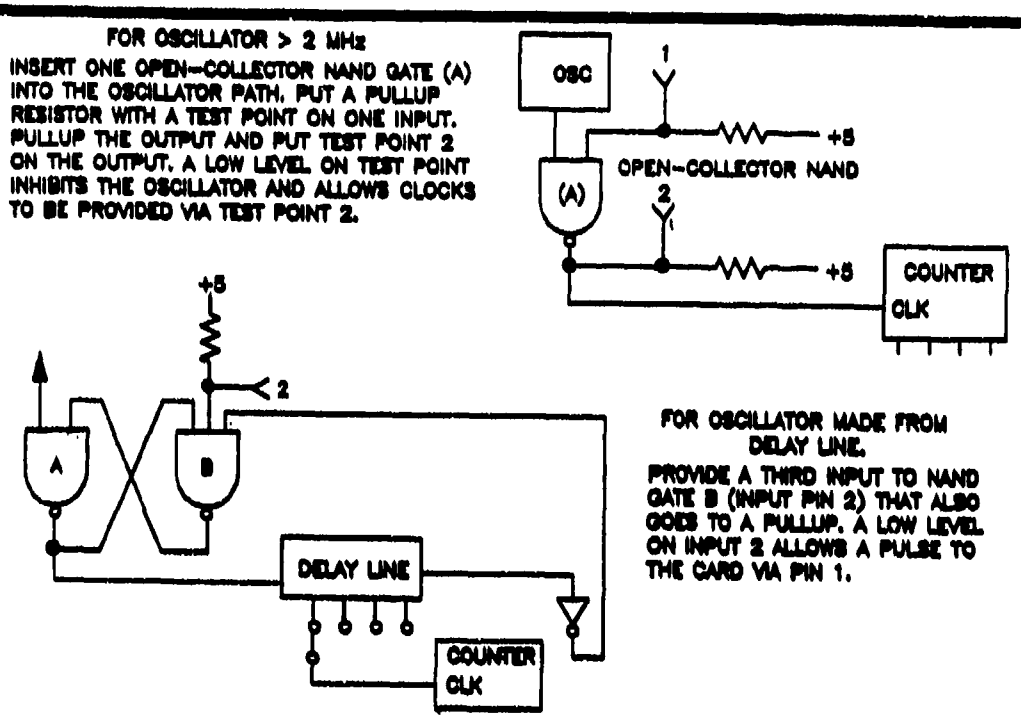


FIGURE 40. DELAY LINE TECHNIQUE OSCILLATOR AND TO SUBSTITUTE EXERNAL CLOCK

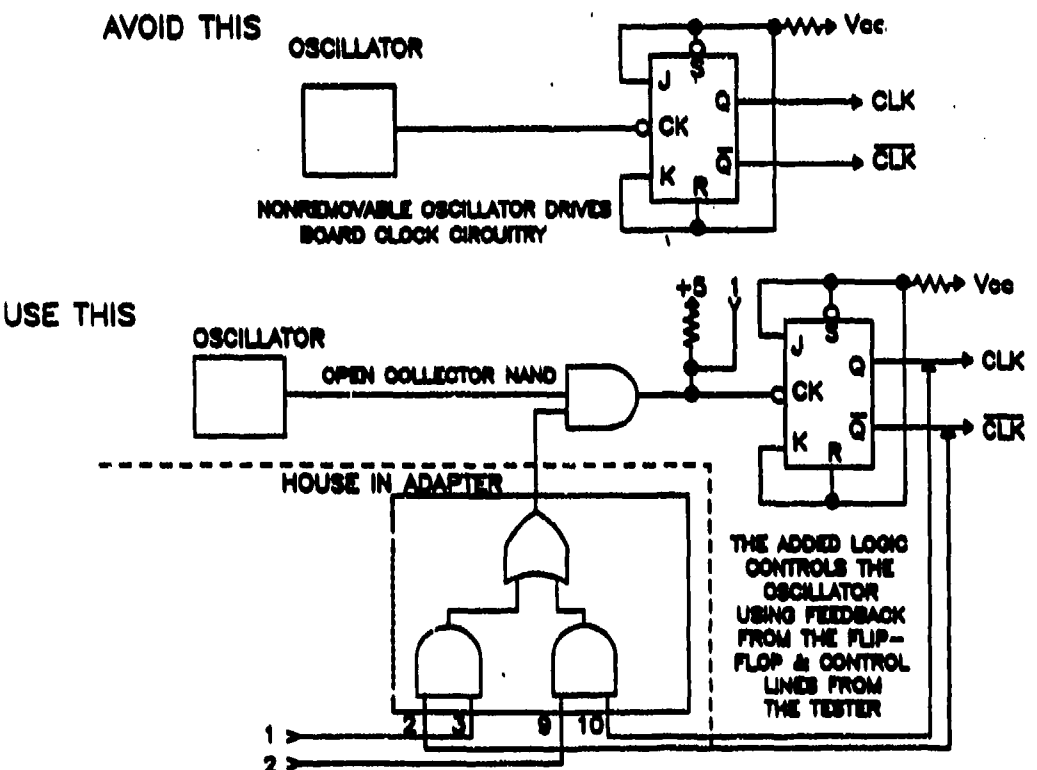
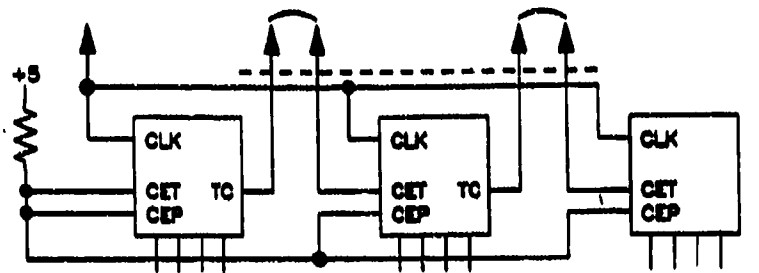
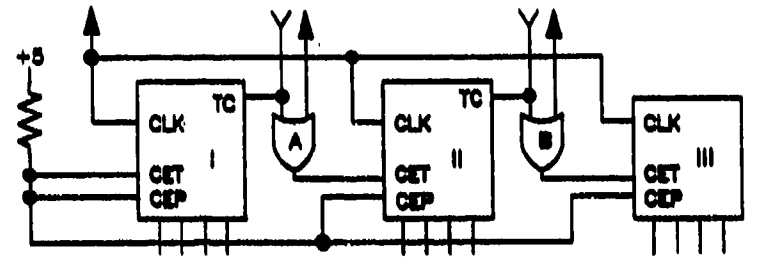


FIGURE 41. TECHNIQUES TO SYNCHRONIZE TWO-RAIL ON-BOARD CLOCK TO TEST EQUIPMENT



BY BRINGING THE TERMINAL COUNTS OFF THE BOARD, BRINGING ANOTHER PIN ONTO THE BOARD AS AN INPUT TO THE CET CONTROL LINES, THE COUNTER STRING IS BROKEN. THESE PINS CAN BE BACKPLANE-JUMPERED IN THE SYSTEM TO COMPLETE THE STRING.



ADD "OR" GATES A AND B BETWEEN THE TERMINAL COUNTS AND CET INPUTS OF THE COUNTERS. BRING THE OTHER INPUT OFF THE BOARD VIA AN UNUSED PIN OR TEST POINT. BY PUTTING A "HI" ON PINS 1 AND 2, YOU ENABLE COUNTERS II AND III TO COUNT AT THE SAME TIME COUNTER 1 DOES. THIS REDUCES THE NUMBER OF CLOCKS REQUIRED TO TEST THE CIRCUIT FROM 4096 TO 16.

FIGURE 42. EXAMPLES OF BREAKING UP COUNTER STRINGS.

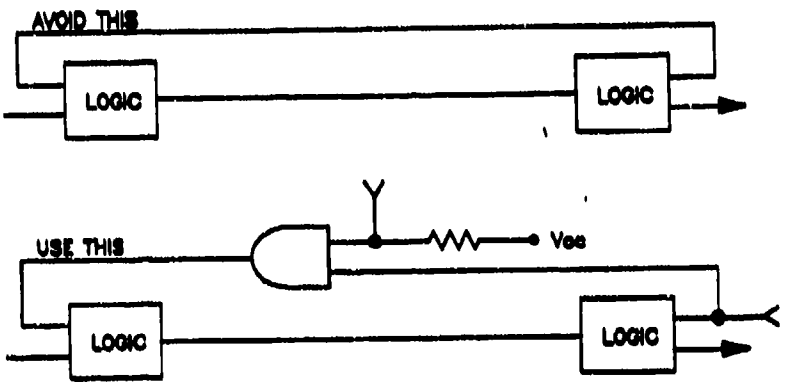


FIGURE 43. EXAMPLE OF INSERTION OF ADDITIONAL LOGIC COMPONENT TO CONTROL FEED-BACK PATH

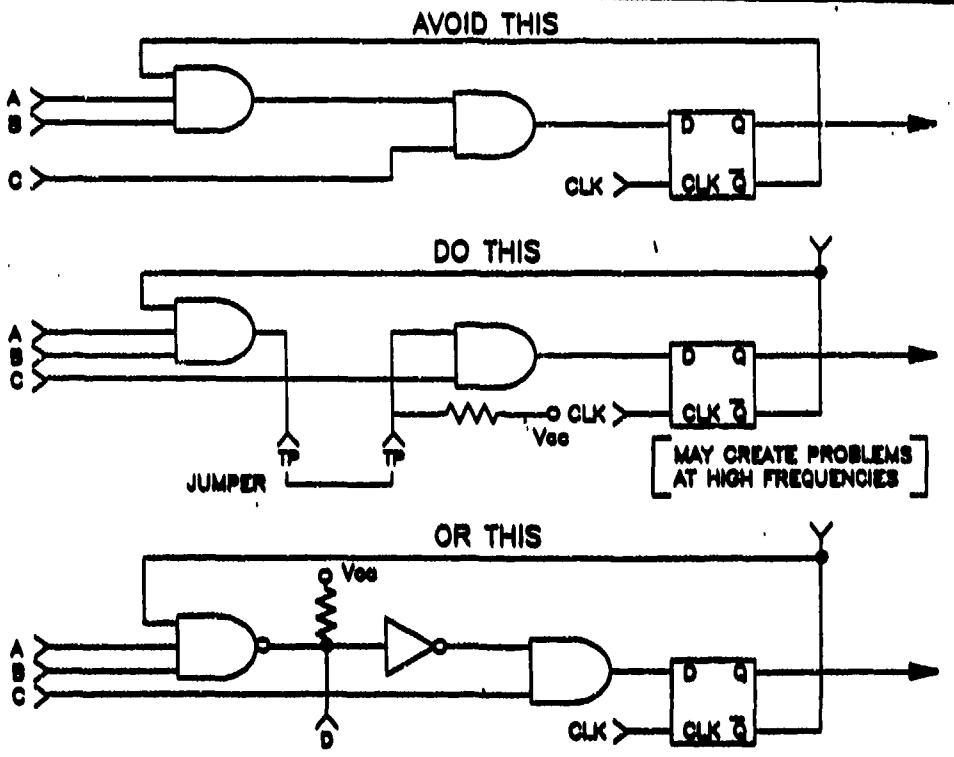
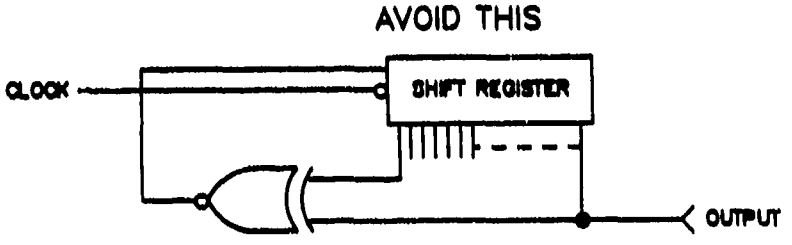


FIGURE 44. EXAMPLES OF DESIGN APPROACHES TO CONTROL FEED-BACK PATHS



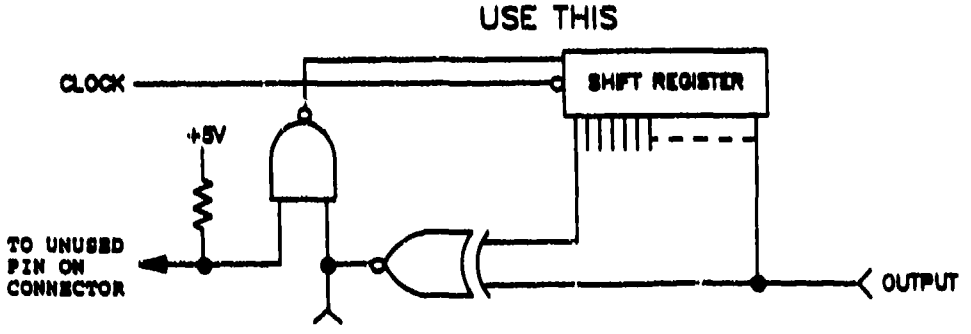


FIGURE 45. ADDITIONAL EXAMPLE OF USING LOGIC COMPONENTS TO PROVIDE FEED-BACK CONTROL

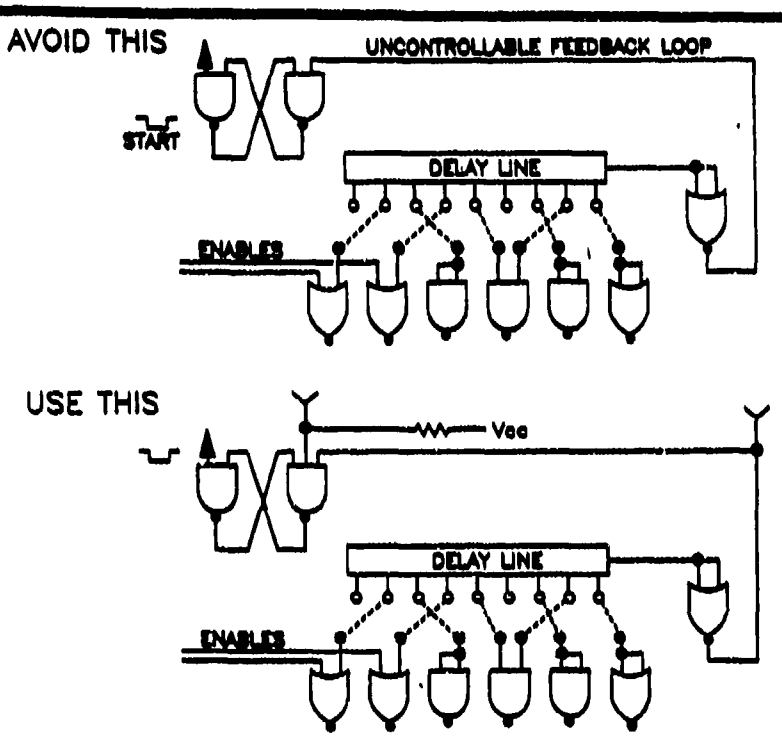


FIGURE 46. ADDITIONAL EXAMPLE OF USING ADDED LOGIC TO CONTROL FEED-BACK PATH

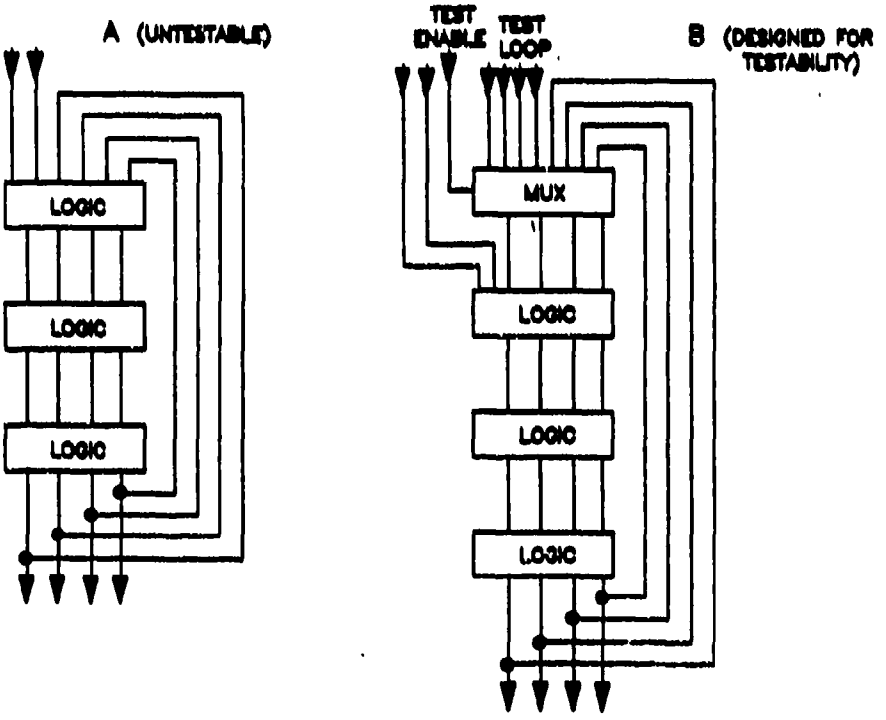


FIGURE 47. EXAMPLE OF BREAKING FEED-BACK LOOPS WITH A MULTIPLEXER

The following list is excerpted from MIL-STD-2165 and is intended to summarize UUT test point selection. The number and placement of such test points is based upon the following:

- o Test points are selected based upon fault-isolation requirements.
- o Test points selected are readily accessible for connection to ATE via system/equipment connectors or test connectors.
- o Test points are chosen so that high voltage and current measurements are consistent with safety requirements.
- o Test point measurements relate to a common equipment ground.
- o Test points are decoupled from the ATE to assure that degradation of equipment performance does not occur as a result of connectors to the ATE.
- o Test points of high voltage or current are physically isolated from test points of low logic level signals.
- o Test points are selected with due consideration for ATE implementation and consistent with reasonable ATE frequency requirements.
- o Test points are chosen to segregate analog and digital circuitry for independent testing.
- o Test points are selected with due consideration for ATE implementation and consistent with reasonable ATE measurement accuracies.

Acceptance testing for modules shall normally be performed using only the module I/O connector, and shall not depend on testing performed previously to assure the module meets its functional and parametric requirements. However, exceptions to this requirement may be granted on a case-by-case basis, as follows:

Test points may be used for acceptance testing on very complex digital modules where the controllability and observability of LSI or VLSI circuits cannot otherwise be achieved to the degree necessary to meet the internal gate-level coverage requirement (usually 95 percent). Test points may be used for acceptance testing on complex analog modules where necessary to test all critical circuit operations.

5.2 Test Header

The test header incorporated in a UUT generally consists of a maintenance-use-only connector which provides the necessary access to test points in the off-line test mode. In all modules, the number of input and output lines which can be dedicated to measurement and stimulus test points is limited. This limitation can be somewhat overcome by using one input-output line for several test points and bringing the line out through the test header. There are seven types of techniques for combining the test point access:

- o Multiplex the measurement test points
- o Convert several test point values into serial data (parallel-to-serial conversion)
- o Compress data taken over several clock steps into a signature (signature analysis)
- o Decode or demultiplex test point stimulus signals
- o Convert serial input data to values on several test points (serial-to-parallel conversion)
- o Use an LSSD or equivalent method
- o Use a digital testability bus.

Most of these design approaches can only be applied to digital circuits and require MSI or larger packages. One or more of these techniques should be used, independently or in combination, in the circuit design of each module.

Test header use is confined to assisting in off-line testing. The header is not to be used in any other application. Any continuous monitoring is accomplished through the use of input and output pins. Many modules will be relatively easy to test and fault isolate, while others may be quite difficult. Therefore, test header usage rules should vary depending upon the needs of the individual module. The following test header usage shall apply to each module, when test connector usage is allowed, and are listed in order of descending preference:

- o Only outputs shall be brought to the test connector to enhance observability. No driving of inputs nor overdriving of outputs shall be performed.

- Only outputs and inputs which do not require critical timing, or are not affected by parasitics (such as tri-state controls) shall be brought to the test connector. No overdriving of active outputs shall be performed.

- Overdriving of active outputs may be performed only as a last resort on the most complex modules, with a test afterwards to verify that the module was not damaged.

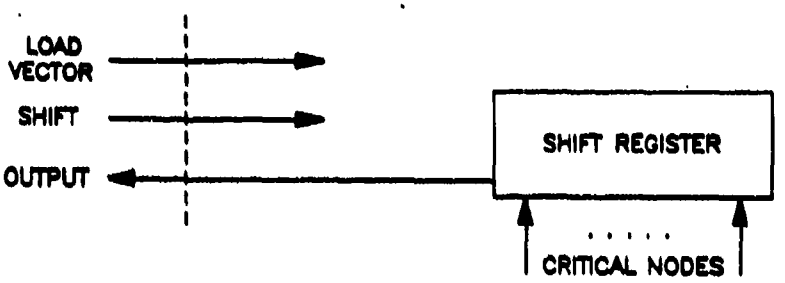
- Physical breaking of signals through the test connector shall not be allowed under any circumstances.

Use of the test header is allowed for all GO/NOGO and fault-isolation testing in the authorized maintenance environment without restriction. However, proper caution should always be exercised so that the test equipment utilized does not present to the header pins an excessive amount of resistive or capacitive loading. Resistive values less than 100 K ohms and capacitance values in excess of 50 picofarads are deemed excessive.

5.3 Increasing I/O Visibility

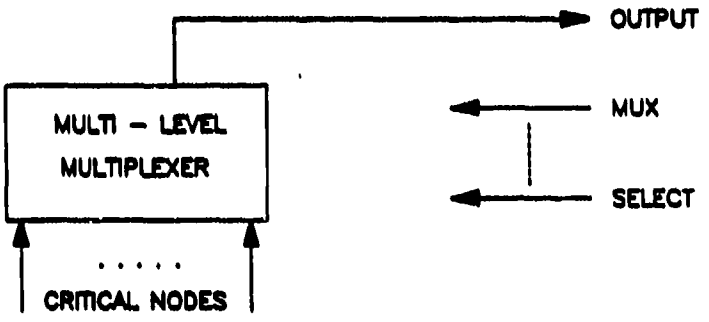
The structured techniques discussed earlier (e.g., scan design) inherently improve the test visibility of the circuitry in question. Less structured techniques are also available for SSI, MSI level designs. Figure 48 shows a shift register used to capture a test vector from multiple critical circuit nodes on a PC board. The number of nodes could be 8, 16, 32 or more depending on the shift register used. The test vector is then shifted serially off the board. Thus, with only a small number of I/O pins (i.e., shift clock, output, etc.) a large test vector readout can be achieved. Since the shift register presents a high input resistance to the nodes and drives off the board via a low output resistance, the electrical situation is optimal. The system penalty is that board real estate must be used to accommodate the shift register chip.

Figure 49 depicts an alternate approach. Here a multiplexer is used to capture critical circuit nodes. By controlling the multiplexer address via MUX select lines on the PC board edge connector, a single node at a time can be selected and read out. The result is the same as with the shift register, many nodes can be accessed with a small penalty in I/O edge pins. Here too the electrical situation is optimal and the penalty to achieve increased visibility similar to the shift register case. Which approach is selected will probably depend on the test philosophy. Is the data usually read out on all nodes during a test cycle or is less than a total read more likely? The shift register is probably preferable in the former case and the MUX in the latter.



- CAPTURES ENTIRE VECTOR

FIGURE 48. IMPROVING OBSERVABILITY WITH LIMITED I/O PINS (SHIFT REGISTERS)



- SELECT ONE NODE AT A TIME

FIGURE 49. IMPROVING OBSERVABILITY WITH LIMITED I/O PINS (MUX)

5.4 Fault Signature/Fault Dictionary

The Fault Signature/Fault Dictionary technique sometimes provides an excellent method for fault isolation of a logic fault within a complex digital device. The application of selected test patterns to the input and a comparison of the output values to the predetermined expected value for a fault-free circuit can identify a failed component provided the fault signature is lodged in the program dictionary. In some cases the fault may be unlisted or isolated to a group of components and require additional application of the guided probe technique to resolve the discrepancy (Refer to Section 5.5).

In order to establish a fault dictionary for a particular circuit card, a logic simulation run is conducted. An inventory of selected faults is established and the test steps where failures are first detected are recorded along with input/output pin states. The pin states that differ from those of a known-good circuit are stored in a dictionary file. Actual failures cause the test system to search the dictionary for a match at the failed test step and matching failed output pins. Since the actual failure mode possibilities are usually too numerous to be simulated, the possibility exists that a fault will occur for which there is no dictionary match. If an unlisted fault does occur, the fault will be detected but may be misdiagnosed. Actually three possibilities exist: correct diagnosis, incorrect diagnosis, and no match.

The problems associated with the standard fault dictionary technique, as described above, can be reduced through the use of dynamic dictionary look-up fault-isolation methods. These methods are implemented by reorganizing the dictionary structure, processing the dictionary in real time and applying the specific algorithms.

5.5 Guided Probe/Clip

Probably the most common fault-isolation technique in use is the guided probe or multi-contact clip that is applied manually to the point under investigation. Accessibility to the point to be tested can sometimes be denied due to circuit layout, modern packaging techniques or the application of conformal coatings. These restrictions are overcome by the careful selection of the best available test point, addition of test points (see Para.5.1) and penetration of the conformal coating.

The overall effectiveness of the guided probe/clip depends upon the circuit testability characteristics, the depth of the software used to direct the operator to the proper test point, and the system program to manipulate the data from the probe to yield accurate fault isolation. One significant disadvantage of the guided probe is the slowness inherent in physically moving the probe numerous times. However, the technique most often does result in the eventual identification of the faulty component. Combinations of fault signature/fault dictionary and guided

probe/clip can often speed up the process via the computer-based attributes of the former technique while still maintaining the accurate isolation properties of the latter.

Node access is needed to fault isolate PCBs by the guided probe technique. To achieve nodal access (measurement and stimulus test points), design consideration and real estate must be provided at the early design stages.

Providing individual probe access to each device pin could require 20-25 percent of the total PCB space and would be a considerable trade-off consideration in the decision of whether to use SMD for a design. Fortunately, a typical digital logical design may have 3 to 4 or more component leads per node. One probe access per node should be sufficient for maintenance diagnostics by guided probe.

One visibility point per node is not enough to provide for isolation of PCB trace opens. However, this failure mechanism is not common for fielded PCBs that have not been mishandled. Fault isolation to the component level, including open input or output connections, is possible with one contact point per node.

It is recommended that contact points be no closer than 50 mil centers. Leads from device pads to test pads should be necked down to a maximum of one half of the solder pad width to provide for thermal isolation during the soldering and de-soldering processes. Examples of probe contact pad layout are shown in Figures 50 and 51.

Physical access to each node may also be obtained by providing test connectors called test headers. The test header provides the additional advantage of making the node available without the need to puncture a PCB's conformal coating. As shown in Figure 52, the PCB area penalty to provide 100 percent access to all device pins might be as much as one third of the total PC area. However, complete nodal access (typically four pins per node) could be provided using one ninth of the PCB space.

Since the guided probe is the predominant fault-isolation technology used at the intermediate and Depot maintenance levels, it is very desirable to provide the ability to physically probe critical nodes as described in Section 5.5. If, however, a determination is made that providing for physical probe points would prevent a system from meeting its program objectives of size and weight, it might be necessary to rely on one or more other fault-isolation or nodal access techniques. The following techniques could be used without nodal access:

No Fault Isolation

In some circumstances, such as when the PCB reliability is very high, and the manufacturing cost is low, it may be desirable to scrap rather than repair defective PCBs.

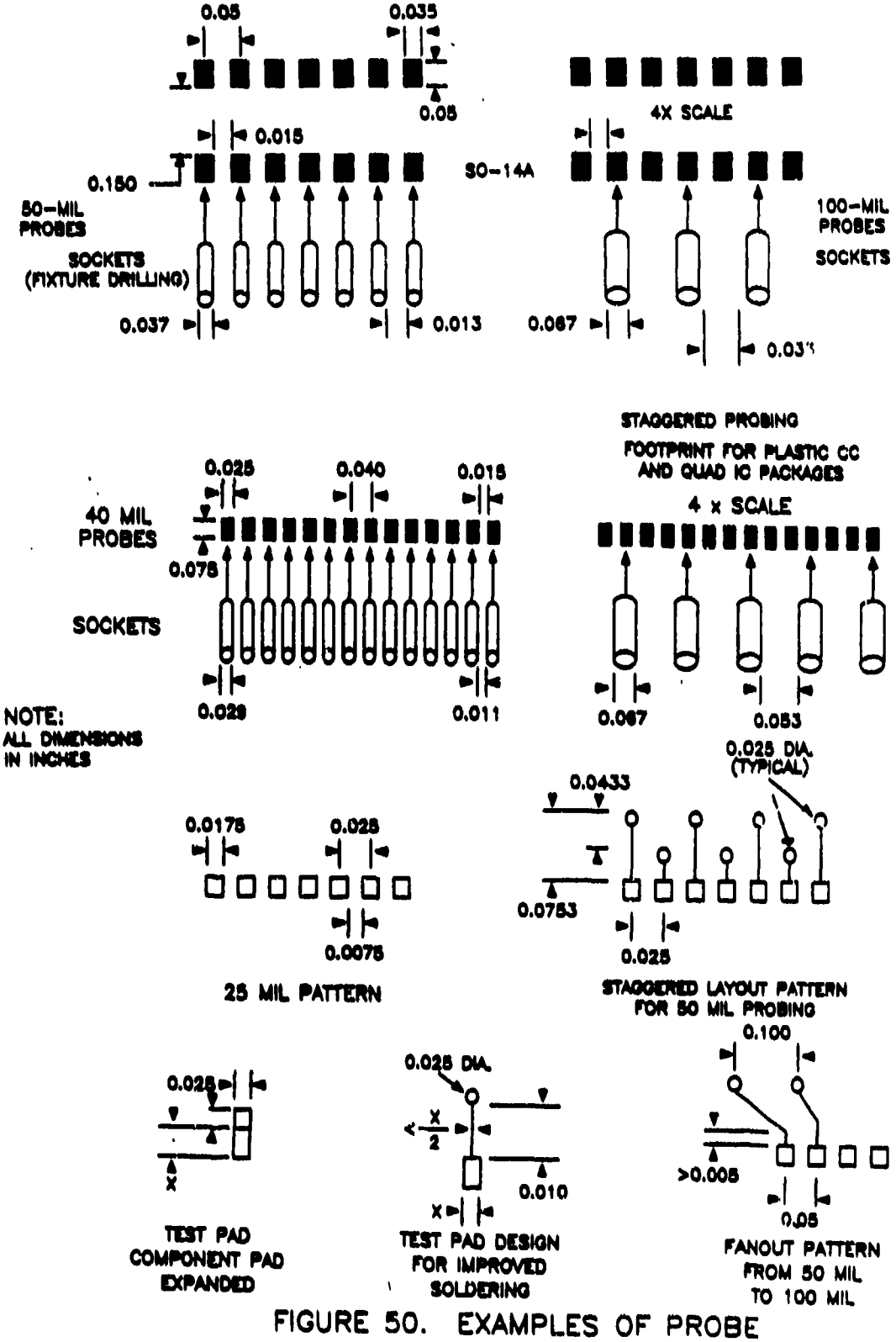
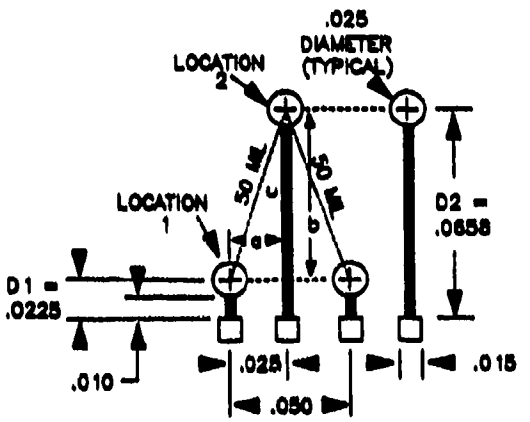
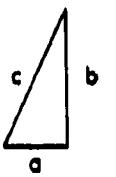


FIGURE 50. EXAMPLES OF PROBE CONTACT PAD LAYOUT



TEST PAD DESIGN FOR
25 MIL LEADLESS DEVICES
USING 50 MIL PROBES

NOTE:
ALL DIMENSIONS
IN INCHES



$$b = \sqrt{c^2 - a^2}$$

$$b = \sqrt{.050^2 - .025^2} = .0433$$

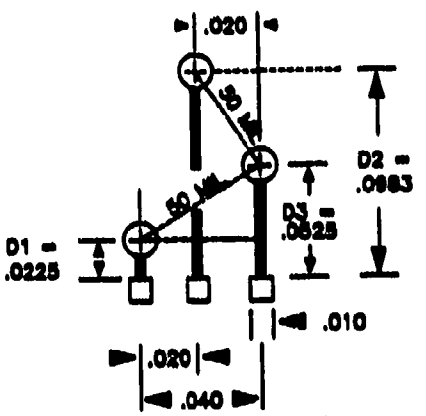
$$D_1 = .010 + \frac{.025}{2}$$

$$D_2 = D_1 + b$$

$$= .010 + \frac{.025}{2} + .043$$

$$D_2 = .0658$$

D_1, D_2 - MINIMUM DISTANCE TO TEST PADS AT STAGGER LOCATIONS ON 25 MIL CENTERS



NOTE:
ALL DIMENSIONS
IN INCHES

$$D_1 = .010 + \frac{.025}{2}$$

$$D_1 = .0225$$

$$D_2 = D_3 + \sqrt{.050^2 - .020^2}$$

$$D_2 = .0525 + .0458$$

$$D_2 = .0983$$

$$D_3 = D_1 + \sqrt{50^2 - 40^2}$$

$$D_3 = .0225 + .030$$

$$D_3 = .0525$$

D_1, D_2, D_3 -
IS THE MINIMUM DISTANCE TO TEST PADS AT STAGGER LOCATIONS 1, 2, 3 ON 20 MIL CENTERS

FIGURE 51. EXAMPLES OF PROBE CONTACT PAD LAYOUT

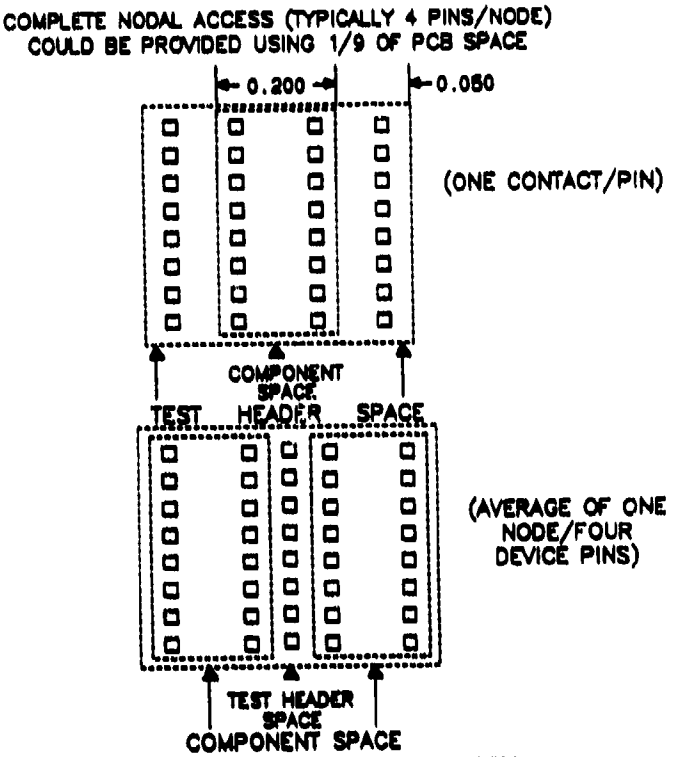


FIGURE 52. PCB AREA

Artificial Intelligence (AI)

AI may be used in combination with other techniques to help determine the most likely cause of the failure.

On-Chip BIT

LSI, VHSIC, and custom gate array devices may have built-in self test capability. A PCB designed exclusively of these devices could control and utilize this test capability to report device failure information. If direct nodal access is not possible,

and the techniques above do not provide the solutions, electrical nodel access can be provided by multiplexing or scan techniques.

5.6 Use of TM Bus

The incorporation of a standard testability bus in the system architecture greatly enhances the capability to fault isolate an assembly to a defective module. The description of this concept and details of its application are contained in Section 7.0.

6.0 PHYSICAL PACKAGING

Automatic electronic assembly technology, currently in use by most military electronic equipment manufacturers, is forcing a definite change to the physical form of the circuit components. The trend toward ICs with high pin counts (above 48) and higher frequency requirements is also dictating a move away from the almost standard Dual In-line Package (DIP) and formed lead components. The advent of VLSI technology requires new and more dense packaging and has accelerated the trend away from conventional PCB layout and assembly.

Numerous new component forms are becoming evident as technology and manufacturing techniques advance. There is a strong trend toward surface mounted devices (SMD) which provide real assembly advantages and superior circuit characteristics at higher frequencies. The move to surface mount assembly is probably the most important factor that will determine which packages are dominant in the industry. There are essentially three distinctive new styles or classes of packaging prominent in the industry to enhance auto assembly and dense packaging strategies. These three are the chip carriers, the small-outline (SO), and the pin grid array (PGA). Of these, the pin grid array is a through-board mounted device.

The following paragraphs will describe the newer packages and address the test related peculiarities.

6.1 Surface Mounted Devices

A surface mounted device or component is a component whose electrical contact to the PCB is on the same side of the board as the component itself. These devices do not require a hole through the PCB. The tighter packing density of surface mount compatible packages is allowing systems manufacturers to meet the function and space demands by reducing PCB real estate by as much as 70 percent. Small packages, called chip carriers, have recently been developed for high-density packaging applications. The benefits of surface mounting, which is the method required for chip carriers, include alleviation of many of the constraints imposed by through-the-board mounting associated with DIPs. For example, one-

sided circuit boards using surface mounted components have a very smooth underside that is well suited to mounting against a heat sink providing even and effective temperature stabilization. Because of their many desirable characteristics, chip carriers are replacing DIPs at all pin counts and the Small-Outline (SO) package is becoming popular for low pin count use.

Surface mounted packages, like chip carriers and small outline packages, are too small to be placed by hand and require automated assembly techniques. Automated assembly techniques have the added advantage of providing more consistent quality, higher production rates, and lower unit assembly cost.

6.2 Chip Carriers

The surface mounted basic chip carrier, as shown in Figure 53, is not a total replacement for through-the-board IC packaging due to imposed limitations on total pin count. For units with over 68 pins a packaging technique called pin grid array is used.

Chip carriers are generally available in four versions:

- o Leadless Ceramic Chip Carrier (LCCC)
- o Leaded Ceramic Chip Carrier (LDCC)
- o Plastic Chip Carrier (PLCC or PCC)
- o Pin Grid Array (PGA)

The term "leadless" indicates that the package has a leadless footprint.

The LCCC is primarily a ceramic substrate with metalized conductors extending from the die-attached cavity to the periphery of the substrate, down the edges and slightly around the underside. Most versions have a die cavity and use normal cavity sealing practices. One of the most common constraints to the LCCC is that the board/package interface is rigid. The thermal expansion of the ceramic is sometimes sufficient to cause the package dimensions to change enough to break the ceramic, or disrupt the electrical connection between the package and the board. Careful mechanical matching techniques can generally alleviate this problem; however, temperature excursions caused by system start up or high performance/high power operation will aggravate the problem.

The LDCC is produced in the same general form as the LCCC except that instead of the conductors being turned under the underside of the board, they are brought out at right angles to the edge as shown in Figure 54.

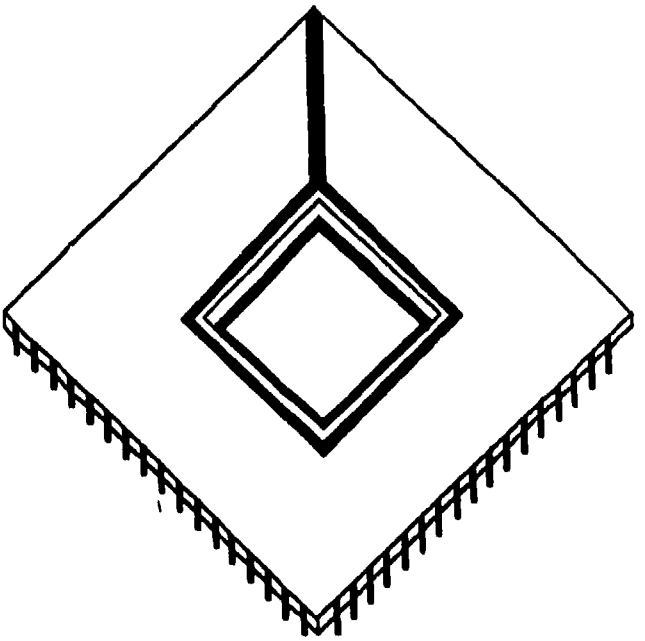


FIGURE 53. SURFACE MOUNTED LEADLESS CHIP CARRIER

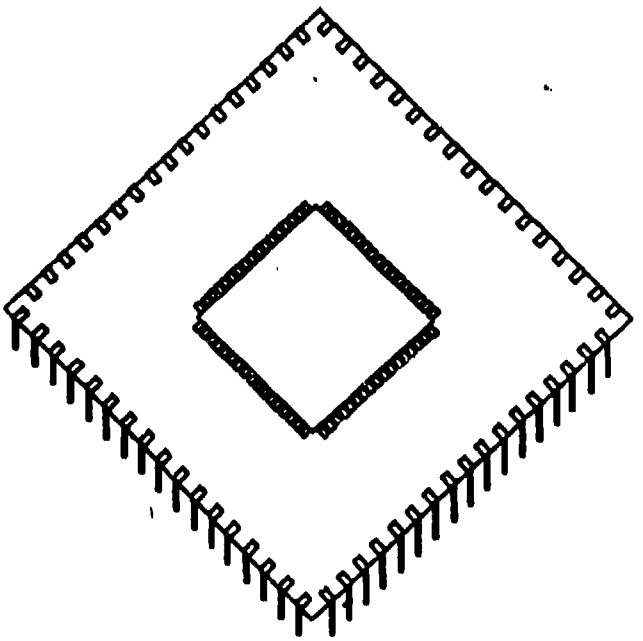


FIGURE 54. SURFACE MOUNTED LEADED CHIP CARRIER

The leads of the LDCC provide the mechanical compliance necessary to permit use of a ceramic chip carrier with an epoxy circuit board.

The plastic chip carrier is constructed using techniques that are virtually identical to the plastic DIP. The principal difference is the positioning of the leads which are turned under the underside to form the "J" lead which effectively produces the same footprint as the LCCC. The PLCC is usually categorized as leadless. Figure 55 provides a summary illustration of JEDEC chip carriers.

6.3 Small Outline (SO) Packages

The second new class of package currently very popular for use where very little PCB real estate is available is the SO package. In appearance it looks like a micro-miniature DIP with leads bent down and out from the body for surface mounting. It occupies approximately one fourth of the surface area of the DIP; however, in configurations of greater than 20 leads or more, it occupies greater space than the PLCC.

The SO is available in 8-, 14-, and 16-pin versions with an 0.150 inch width and in 20-, 24-, and 28-pin versions with an 0.30 inch width. Figure 56 illustrates an SO package.

6.4 Pin Grid Arrays

The third new class of packaging, the pin grid array, is not a true surface mount device although it is similar in outline to the chip carrier. It is a through-the-board mounting device capable of accommodating a large number of pins. Although there are numerous mounting problems, it does provide the best solution for mounting situations requiring more than 124 leads. It is widely used in large main frame computer applications.

The PGA, shown in Figure 57, is normally constructed from a ceramic substrate with metalized conductors from the IC attachment area to an array of pins located on the underside of the package. The pins, used for through-the-board mounting, are spaced in a regular rectangular pattern with 0.10 inch spacing and 0.10 inch pitch.

The dense array of pins creates significant board layout problems and soldering and desoldering problems are prevalent. On the positive side, however, the PGA is compatible with assembly methods used for DIPs, is extremely rugged, is highly area efficient, and can dissipate significant amounts of power (12 watts).

JDEC DESIGNATION	CHARACTERISTICS
LEADLESS TYPE A (MS002)	1. INTENDED FOR SOCKET 2. NOTCHES IDENTIFY AS SOCKET VERSION ONLY 3. SINGLE OR MULTIPLE LAYER CONSTRUCTION 4. USED CAVITY DOWN FOR HEAT DISSIPATION FROM PLANE WITH OR WITHOUT A HEAT SINK
LEADLESS TYPE B (MS003)	1. INTENDED FOR SOCKETS OR DIRECT SOLDER 2. SINGLE OR MULTIPLE LAYER CONSTRUCTION 3. NOTCHED CONFIGURATION FOR SOCKET COMPATIBILITY
LEADLESS TYPE C (MS004)	1. INTENDED FOR DIRECT SOLDER SOCKETING IS NOT RECOMMENDED 2. SINGLE OR MULTIPLE CONSTRUCTION
LEADLESS TYPE D (MS005)	1. INTENDED FOR SOCKET OR DIRECT SOLDER 2. USED CAVITY DOWN FROM PLANE WITH OR WITHOUT A HEAT SINK 3. NOTCHED CONFIGURATION FOR SOCKET COMPATIBILITY
LEADED TYPE A (MS007)	1. INTENTED FOR SOCKETS OR DIRECT SOLDER 2. PREMOLDED AND POST MOLDED VERSIONS
LEADED TYPE B (MS008)	1. LEADLESS TYPE A WITH CLIPS FOR DIRECT SOLDER ATTACH 2. GENERALLY USED CAVITY DOWN WITH HEAT DISSIPATION FROM PLANE WITH AND WITHOUT HEAT SINK 3. SINGLE OR MULTIPLE LAYER CONSTRUCTION

FIGURE 55. CHIP CARRIER SUMMARY

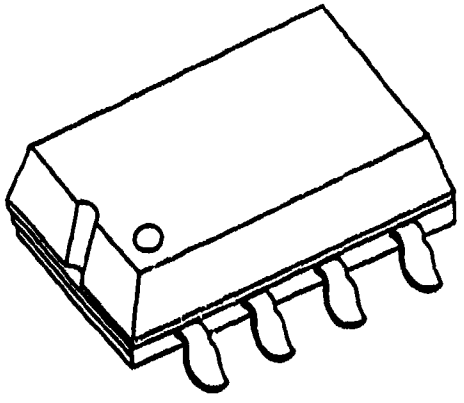


FIGURE 56. SMALL OUTLINE (SO) PACKAGE

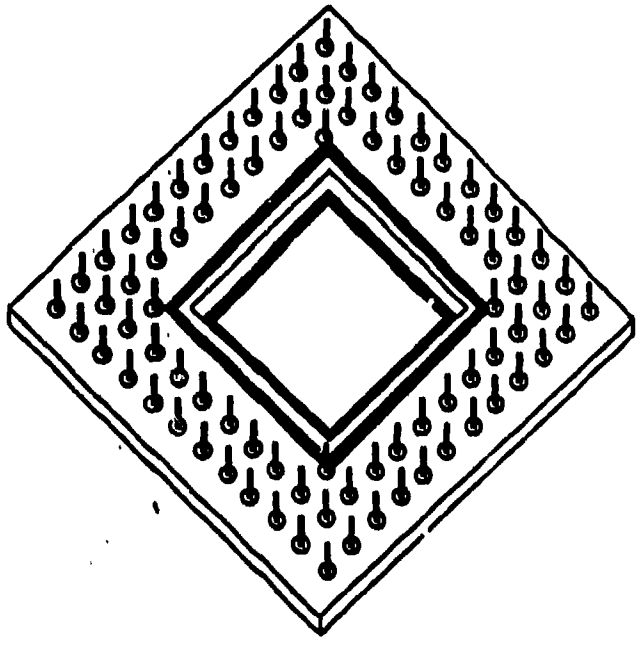


FIGURE 57. PIN GRID ARRAY (PGA)

There is a surface mounted version of the PGA which uses a pad grid array on the underside of the carrier. This unit does have the thermal expansion problems of the LCCC and the inspection of the soldered joint is impossible.

6.5 Packageless Configurations

A fabrication technology wherein the IC device unpackaged is attached to the board with adhesive, and the leads from the die are bonded directly to the board traces, is called chip-on-board packaging. This process in inexpensive applications is especially useful where a high level of protection is not required.

Tape Automated Bonding (TAB) is another assembly technology which is becoming a significant packaging technology. A sandwich of conducting material, usually copper, and a stable dielectric film, such as a polymide, is formed. The film is accurately removed and the beams etched to match the package lead frame at an end and the IC bonding pads on the other. The sandwich is bonded to the IC and attached to the lead frame. A critical advantage of this process for VLSI circuits is the beam separation maintenance by the tape.

In the late 1970s military systems used flat packs for nearly 45 percent of IC requirements. Today, chip carriers are the normal mode in nearly all new military systems design especially in airborne systems, with their need for extreme miniaturization and excellent thermal transfer characteristics. The next generation of avionics systems will likely use chip carriers exclusively. The leadless carrier is not expected to be prevalent due to the mechanical problems associated with its lack of compliance to absorb stresses imposed by thermal shock. The leaded ceramic chip carrier is becoming more and more common and is expected to displace the leadless version. Military systems manufacturers are increasingly using chip carriers to house multi-chip assemblies built in captive hybrid facilities.

The trends in military computers are similar to those noted above but there is more interest in plastic packages. The leaded carrier is being accepted here due to its good thermal conductivity compliant leads and hermetic seals.

6.6 Testing Surface Mounted Devices

With the increased use of SMD PCBs, the Intermediate and Depot repair processes will change to adapt to this new technology. Equipment presently being used to test and repair conventional PCBs may have limited or no capability to handle SMDs.

Surface mount technology causes test problems and, as a result, test techniques will change to accommodate these problems. The predominant fault-isolation technique at the Intermediate shop and Depot level has been the guided

probe. Significant testability enhancements must be made to SMD PCBs to make effective use of the guided probe for fault isolations to an ambiguity group of one at the component level.

For many SMD packages, the actual device contact surface may not be accessible. Additionally, the contacts may be so close or fragile that probing could damage the component, lead, or PCB. One increasingly popular concept is to mount the base IC chip directly on the PCB, either using wire bonds, or a reflow solder technique in which the IC chip pads directly solder to the PCB.

Often with SMT PCBs an individual component may only be replaced once before the PCB may be scrapped. This is caused by the fact that smaller traces and pads cannot endure as much heat as larger pads or traces. Also, more heat must often be applied to remove or install an SMD since all device connections may have to be heated at one time. This requires that the entire repair process be considered when making logistic and testability trade-offs.

7.0 TEST AND MAINTENANCE BUS

7.1 Overview

The TM bus has been discussed in various preceding paragraphs. This section provides additional detail on the VHSIC TM bus although the document entitled "VHSIC Phase 2 Interoperability Standards, TM Bus Specification" should be obtained by anyone desiring the complete and official VHSIC TM Bus Specification. Also discussed in this section are other TM Bus initiatives by the JTAG and IEEE Committees.

7.2 VHSIC TM BUS

A TM bus consists of a set of signal lines that provide a serial path for test and maintenance control and data information. It is a linear, multi-drop communications media which transfers bit serial data between a "Master" module and a number of slave modules residing on a single back plane.

Three contractors - Honeywell, TRW, and IBM - through their VHSIC Phase 2 efforts, have generated a specification for a standard TM bus utilizing only four I/O pins. A review of this specification supports the conclusion that it provides an excellent standard approach accomplishing all benefits cited in Section 3.5 with the least real estate penalties. A brief description of this bus follows. For complete details, refer to the official VHSIC TM Bus Specification.

7.2.1 Physical Requirements

The specification provides for a serial path consisting of four lines between the master and slave modules with capacity for paralleling up to a total of 32 slave modules Each of these lines is dedicated to a particular bus signal. These signals are:

- o Clock
- o Master Data
- o Slave Data
- o Control

Figure 58, TM Bus Signals, depicts the general master-slave module relationship and indicates signal flow. The bus signal characteristics are described in the following paragraphs.

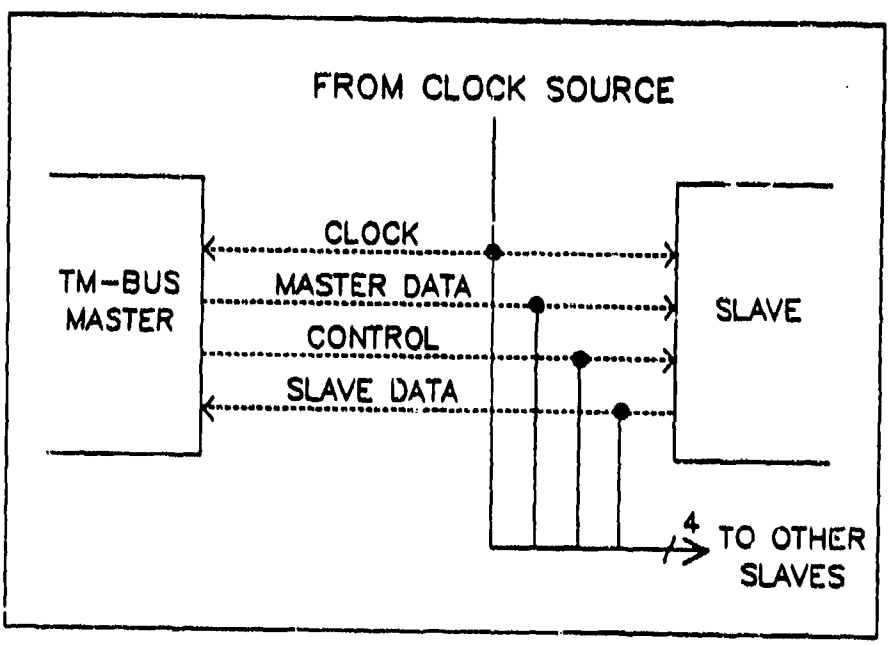


FIGURE 58. TM SIGNALS

7.2.2 Electrical Requirements

As stated previously, there are four signal types that make up the TM bus. All bus signals use negative logic, i.e., the logic '1' state (or asserted state) is the lowest voltage level and the logic '0' state (or released state) is the higher voltage level on the bus.

The TM bus CLOCK signal is a single phase clock. All control and data transfer operations are synchronized with the TM bus CLOCK signal. All data and commands are placed on the TM bus on the high to low transition of the clock and latched-in on the next high to low transition. The CLOCK signal is typically 6.25 MHz and single phase. Specific voltage levels, rise and fall time and duty cycle data are contained in The VHSIC TM Bus Specification.

The TM bus MASTER DATA signal is a single uni-directional line used to transmit device addresses, instruction data, and/or scan data from the MASTER to the SLAVE(s). The MASTER DATA line is also used in conjunction with the CONTROL line to indicate bus states.

The TM bus SLAVE DATA signal is used to transmit acknowledgements, data, and/or interrupts from the SLAVE(s) to the MASTER. The TM bus SLAVE DATA line supports a wired OR configuration.

The TM bus CONTROL signal is a single uni-directional line from the MASTER to the SLAVE(s). When the CONTROL line is asserted, the bus is placed in the DATA TRANSFER state. When the CONTROL is released, the bus is in the PAUSE or IDLE state.

The bus signal lines have a characteristic impedance of between 20 and 50 ohms and are terminated in the Thevenin-equivalent of a terminating resistor of 30 - 40 ohms in series with a voltage source of between +1.9 and +2.1 volts. Specific module requirements for line input capacitance and inductance, AC and DC voltage ranges, timing relationships, etc. can be found in the VHSIC TM Bus Specification.

7.2.3 Data Link Requirements

The TM Bus is the channel for control and data information flow between a maintenance controller and the modules within a system. The module in control of the TM Bus is referred to as the "MASTER". All other modules on the bus are referred to as "SLAVES". The information transferred and the scheduling of data and commands is system dependent and is not addressed in the TM bus Specification. Table 3 - TM Bus Design Parameters and Characteristics - summarizes the TM Bus design.

The master module transmits a header packet and selected data packets. The slave modules respond by acknowledging the command and/or transmitting any data jackets requested. The slave will only transmit data packets on command and will indicate interrupts with an appropriately placed interrupt bit. Complete details of TM bus operation are outlined in the VHSIC TM bus Specification.

7.2.4 Element TM Bus

The TM bus serves at a board to board level. The Element TM bus (ETM) is a 6- wire bus used between chips on a board. Thus, each VHSIC chip has 6 pins allocated for this bus. The functions include a clock, data in, data out, interrupt, mode and select.

TABLE 3

TM Bus Design Parameters and Characteristics

o Performance Characteristics	o Protocol Characteristics
- 4-pin bus signals	- 8 reserved address bits
- Synchronous Operation	- 32 module addresses (maximum)
- Two Data Lines	- 8 sub-addresses per module address
- Module/board compatible	- Multi-drop Configuration
- SLAVE status register	- Interrupt Capability

MISSION
of
Rome Air Development Center

RADC plans and executes research, development, test and selected acquisition programs in support of Command, Control, Communications and Intelligence (C^3I) activities. Technical and engineering support within areas of competence is provided to ESD Program Offices (POs) and other ESD elements to perform effective acquisition of C^3I systems. The areas of technical competence include communications, command and control, battle management information processing, surveillance sensors, intelligence data collection and handling, solid state sciences, electromagnetics, and propagation, and electronic reliability/maintainability and compatibility.