

NASA CR:

150920

T 76-10061

(NASA-CR-150920) SIMULATION VERIFICATION  
 TECHNIQUES STUDY: SIMULATION SELF TEST  
 HARDWARE DESIGN AND TECHNIQUES REPORT  
 (McDonnell-Douglas Astronautics) 487 p HC  
 \$12.50

N76-32856

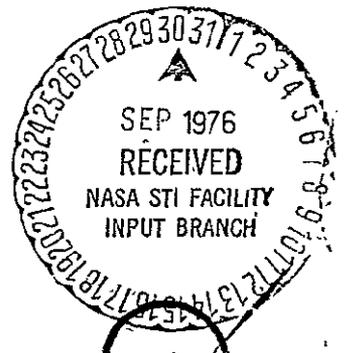
CSCL 09B G3/60

Unclas  
02454

RY COPY

DEC 31 1975

HINSON SPACE CENTER  
HOUSTON TEXAS



**MCDONNELL DOUGLAS**

**CORPORATION**

**SIMULATION VERIFICATION  
TECHNIQUES STUDY**

**SIMULATION SELF TEST HARDWARE  
DESIGN AND TECHNIQUES REPORT**

**1 NOVEMBER 1974**

**MDC E1150**

**PREPARED FOR NATIONAL AERONAUTICS AND SPACE ADMINISTRATION  
JOHNSON SPACE CENTER  
HOUSTON, TEXAS 77058**

**SUBMITTED UNDER CONTRACT NAS 9-13657**

**MCDONNELL DOUGLAS ASTRONAUTICS COMPANY - EAST  
HOUSTON, TEXAS 77058 (713 488-5660)**

TABLE OF CONTENTS

	Page
List of Figures . . . . .	vi
List of Tables. . . . .	xi
List of Symbols . . . . .	xiii
List of Pages . . . . .	xiv
SECTION 1 SUMMARY . . . . .	1-1
SECTION 2 INTRODUCTION. . . . .	2-1
SECTION 3 HARDWARE VERIFICATION TASK OBJECTIVES (TASK 1.0). . . . .	3-1
3.1 Subtask 1.1 Objectives . . . . .	3-1
3.2 Subtask 1.2 Objectives . . . . .	3-1
3.3 Subtask 1.3 Objectives . . . . .	3-1
SECTION 4 GROUND RULES AND ASSUMPTIONS. . . . .	4-1
SECTION 5 DESCRIPTION OF SIMULATOR HARDWARE (SUBTASK 1.1) . . . . .	5-1
5.1 Reference Configuration. . . . .	5-1
5.2 Subsystem Reference Configurations . . . . .	5-3
5.2.1 Host Computer Complex . . . . .	5-3
5.2.2 Minicomputers or Other Ancillary Computers. . . . .	5-4
5.2.3 Crew Station. . . . .	5-4
5.2.4 Instructor/Operator Station . . . . .	5-5
5.2.5 Data Conversion Equipment . . . . .	5-5
5.2.6 Visual Systems Configuration. . . . .	5-7
5.2.7 Motion Base System. . . . .	5-13
5.2.8 Flight Computer and Interface . . . . .	5-13
SECTION 6 DATA SOURCES FOR SURVEY OF CURRENT HARDWARE SELF TEST TECHNIQUES (SUBTASK 1.2). . . . .	6-1
6.1 Simulator Users/Developers . . . . .	6-1
6.1.1 Spaceflight Training Simulators . . . . .	6-2
6.1.2 Military Training Simulators. . . . .	6-4

6.1.3	Commercial Airline Simulators . . . . .	6-6
6.1.4	Engineering Development Simulators . . . . .	6-7
6.2	Bibliographic Data Sources . . . . .	6-8

SECTION 7

	RESULTS OF SURVEY OF CURRENT HARDWARE SELF TEST TECHNIQUES . . . . .	7-1
7.1	Test Design Techniques . . . . .	7-2
7.1.1	Readiness Tests (Fault Detection) . . . . .	7-2
7.1.2	Fault Isolation Techniques . . . . .	7-11
7.1.3	Incipient Fault Detection Techniques . . . . .	7-21
7.2	Test Implementation Techniques . . . . .	7-28
7.2.1	Test Timing Techniques . . . . .	7-28
7.2.2	Test Signal Generation . . . . .	7-30
7.2.3	Test Data Acquisition Techniques . . . . .	7-33
7.3	Test Data Processing . . . . .	7-34
7.3.1	Frequency Response from Sampled Dynamic Test Data (Fourier Transform Techniques) . . . . .	7-35
7.3.2	Frequency Response from Logged Operational Data . . . . .	7-40

SECTION 8

	DEFINITION OF HARDWARE AND SOFTWARE TECHNIQUES FOR SIMULATOR CHECKOUT (SUBTASK 1.3) . . . . .	8-1
8.1	Ancillary Computers and Interface Equipment . . . . .	8-3
8.1.1	Flight Hardware Interface Device . . . . .	8-3
8.1.2	Flight Computer . . . . .	8-34
8.1.3	Other Minicomputers . . . . .	8-38
8.2	Data Conversion Equipment . . . . .	8-40
8.2.1	Data Conversion Equipment Description and Characteristic Parameters . . . . .	8-40
8.2.2	Failure Mode Analysis . . . . .	8-49
8.2.3	Checkout Techniques . . . . .	8-56
8.3	Controls and Displays . . . . .	8-77
8.3.1	Displays Overview . . . . .	8-78
8.3.2	Galvanometers . . . . .	8-88
8.3.3	DC Servo Driven Meters . . . . .	8-100

8.3.4	Synchro/Resolver Driver Meters. . . . .	8-112
8.3.5	Lighted Indicators. . . . .	8-134
8.3.6	Bilevel Electromechanical Flags . . . . .	8-141
8.3.7	CRT Graphics Display. . . . .	8-147
8.3.8	Controls Overview . . . . .	8-153
8.3.9	Toggle Switches . . . . .	8-155
8.3.10	Push Button Switches. . . . .	8-170
8.3.11	Rotary Switches and Potentiometers. . . . .	8-174
8.3.12	Throttles, Hand Controllers, Rudder Pedals. . . . .	8-185
8.3.13	Crew Station/IOS Controls and Displays Self Test Summary . . . . .	8-187
8.4	Visual Simulation Subsystems . . . . .	8-192
8.4.1	Image Transmission System Self Test Techniques. . . . .	8-195
8.4.2	Display Equipment . . . . .	8-217
8.4.3	Image Generation Equipment Self Test Techniques. . . . .	8-223
8.4.4	Visual System Self Test Software Requirements and Data Base Impact. . . . .	8-23
8.5	Motion System. . . . .	8-240
8.5.1	Description . . . . .	8-240
8.5.2	Failure Mode Analysis . . . . .	8-246
8.5.3	Checkout Techniques Evaluated . . . . .	8-246
8.5.4	Recommended Checkout Approach . . . . .	8-258
8.6	Miscellaneous Equipment. . . . .	8-261
8.6.1	Aural Cues System (ACS) . . . . .	8-261
8.6.2	Power Supply System . . . . .	8-267
8.6.3	External Clocks and Timing Equipment. . . . .	8-268

SECTION 9	INTEGRATED SYSTEM SELF TEST SUMMARY . . . . .	9-1
9.1	Simulator Self Test Sequencing . . . . .	9-2
9.2	Simulator Self Test Software Summary . . . . .	9-7
9.2.1	Self Test Executive Software . . . . .	9-9
9.2.2	Subsystem Test Software . . . . .	9-12
9.3	Self Test Hardware Summary . . . . .	9-13
9.4	Data Base Impact . . . . .	9-18

9.5	Self Test System Relative Costs. . . . .	9-20
9.6	Self Test System Impact by Requirements or Design Changes . . . . .	9-23
9.6.1	Sources of Change . . . . .	9-23
9.6.2	Impact of Simulator Changes . . . . .	9-26
SECTION 10	CONCLUSIONS AND RECOMMENDATIONS. . . . .	10-1
SECTION 11	REFERENCES . . . . .	11-1
APPENDIX A	SIMULATOR REFERENCE CONFIGURATION COMPONENT LISTINGS . . . . .	A-1
APPENDIX B	A GLOSSARY OF TERMS OF INTEREST FOR SIMULATOR SELF TEST APPLICATIONS. . . . .	B-1
B.1	Specialized Vocabularies used in Compiling This Glossary . . . . .	B-2
B.2	Definitions of Words and Phrases. . . . .	B-3
B.3	Definitions of Abbreviations and Acronyms . . . . .	B-16
APPENDIX C	SIMULATOR VERIFICATION TECHNIQUES BIBLIOGRAPHY . . . . .	C-1
C.1	Subject Index to Bibliographic Material . . . . .	C-2
C.2	Bibliographic Document Listing. . . . .	C-33

LIST OF FIGURES

NUMBER		PAGE
5.0-1	Reference Simulator Configuration.....	5-2
5.2-1	Simulator Data Conversion Equipment.....	5-8
5.2-2	Minicomputer to Analog/Digital Interface.....	5-9
5.2-3	Typical Digital Output Data Path.....	5-11
5.2-4	Typical Digital Input Data Path.....	5-12
5.2-5	Typical Analog Input Data Path.....	5-12
5.2-7	Synchro/Resolver Output Data Path.....	5-13
5.2-7	Typical Analog Output Data Path.....	5-13
5.2-8	Visual System Block Diagram.....	5-14
5.2-9	DC Analog Servo System.....	5-15
5.2-10	Servo System with Digital Comparison.....	5-15
5.2-11	Synchro/Resolver Servo System.....	5-16
5.2-12	Flight Computer Interface.....	5-19
7.1-1	Step Response of an Underdamped Control System.....	7-8
7.1-2	Transient Response to Step Input (STRSP).....	7-10
7.1-3	Fault Isolation by Transfer Function Derivation.....	7-17
7.1-4	Fault Isolation by Fault Dictionary (FIFD).....	7-20
7.1-5	Degradation Rate Analysis for Incipient Fault Detection.....	7-26
7.3-1	Use of Averaging to Improve Signal-To-Noise Ratio.....	7-39
7.3-2	Illustration of "Aliasing": The High-Frequency Sinusoid looks like its low frequency alias when sampled at too low a rate...	7-39
7.3-3	Leakage Example: Fourier Transform of a Cosine Waveform; Truncation Interval not equal to a Multiple of the Period.....	7-41
7.3-4	Use of "Windowing" to Reduce Leakage.....	7-42
7.3-6	Flow of Noise Injection Analysis.....	7-44
7.3-7	Correlation Processing to Obtain Frequency Response.....	7-45

LIST OF FIGURES (continued)

NUMBER		PAGE
8.1-1	Field External Interfaces.....	8-4
8.1-2	FHID Internal Interfaces.....	8-6
8.1-3	FHID Self Test Software - Host Executed.....	8-22
8.1-4	FHID Self Test Software - Processor Executed.....	8-23
8.1-5	Flight Hardware Interface Test Flow (FITST).....	8-30
8.1-6	Flight Computer Block Diagram.....	8-35
8.2-1	Data Conversion Equipment Configuration.....	8-41
8.2-2	Equivalent Circuit With A Buffer Amplifier Per Input.....	8-42
8.2-3	DCE Analog Test Configuration.....	8-57
8.2-4	Equivalent Circuit With One Buffer Amplifier.....	8-58
8.2-5	DCE Configuration With Crossbar Switching.....	8-61
8.2-6	DCE Configuration With Relay Switching.....	8-63
8.2-7	Discrete DCE Test Configuration.....	8-64
8.2-8	Additional Test Hardware For Bit-0 For Discrete Loop Closure.	8-66
8.2-9	OR'D Input Gates.....	8-67
8.2-10	Readiness And Fault Isolation Test Software Flow, DCETST.....	8-73
8.3-1	Typical Silicon Phototransistors And Reflective Sensors (Spectronics Inc.).....	8-81
8.3-2	Example Of A Current Sensing Circuit.....	8-86
8.3-3	Typical Galvanometer Case, Circuit And Equation Of Motion....	8-89
8.3-4	Sample Photocell Layout And Circuit.....	8-94
8.3-5	Galvanometer Test Software.....	8-96
8.3-6	Typical Servo Driven Instrument.....	8-101
8.3-7	Typical Instrument Electrical Diagram.....	8-102
8.3-8	DC Servo Driven Meter.....	8-103
8.3-9	Tape Meter Construction.....	8-104
8.3-10	Tape Meter Analog Servo System Modified For Automatic Checkout	8-105
8.3-11	Servo Meter Self Test Software Flow, SERTST.....	8-111
8.3-12	Face Of Attitude Director Indicator.....	8-114
8.3-13	Typical Rate/Error Pointer Circuit.....	8-114
8.3-14	Typical Circuit Of One Axis Of Attitude Ball.....	8-115
8.3-15	Horizontal Situation Indicator.....	8-118

LIST OF FIGURES (continued)

NUMBER		PAGE
8.3-16	DC Analog Servo System As Used For Glidescope, Deviation Bar, And Compass Card.....	8-119
8.3-17	Solenoid Operated To/From Indicator.....	8-119
8.3-18	Technique For Measuring Current Through DME Seven-Element Digital Readout Assemblies.....	8-120
8.3-19	Heading Select And Course Select Electrical Diagram.....	8-120
8.3-20	ADI Calibration Test Software Flow, ADIST.....	8-131
8.3-21	ADI Dynamic Test Software Flow, ADIDT.....	8-132
8.3-22	HSI Test Software Flow, HSITST.....	8-133
8.3-23	Lighted Indicators And Push Button Switches.....	8-135
8.3-25	Current Sensing Diagram For AC Circuits.....	8-136
8.3-26	Current Sensing Diagram For Push Button Switch.....	8-137
8.3-27	Lamp Life And Candlepower Vs. Applied Voltage From Chicago Miniature Lampworks.....	8-137
8.3-28	Software Functional Flow Diagram.....	8-140
8.3-29	Assembly Of Bilevel, Electromechanical Flag.....	8-142
8.3-30	Schematic Of Electromechanical Flag.....	8-143
8.3-31	Display System Functional Block Diagram (ADAGE/400).....	8-148
8.3-32	Mechanical Data For 2TLI Switches.....	8-156
8.3-33	Resistances Of Bilevel Input Line.....	8-159
8.3-34	Switch Checkout By A Comparator.....	8-162
8.3-35	Software Functional Flow Diagram.....	8-163
8.3-36	Comparator Circuit For Component Switching Pole.....	8-163
8.3-37	Direct Measurement Of Voltage Drop Across Switch Contacts...	8-164
8.3-38	Modified Cylinder Actuator For Pair Of Toggle Switches.....	8-167
8.3-39	Moveable Overlays To Operate Toggle.....	8-168
8.3-40	Selected Vendor Data On Microswitch Series 2 Pushbutton Switch.....	8-172

LIST OF FIGURES(continued).

NUMBER		PAGE
8.3-41	Software Functional Flow Diagram.....	8-173
8.3-42	Typical Commercial Rotary Switches.....	8-175
8.3-43	Potentiometer Checkout Technique.....	8-177
8.3-44	Comparison Of Contact Voltage Drop.....	8-179
8.3-45	Software Functional Flow Diagram.....	8-180
8.3-46	Comparator Circuit For Component Switching Pole.....	8-182
8.3-47	Direct Measurement Of Voltage Drop Across Switch Contacts.....	8-183
8.3-48	Crew Station Test Software Flow, CSTST.....	8-189
8.4-1	Shuttle Orbiter Crew Station Fields Of View.....	8-192
8.4-2	Simulator TV System Schematic.....	8-196
8.4-3	Typical RGB System Schematic.....	8-198
8.4-4	Schematic Diagram Of Articulated Optical Probe (Simulates Spacecrafts Three Angular Degrees Of Freedom).....	8-199
8.4-5	An Optical Interface Of Optical Probe And RGB Camera.....	8-199
8.4-6	Characteristic Parameters For TV Readiness And Fault Isolation Tests.....	8-201
8.4-7	One Horizontal Line Of TV Test Signal(Typical).....	8-203
8.4-8	TV Test Signals And Their Images On The Display.....	8-205
8.4-9	Gamma Degradation As Apparent From A Stair Step Test Signal.	8-209
8.4-10	Tektronix Digital Processing Oscilloscope System.....	8-213
8.4-11	TV Self Test Hardware Schematic.....	8-216
8.4-12	Shadow Mask Color Kinescope Details.....	8-218
8.4-13	Reflective Non Pupil Forming Infinity Display System.....	8-218
8.4-14	Relative Capabilities Of Alternate Display Scanning Concepts	8-221
8.4-15	Earth Scene Generation - Basic Components.....	8-224
8.4-16	Servo System With Digital Comparison.....	8-226
8.4-17	Synchro/Resolver Servo System.....	8-227
8.4-18	Visual Subsystems Test Software, VISTST.....	8-232
8.4-19	Video Equipment Test Software, VIDTST.....	8-235
8.4-20	Scene Generation Equipment Self Test Software Flow, GENTST..	8-238

LIST OF FIGURES ( CONTINUED )

NUMBER		PAGE
8.5-1	Motion Base Geometry.....	8-241
8.5-2	Motion Base Hydraulic Schematic.....	8-243
8.5-3	Static Checkout Module.....	8-254
8.5-4	Step Input Checkout Module.....	8-255
8.5-5	Frequency Response Checkout Module.....	8-256
8.5-6	Master Sequencing Program.....	8-257
8.6-1	Functional Diagram Of Aural Cue Synthesizer.....	8-262
8.6-2	Aural Cue Speaker Locations.....	8-264
9.1-1	Readiness Test Elements and Execution Sequence.....	9-3
9.1-2	Daily Self Test Sequence.....	9-6
9.2-1	Simulator Self Test Software Tree.....	9-8
9.2-2	Self Test Executive Software Flow.....	9-10

LIST OF TABLES

NUMBER		PAGE
6.1-1	Simulation Verification Techniques Study Simulation Facility Survey Summary.....	6-3
6.1-2	Existing FDS Checkout Software.....	6-5
7.1-1	Typical Fault Dictionary for a Simple RC Network .....	7-18
8.1-1	List Of FHID Line Replaceable Units.....	8-7
8.1-2	Characteristic Parameters Of FHID LRU's.....	8-11
8.1-3	Failure Modes Of FHID LRU's.....	8-15
8.2-1	DCE LRU Characteristic Parameters.....	8-44
8.2-2	Possible LRU Failure And Internal Symptom.....	8-50
8.2-3	Characteristics Of Candidate Checkout Techniques.....	8-69
8.3-1	Analog Meters.....	8-90
8.3-2	Failure Modes.....	8-90
8.3-3	Characteristics Of Candidate Techniques.....	8-92
8.3-4	DC Servo Driven Meters.....	8-103
8.3-5	Tape Meters.....	8-105
8.3-6	Failure Modes Of Servo Meters.....	8-107
8.3-7	Failure Modes Of Tape Meters.....	8-107
8.3-8	Characteristics Of Candidate Techniques For Tape Meters.....	8-109
8.3-9	Characteristics Of Candidate Techniques For Servo Meters.....	8-110
8.3-10	Attitude Director Indicator.....	8-117
8.3-11	Functional Description Of HSI Elements.....	8-118
8.3-12	ADI Peculiar Failure Modes.....	8-122
8.3-13	Failure Modes Of The HSI.....	8-123
8.3-14	Characteristics Of Candidate Techniques For ADI Checkout.....	8-124
8.3-15	Characteristics Of Candidate Techniques For HSI Checkout.....	8-125
8.3-16	Lighted Indicators.....	8-138
8.3-17	Bilevel, Electromechanical Flags.....	8-142
8.3-18	Failure Modes.....	8-143
8.3-19	Characteristics Of Candidate Techniques.....	8-145
8.3-20	Toggle Switches.....	8-157
8.3-21	Failure Modes.....	8-160
8.3-22	Characteristics Of Candidate Techniques.....	8-169

LIST OF TABLES ( continued )

NUMBER		PAGE
8.3-23	Pushbutton Switches.....	8-171
8.3-24	Rotary Switches.....	8-174
8.3-25	Characteristics Of Candidate Techniques.....	8-184
8.3-26	Controller Motion Sensors.....	8-186
8.3-27	Crew Station Test Software Data Base Impact.....	8-191
8.4-1	Failure Modes For The Globe System.....	8-228
8.5-1	Parameter Characteristics.....	8-244
8.5-2	Failure Modes.....	8-247
8.5-3	Checkout Test Technique Which Will Evaluate Specific Malfunction Mode.....	8-248
8.5-4	Characteristics Of Candidate Checkout Testing Techniques....	8-249
8.5-5	Typical Hydraulic Motion Base Instrumentation.....	8-259
8.5-6	Incipient Fault Detection-Hydraulic Motion Base.....	8-260
8.6-1	Aural Cues System Characteristic Parameters.....	8-265
9.3-1	Hardware Added for Self Test.....	9-14
9.3-2	Additional DCE Channel Requirements for Self Test.....	9-15
9.5-1	Relative Cost Factors of Self Test System.....	9-21
9.6-1	Impact of Simulator Changes.....	9-27

## LIST OF SYMBOLS

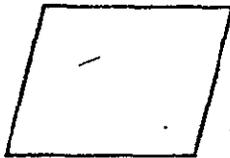
### 1. FLOW CHART SYMBOLS

Flow charts shown in this report conform to American National Standard X3.5-1970, "Flow Chart Symbols and Their Usage in Information Processing".

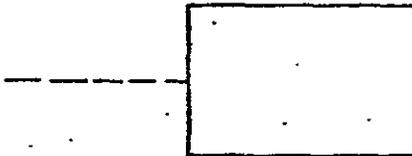
Symbols of interest are defined below.



Process: Computation, storage move, etc.



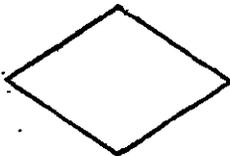
Input/Output



Comment; annotation



On-line storage (e.g., disk)



Decision



Predefined process, subroutine

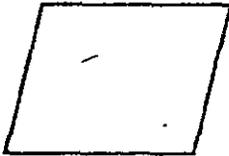
## LIST OF SYMBOLS

### 1. FLOW CHART SYMBOLS

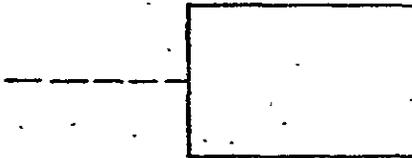
Flow charts shown in this report conform to American National Standard X3.5-1970, "Flow Chart Symbols and Their Usage in Information Processing". Symbols of interest are defined below.



Process: Computation, storage move, etc



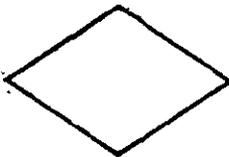
Input/Output



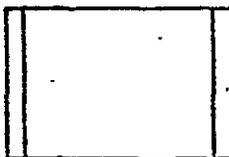
Comment, annotation



On-line storage (e.g., disk)

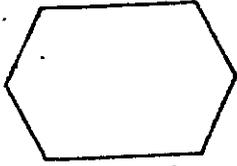


Decision

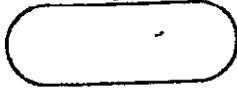


Predefined process, subroutine

LIST OF SYMBOLS (continued)



Preparation, initialization



Program terminus: begin, end, return, stop, etc.



Flowline connector



Parallel mode: multitasking, databus

LIST OF PAGES

Title Page  
ii to xiv  
1-1 to 1-2  
2-1  
3-1  
4-1 to 4-2  
5-1 to 5-16  
6-1 to 6-8  
7-1 to 7-47  
8-1 to 8-268  
9-1 to 9-27  
10-1 to  
11-1 to 11-3  
A-1 to A-12  
B-1 to B-18  
C-1 to C-66

## SECTION 1

### SUMMARY

This report presents the final results of the Hardware Verification Task, Task 1.0, of the Simulation Verification Techniques Study. As noted in Section 3, the objective of this task was to define software and hardware techniques for checking the operational status of the various subsystems anticipated in future NASA training simulators. This Task was implemented by the execution of three basic subtasks consisting of the definition of expected simulation hardware, Subtask 1.1, a survey of current self test techniques, Subtask 1.2, and the definition of specific techniques applicable to the various simulator subsystems, Subtask 1.3. This report presents the results of all three subtasks.

Section 3 reviews the basic objectives of the various subtasks while Section 4 reviews the groundrules under which the overall task was conducted and which impacted the approach taken in deriving techniques for hardware self test. The results of the first subtask, the definition of simulation hardware are presented in Section 5. The hardware definition is based primarily on a brief review of the simulator configurations anticipated for the Shuttle training program. However, these subsystems virtually encompass the full range of simulator subsystems currently being implemented for modern training simulators.

The results of the survey of current self test techniques, Subtask 1.2, are presented in Sections 5 and 6. Section 5 reviews the data sources that were considered in the search for current techniques while Section 6 presents the results of the survey highly condensed and structured in terms of the specific types of tests that are of interest for training simulator applications. Specifically, these types of tests are readiness tests, fault isolation tests and incipient fault detection techniques. The most applicable techniques have been structured into software flows that are then referenced in subsequent discussions of techniques for specific subsystems.

The results of the third and major subtask, the definition of hardware and software techniques, Subtask 1.3, are presented in Section 8. In this section each of the major simulator subsystems is addressed specifically and the techniques for implementing the self test of these subsystems are treated. The

simulator subsystems consist of the ancillary digital computers and their interfaces, the data conversion equipment, controls and displays for the crew station and the instructor operator stations, visual simulation subsystems, the motion base subsystem and several items of miscellaneous equipment.

The integration of the various subsystem tests is addressed in Section 9. This section primarily reviews the scope of the hardware and software requirements if all of the subsystem tests defined in Section 8 are implemented and also considers the type of software structure or executive that is required for fully automating these tests. Finally the overall aspects of the software requirements in terms of data base impact and the sensitivity of the self test systems to simulator changes are discussed.

The conclusions and recommendations derived from the considerable effort expended in the execution of the Hardware Verification Techniques Task are presented in Section 10. In addition to the references listed in Section 11, an extensive bibliographic listing is included in the appendices.

## SECTION 2

## INTRODUCTION

This report presents software and hardware techniques for implementing simulator hardware self test as defined during the Simulation Verification Techniques Study being conducted for NASA's Johnson Space Center under Contract NAS9-13657. This study is being performed by McDonnell Douglas Astronautics Company - East at its McDonnell Douglas Technical Services Company - Houston Division. Keith L. Jordan, Simulation Development Branch, FSD, is Technical Monitor of the contract for the NASA.

The Simulation Verification Techniques Study is one of a number of studies recently conducted by NASA-JSC in support of the development of training and procedures-development simulators for the Space Shuttle Program. The other studies have included the following: Shuttle Vehicle Simulation Requirements Study, NAS9-12836; Space Shuttle Visual Simulation System Design Study, NAS9-12651, performed by McDonnell Douglas Electronics Company; Development of Simulation Computer Complex, NAS9-12882; and Crew Procedures Development Techniques, NAS9-13660. The last two of these were performed by McDonnell Douglas Astronautics Company - East.

The present study, the Simulation Verification Techniques Study, consists of two major Tasks. These are the Hardware Verification Task and the Performance Verification Task. The present report is the final report for the Hardware Verification Task. This report summarizes the techniques found useful for simulator hardware verification and describes the software and hardware provisions that are required for implementing these techniques for the various simulator subsystems.

## SECTION 3

### HARDWARE VERIFICATION TASK OBJECTIVES

The objective of Hardware Verification Task was to define software and hardware techniques for checking the operational status of all simulator equipment anticipated in future NASA simulators. These techniques will determine the state of equipment readiness, identify incipient failures, and provide for fault isolation to a repairable or replaceable piece of equipment. This Task required the definition of anticipated simulator hardware, the survey of current self test techniques, the definition of parameters characterizing simulator subsystems and replaceable components, the definition of techniques for acquiring data representing these parameters, the definition of processing requirements for determining hardware status, and the collective integration of data acquisition and processing requirements into the context of a simulator configuration.

#### 3.1 SUBTASK 1.1 - DEFINITION OF SIMULATION HARDWARE

The objective of this subtask is to survey current spacecraft and the current Shuttle vehicle, as well as state-of-the-art spacecraft and aircraft simulations to determine the hardware that may be required for implementation of future spacecraft, real-time simulations.

#### SUBTASK 1.2 - SURVEY OF CURRENT HARDWARE SELF TEST TECHNIQUES

The objective of this subtask was to survey current and near-future self test methods and practices in order to establish what can be accomplished within the current state-of-the-art.

#### SUBTASK 1.3 - DEFINITION OF HARDWARE AND SOFTWARE TECHNIQUES FOR SIMULATOR CHECKOUT

The objective of this subtask was to define software and hardware techniques for checking the operational status of all simulator equipment. In addition, the techniques developed shall provide for fault isolation to a piece of repairable or replaceable equipment. This task requires the definition of parameters characterizing the outputs of subsystems and components; the definition of anticipated failure modes for components and subsystems; the development of means for sensing the performance parameters and transmitting them back to a computer for processing; and the definition of processing requirements for verifying proper performance as well as for isolating faults to a defective unit.

## SECTION 4

### GROUNDRULES AND ASSUMPTIONS

The Simulator Verification Techniques study was conducted in accordance with the groundrules established by the contract statement of work. This section discusses those groundrules and their significance in carrying out the study effort. Also discussed in this section are some assumptions that were made in the course of the study and which were necessary to concentrate our effort in developing a self test concept.

The study groundrules were listed in the study statement of work as follows:

1. The design of hardware or software used for checkout shall be such as not to endanger personnel or equipment by its use or through malfunction.
2. Malfunction of the self-check hardware shall not hinder normal operation of the simulation.
3. The design of hardware or software used for checkout shall minimize human intervention.
4. The checkout techniques shall make maximum use of the simulation and its computer to perform the checkout.
5. The checkout and verification techniques shall minimize the requirements on computer resources.
6. The checkout time shall be minimized.

Safety of personnel and equipment is a primary consideration in the design of aerospace systems. Addition of self test capabilities to these systems must, therefore, be consistent with safety requirements of the basic system; in particular, special care must be exercised in the design of self test system elements so that a failure in any of these elements would not endanger training or operating personnel, or the basic equipment or facility being used.

The main objective of incorporating a self test capability in a simulator is to increase availability of the system. Availability is a function of reliability and mean time to repair. If the addition of self test elements degrades

reliability or maintainability of the system, the objective of self test is defeated. Therefore, self test elements added must be made a part of the system in such a way that a failure of any of these elements would not interfere with basic system operation; also, mechanical and functional integration should not interfere with accessibility of operational components in any way that time to repair would be increased.

In keeping with the objective of increasing system availability by addition of a self test capability, it is mandatory that the amount of time required to exercise this capability be held to a minimum. Manual activities generally to be longer times than automated activities. Wherever possible, therefore, the self test process should be carried out by automatic means that minimize human operator intervention.

The hardware and software resources available for command and control functions in a simulator are quite extensive and powerful. These resources are in heavy demand during complex simulation sequences but are relatively idle at other times. In order to minimize design, development, and manufacturing cost, it is desirable that no self test oriented hardware or software elements be added to the system when an already existing element of the basic simulator can perform a similar verification task. Consequently, use of the simulator host computer and auxiliary computers has been assumed for test control and execution functions.

Programming and operation of a large Host Computer, such as the type of equipment envisioned for Shuttle Simulators, are quite costly in terms of man-hours and equipment required. During simulator development, as well as during operational phases, computer time availability is likely to be limited either because of software development requirements or training requirements. The software structures considered for test software have been generalized and modularized to minimize both the development and operational demands for computer resources.

The time required for hardware checkout is a critical factor in determining how often the tests of the hardware can be exercised. We have assumed the adequacy of end to end readiness tests for daily, pre-run testing and have structured the software for timely implementation of these types of tests. This has separated out the operations required for fault isolation, when necessary, as well as incipient fault detection.

## SECTION 5

### DESCRIPTION OF SIMULATOR HARDWARE (SUBTASK 1.1)

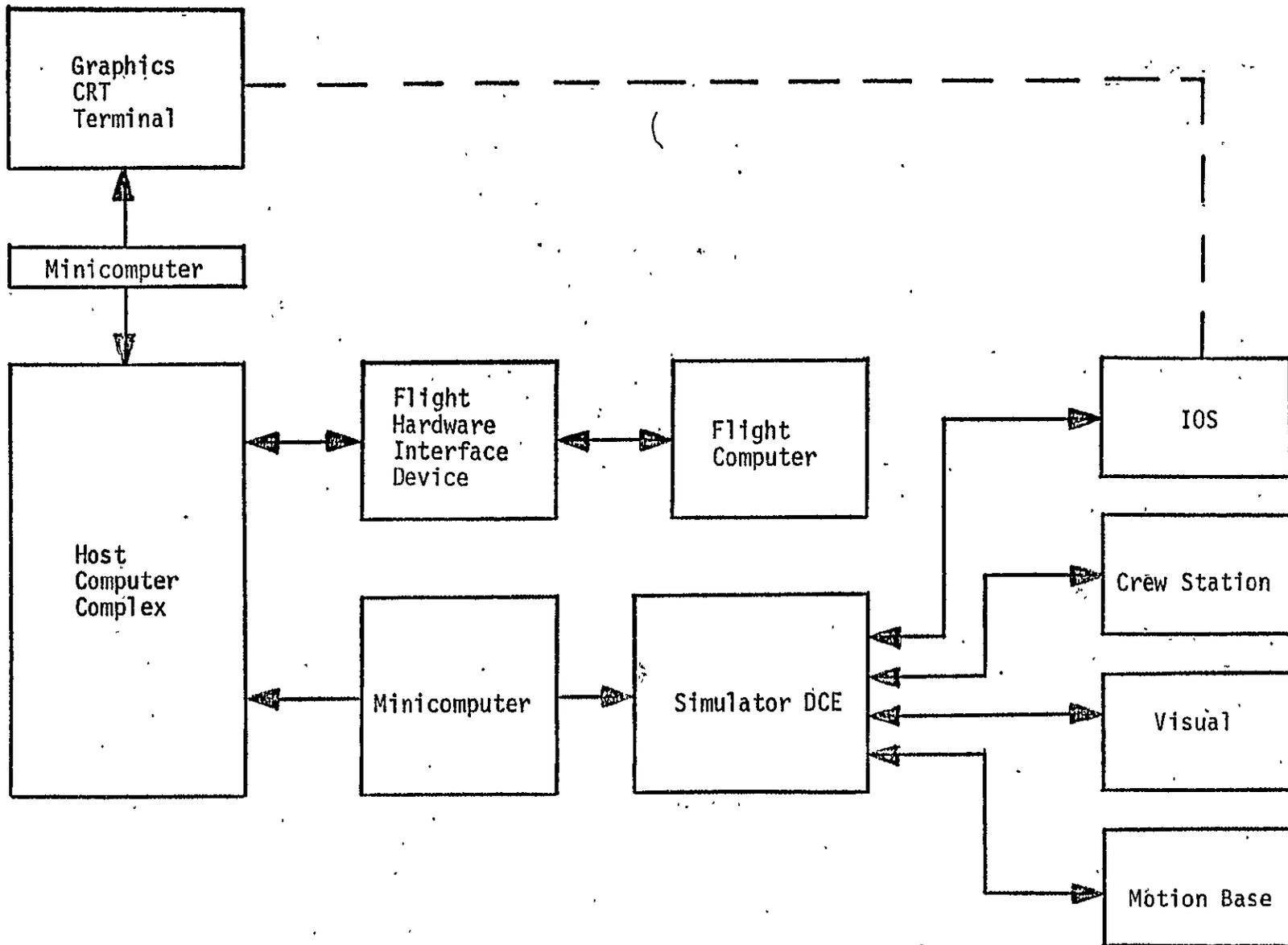
A large amount of data has been reviewed to establish the types and quantities of hardware components and the most likely arrangements or configurations anticipated on near future NASA simulators. In order to develop hardware verification concepts on a realistic and useful basis, it is useful to define a set of components of interest, and the relative quantities and likely configurations of these components at least within subsystems or strings of hardware. This has been accomplished by the definition of a simulator Reference Configuration, described in this section. The study activities related to the Hardware Verification Task address the implementation of self test ideas within the context of this Reference Configuration. In summary, the Reference Configuration establishes the following aspects of the hardware verification task:

- The specific hardware component level to which verification techniques will be carried
- The specific component types to be considered
- The typical configurations of components within subsystems or strings
- The simulator facilities expected to be available for implementation of automated verification techniques (i.e., minicomputers, test stations, data paths, etc.)

The Reference Configuration was established by using the schematic diagrams assembled for each of the upcoming Shuttle training simulators and component lists for these simulators. This information was used to establish the top level configuration shown in Figure 5.0-1, and to define an extensive component listing with appropriate descriptive data. The list is presented in Appendix A and is discussed in the following paragraphs. Section 5.1 reviews the top level aspects of the Reference Configuration. Section 5.2 reviews the details of the subsystem configurations with appropriate schematic diagrams.

#### 5.1 REFERENCE CONFIGURATION

The Reference Configuration shown incorporates all of the major elements anticipated for the forthcoming NASA Shuttle training simulators. The arrangement of the flight computer, interfaced directly to the host computer, is typical



5-2

FIGURE 5.0-1 REFERENCE SIMULATOR CONFIGURATION

MDC E1150  
1 November 1974

also of such simulators as the SLS, the MCAIR F-15 engineering development simulation in St. Louis, anticipated configurations of Shuttle software development simulators, and the Shuttle Avionics Integration Laboratory, SAIL. The use of a minicomputer to off-load display processing for a graphics system is expected for the SMS and has been done for the Flight Simulation Laboratory at MCAIR. For the SPS graphics display processing will be done on the XDS 930.

The crew station, IOS, visual displays and motion base are driven by a single minicomputer as proposed for the Shuttle Mission Simulator in Reference 5.1-1. However, no impact on the results of this study is foreseen for a different arrangement. The approach taken effectively analyses the verification of the minicomputer and its interface to the host computer as one set or string of hardware. The subsystems driven by the minicomputer are then approached individually with no interference expected between the verification techniques for the separate subsystems. The software requirements established will, therefore, be available for implementation in either the minicomputer or the host computer dependent only on the particular configuration being tested.

## 5.2 SUBSYSTEM REFERENCE CONFIGURATIONS

The hardware verification techniques were developed by addressing realistic configurations of subsystem hardware and components. This approach places more importance on the nature of the subsystem configurations than on the configuration of the overall simulator. Consequently, in this section, the subsystem configurations are described and defined to more detail than for the overall simulator configuration.

### 5.2.1 Host Computer Complex

The Host computers for simulators of interest to this study are major scientific computing systems. Maintenance of these systems is provided on a regularly scheduled basis by the computer vendor or other contractor. Effective and efficient maintenance is a continuing concern of major importance to the major computer vendors. Consequently, large digital computer checkout technology is beyond the scope of this study. The study emphasis was therefore placed on that particular hardware peculiar to simulator systems and for which there has been limited past effort directed toward development of automated checkout features.

### 5.2.2 Minicomputers or Other Ancillary Computers

Minicomputers have been incorporated increasingly within many large simulator configurations as their technology becomes more advanced and economical. However, the function that the minicomputer or ancillary computer performs is not of any consequence when the problem of verifying the hardware interface and the minicomputer mainframe is addressed. The parameters or characteristics of primary concern are the word sizes of the host and minicomputers, the data word formats, the parity checking and control signal requirements. These parameters are unique to each particular installation and do not impact the conceptual approaches that are taken to establish proper functioning of the mini and its interface once the assumption is made that the host computer and its data channel are functioning properly.

Consequently, the problem addressed for this specific hardware is the requirements for software to verify that the minicomputer and its interface are functioning properly. This requirement differs from the general vendor's maintenance requirements to isolate faults within a computer system. The functional software requirements are established without definition or limitation to a specific configuration.

### 5.2.3 Crew Station

The crew station configuration is of importance to the hardware verification task only with respect to the components in the crew station. That is, the components in the crew station do not interact between each other but represent the terminations of data paths going to or from the host computer and routed through the data conversion equipment. The configuration of the data conversion equipment is, therefore, the pacing consideration in defining the crew station and IOS checkout process. Consequently, the crew station configuration is defined only by the control and display components listed in Appendix A. The list consists mainly of components anticipated for the Shuttle Mission Simulator. However, the component list is also representative of the Horizontal Flight Simulator from a verification viewpoint. There are no components or quantities in the Horizontal Flight Simulator that present verification problems not encountered in the reference configuration.

Simulator instruments are implemented with either actual flight hardware or simulated flight hardware. The simulated flight instrument consists of flight instrument exteriors with the insides modified to accept analog DC inputs. Synchro/resolver drive circuitry and DC circuitry are both included in the reference configuration.

#### 5.2.4 Instructor/Operator Station

The components that comprise the instructor/operator station are also shown in Appendix A. The basic unit of communication with the system from the instructor station is the CRT graphics displays and keyboards. CRT displays and keyboards provide maximum flexibility both for monitoring and for introducing inputs. Fault insertion can also be accomplished through these graphics terminals. Mode select switches are provided for additional computer inputs. Instrumentation and displays that duplicate crew station hardware are assumed wired in parallel directly from the crew station. Panels are provided for monitoring hand controller positions, communications and recording, and simulator status, including visual, motion base, computer, DCE, lighting, and sound. Television monitors are also provided for monitoring the out-the-window scenes.

The verification requirements for instruments in the instructor/operator station are assumed identical to those in the crew stations.

#### 5.2.5 Data Conversion Equipment

Verification requirements for the data conversion equipment are particularly configuration sensitive. Consequently, a complex data conversion system structure, typical of the Shuttle Mission Simulator, has been expanded to a level of detail which shows schematically the component arrangements down to the Least Replaceable Unit, LRU, level.

The data conversion equipment between the host computer and simulator is shown in Figure 5.2-1. A minicomputer functions as a control processor between the data conversion equipment and the host computer. The minicomputer buffer enables the host computer to transfer its data in continuous blocks at the beginning or end of a time frame and thus minimize interrupt processing by the host computer.

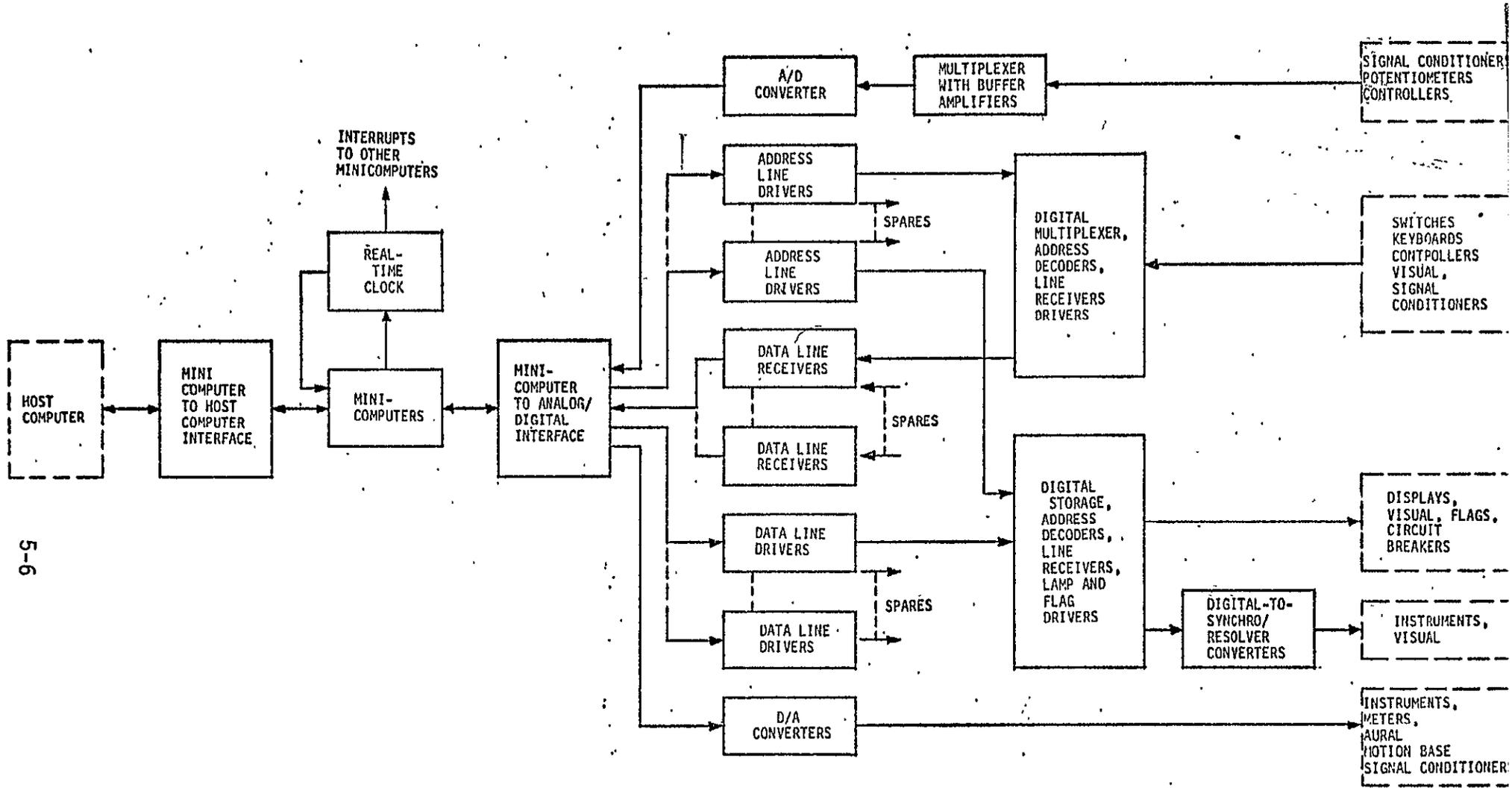


FIGURE 5.2-1 SIMULATOR DATA CONVERSION EQUIPMENT

The mini-computer to analog/digital interface shown in Figure 5.2-2 provides data distribution between the minicomputer and the data conversion components. This interface generates the device addresses and provides the signal conditioning as the data words are transmitted serially to/from the minicomputer. The interface also decodes the addresses and routes output signals and multiplexes input signals to/from the data conversion components. The data conversion components include analog-to-digital converters with input multiplexers, digital-to-analog converters, and data line drivers and receivers. Appendix A lists the quantities and characteristics of these components.

The data conversion equipment also includes digital storage, digital multiplexers, address decoders, lamp and flag drivers, line drivers and receivers, and signal conditioning for the simulator controls and displays which are assumed to be located on the simulator. Locating this equipment on the simulator adds line drivers and receivers but drastically reduces the number of cables between the simulator and interface equipment. The data paths between the simulator and the analog/digital interface for the various types of signals are shown in Figure 5.2-3, 5.2-4, 5.2-5, 5.2-6, and 5.2-7.

#### 5.2.6 Visual Systems Configuration

The visual systems analyzed for automated verification techniques are represented schematically in Figure 5.2-8. This configuration encompasses basic electronic and electro-mechanical components anticipated for near future visual systems. The orbital earth model incorporates those servo, optical, and TV systems also required for implementation of star fields with a star ball. A change of scale in the flat earth model does not change the servo or TV system configuration requirements although the hardware performance specifications may be reduced as opposed to a terminal area model system. Checkout requirements for the computer image generation system are not affected by the type or mix of scenes generated. The remaining functions verified are the TV controls and video mixing and distribution units. Testing of these units are addressed as part of the visual systems checkout problem. Component descriptions may be found in the component table, Appendix A.

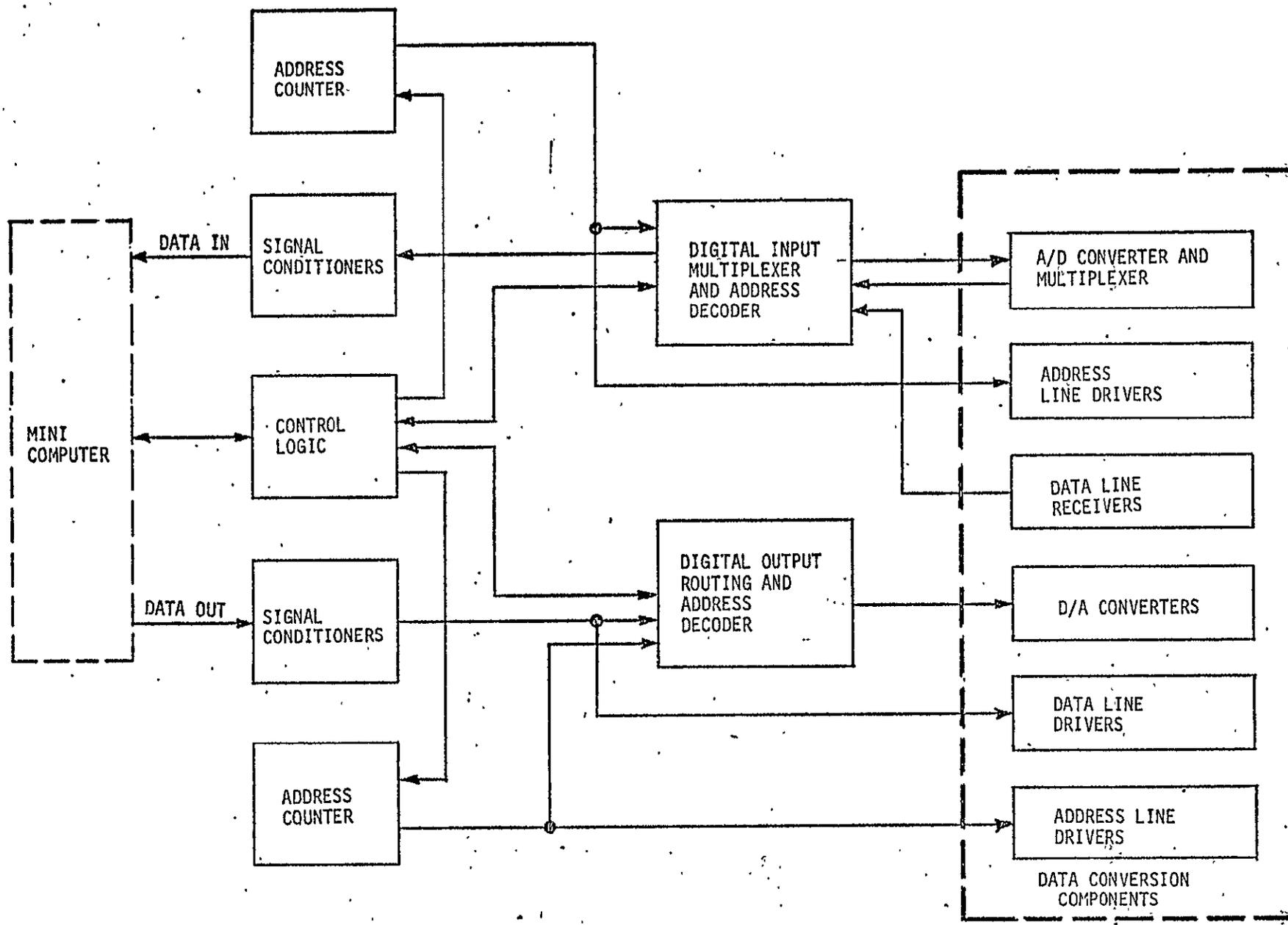


FIGURE 5.2-2 MINICOMPUTER TO ANALOG/DIGITAL INTERFACE

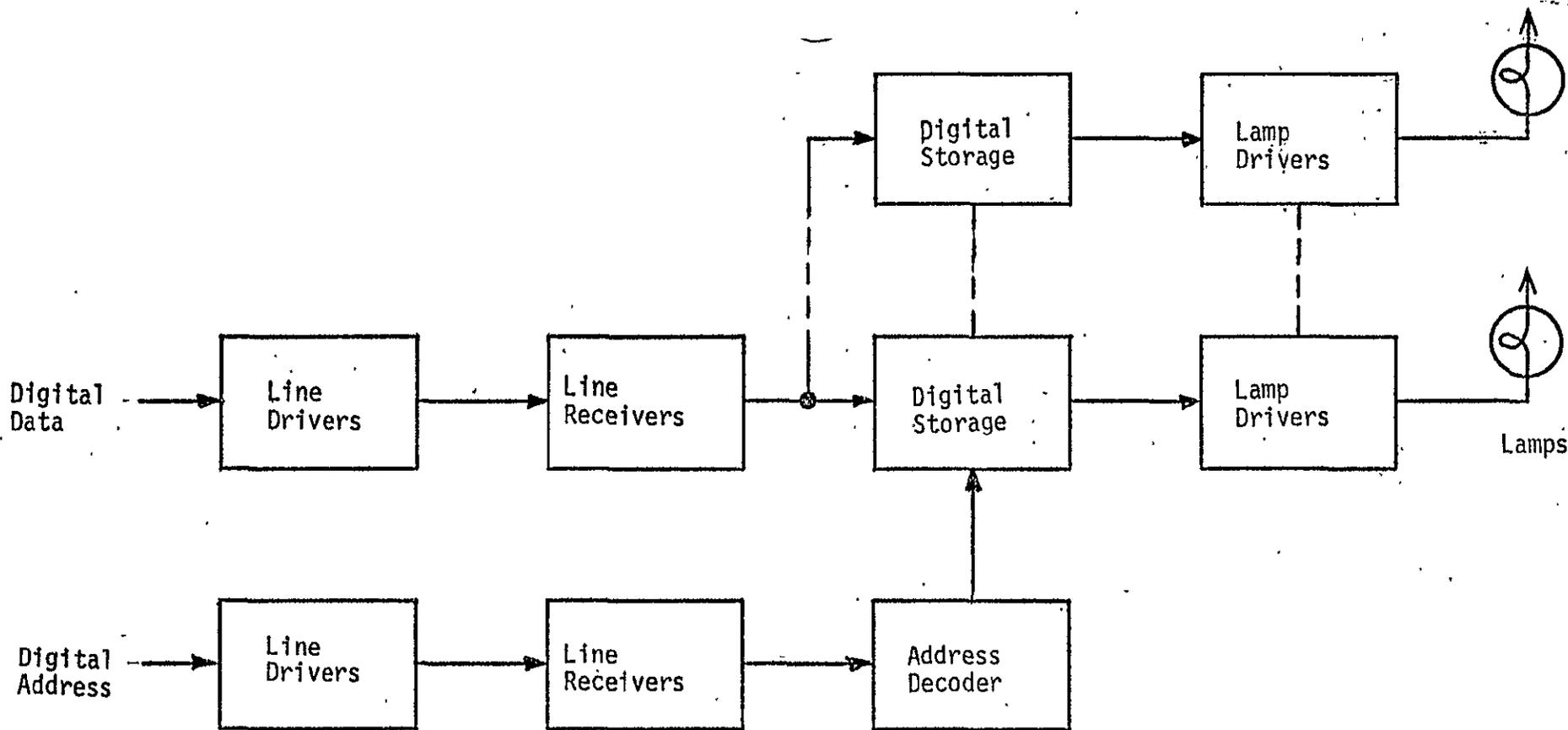


FIGURE 5.2-3 TYPICAL DIGITAL OUTPUT DATA PATH

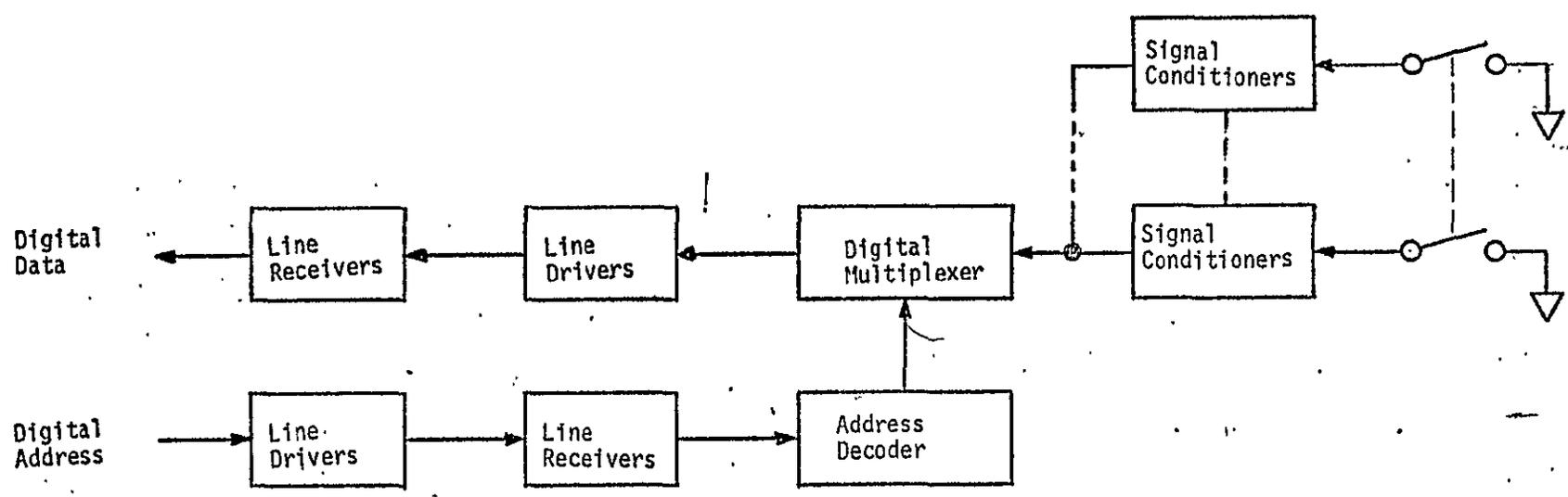


FIGURE 5.2-4 TYPICAL DIGITAL INPUT DATA PATH

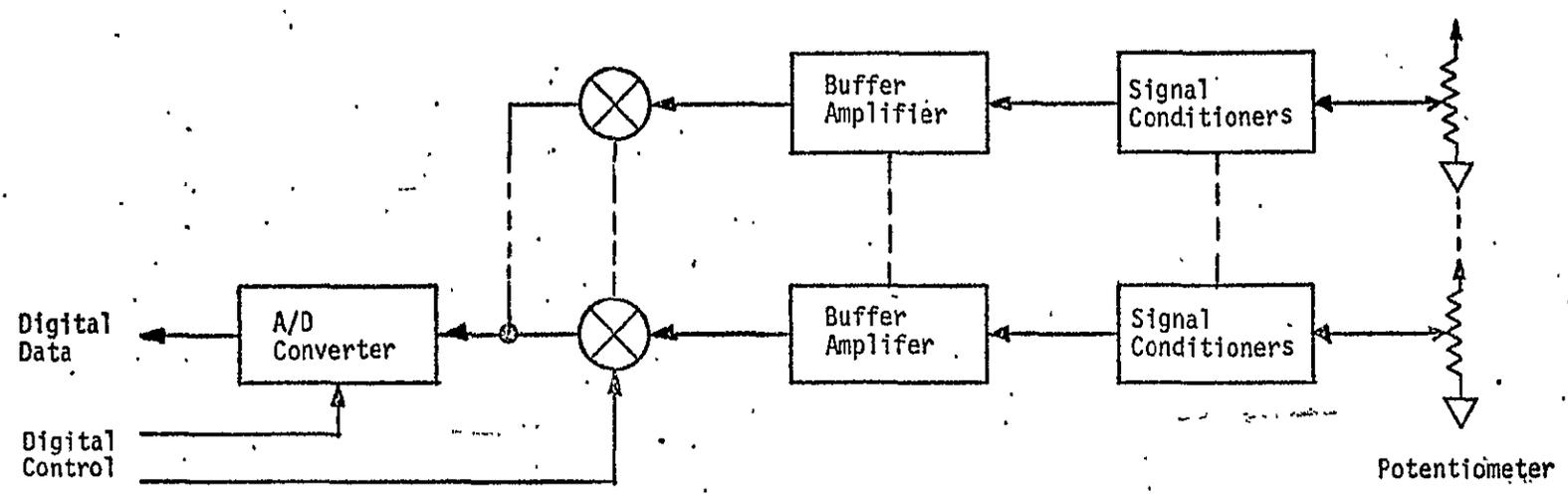


FIGURE 5.2-5 TYPICAL ANALOG INPUT DATA PATH

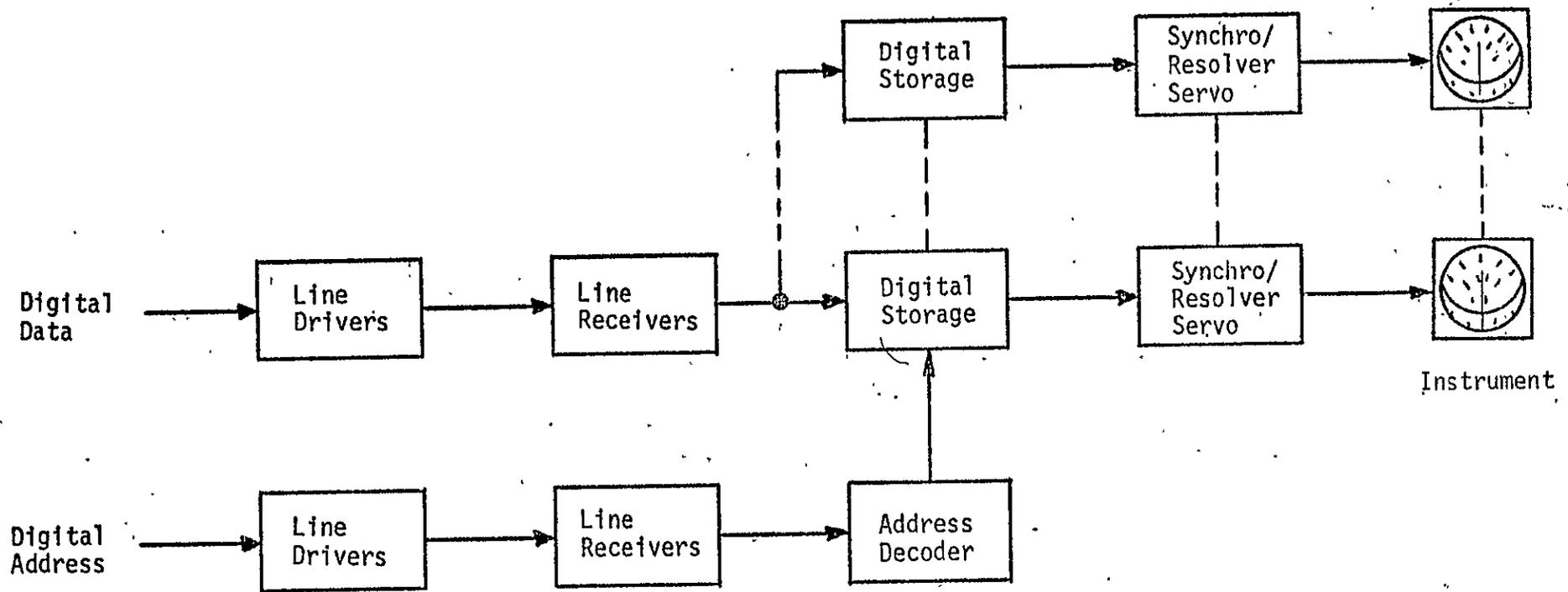


FIGURE 5.2-6 SYNCHRO/RESOLVER OUTPUT DATA PATH

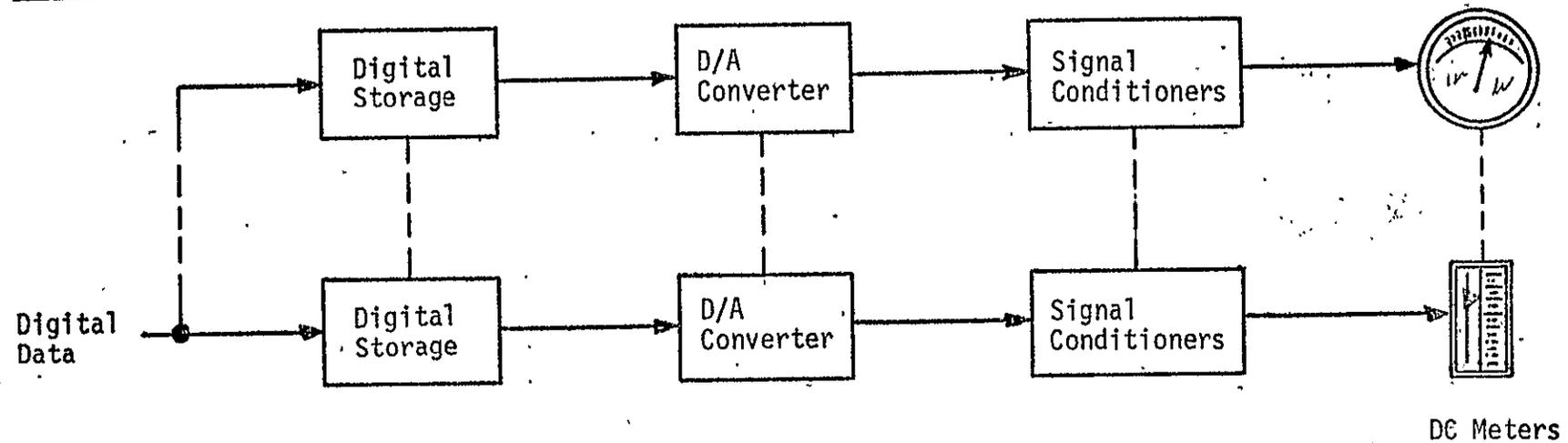


FIGURE 5.2-7 TYPICAL ANALOG OUTPUT DATA PATH

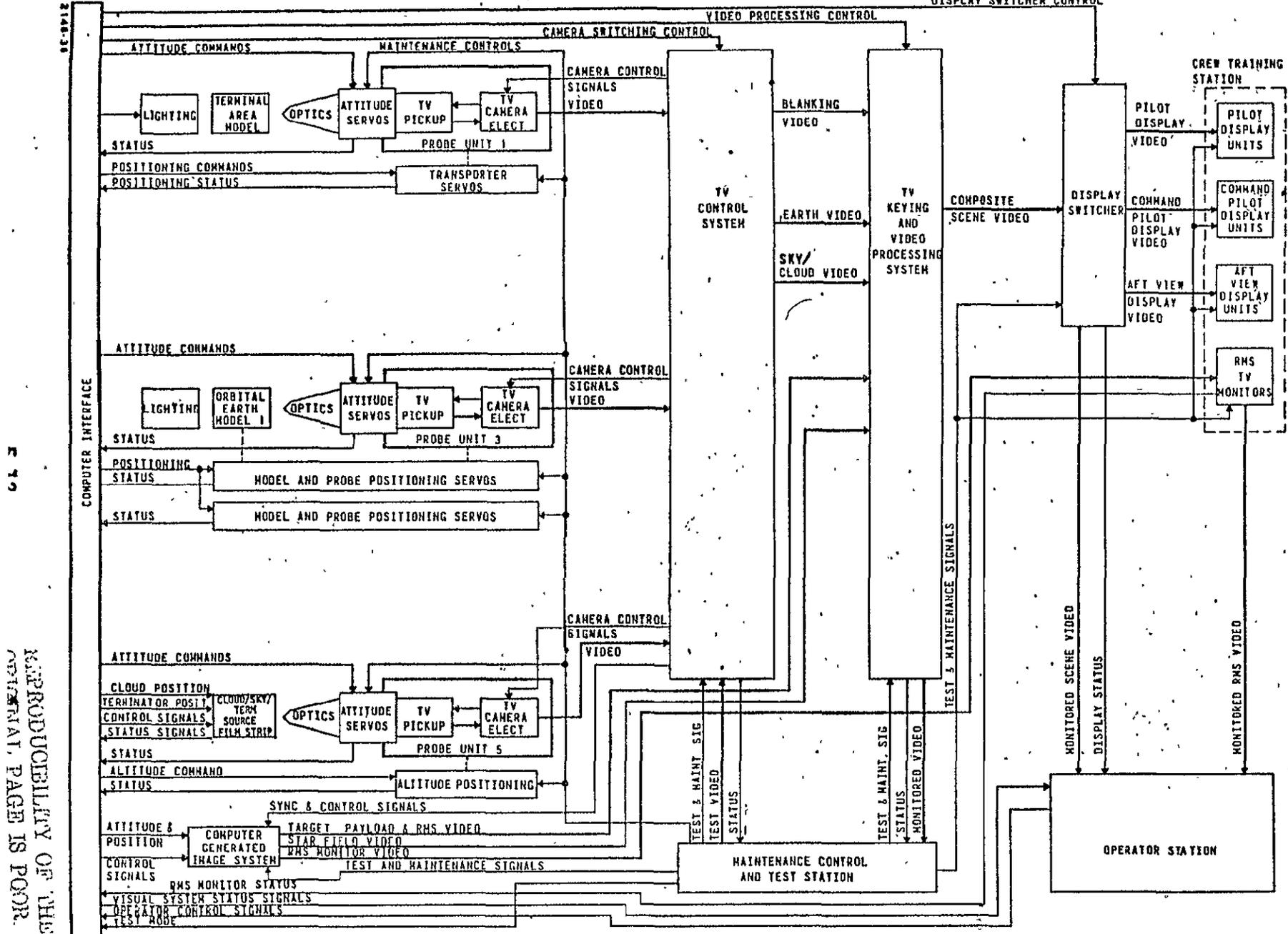


FIGURE 5.2-8 VISUAL SYSTEM BLOCK DIAGRAM

MDC E1150  
1 November 1974

Three methods for implementing servo systems are considered. These methods are commonly used on all types of simulators and consist of the following:

- A DC analog system, Figure 5.2-9
- A digital system with digital encoders for position feedback, Figure 5.2-10
- A 400 hz synchro/resolver system, Figure 5.2-11

#### 5.2.7 Motion Base System

Survey of various simulator installations, both for aircraft and spacecraft, reveal a preponderance of hydraulic driven motion platforms. Electrical motion systems are more common where large displacement and accurate positioning are important factors, such as in model and camera visual systems, or in the 100 foot lateral displacement motion base of the Flight Simulator for Advanced Aircraft at NASA-ARC.

Studies performed for NASA-JSC for baseline hardware definition of Shuttle simulators, (Reference 5.2-1) have reinforced the concept that a moving platform powered by hydraulically driven actuators has advantages over other possible drive schemes. These advantages are especially attractive in mass movement capability, range of displacement, and velocity and acceleration capabilities within the facilities envelope allowed for the simulator, while meeting the simulation requirements for the various flight regions of the Shuttle vehicle.

The motion base system configuration used for reference in this study is a six-degree-of-freedom, synergistic actuator, hydraulic system. A detailed description of this configuration is shown in Section 8.5.1. Figure 8.5-1 shows the actuator arrangement used to drive the motion platform, while Figure 8.5-2 presents the hydraulic system layout for providing kinematic drive for this platform.

#### 5.2.8 Flight Hardware and Interface

The flight hardware of concern for purposes of simulator reference configuration definition includes:

- Three flight computers
- A mass memory device

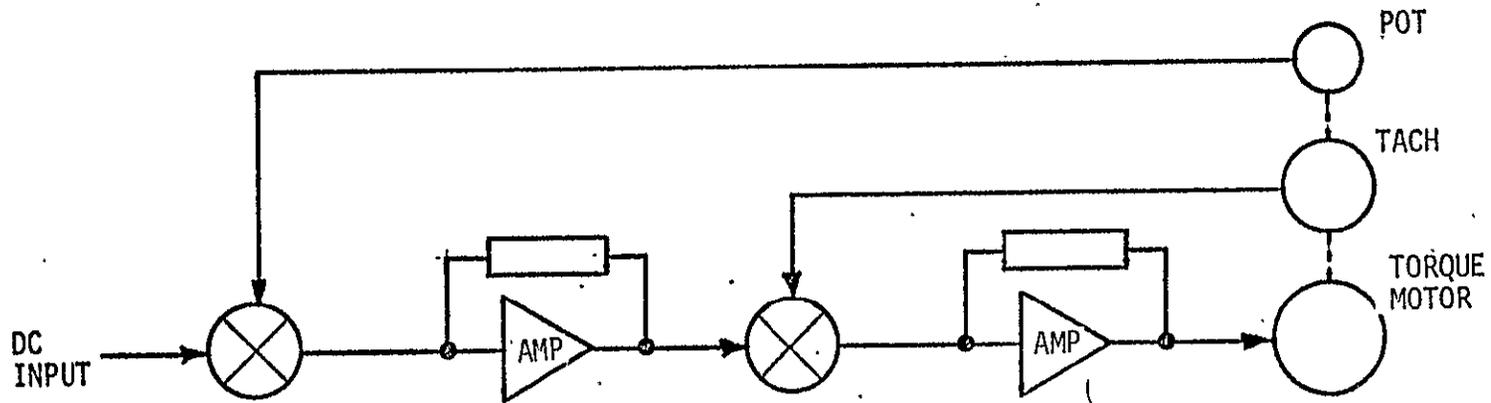


FIGURE 5.2-9 DC ANALOG SERVO SYSTEM

5-14

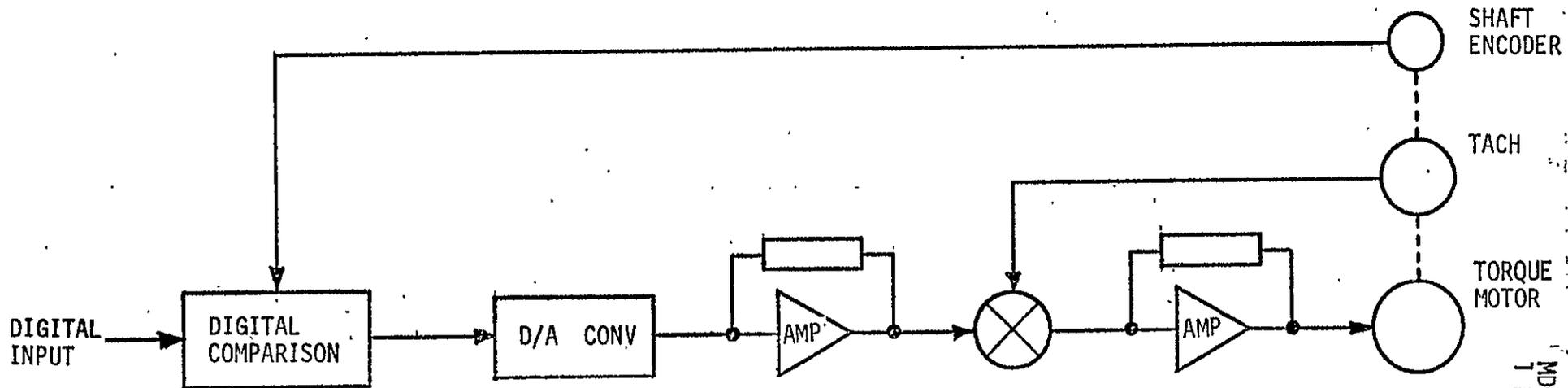


FIGURE 5.2-10 SERVO SYSTEM WITH DIGITAL COMPARISON



- The onboard graphics CRT displays
- The keyboard for flight computer entries
- The display electronics unit that provides interface services between the flight computers and the keyboard and displays

In the simulator, this hardware is interfaced to the HOST computer by means of the Flight Hardware Interface Device. In the actual Orbiter, the flight computers interface with various Shuttle systems. In the simulator, elements of these systems or their functions are simulated in the HOST computer. This variety and complexity of functions places a heavy demand in rate and data handling versatility at the HOST to flight hardware interface. The FHID provides this versatility, while at the same time providing time generation and time tagging functions for flight data handled by the flight computer. Figure 5.2-12 shows the equipment arrangement and interfaces for the flight hardware used in the Reference Simulator Configuration.

## SECTION 6

### DATA SOURCES FOR SURVEY OF CURRENT HARDWARE SELF TEST TECHNIQUES

Initial efforts of the Simulation Verification Techniques Study focused on obtaining information regarding the techniques currently being used in simulator verification. This section summarizes the results of the survey in terms of the data sources that were sought out or contacted for information of interest to the present study. Section 7 presents the self test techniques found to be pertinent to simulator applications. Self test system data sources surveyed included simulator developers and users, as well as the literature of relevant technology areas.

#### 6.1 SIMULATOR USERS/DEVELOPERS

Requirements for hardware self-test techniques as well as past experience vary widely with simulator users at least partly as a function of the basic nature of the simulator activity with which they are concerned. Aerospace simulators of interest to this study may be grouped into the following four categories:

- o Spaceflight training simulators
- o Military training simulators
- o Commercial airline training simulators
- o Engineering development simulators

Spaceflight training simulators as developed and utilized in the past at Johnson Space Center represent the most sophisticated and complex simulators assembled for any purpose. This complexity derives in part from the complexity of the vehicles being simulated and in part from the fidelity of the simulation required. The stringency of the fidelity requirements is imposed by the fact that spacecraft crews, up until now, have not had the opportunity for any actual flight training in the vehicles they are to fly, prior to the time of their operational flight or mission.

Military training simulators are currently being procured in increasing numbers as an economy measure to conserve both fuel and money, since these

simulators are used for crew training functions previously accomplished during flight training phases of crew training programs. As a consequence, the fidelity requirements for these simulators seem to be increasing and motion base requirements and visual display requirements are becoming more sophisticated.

Commercial airline training simulators have also increased in fidelity in recent years. The objective of fidelity improvements for these simulators is to reduce crew flight training hours required for transition training to new aircraft. Airline simulators probably have as high a utilization factor as any simulators in use.

Engineering development simulators are a standard tool for today's airframe developers and manufacturers. The use of simulators for development and advanced development purposes is also promoted at NASA's Ames Research Center, Mountain View, California.

Simulation facilities and/or simulators reviewed or visited during survey activities are summarized in Table 6.1-1. The general nature of the facilities and their activities is described further in the following paragraphs:

#### 6.1.1 Spaceflight Training Simulators

The spaceflight training simulators reviewed were basically the simulators at JSC although a visit was made to the MSFC facilities. Specifically, the NASA JSC simulators included in the survey were the Command Module Simulator (CMS) and the Lunar Module Simulator (LMS) in Building 5; the Skylab Simulator (SLS) in the same building; and the Command Module Procedures Simulator and the Lunar Modules Procedures Simulator in Building 35.

The Command Module Simulator (CMS) is the oldest simulator at JSC; as such, it employs very little checkout hardware or software. Software diagnostics are used for the DDP computers, which have also been modified with a "traceback" register to aid post-crash diagnosis. All other checkout and maintenance is implemented by straightforward manual methods.

FACILITY	LOCATION	CONTACT	EQUIPMENT/ACTIVITIES
Flight Simulation Lab. Flight & Guidance Simulation Lab.	Ames Research Center Moffett Field, California	Joe Rathert Chief, Simulation Science Division	18 different installations, four of which are considered major, and supporting computers and interface equipment. These include a 6 degree-of-freedom motion base with automated checkout software; additional 5 and 6 degree of freedom motion bases; visual systems; four hydraulic control loader systems for force feel simulations; aircraft sound simulators. Equipment used for research and development.
Computation Lab. Computer and Simulation Division	Marshall Space Flight Center, Huntsville, Alabama	Frank Vinz	EAI 8900 Hybrid Computer System, 6 Degree-of-freedom motion base terrain models, visual displays, etc. No specific activity in automated checkout.
Avionics Laboratory; Human Resources Lab; Flight Simulation Branch, Training and Checkout Division.	Wright Patterson AFB Ohio 45433	Col. Scolatti, Jim Henderson Art Dody	Numerous simulators for both training and R&D. Activities include procurement of AF simulators, specification of validation, verification, reliability and maintainability requirements. Work on development of "Morning Readiness Test" concept.
Naval Training Equipment Center	Orlando, Florida	F. R. Cooper	1 F-4 simulator for research and development. Primary activity is procurement of training equipment including simulators. Advanced work done on checkout of visuals and vehicle dynamics simulation to isolate faults between visuals and the aircraft simulation.
Computer Simulation Facility	Naval Air Development Center Warminster, Pennsylvania	Bob Jones	A number of large hybrid simulators. Two CDC 6600's share extended memory. Four DADIOS units for interface. Simulators used for engineering development. No new or sophisticated checkout techniques in development.
American Airlines Flight Academy	Ft. Worth, Texas	R. D. McLure Mgr., Simulator Technical Support	Training simulators for 727, DC-10, BAC 1-11. Simulators operational 16 hours/day. Maintenance and checkout primarily on third shift. Pre-training test run for daily checkout.
Flight Training Center	United Air Lines Denver, Colorado	Dale Seay Mgr., Simulator Operation	Training simulators for 727, DC-10, 747, DC-8. Maintenance on third shift. Paper on performance verification presented at AIAA symposium.
USAF F-15 Simulator	MCAIR St. Louis, Missouri	Ken Kury	Two Datacraft 6024's will be used to drive simulator including simulation of the flight computers. Reliability/maintainability requirement specified.
MDC Flight Simulation Laboratory	MCAIR, MCAUTO St. Louis, Missouri	Roger Mathews Steve Rayhawk Gene Brown Frank Brown	CDC 6600 supports a variety of engineering development simulations. Equipment include swinging arm motion base; dual cockpit air combat simulator, No formal verification or hardware checkout activity.
MDEC Training Simulators	MDEC, St. Louis, Missouri	Howard Barnes Bill Boring Tom Franz John Mackey	MDEC has developed simulators for the A-7D and DC-10 most recently. The A-7D uses a morning readiness test.

TABLE 6.1-1 SIMULATION VERIFICATION TECHNIQUES STUDY  
SIMULATION FACILITY SURVEY SUMMARY

The Lunar Module Simulator (LMS) was reviewed but no special checkout techniques were found to be in use for this simulator.

The Skylab Simulator (SLS) began as a MOL simulator for the Air Force, and therefore has extra-cost features which make hardware checkout very convenient. The hardware and software architecture allows maintenance personnel to decouple an individual functional unit from the system and diagnose that unit with built-in stimulus/response hardware, while the rest of the simulator system operates normally.

Until recently, the Command Module Procedures Simulator (CMPS) and Lunar Module Procedures Simulator (LMPS) shared the CDC 6400 host computer. The Shuttle Procedures Simulator (SPS) currently under development also uses the CDC 6400, augmented by XDS 930 and 9300 computers. Three distinct hardware checkout programs have been developed for the CMPS and LMPS. These are PSALT, GREMLIN, and QUIKCHECK. The self test functions implemented with these software packages are summarized in Table 6.1-2.

#### 6.1.2 Military Training Simulators

Current military simulator procurements show the effect of the Complete Cycle Cost (C<sup>3</sup>) philosophy common in other military procurements. As a result, specifications for new simulators include highly formalized requirements for configuration management, documentation, initial verification and validation, hardware checkout, reliability, and maintainability. To a great extent, the specifications simply define requirements, leaving it to the contractor to develop the techniques to satisfy these requirements.

The USAF specification for an F-15 training simulator as an example, calls for an MTBF of 90 hours and an MTTR of one-half hour. This corresponds to an availability of 99.4%. In the area of methods to achieve this low MTTR, the specification includes high-level requirements for access panels, test points, and software diagnostic/exercise routines.

The principal sources contacted for information on military simulators were the F-15 simulator procurement being managed by McDonnell Douglas Aircraft Company

TABLE 6.1-2 EXISTING FSD CHECKOUT SOFTWARE

PROGRAM NAME	HARDWARE TESTED	CHECKOUT FUNCTIONS
PSALT (PROCEDUPES SIMULATOR ANALOG LINKAGE TEST)	SEE BELOW	<ul style="list-style-type: none"> <li>o EXECUTIVE CONTROL OF FLINT, TAMS, SWORD, CDT, AND EQPCK.</li> <li>o CDC 211 (CRT TERMINAL) OPERATOR INTERFACE.</li> <li>o CDC 6400 SCOPE (OPERATING SYSTEM) INTERFACE.</li> <li>o PRINT OUTPUT.</li> <li>o MISCELLANEOUS MONITORING AND DISPLAY FUNCTIONS.</li> </ul>
FLINT (FUNCTION LINK INTERPRETER)	DCE: A/D, D/A, AND DISCRETE I/O	<ul style="list-style-type: none"> <li>o SELECTION OF INPUT AND OUTPUT SIGNAL PATHS.</li> <li>o REPETITIVE OUTPUT OF OPERATOR-SUPPLIED TEST STIMULI AND COMPARISON WITH EQUIPMENT RESPONSE (USING OPERATOR-SUPPLIED TOLERANCE).</li> <li>o FAULT DETECTION AND ERROR-FREQUENCY COUNT.</li> <li>o DISPLAY AND PRINT FAULT-DETECTION RESULTS.</li> <li>o FAULT ISOLATION (IN COOPERATION WITH TEST OPERATOR).</li> </ul>
TAMS (TEST ANALOG MODE SELECT) <sup>a</sup>	EAI 231R (ANALOG COMPUTER) MODE CONTROL ADAPTER (MCA) AND ANALOG/DIGITAL LINKAGE CONTROLLER (ADLC)	<ul style="list-style-type: none"> <li>o COMMAND EACH MODE OF THE ANALOG COMPUTER; CHECK WHETHER DESIRED MODE WAS SELECTED.</li> <li>o TEST MCA ALARM SIGNAL.</li> <li>o TEST ADLC CLOCK START/STOP.</li> <li>o TEST STATUS OF DATA CHANNEL.</li> <li>o GENERATE APPROPRIATE ERROR ALARMS.</li> </ul>
SWORD (STATUS WORD DIAGNOSTIC)	LINK STATUS WORD (12-BIT REGISTER USED BY PERIPHERAL PROCESSOR)	<ul style="list-style-type: none"> <li>o COMMAND SET/RESET OF EACH INDIVIDUAL BIT OF THE LINK STATUS WORD; CHECK FOR PROPER RESPONSE.</li> <li>o TEST REPETITIVELY; GENERATE AND DISPLAY ERROR COUNT.</li> </ul>
CDT (CLOCK DIAGNOSTIC TEST)	ADLC CLOCK (HARDWARE INTERVAL TIMER)	<ul style="list-style-type: none"> <li>o TEST CLOCK START/STOP RESPONSE.</li> <li>o TEST CLOCK RESET MODE.</li> <li>o TEST CLOCK CAPABILITY TO GENERATE DATA-TRANSFER PULSES.</li> <li>o TEST CLOCK FREQUENCY COUNT AGAINST CDC 6400 INTERNAL CLOCK.</li> </ul>
EQPCK (EQUIPMENT CHECK)	DISPLAY HARDWARE IN END-TO-END CONFIGURATION, INCLUDING ASSOCIATED DCE (D/A, A/D, D/R, ETC.)	<ul style="list-style-type: none"> <li>o APPLY OPERATOR-DEFINED CONSTANT SIGNALS TO D/A OR A/D CHANNELS.</li> <li>o APPLY SINE OR TRIANGULAR WAVE TO D/A CHANNELS.</li> <li>o DRIVE DISCRETES ON/OFF AT A SPECIFIED RATE.</li> <li>o CYCLE DIGITAL READOUTS OVER ALL DIGITS.</li> <li>o DRIVE D/R CHANNELS AT A SPECIFIED RATE.</li> <li>o SET D/R UNITS TO A SPECIFIED ANGLE.</li> </ul>
ADG (AVERAGE DEVIATION GRAPH)	A/D AND D/A CALIBRATION	<ul style="list-style-type: none"> <li>o SELECT D/A AND A/D SIGNAL ROUTING.</li> <li>o DRIVE EACH CHANNEL THROUGH A <math>\pm 128</math> VOLT RANGE.</li> <li>o DISPLAY CALIBRATION CURVES.</li> <li>o GENERATE ALARM FOR EXCESSIVE DEVIATION.</li> </ul>

<sup>a</sup> TAMS IS NOT IN USE AT PRESENT.

St. Louis; and the information available from the Naval Training Equipment Center, Orlando, Florida. NTEC manages the procurements of many of the simulators acquired for the Navy and Marine training programs and is also involved in certain Army procurements as well.

### 6.1.3 Commercial Airline Simulators

Airlines are highly cost-conscious in their approach to training simulator procurement and operation. Their management demands, and achieves, high availability of simulation equipment with straightforward manual methods (e.g., man-in-loop preflight) and few checkout aids. Airline personnel interviewed stated that the key to simulator maintenance is simply "good people". Looking at their recent operating history (generally, 99% or better availability), they see little need for any automation of checkout. However, when pressed, they concede that they had significant "teething" problems when their current simulators were new.

In one respect, airline simulators are well suited to incorporation of automatic checkout techniques. That is, their configurations and operational procedures are very stable and well controlled. Countering that, however, is the fact that to reduce initial cost, airline simulators are very centralized, and allow small space and time reserves in the central computer. This would make it rather difficult to retrofit checkout hardware and software. Only one airline simulator with mass storage and overlay capability was identified.

When a particular simulator subsystem presents a downtime problem, the tendency is to improve maintenance procedures, change spares provisioning policies, and/or redesign the equipment to achieve higher reliability, so that the subsystem will not require as much maintenance.

Principal sources contacted for information on airline activities were American Airlines Flight Academy, Ft. Worth and the United Air Lines facilities in Denver, Colorado, both of which were visited by study personnel.

#### 6.1.4 Engineering Development Simulators

Engineering simulators considered in this study included simulators built up to support a particular aircraft or spacecraft design project (such as the Space Shuttle, DC-10 or F-15), as well as multi-user simulation facilities intended to support a number of different aircraft or spacecraft development programs (such as the Flight Simulation Facility at MCAIR and the Simulation Sciences Division of NASA-Ames).

The major obstacle to implementation of advanced checkout techniques on such simulators is the steady stream of changes to the simulator hardware and software. Typically, the simulator support staff has difficulty just keeping up with the changes and capability improvements requested by the using group(s), leaving little time to devote to incorporation of checkout features.

## 6.2 BIBLIOGRAPHIC DATA SOURCES

The present state of the literature in the field of automatic test equipment was aptly described in Reference 6.2-1, in September of 1974.

"Because testing has become a rather specialized field with its own in jargon, most engineers who are confronted with a testing assignment may not be able to assess the entire testing problem. The problem is compounded further by the non-existence of a unified body of knowledge in the form of tutorial texts on automatic testing; hence the engineer must wade through voluminous and not so easily digested technical journals before finding something that can be readily applied. The Engineer's most frequent compromise, therefore, is to take what is available commercially (after an inexhaustive study of the field) and tailor whatever he buys to fit the product that is to be tested. As a result, there is a tendency to buy ATE haphazardly or build the whole ATS from scratch with great expenditure of time and money."

The absence of an integrated, cohesive literature on the broad area of automatic testing prompted us, even during our self sponsored studies, to establish subject indexed reference lists in an attempt to organize literature of interest. In the present study, we have extended these early results to bring them up to date and to make them more useful to the reader.

The results of literature surveys are presented in the Appendices to this report. Appendix B contains a glossary of terms particularly useful in the area of simulator self test techniques. In Appendix C, we have attached a comprehensive bibliography of material which we have reviewed during or before this study. In the first part of the bibliography, the source documents are referenced by subject matter. In the second part of the bibliography, the documents are listed in more complete bibliographic form, alphabetically, by authors' last name.

## SECTION 7

### RESULTS OF SURVEY OF CURRENT HARDWARE SELF TEST TECHNIQUES

Generic techniques for hardware self test that are of particular potential interest to simulator users and developers are presented in this section. These techniques are limited to those the simulator user might apply himself to various simulator subsystems as opposed to very sophisticated and esoteric techniques that would be of primary interest to more specialized subsystems manufacturers. An example of the latter is the digital logic simulations often used by computer manufacturers further discussed in paragraph 7.1.1.1.

The techniques of interest to the present study fall into the following three major categories:

- Test design
- Test implementation
- Test data processing

Test design techniques are primarily influenced by the objective of the test being designed. The tests for which design techniques have been considered include fault detection tests, fault isolation tests and incipient fault detection tests.

Test implementation techniques are concerned with the means of generating and inserting test signals, into the hardware being tested at the proper points to accomplish the test objectives. Test implementation is also concerned with the acquisition of the required test data. The latter problem becomes one of either sensing a parameter that is difficult to measure or switching a signal without introducing an additional failure point into the hardware being tested.

Test data processing is that processing that is required to manipulate the data acquired from the test in order to make it reveal the information that is required in fulfillment of the test objectives. Test data processing, in some of its forms, is a heavily documented problem area. Unfortunately, some of the more mundane, though more useful techniques in this area, are hardly documented at all.

Test design techniques are discussed in Section 7.1, test implementation techniques in Section 7.2, and data processing techniques in Section 7.3.

## TEST DESIGN TECHNIQUES

Test design techniques are methods and processes that can be used to accomplish one or more of the following:

- Identify a test approach for a subsystem
- Develop a test sequence
- Establish test signal requirements
- Identify characteristic parameters for correlation of test results

Test design techniques are highly dependent on the objective of the test being designed as well as on the basic type of hardware being tested. Consequently, in this section, we have addressed the basic types of tests of interest to this study. These three types are readiness tests, whose primary objective is fault detection; fault isolation tests, and incipient fault detection test techniques.

Readiness tests are discussed in Section 7.1.1 in terms of the various hardware categories to which they may be applied. Fault isolation test discussions are organized similarly in Section 7.1.2. Incipient fault detection tests discussed in Section 7.1.3 are addressed in terms of the various approaches that have been identified for implementing an incipient fault detection capability.

### 7.1.1 Readiness Tests (Fault Detection)

Readiness tests are used to verify the readiness of simulator subsystems to perform according to design and operational requirements. Generally these tests check each of the simulator subsystems and, whenever possible, use already verified subsystems to check other subsystems of the simulator. Ideally, readiness tests are end to end tests. They are primarily concerned with testing a complete string of hardware.

In order to implement an end to end test for a hardware subsystem and arrive at any degree of confidence as to its proper operability, it is generally necessary to first consider the likely failure modes of the hardware and the symptoms of those failures. For an end to end test this requires the identification of some manifestation of the failure that is going to be apparent at the end of the hardware string at which the data acquisition process will be

implemented.

Once the failure modes and symptoms are recognized, then it is logical to try to identify the characteristic parameters that will enable detection of the failures being tested for. The characteristic parameters are not necessarily quantities that are being sampled directly by the data acquisition system. For example, the data acquired may be the position of a device as a function of time. The characteristic parameters for evaluating the system's performance may be rise time, peak overshoot and settling time. The necessary operations between the data acquired and the characteristic parameters represent the data processing discussed in Section 7.3.

Test design techniques contribute directly to the identification of symptoms of faults as well as to selection of characteristic parameters for readiness testing. Test design techniques depend heavily on the type of hardware being considered. Consequently, the remaining discussion is organized in terms of the basic hardware types considered. These three types are digital electronics, analog electronics and electro-mechanical systems.

#### 7.1.1.1 Digital Electronics (Fault Detection)

Digital circuit elements are found in a simulator in the data conversion equipment, the host computer and flight computers as well as the devices which interface the computers of various types. Some of these interfaces are themselves computers.

Failures of digital circuit elements manifest themselves in one of three different modes:

- Constant logic 1 output, unable to switch to a logic 0
- Constant logic 0 output, unable to switch to a logic 1
- Output which does not represent the inputs and is intermittently erroneous

The failure may occur in data storage or in gating circuitry. In this case the accuracy of calculations is affected. On the other hand, a failure may develop in address generation or recognition circuitry. This type of error causes data or commands to be routed to improper devices, or not to be transferred at all. The first case may not be immediately recognized, but instead may be processed

having the same effect on the system as that caused by presenting wrong data to the right device.

The limited choice of failure modes at the logic level belies the complexity of the problems, both of detecting and isolating faults in digital circuit elements when combined into the major simulator subsystems previously noted.

These problems arise because of the logical complexity of the circuits any time a substantial path length is considered. As a consequence it is entirely possible and frequently likely for the hardware logic structure to effectively conceal faults from casually designed functional tests of the basic functional units. For example, exercising all of the hardwired instructions in a computer and checking the answer of the combined operations may or may not reveal the presence of a fault. The reason this can occur is that for any particular combination of numbers to be processed, there may be bit cells that remain unchanged during the process or whose values are dropped during round off operations.

There is a solution to this test design problem. That solution is the gate level simulation of the entire digital logic circuit in the string. With the proper simulation, it becomes possible to insert faults at any critical point identified, simulate a test and evaluate whether the test detects that fault and propagates it in a manner that can be discerned in the test results that are produced.

Unfortunately, the gate level simulation of digital circuits is a large and sophisticated undertaking that is best left to the manufacturers of digital circuits. Not only is the required effort substantial, but the talent required is sophisticated, and in addition, if a major computing system is involved in the path, the problem of obtaining an accurate representation of the computer's operation is itself a major undertaking. This is due to the proprietary importance of this very detailed information on the operation of a vendor's computer system. This latter factor also encourages assignment of the diagnostic development task for digital equipment to the people who make it.

Nevertheless, there is a limited amount of action that may be taken by the simulator user for implementation of readiness tests. This action is the specification of diagnostic software capabilities as part of computer and digital system procurements. The form that any such specification may take is necessarily functional. For example, it is relatively easy to specify the provision of test software to test all memory cells for their ability to assume both true and false conditions; and to exercise the hardware instruction set not only for numerical results but also for timing which may also be revealing; and to exercise communication paths to assure that all bits, both data, address, and parity, and inter-subsystem communication are being transmitted and received correctly. The functional tests required in any specific application may be directly derived from the device's operational requirements. The availability of this software can never give a one hundred per cent confidence level that all of the hardware is functioning properly but it can give a much higher level than no test at all. In addition, depending on the hardware in question, the manufacturer may be able to specify what per cent confidence may be realized from exercising certain specific parcels of his self test software as part of a daily readiness test.

#### 7.1.1.2 Analog Electronic Circuits (Fault Detection)

Analog electronic circuits are found in the data conversion equipment, the television systems for the visual simulation, and in the aural simulation. Analog circuits do not generally exhibit the logical complexity that is found in digital circuits. As a consequence, it is usually possible to identify characteristic parameters that are effective in evaluating the circuits on an end to end basis for readiness testing.

Failure modes of analog circuits can be cataloged and because of the large variety of component types to be considered the catalog would be extensive. For purposes of designing readiness tests it is simpler and more efficient to consider the basic nature of electronic equipment of this type. The electronic subsystems we are concerned with for the simulator hardware environment consist of a number of lower level modules that are considered for self test purposes to be LRU's or Least Replaceable Units. For purposes of readiness testing we are concerned with verifying the proper operation of a relatively simple

arrangement or collection of these LRU's which might also be described as "black boxes." These boxes as well as the subsystems into which they are integrated have fairly well defined functional performance specifications. The objective of the readiness test can then readily be interpreted to be the verification that these subsystems or LRU's are performing per their performance specification. This effectively amounts to direct identification of characteristic parameters for the subsystems or LRU's with only an intuitive consideration of their potential failure modes.

Typically the choice of the characteristic parameters which are likely to be of interest for measurement in a self test system is relatively limited and will consist of some combination of the following:

- Bandwidth
- Signal/Noise Ratio
- Linearity
- Amplitude and phase response versus frequency
- Sensitivity
- Differential gain

Many of these parameters may be incidental properties of the circuit or because of the circuit's functional purpose, they may be performance criteria and consequently, for test purposes, they become characteristic parameters of primary importance.

Since the input as well as the output of an electronic circuit is basically an electrical signal, the specification of characteristic parameters is the major part of the readiness test problem aside from some possible switching requirements. The stimulus is readily generated with electronic signal generators and the output or response is acquired as another electrical signal ready for processing.

#### 7.1.1.3 Electro-Mechanical Systems (Fault Detection)

Electro-mechanical systems are found in simulators in many subsystems. These subsystems include the galvanometer and servo driven meter movements in the crew station and IOS, as well as the model drives for the visual simulation, and the electro-hydraulic motion base servo systems. These systems are all applications of classical control theory design techniques and are susceptible

to analysis and testing using the concepts of modern control theory.

Readiness tests for these systems are again only concerned with determining whether the overall end to end performance of the unit or subsystem is within performance specifications. In the case of electro-mechanical systems it is desirable to look at the malfunction modes and the associated symptoms that characterize the system. This analysis serves two purposes. It helps to define the characteristic parameters which are required to reveal any extent malfunctions and it also helps to assure that determination of those characteristic parameters by testing is necessary for the purpose of detecting any potential faults.

Hydraulic motion base systems are an interesting case in point in this regard. It is possible to directly instrument many of the properties of the motion base that need to be checked. For example, line pressures, reservoir levels, pressure drops, etc. This direct instrumentation enables verification of these parameters during static checks. Unless other failure modes, which cannot be tested statically, are considered essential for daily testing, there is no requirement to implement an additional dynamic test although the motion base system is a straightforward easily tested hydraulic feedback control system.

Control systems whose performance can be analyzed at least approximately by conventional linear control theory are generally evaluated dynamically in terms of their response to several basic input signals. These common test signals are the unit impulse, the step, the ramp, and the parabolic input. These signals represent different alternates for assessing the transient response of the system. Frequency response measurement is an alternative to transient response analysis. The frequency response of a system is defined as the steady state response of the system to a sinusoidal input signal.

The most common means for specifying the time domain or transient performance of a control system is in terms of its response to a step input. The response of a typical underdamped second order system to a step input is shown in Figure 7.1-1. This figure has been used to identify or define the common, time domain, performance parameters. These are the rise time or time to 100 per cent of the commanded value. For an overdamped system, there is no

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

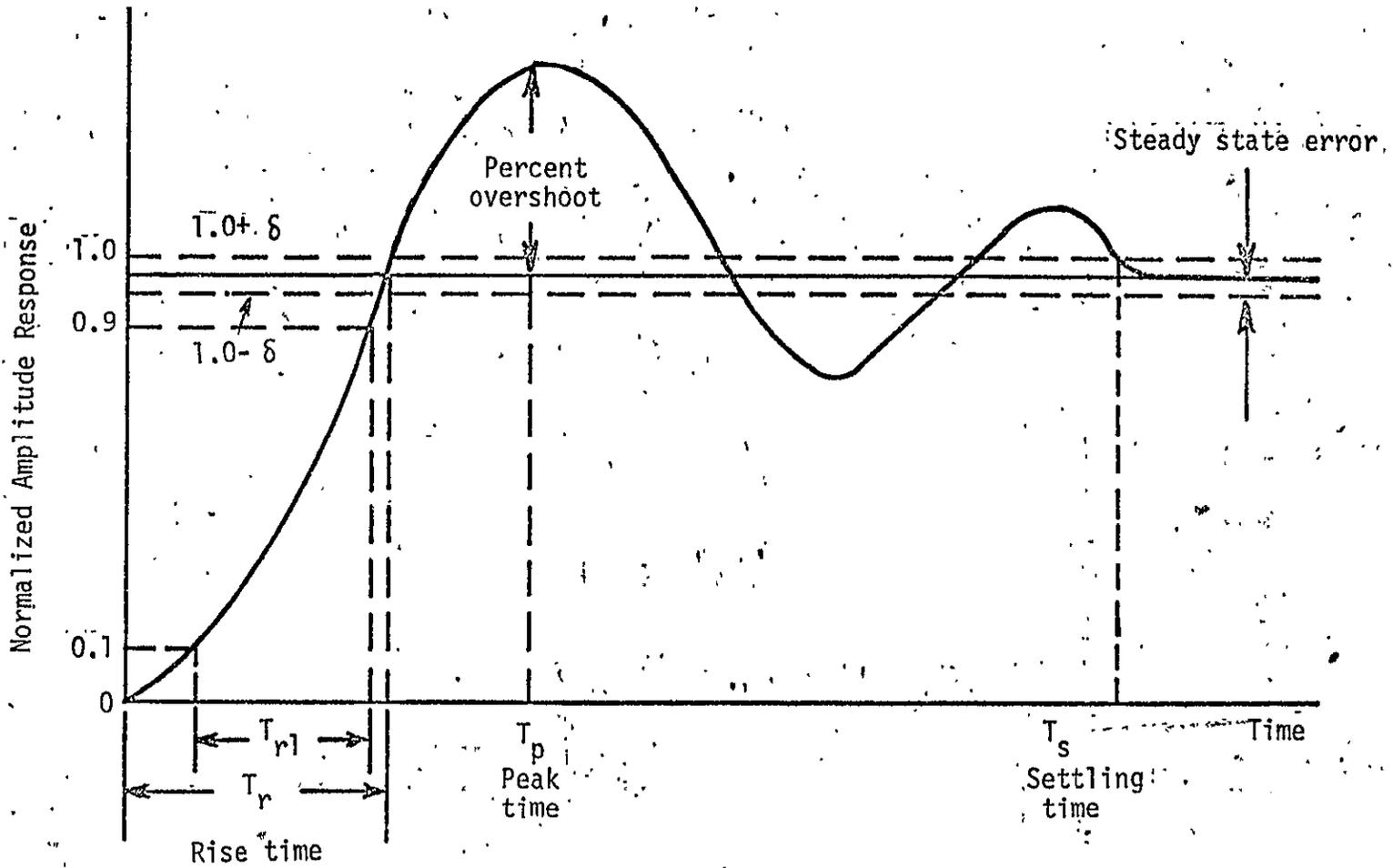


FIGURE 7.1-1 STEP RESPONSE OF AN UNDERDAMPED CONTROL SYSTEM

1 November 1974

overshoot and the rise time is defined with reference to some fixed per cent of the full deflection, for example 10-90 per cent. The remaining parameters for the underdamped system are the per cent overshoot, the time of peak overshoot, and the settling time. The settling time is the time required for the response to become damped within some nominal boundary of the commanded value. Use may also be made of the residual steady-state error.

Although the parameters described above are defined for a linear, second order system, the wide spread use of these parameters is due to the fact that for many higher order and/or nonlinear systems, the dynamic response is quite often primarily impacted by a pair of dominant roots. Consequently, the response of these systems is not entirely different from that shown in Figure 7.1-1.

Readiness testing of electro-mechanical systems, when dynamic tests are necessary, should be adequately addressed by evaluating the parameters described above following the application of a step input as a stimulus. Frequency response testing should almost always be unnecessary for this purpose. One reason for this is that there is a certain amount of correlation between the frequency response of a system and its transient performance. For example, loss of high frequency response will show up in the degradation of rise time. Similarly, the magnitude of the per cent overshoot, also relates to the magnitude of the resonant peak amplitude response in the frequency response characteristics.

The utility of determining the response to a step input of dynamic systems is so great that a utility routine has been flow charted for this purpose and for reference use in other parts of the study. This flow chart is presented in Figure 7.1-2.

The utility, step response test routine STRSP, shown in Figure 7.1-2 is based on the following assumptions as to its implementation and utilization:

- The routine tests a block of devices whose scale factors and other characteristics are communicated to the routine in data arrays.

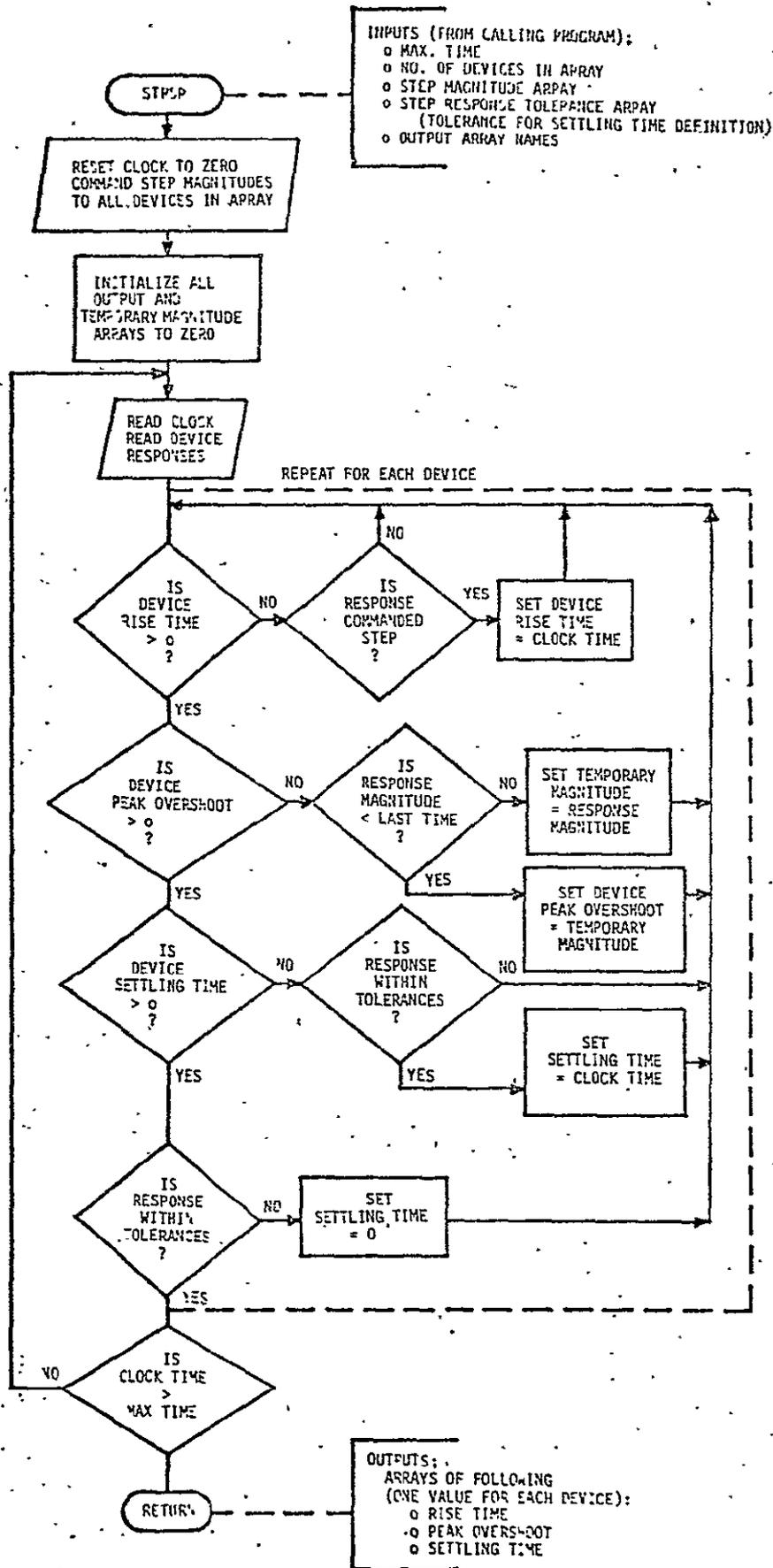


FIGURE 7.1-2 TRANSIENT RESPONSE TO STEP INPUT (STRSP)

- Time reference is provided by an external clock which is programmable to the extent that it can be reset on command.
- The command for clock initialization and the step signal magnitudes are communicated to the equipment being tested concurrently, making the clock an absolute time reference for the test.
- The characteristic parameters determined by the test are communicated back to the calling program in arrays also designated by that program.

The calling program is responsible for processing of the data generated by the test routine, STRSP. For readiness tests this consists only of comparison of the test results with nominal values and tolerances that are available for this purpose.

### 7.1.2 Fault Isolation Techniques

Fault isolation tests or procedures are either special tests or data analyses that are implemented for the purpose of identifying the failed unit in a system or subsystem. For the purposes of the present study we are concerned with isolating failures to what we call the LRU level. An LRU is the last step in the automatic testing process once a fault has been detected. LRU's are defined by their natural modular nature and by the need at some point to remove a unit to a workbench to accomplish the final repair or replacement.

Ideally, it would be desired to perform further analyses on the data obtained from the readiness tests and establish the designation of the defective LRU. In subsystems in which LRU's are combined in circuits with any degree of complexity, this is seldom possible unless the functional performance of the LRU imposes a unique signature on the response obtained from the readiness test.

There are simulator subsystems with many LRU's essentially arranged in parallel. In this instance readiness testing requires testing each of the parallel units and fault isolation is accomplished incidentally to the readiness test. Subsystems of this type are crew station and the instructor operator station.

The fault isolation problem is also minimized when functional testing is implemented at the LRU level. For example, testing for the proper functional operation of a digital subsystem such as a data channel effectively isolates a

fault to as low a level as it is desirable to go with a self-test system.

Like readiness tests, fault isolation tests are sensitive to the type of hardware being tested. The following discussion considers digital electronic circuits, analog electronic circuits and electro-mechanical subsystems.

#### 7.1.2.1 Digital Electronics (Fault Isolation)

The readiness testing of digital circuit subsystems is limited to basic functional testing in Section 5.1.1.1. Functional testing in this instance involves exercising the basic functional elements of the digital subsystem such as the data channels, the interface hardware, the memory, the hardware instruction set, etc. Detection of a fault by any of these tests automatically identifies a faulty functional unit to the lowest level it seems necessary to go for a simulator self test concept.

Techniques that are more suitable for manual or semi-automatic testing of digital systems include hardware probe insertion and the monitoring of discrete bits at certain key interface points. Probe techniques require development even for semi-automatic application and seem far beyond the scope of what should be attempted for a simulator self test system.

The only techniques that seem useful and desirable for simulator implementation are functional testing which effectively isolates faults to specific digital functions and the use of intuitive or heuristic techniques for subsystems such as the data conversion equipment.

#### 7.1.2.2 Analog Electronics (Fault Isolation)

Electronic equipment that processes basically analog signals is characterized by relatively simple arrangements of its LRU's. The characteristic parameters identified for end to end readiness testing of an electronic system are also applicable to testing of the LRU's individually. The basic fault isolation problem for electronic systems is consequently solvable by introducing additional switching to enable testing of each LRU once the readiness test has detected a fault. This switching must of course be implemented in a fail safe manner. That

is, test circuit switch failures should not interfere with nominal simulator operations.

Although analog electronic systems configurations are basically less complex than digital logic networks, it may not always be possible or reasonable to rely on intuitive or heuristic techniques for derivation of the most effective or most direct test sequence for fault isolation. The techniques that are most useful and applicable to the test design problem for fault isolation in electronic systems are graph theory and related modern network theory. Modern network theory and its relation to graph theory is very clearly presented in Reference 7.1-1.

The basic problem that these techniques can address is that of test design. Since we have restricted readiness testing to end to end testing, we must identify the test design problem addressed by these techniques to be fault isolation. The advantage of using these techniques is that they can identify methods for avoiding the necessity of inserting test signals before each LRU and sensing response immediately past the LRU. In other words, the application of these techniques, when warranted, can potentially avoid the addition of switching to effectively enable testing each LRU individually. If the system is relatively simple, the necessary switching is not generally a problem. However, as the system complexity increases the switching requirements and the number of separate tests that are required can increase accordingly and reach a level where the application of somewhat more sophisticated test design techniques pay off.

#### 7.1.2.3 Electro-Mechanical Systems (Fault Isolation)

Fault isolation for electro-mechanical systems is of primary interest with respect to the servo drives for the models and cameras in the visual simulation and for the motion base. The servo driven instruments in the crew station and the IOS are themselves LRU's and readiness tests effectively accomplish fault isolation at least to the level of interest for simulator self test.

If the response to a step input is assessed as part of the readiness test for these remaining systems, then some insight is already available as to the type of failure that is present. This insight is attributable to the fact that

The servo systems for the models and the motion base are relatively straightforward control systems whose transfer functions are known and whose performance is relatively linear and corresponds with what is predicted by linear control theory.

This insight is available to the control systems engineer but not necessarily to the technicians or operators responsible for operation and/or maintenance of the equipment. Consequently, it may be desirable to implement additional computerized analyses to provide the operator or maintenance crew with the same insight. There are several techniques well documented in the literature for this purpose. These techniques are not very complex but because of the level of detail required with respect to particular subsystem description, it has not been feasible to carry out detailed example analyses for simulator applications.

The fault isolation techniques for electro-mechanical systems that are of interest for simulator applications are based on the further analysis of the frequency response data for the tested systems. The following assumptions are implicit in both of these techniques:

- The system nominal transfer function is known
- The system performs linearly in the region of the test
- The system is relatively simple

The requirement for relative simplicity cannot itself be delineated very precisely. However, as the techniques are considered, the limitations on complexity should become evident. One interpretation of simplicity is that the systems performance be primarily governed by a dominant set of roots of the characteristic equation or effectively by a lower order dominant transfer function. This is simply the equivalent of saying that the presence of many higher order terms in the transfer function does not rule out the use of these techniques as long as the performance of the system is dominated consistently by the lower order roots and the others may be neglected.

The two techniques to be described may be labeled the "transfer function coefficient derivation method" and the "frequency response pattern recognition method"

### Transfer Function Coefficient Derivation Method

The frequency response of a system whose transfer function is known is readily computed using basic control theory and may be plotted in a Bode diagram. The breakpoints in the linearized frequency response are, of course, at the frequencies corresponding to the locations of poles or zeroes of the transfer function. The changes in slope of the amplitude response are governed by the following two properties:

- At each simple pole or zero, there is a 20 db/decade change in slope [plus (+) for a zero, minus (-) for a pole]
- At each complex pole or zero, there is 40 db/decade change in slope [again plus (+) for a zero, minus (-) for a pole]

The coefficient derivation method is based on analysis of the linearized or line segment approximation to the frequency response of the system as follows. If the breakpoints in this frequency response are known (and they are readily computed) for the nominal system, then it is possible to select several test frequencies between each breakpoint and obtain the amplitude response by testing. From the test data, it is possible to fit linear segments of the amplitude response between each breakpoint. The intersections of these linear segments are now the new breakpoints of the system's transfer function. Based on the sign of the change in slope at the breakpoint, the breakpoint frequencies may be inserted into the experimentally derived transfer function as either poles or zeroes.

From an analysis of the system, the nominal numerator and denominator polynomials are known. The coefficients of the terms in these polynomials are simple algebraic expressions involving the basic physical properties of the system. For example, these properties are gains, coefficients of friction, spring constants, etc. Once the actual transfer function is derived from test data, it is only necessary to solve the simultaneous algebraic coefficient expressions backwards or inverted in order to determine what the values of the physical properties are currently. These physical property values may then be checked against the acceptable values and tolerances for these parameters to isolate to the characteristic that has degraded sufficiently to cause a failure in performance level during the readiness test.

The process for accomplishing this analysis, including the portion that may be implemented on the computer, is flow charted in Figure 5.1-3. Further information on the selection of test frequencies, as well as example problems, may be found in References 7.1-2 and 7.1-3.

#### Frequency Response Pattern Recognition Method

The coefficient derivation method literally solved for the system properties based on its present performance. The pattern recognition method effectively accomplishes the same degree of isolation but on a more experimental basis. The results require substantially less computation during the development as well as after implementation.

The pattern recognition method also considers the impact on the Bode diagram of changes in values of system parameters. The approach in this case is as follows.

First, a limited set of test frequencies are selected, based on reference to the nominal frequency response properties of the system. Next the amplitude and phase response characteristics of the system are determined for each test frequency for some substantial deviation from the nominal value for each property that enters into the transfer function. These parameters would again be gains, spring constants, friction terms, etc. The determination of the impact of the component deviation may be established analytically or by performing a small parametric evaluation with a simulation of the system. Knowing the effect on the amplitude and phase response, it is now possible to construct a table. The table can be constructed by listing the test frequencies along one side with provision for phase and amplitude characteristics at each frequency. The components of the system are listed across the other dimension of the table. If components can fail with either positive or negative characteristic changes than both contingencies must be allowed for. Table 7.1-1 is an example of what such a table could look like. The table is filled out with whole numbers because the fault levels at the bottom of the table were divided into the phase and amplitude deviations and the result rounded to the nearest whole number. The same normalization can, of course, be done for the data obtained from a test.

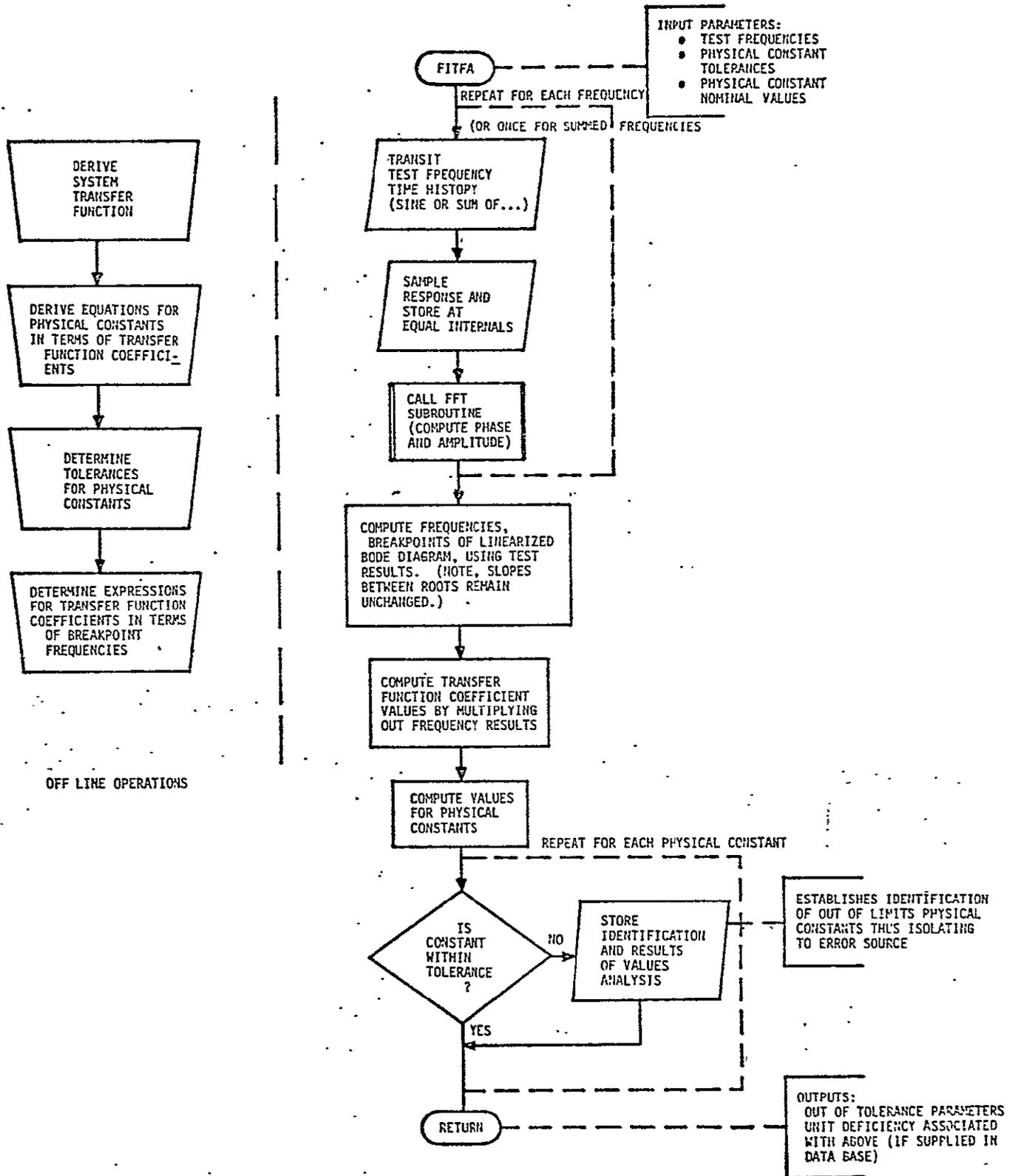


FIGURE 7.1-3 FAULT ISOLATION BY TRANSFER FUNCTION DERIVATION

TABLE 7.1-1 TYPICAL FAULT DICTIONARY FOR A SIMPLE RC NETWORK\*

FREQUENCY RAD./SEC.		-50% CHANGE							+50% CHANGE						
		R <sub>1</sub>	R <sub>2</sub>	R <sub>3</sub>	R <sub>4</sub>	C <sub>1</sub>	C <sub>2</sub>	C <sub>3</sub>	R <sub>1</sub>	R <sub>2</sub>	R <sub>3</sub>	R <sub>4</sub>	C <sub>1</sub>	C <sub>2</sub>	C <sub>3</sub>
10	AMP RATIO	1	-1	0	0	0	0	0	-1	1	0	0	0	0	0
	PHASE	0	0	0	0	0	0	0	0	0	0	0	0	0	0
40	AMP RATIO	1	-1	0	0	0	0	0	-1	1	-1	0	0	0	-1
	PHASE	0	0	1	0	0	0	0	0	0	-1	0	0	0	0
90	AMP RATIO	1	-1	1	0	0	1	1	0	0	-1	0	0	-1	-1
	PHASE	0	0	1	1	0	1	1	0	0	-1	0	0	0	-1
200	AMP RATIO	0	0	1	0	-1	1	1	0	0	-1	0	1	-1	-1
	PHASE	0	0	1	1	0	1	1	0	0	-1	0	0	-1	-1
500	AMP RATIO	0	0	1	1	0	1	1	0	0	-1	0	0	-1	-1
	PHASE	0	0	1	1	0	1	1	0	0	-1	-1	0	-1	-1
3000	AMP RATIO	0	0	1	1	0	1	1	0	0	0	0	0	0	0
	PHASE	0	0	1	1	0	1	1	0	0	0	-1	0	-1	-1

AMPLITUDE RATIO FAULT LEVEL = .025  
PHASE ANGLE FAULT LEVEL = .0785 RAD.

\* This table, based on Reference 5-2, shows the patterns of performance deviations for  $\pm 50\%$  component value deviations in a simple RC network. The round numbers appear because the actual deviations have been divided by the fault levels noted, and the results have been rounded to the nearest whole number.

When the normalized data from a test is obtained, then the results at each frequency can be compared with the results obtained experimentally. This is an extremely simple computation and immediately pinpoints the likely source of the fault. The question of multiple failures immediately arises and admittedly the approach as outlined is oriented toward single point failures. The coefficient derivation method previously discussed does not have any limitations on the number of concurrent deviations that can be detected and would be a more appropriate choice if this situation is likely to occur frequently.

The pattern recognition technique lends itself readily to the solution of simple problems and to small studies that assess alternate options or modifications to the basic techniques. Consequently, there are a large number of papers that discuss this approach and its various ramifications at great length. Some of the more interesting discussions may be found in References 7.1-2 and 7.1-3. A flow chart of the entire process is shown in Figure 7.1-4.

#### Frequency Response Analysis for Above Fault Isolation Techniques

The two techniques discussed above are both based on further analysis of Bode diagram information. The frequency response data for constructing a Bode diagram may be derived from tests conducted in a number of ways. If one test frequency is applied at a time and the response of the system measured, the data is ready for plotting immediately. However, if more complex signals are used to reduce test time, for example, or if normal operating signals are monitored for purposes of frequency response analysis, then certain data reduction techniques may be required. These techniques are discussed in Section 7.3. In the following Section, we address the problems associated with incipient fault detection.

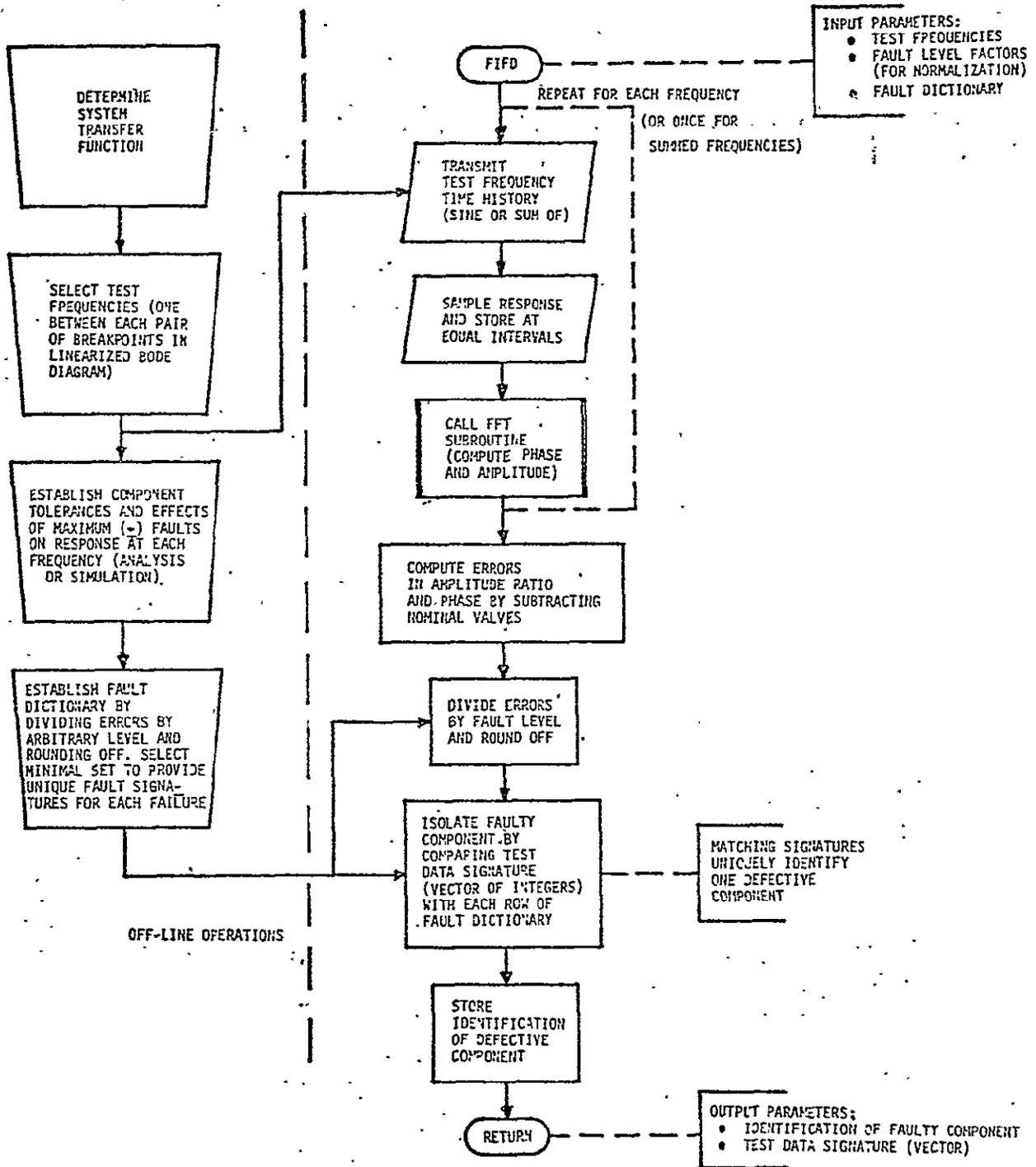


FIGURE 7.1-4 FAULT ISOLATION BY FAULT DICTIONARY (FIFD)

### 7.1.3 Incipient Fault Detection Techniques

Readiness tests are designed to detect existing faults in a system. A fault is an existing defect or deficiency which prevents the tested system from achieving a satisfactory performance level. An incipient fault is a fault that is likely to happen in the near future and whose occurrence we would like to plan for. That is, we would like to order replacement components or schedule maintenance at a time when it does not interfere with our normal everyday operations. Since readiness tests are intended to be executed daily in order to assure proper operation of the simulator for the days training activities, it is most advantageous if we can concurrently acquire any additional data that may be needed to obtain a warning of a fault about to happen, or at the least, look at the data obtained in a more quantitative sense than the simple go-no go condition.

We have identified a number of basic techniques for detecting incipient faults. These techniques are applicable to all of the analog circuits and systems and to the servomechanisms in the various simulator subsystems. We have not found any techniques useful for digital electronics or for discrete signal lines. The primary difficulty with discrete elements is that failure prediction techniques are in one way or another concerned with the performance degradations that may be observed. With any system that exhibits only two states as opposed to a continuously varying, infinite variety of states, it is impossible to detect trends or gradations.

Since the incipient fault detection techniques discussed are only applicable to analog electronics or electro mechanical systems, we have not organized this section by system type. Instead, we proceed directly to a description and evaluation of the various incipient fault detection techniques. These are overstress testing, marginal testing, gray area performance evaluation and degradation rate analysis.

#### 7.1.3.1 Overstress Testing

Overstress testing is based on the operation of the tested system or LRU at the high end or beyond the high stress limit of its operational band. The

Overstress operation must, of course, be kept within safety limitations that are established for safe operation. For simulator applications, we are not so much interested in exceeding structural limitations, which is what the term stress tends to imply, but rather power limitations and possibly frequency limitations.

The bandwidth quoted for systems is conventionally taken to be the lowest resonant frequency of the system. If the system is only required to have linear response out to that bandwidth, there may not be a performance level specified at the higher frequencies. However, testing in this area could be useful for incipient fault detection and could be considered as one type of overstress testing, although it would not be expected to induce any failures.

More frequently, overstress testing might be applied in terms of excessive electrical power levels. Operation at higher than normal voltage levels can cause marginal components to burn out at a time when repair or replacement is more opportune. For simulator applications, this type of approach is more suitable for the regular maintenance shift when it is appropriate. There is no requirement for data storage or processing and no value in automation of the procedure. In addition, it is definitely an undesirable approach to use as part of a readiness test just prior to a training session.

#### 7.1.3.2 Marginal Performance Testing

Marginal performance testing is based on operation of a system at a marginal performance level such that minor irregularities in performance may be revealed. Examples of this technique would be commanding an extremely low rate motion from a servo and then monitoring the servo response to detect irregularities in motion or the threshold command levels required to stimulate a response. Either of these approaches should reveal high friction levels or potential motor degradations that could result in a subsequent failure.

Marginal performance testing is in distinct contrast with readiness testing requirements. Readiness tests, as defined herein, are concerned with

determining if a system is functioning properly. This implies testing the unit over its normal performance range, if not at its upper performance limits. Consequently, the use of marginal performance testing implies the execution of separate tests expressly for the purpose of incipient fault detection and of no value to the basic readiness test requirement.

Although this type of testing must be recognized as one approach that can be implemented when nothing else will suffice, it is recommended that either of the remaining two types of incipient fault detection techniques be applied first. Although directly applicable experience data is not available, it is anticipated that the following techniques will be adequate for future simulator subsystems.

#### 7.1.3.3 Gray Area Performance Degradation

The simplest approach for incipient fault detection for either electronic (analog) or electromechanical subsystems is the detection of the fact that a subsystem has moved into a gray performance level margin or boundary and its performance in the near future is likely to drop below acceptable levels. In order to do this, it is primarily necessary to store in the data base an additional set of tolerances for each characteristic parameter. During the readiness test performed before each training session, the units performance is compared not only with an absolute tolerance level which establishes the unit as failed, but the performance is also compared with a second tolerance which, if exceeded, says that the device is going to exceed its required performance limits within some predetermined time period, such as 10 or 30 days.

The "gray area performance" technique has some strong advantages, which include the following:

- Data base requirements are minimized by the need to store only one additional tolerance for each characteristic parameter.
- Computer software development and computer computations are minimized.

There are also several disadvantages for the "gray area performance" techniques. These disadvantages include the following:

- Complete knowledge of the systems performance degradation characteristics must be available before the technique can be implemented.
- The systems performance degradation characteristics must be predictable and well behaved; otherwise either false warnings or "too late" warnings are likely to be received.
- If the test data analyzed are noisy or erratic on a day to day basis, false indications will be generated.

The gray area performance degradation technique may be adequate for a number of reliable subsystems or components but, nevertheless, a more sophisticated technique may be required. The final approach to be discussed resolves some of the difficulties mentioned above and has been designated the "degradation rate analysis technique".

#### 7.1.3.4 Degradation Rate Analysis Technique

The incipient fault detection requirements for training simulators are oriented to relatively reliable subsystems or components. That is, we are concerned with units that may fail once or twice a year at the most, and for which failures we would like somewhere between 10 and 30 days warning. This time scale is of interest and a factor in predicating a degradation rate analysis technique inasmuch as it must be considered in determining over what sort of elapsed time period data should be accumulated and retained for failure prediction. The accumulation of data, of course, impacts data base requirements.

The degradation rate analysis technique consists simply of storing performance data from day to day and each day reviewing the accumulated history of the units performance levels and using this information to predict a failure date. The most reasonable algorithm that suggests itself for processing the data is a least squares curve fit of a linear or first order curve. The daily sequence would proceed as follows:

- Execute readiness tests and store necessary performance characteristics.
- Store data for degradation rate analysis by shifting all data points back one location in the data array in order to drop the oldest data point and make room for the new one.
- Call a least squares curve fit routine to fit a linear curve to the total set of data.

- Check the slope of the resulting curve. If negative (performance decaying), continue. If positive, select a more recent set of data, i.e., the most recent half of the array. Repeat curve fit and slope check.
- Extrapolate curve to find date at which unit performance will exceed its tolerance.
- Check predicted failure date against warning require reference.
- If failure is within critical period, print warning.

This entire procedure is flow charted in Figure 7.1-5.

There are a number of advantages for this type of technique that are in contrast to the disadvantages of the previous approach. These consist of the following:

- The linear curve fit technique filters out perturbations that result from day to day test variations.
- There is no need to know what the performance degradation history for a unit is likely to be.
- There is no input data required that corresponds to the gray area tolerances of the previous method.
- Experience acquired from this technique can be used to eventually implement the gray area technique and thereby conserve computer and data base resources.

There are several disadvantages to this technique that should not be overlooked. These consist of the following:

- Storage of data for any period (10, 20 or 30 days) impacts the data base requirements substantially.
- The accuracy of the prediction, especially the timeliness, is going to be sensitive to the choice of sampling period. That is, a rapid ten day decline is going to be moderated by the curve fit if the data base goes back 30 days.

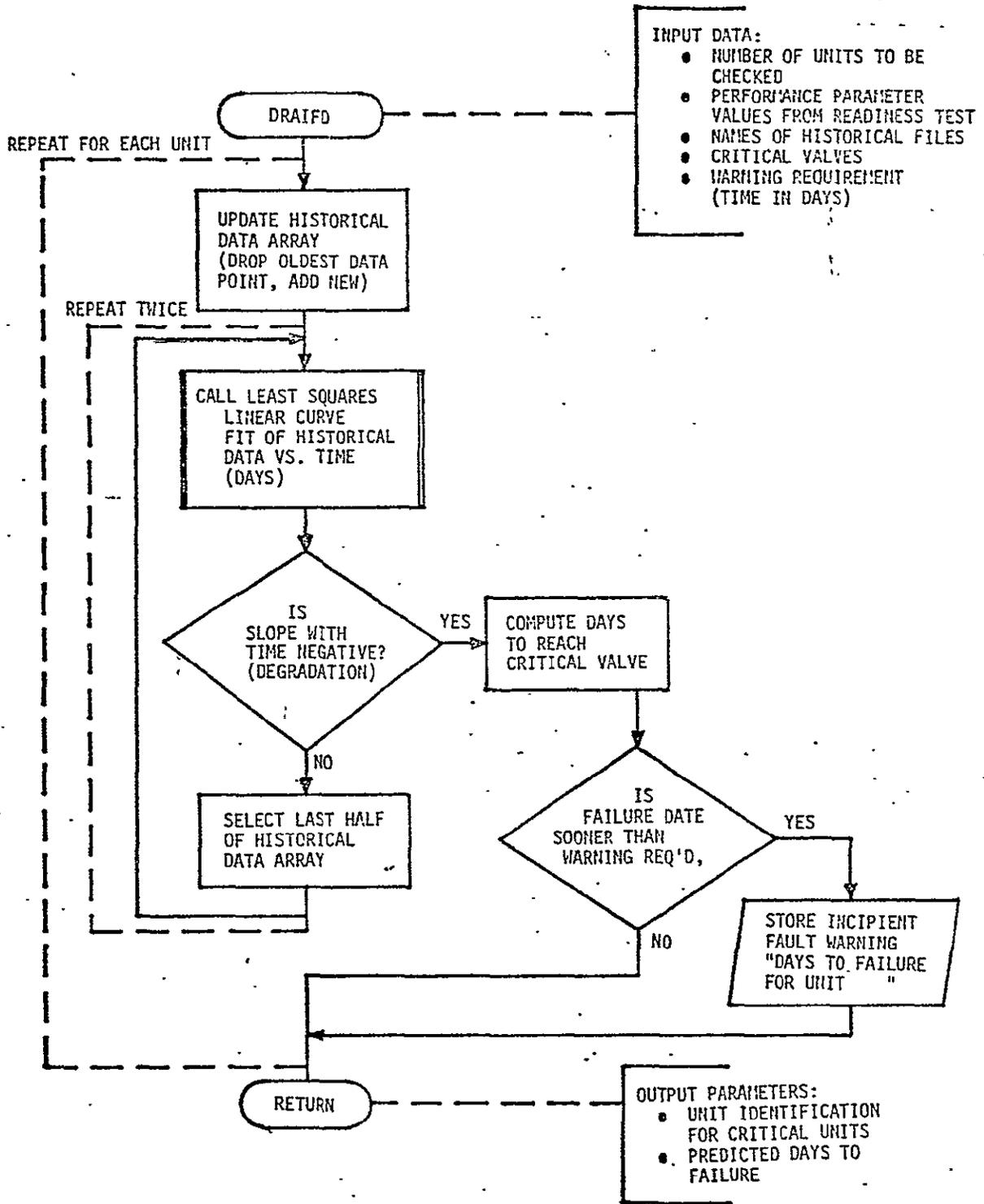


FIGURE 7.1-5 DEGRADATION RATE ANALYSIS FOR INCIPIENT FAULT DETECTION

Degradation rate analysis techniques require development for any units to which they are applied. Although the computer resource requirements are large, they need not be large indefinitely, since well behaved systems may be converted to the gray area technique as historical data is accumulated.

## 7.2 TEST IMPLEMENTATION TECHNIQUES

Test implementation techniques are considered for the present report to include all of the techniques utilized in implementing a test that is required to fulfill one of the objectives discussed in the previous section. These objectives are fault detection, fault isolation and incipient fault detection. Test implementation is an extremely broad subject if addressed generically in this context. Consequently, the bulk of the test implementation analyses accomplished during the course of the present study are presented in the context of the particular systems addressed. That is, test implementation is the major concern in addressing data conversion equipment testing or motion base testing or TV system testing or whatever the subsystem might be. In this section, we present a limited amount of information on several topics that apply to all of the simulator equipment tests or to substantial numbers of them.

The topics discussed in this section are test timing, test signal generation and test data acquisition.

### 7.2.1 Test Timing Techniques

The alternative techniques for obtaining time reference information that were considered included the following:

- Internal computer clocks
- Interrupt generation by simulator central timing unit
- Programable external clock

These alternative techniques were explored in terms of how they would affect the implementation of the test software, what the impact of test changes might be on the test software, and what would be the accuracy and reliability of the time information associated with the data obtained.

The use of internal computer clocks seems a low accuracy and low reliability approach for several reasons. First, the feasibility and the approach for its implementation are very highly sensitive to the particular make and model of computer system that is controlling the test. Secondly, in the simulator configurations considered for the study there is always assumed to be an interfacing digital computer, probably a minicomputer, between the host computer and any simulator equipment. If the test is controlled from the host then there is

a substantial time delay between the time at which the host commands an output or receives a data word and the time at which that information was active in the simulator hardware. In addition, unless only one device is addressed at a time, this time lag is likely to be variable depending on the amount or quantity of information that was communicated in the data transfers. Consequently, the disassociation of the time reference in the computer and the time of events happening at the simulator equipment was considered an undesirable and unpredictable situation that is best avoided.

The central timing equipment for the simulators will generate interrupts at various time intervals to control and synchronize various intercomputer data transfers and to maintain a common real time reference during the actual simulation. One approach to timing test operations is to have all test events triggered by an interrupt generated by this equipment. This results in a test software load that is in some ways similar to the simulation software load and presents a familiar programming problem to the simulator programming support personnel. Unfortunately, interrupt processing does not provide a time reference in the absolute sense. The time must be calculated in the computer by updating a time counter. Concurrently, the interrupt can trigger the sampling of the analog signals by the DCE. This has the potential of providing a good time reference for data acquisition. However, if the computer commands a stimulus, the time at which the stimulus arrives at the equipment being tested is known accurately only to an accuracy of one interrupt interval. Interrupt processing is of course vulnerable to the peculiarities of each computers interrupt processing logic.

Implementation of a programmable digital clock external to the computing equipment is an inexpensive and simple alternative. A programmable clock can be implemented by using a high frequency pulse rate from the central timing unit to drive a counter which can be assembled from commercially available logic circuit elements. The interrupt pulses used to drive the counter can be of much higher frequency than would probably be used for interrupt processing as described in the previous paragraph and the clock would therefore have inherently better resolution. The programming requirement is only to have the ability to reset to zero on command. The clock output can be tied permanently to a register in the DCE and that register can be addressed and read every time that test data is

sampled from the analog or discrete channels. The programmable reset feature enables the clock to be reset every time a transient response test is initiated. This makes the clock time an absolute time reference from the time of the test signal input. The reset feature is unnecessary for frequency response testing because an absolute time reference is not needed. The clock will cycle through its own reset point when it reaches the limit of its count which will require some software logic to adjust accordingly.

The external digital clock is simple, inexpensive and insensitive to the type of computer system incorporated in the simulator. It can also be used as a time reference for checking the elapsed time of digital processes independent of the computer clock. This affords further opportunities for simulator timing checks.

### 7.2.2 Test Signal Generation

The classic stimulus/response approach for testing a component or subsystem assembly requires the application of external inputs to the unit under test, and the subsequent monitoring, measuring, and analysis of the outputs elicited. This section discusses various techniques for generation, distribution, and application of test stimuli.

There are five basic types of test stimuli used in the process of hardware verification and checkout. These are analog DC signals, repetitive continuous signals, transient test signals, special purpose digital patterns, and noise signals. Each of these signals may be generated by either software or hardware techniques. Software techniques, in the context of a simulator configuration, refers to generation of the signals in the host or ancillary computers and routing to the simulator equipment through the DCE if the final signal is analog. In the following discussion we address the requirements for all five types of signals and the alternative means for generating them.

#### 7.2.2.1 Analog DC Signals

The analog DC signals are non-repetitive but may be time varying at a very low rate. Analog DC signals are used to test discrete channels to such devices as indicators or flags, as well as meters and servo systems including the motion

base servos. Analog signals can be used to check the data conversion equipment channels for linearity as well as for calibration.

Analog DC signals could be generated by special purpose or already available hardware, for example, by switching signals from a programmable calibrated power supply. However, in the context of the simulator configuration, commanding analog outputs on the DCE channels in the same manner as regular outputs on those channels is most economically and expeditiously accomplished by the test software.

#### 7.2.2.2 Repetitive Continuous Signals

Repetitive signals of interest include sine waves, square waves, sawtooth waves and other wave forms that are commonly generated by test hardware signal generators for test purposes. There are two sets of test signals in this category that are of particular interest for purposes of simulator self test. These are the low frequency sine wave signals required for frequency response testing of the servo systems in the visual simulation and for the motion base, and the complex high frequency wave forms required for video system self test.

Low frequency sine waves and other basic wave shapes can be generated by software and transmitted to the simulator hardware through the regular DCE channels. Although the converted signal will be stepwise discontinuous, the maximum conversion rate of the D/A converters is so high, about 10KHz, that it will not be detected by the low frequency systems to be tested. The natural frequencies of the servo systems should be no more than several hundred Hertz even for the pitch and yaw servos in the probe. The irregularity of the D/A converted signal will not be perceptible to these devices.

The high frequency wave forms required for the TV system tests are so high in frequency, 20 MHz, that there is no need to consider software techniques for this purpose. Instead, it is necessary to incorporate a standard television test pattern generator into the test equipment. Switching may be required for turning on power or selecting the desired test patterns but this can be accomplished with commercially available solid state switches.

### 7.2.2.3 Transient Test Signals

The primary signal of interest for transient response testing is the step function. Fortunately, this signal can be generated accurately by transmitting the required step magnitude through the appropriate D/A channel. The low natural frequency of the simulator systems being tested insures that the signal that arrives will be sensed as an instantaneous step. This is also an aspect enhanced by processing through the D/A hardware. On one cycle the signal communicated will be the zero level. On the next cycle, the signal is the step amplitude. The settling time of the D/A is approximately the rise time of the step and should be about 100 microseconds.

### 7.2.2.4 Digital Patterns

Digital patterns required for testing of digital interfaces, or addressing and conversion hardware in the DCE, can be generated either by hardware or software. In the simulator environment, the use of the digital computer to generate these patterns for testing everything (except maybe itself) seems like the logical choice. However, if a system interface hardware develops a troublesome history, it may be desirable to introduce a hardware pattern generator. There are available programmable digital pattern generators that can be programmed to issue 1 to 1024 bits of data either in a serial stream or in parallel groups of words of up to 16 bits each.

### 7.2.2.5 Noise Signals

White noise signals have an equal amount of energy in every unit of the frequency bands within the range of interest. White noise signals are used in either a continuous or impulsive mode in order to verify the response of a component or subsystem to different frequencies of input signals.

The use of noise signals for frequency response testing permits concurrent verification of response to various frequencies with application of a single stimulus. In order to obtain the same response information with individual frequency inputs, many single frequency stimuli would be required, with the number depending on the range of interest.

The principal disadvantages of the white noise signal are that it distributes the signal energy over all frequencies in the bandwidth which adversely effects the signal to noise ratio that might otherwise be achieved if only discrete frequencies were of interest. Since a relatively limited number of frequencies are likely to be considered for subsequent processing, this factor may be significant when practical applications are considered. The filtering of unnecessary high frequency signals is a requirement for limiting the noise in the test process.

White noise with a bandwidth of several hundred cycles, which is all that is required for potential simulator subsystems tests, can be readily generated by software techniques and communicated to the systems being tested through the available DCE.

### 7.2.3 Test Data Acquisition Techniques

As previously noted, test data acquisition is a primary topic of the analyses of each of the simulator subsystems and components addressed in this study. In this section a few general and broad aspects of test data acquisition are considered inasmuch as these factors are applicable to all of the study hardware self test activity.

The three areas which seem appropriate to address are sensors, switching, and data storage.

#### 7.2.3.1 Sensors

Early in the study it became apparent that sensing of the response of simulator subsystems to commands from the computer was going to be a primary problem area. Consequently, sensor technology was given a considerable amount of attention. Assembly of generic sensor data could easily reach encyclopedic proportions and is best left to other sources such as the industry directories for sensing equipment and transducers.

The basic reason that sensing is a problem area is that simulator self test requires observing and recording parameters that are intended to be observed and recorded manually if necessary by a crew member. In addition we are concerned with devices that were designed to be manipulated by the crew member which also created a problem of automatic manipulation in a calibrated manner.

The manipulation of controls can be addressed largely by servomechanisms and solenoids but sensors are available in many more varieties. Sensors considered for the present study included the following generic types:

- Mechanically activated
- Magnetically activated
- Light activated
- Thermally activated
- Current activated
- Pressure activated (fluidic devices)
- Acoustically activated

A more detailed discussion of sensors of interest may be found in Section 8.3.1.2.

#### 7.2.3.2 Switching Techniques

The switching requirements for implementing self test techniques for the various simulator subsystems are substantial. Particular attention was paid to this area in the development of the data conversion equipment self test techniques. Switches considered included cross bar switches, reed relays, magnetic switches and solid state (MOS/FET) switches. The latter seemed the most suitable for reliability, cost and failsafe characteristics. Switching concepts are discussed further in Section 8.2.3.

#### 7.2.3.3 Data Storage (Temporary)

There was no evaluation made of any means of storing large volumes of data for either a long term or short term (minutes) basis. The nature of the tests anticipated is such that the temporary storage capacity of the computer memories available is not strained. Incipient fault detection data is assumed to be stored in a data base system with media undefined. The accessibility of the incipient fault detection data to the computer on a daily basis is a prerequisite for implementation of the analyses as planned. However, the data base requirements are not expected to be so great that this should present any major problem with respect to the availability of resources for data storage.

### 3 TEST DATA PROCESSING

In Section 7.1 we addressed the various types of tests that were of interest to this study and then considered the techniques that were found during the course of survey activities for accomplishing the objectives of those tests. It was established that fault detection for the low frequency dynamic systems could be accomplished with transient response testing. Incipient fault detection was also addressed by the same techniques. However, fault isolation for these systems required analysis of the frequency response data. In addition, it is anticipated that check out of the aural simulation system will be based on a harmonic analysis of the various sounds being simulated. Both of these objectives require the definition of frequency response data for the systems involved.

In this section we address the problem of obtaining the frequency responses of a system from test data. If the frequency response test is performed by stimulating the tested system with one frequency at a time, then there is no data processing required to arrive at the frequency response. However, if the frequencies of interest are mixed (summed) to reduce the time required for implementing the test, or if actual run time data is logged at opportune moments to obviate the need for operating the system solely for test purposes, then more sophisticated processing techniques are required merely to arrive at the frequency response data required for subsequent fault isolation analysis. In addition, definition of the spectral signature of the aural simulation will require decomposition of complex sounds into their spectral components in order to assess the correct simulation of the source sound signature.

Consequently, in this section we consider the application of the Fourier Transform, its implementation with the Fast Fourier Transform algorithm, the error sources that must be considered in sampling data for Fourier analysis, and finally the techniques for deriving the frequency response of a system from normal operating time histories logged at an appropriate time in a simulator training exercise.

#### 7.3.1 Frequency Response from Sampled Dynamic Test Data (Fourier Transform Techniques)

The traditional method of generating frequency response or a Bode plot is serial. The system is forced with a pure sinusoid at a certain frequency until reaching a steady-state oscillation, at which time the gain and phase are recorded. The process is repeated for as many frequency points as desired. In certain instances, it may be more convenient to generate a frequency response

1 November 1974

using a broad-band test signal, containing components over the entire frequency range of interest. This could be either a sum-of-sinusoids signal, Reference 7.3-1 or a "white" pseudo-random noise signal, References 7.3-2 and 7.3-3. For a linear system (or a non-linear system driven in its nearly-linear range), the response to each frequency component of the composite signal will be the same as its response to that signal supplied in pure form; therefore, the complete frequency response can be developed by a frequency analysis of the system's response to the composite signal.

This frequency analysis, or decomposition of the composite response into its harmonic components, can be accomplished by use of discrete filters or a single sweep filter, but today is more commonly implemented in software, by use of the Fast Fourier Transform (FFT), described in the following subsections.

#### 7.3.1.1 Fourier Transform Definitions

The basic Fourier transforms for a time function  $f(t)$  are given by

$$F(i\omega) = \int_{-\infty}^{\infty} e^{-i\omega t} f(t) dt$$

$$f(t) = \int_{-\infty}^{\infty} e^{+i\omega t} F(i\omega) d\omega$$

provided that the integrals exist. If the function  $f(t)$  is non-periodic, then its transform (spectrum) will be continuous. If the function is periodic with period  $T$ , then integration need only be performed over a single period of the function, and its spectrum will consist of discrete points (a "line spectrum"). In practice, there is some difficulty in obtaining a sample record whose length is precisely equal to one or more complete periods of the time function; this problem is discussed in the next subsection.

For digital processing, the time function must be sampled. The transformation is then implemented by the Discrete Fourier Transform (DFT), in which the integrals are replaced by sums. The computation time required for straightforward evaluation of the DFT (proportional to  $N^2$ , where  $N$  is the number of samples) tended to inhibit its application until the development of the Fast Fourier Transform (FFT). The FFT algorithm takes advantage of certain symmetries in the trigonometric

functions, making the computational load proportional to  $N \log N$  instead of  $N^2$ . When the number of samples is a power of two, the computational savings are particularly significant; e.g., a factor of 200 for  $N=1024$ . (A thorough derivation of the FFT algorithm, as well as FORTRAN listings, is provided in Reference 7.3-4 or Reference 7.3-5.)

The Fast Fourier Transform subroutines are applied by setting up data arrays in the next higher level program, that is, the one that calls the FFT subroutine. These arrays include the sampled data time history, as well as arrays for storage of the results of the Fourier transform analysis. When the subroutine is called it computes the FFT and stores the amplitude and phase angle relationships for each frequency derived from the sample. The correlation of the frequency data with real time frequencies must be accomplished by the calling program which only needs the real time sample rate at which the samples were taken to do this.

#### 7.3.1.2 Sampled Data Error Sources and Their Correction

Certain error sources are introduced into sampled data analyses because the sampled data is only an approximation to the continuous signal from which it was derived or because the duration of the sample must be limited rather than infinite. In addition, there is the problem of noise commonly found in the real world, that has a tendency to upset analytical expectations. Consequently, for the frequency response analyses described in the previous subsection, we need to be concerned with several error sources. These are noise, aliasing and leakage.

##### Noise

Over long time spans, random noise will contribute spurious components to the output spectrum, mildly distorting its the shape. For short data records, however, significant sampling error in the frequency components of the noise will badly distort the spectrum in an erratic manner. Thus it is important to remove as much of the noise as possible before FFT processing.

The basic method of removing noise is by low-pass filtering of the continuous waveform, before sampling and digitizing. The breakpoint on this filter must of course be established as a compromise between allowing noise to pass, and

attenuating the higher frequencies present in the signal (see also the discussion of aliasing, below). If pre-filtering of the signal is not feasible, some smoothing can be achieved after the sampling and digitizing, by first differentiating and then integrating the digital data. (Since this operation will destroy the amplitude reference, it is necessary to save the first amplitude data point before differentiating, and then use this point as an initial condition in integrating.)

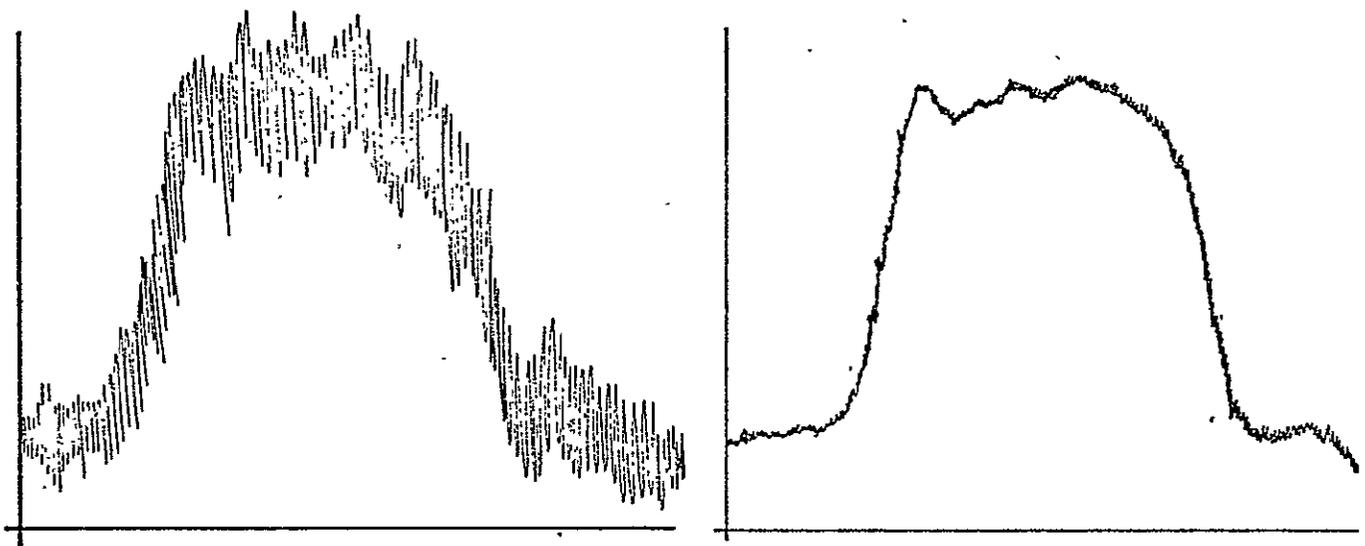
For repetitive data which are synchronously triggerable (e.g., TV waveforms referenced to a line-sync pulse), averaging is a powerful technique for noise attenuation. The randomness of the noise implies that noise samples in different records will cancel, making the underlying signal easier to detect and analyze, as shown in Figure 7.3-1. Assuming perfect synchronization of consecutive data records (i.e., negligible time jitter), the theoretical improvement in signal-noise ratio is  $\sqrt{M}$ , where M is the number of data records averaged together. For example, a signal derived from the average of 100 data records will show a factor-of-ten improvement in signal/noise ratio.

### Aliasing

In sampling data, it is essential to sample the incoming data at a sampling frequency at least twice the highest frequency present in the data. Power present at high frequencies will not be filtered out by the sampling process. It will be "folded over," and appear as spurious power at lower frequencies. This phenomenon, known as "aliasing," is explained by Figure 7.3-1. This figure shows two pure sine waves, differing in frequency by a multiple of  $1/T$ , which cannot be distinguished from each other by samples at a spacing T. For example, if a signal is sampled 20 times per second, the apparent power at 9 Hz will actually include all power present at 11, 29, 31, 49, ... Hz. Aliasing cannot be compensated for after the data has been sampled. It must therefore be prevented, either by sampling at an adequate rate, or by pre-filtering the continuous data to remove high-frequency components which are not of interest in the analysis (e.g., random noise).

### Leakage

In the real world, we always work with finite-length records. If this time-domain truncation can be performed in a manner such that the data record consists



(a) Typical raw-data record

(b) Average of 1000 records

FIGURE 7.3-1 USE OF AVERAGING TO IMPROVE SIGNAL-TO-NOISE RATIO

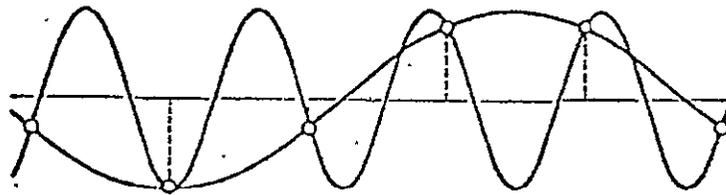


FIGURE 7.3-2 ILLUSTRATION OF "ALIASING": THE HIGH-FREQUENCY SINUSOID LOOKS LIKE ITS LOW FREQUENCY ALIAS WHEN SAMPLED AT TOO LOW A RATE

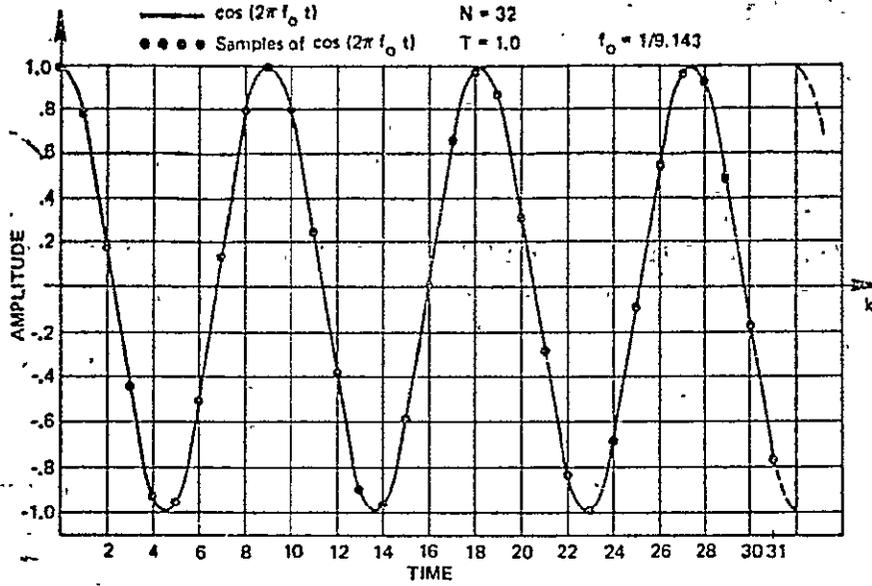
of a precise integral multiple of the base period of the function, the accuracy of the transform is unaffected by truncation. Truncation to any other time-span results in "leakage" of some signal power from its correct location into "side-lobes" of the line spectrum. This degradation of accuracy (line "smearing") is depicted in Figure 7.3-3.

Leakage is reduced and spectral lines are sharpened by a technique called "windowing," whereby the digital data is multiplied by a window function which tapers to zero toward both ends of the data record, before transformation. This makes the time function look more nearly periodic in the window; typical results are shown in Figure 7.3-4. Further discussion of the applications and techniques of windowing may be found in Reference 7-8.

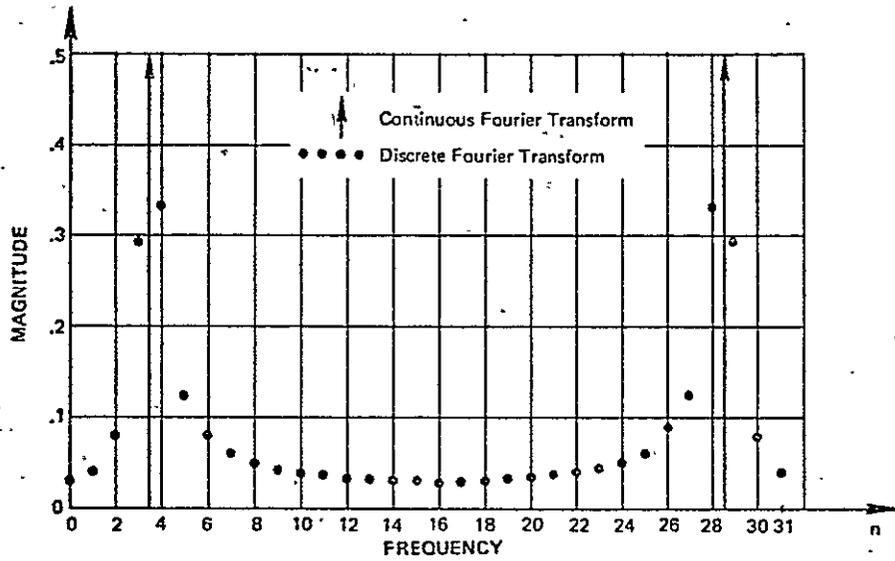
### 7.3.2 Frequency Response From Logged Operational Data

The application of the Fast Fourier Transform as discussed in the previous section produces the frequency response of a system under test when the input or stimulus signal is a known quantity such as a sum of sinusoids or white noise signal. If the amplitude and phase characteristics of a signal that occurs during normal operation of the system could be established, it would appear that the response of the system to this random signal could be computed as before and then consequently the frequency response. The advantages are several. First, there is no requirement for dedicated system test time. Secondly, there is no need to generate special test signals. Finally, there is no wear and tear incurred on the system just for the purpose of testing. For a simulator subsystem such as the motion base, the latter is an important factor.

Unfortunately, although the Fourier transform can be applied to both input and output signals, there is no timewise correlation involved between the two and hence no phase information available unless something further is done. During the survey activity, three different approaches were found which are described briefly below. These approaches are susceptible to the quality of the signals that are normally available from the systems being analyzed. In addition, because the mathematical operations required are rather complex, it is not determinable without some development effort how well conditioned the basic mathematical approaches are. Inherent mathematical instabilities may preclude the successful

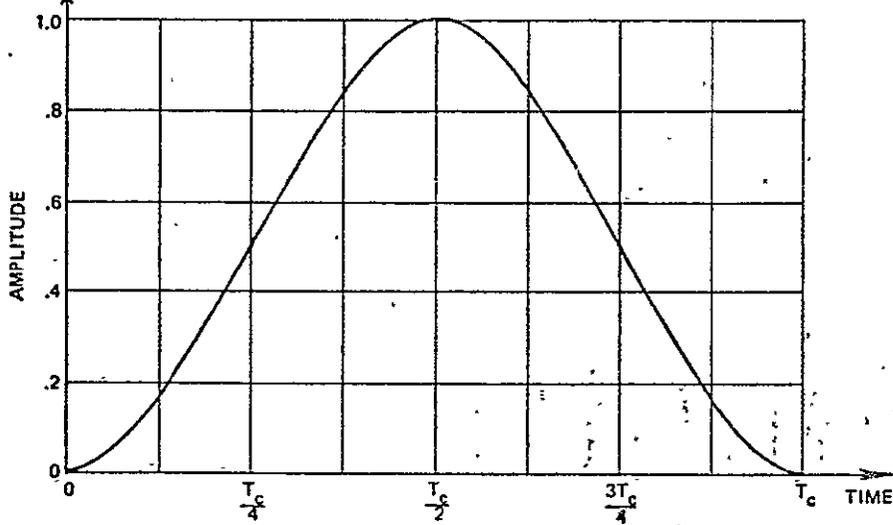


(a) Time-domain Samples

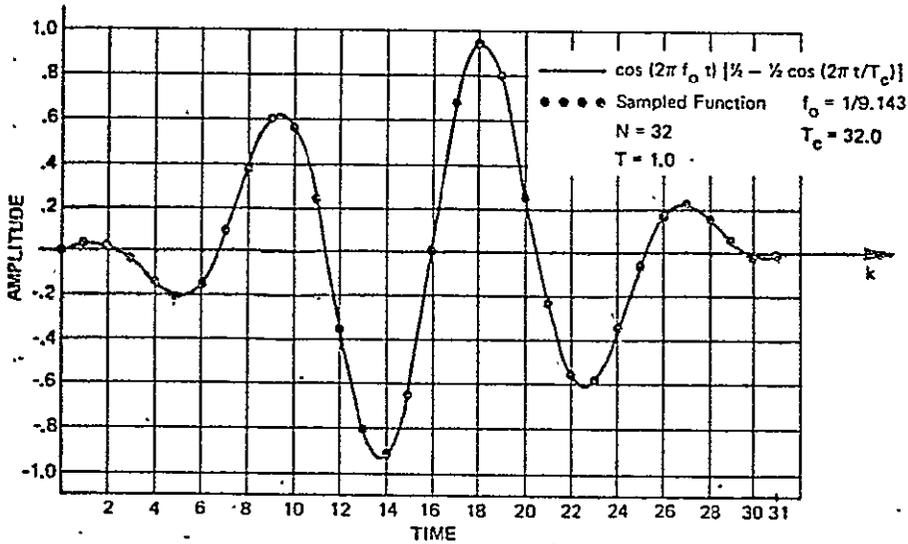


(b) Resulting Fourier transform

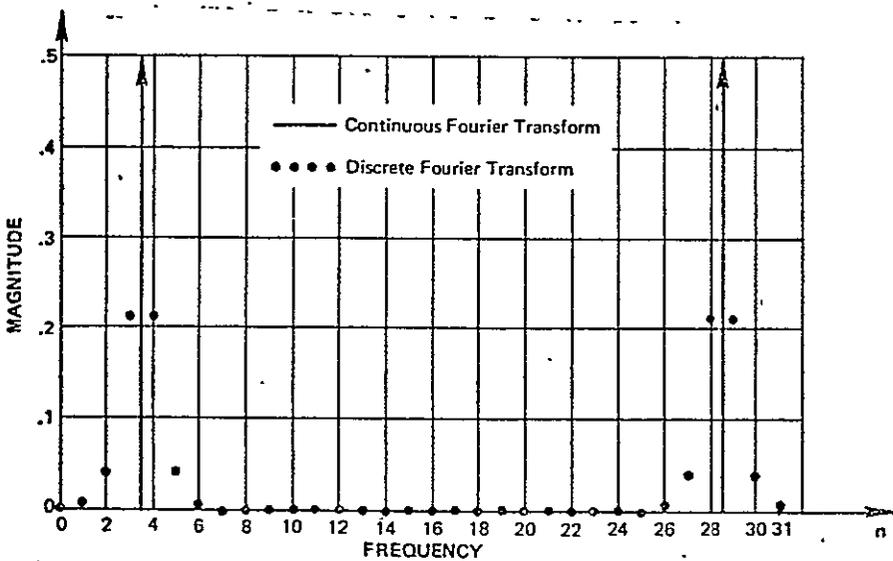
FIGURE 7.3-3 LEAKAGE EXAMPLE: FOURIER TRANSFORM OF A COSINE WAVEFORM;  
 TRUNCATION INTERVAL NOT EQUAL TO A MULTIPLE OF THE PERIOD



(a) Hanning window function



(b) "Windowed" time function



(c) Resulting Fourier transform

FIGURE 7.3-4 USE OF "WINDOWING" TO REDUCE LEAKAGE

application of these techniques. These techniques are nevertheless described below because they represent a technology which has definite potential in the simulator area and should be considered for future development effort.

These techniques are referred to as the running left-side Laplace Transform technique, the noise injection technique and the correlation processing technique.

### 7.3.2.1 Running Left-Side Laplace Technique

The running left-side Laplace transform technique, Reference 5-9, is applicable to monitoring of single-input, single-output linear system, satisfying a differential equation of the form

$$x + p_1 \dot{x} + p_2 \ddot{x} + \dots + p_n \frac{d^n x}{dt^n} = z_0 u + z_1 \dot{u} + \dots + z_m \frac{d^m u}{dt^m}$$

The overall operation of the process, as shown in Figure 5.3-5, is as follows:

- Raw input and output data are transformed to generate a set of algebraic coefficients for input to a simultaneous-equation solver;
- The equations are solved to generate estimates of the constants in the differential equation representing the subsystem operation;
- Finally, the estimated constants are compared to nominal values, drawn from the data base, resulting in a good/bad assessment of the subsystem operation.

The coefficients for the simultaneous equations can be formed by a variety of methods, e.g. repeated differentiation, repeated integration. The method Garzia favors is the use of a "running left-side Laplace transform," (RLLT), a generalization of the well-known Laplace transform, defined as:

$$F(s, t) = \int_{t_0}^t f(v) e^{s(t-v)} dv$$

The properties of this transform are similar to those of the basic Laplace transform, with modifications as required by the finite limits of integration. The desired equations are formed by forming a set of RLLT's for distinct values of  $s$ , chosen to span the frequency ranges appropriate to the input and output. Conveniently, these RLLT's can be formed by hardware low-pass filters (or equivalent software difference equations), making it possible to generate the requisite  $(m+n)$  simultaneous equations simultaneously, in parallel, with relatively little computational load.

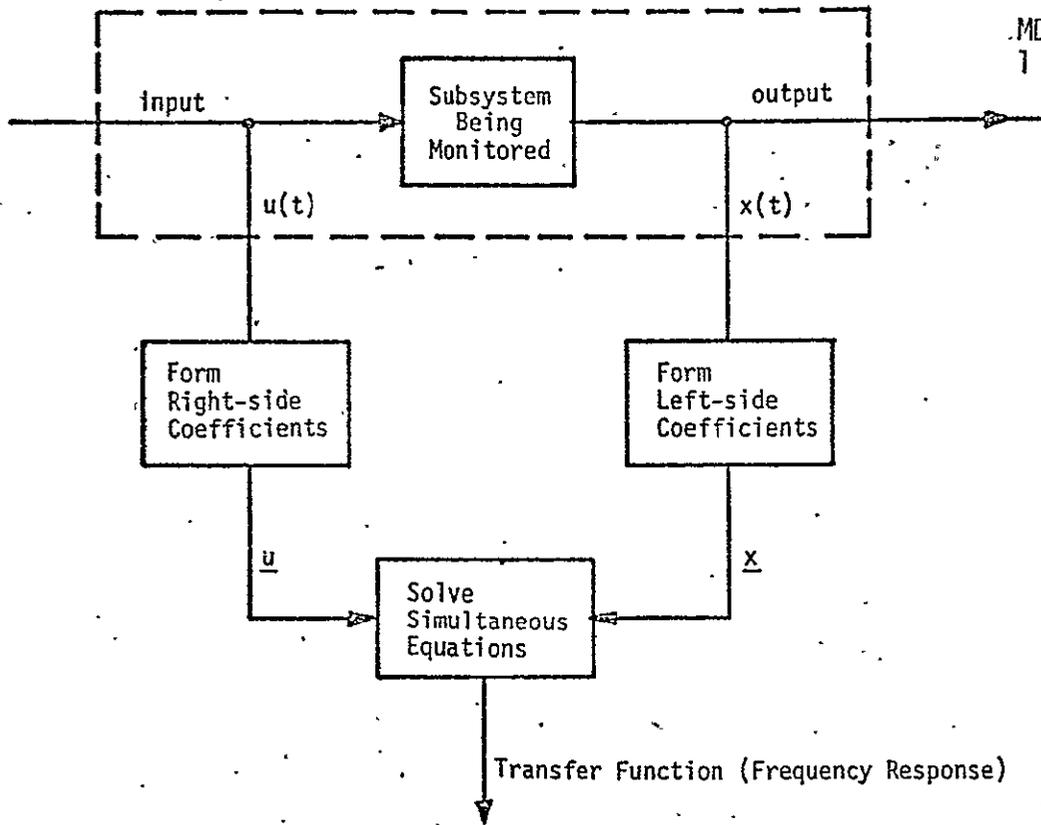
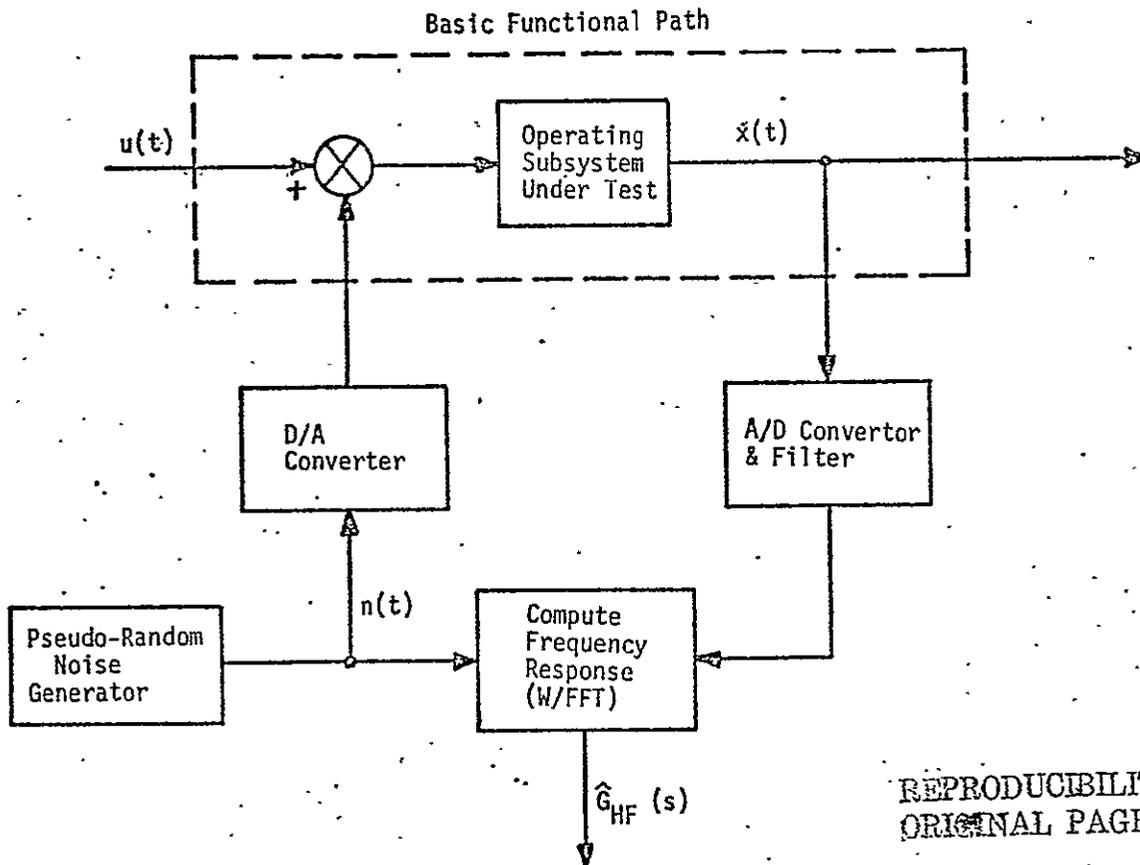


FIGURE 7.3-5 GARZIA'S RUNNING LEFTSIDE LAPLACE TRANSFORM PROCEDURE



REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

FIGURE 7.3-6 FLOW OF NOISE INJECTION ANALYSIS

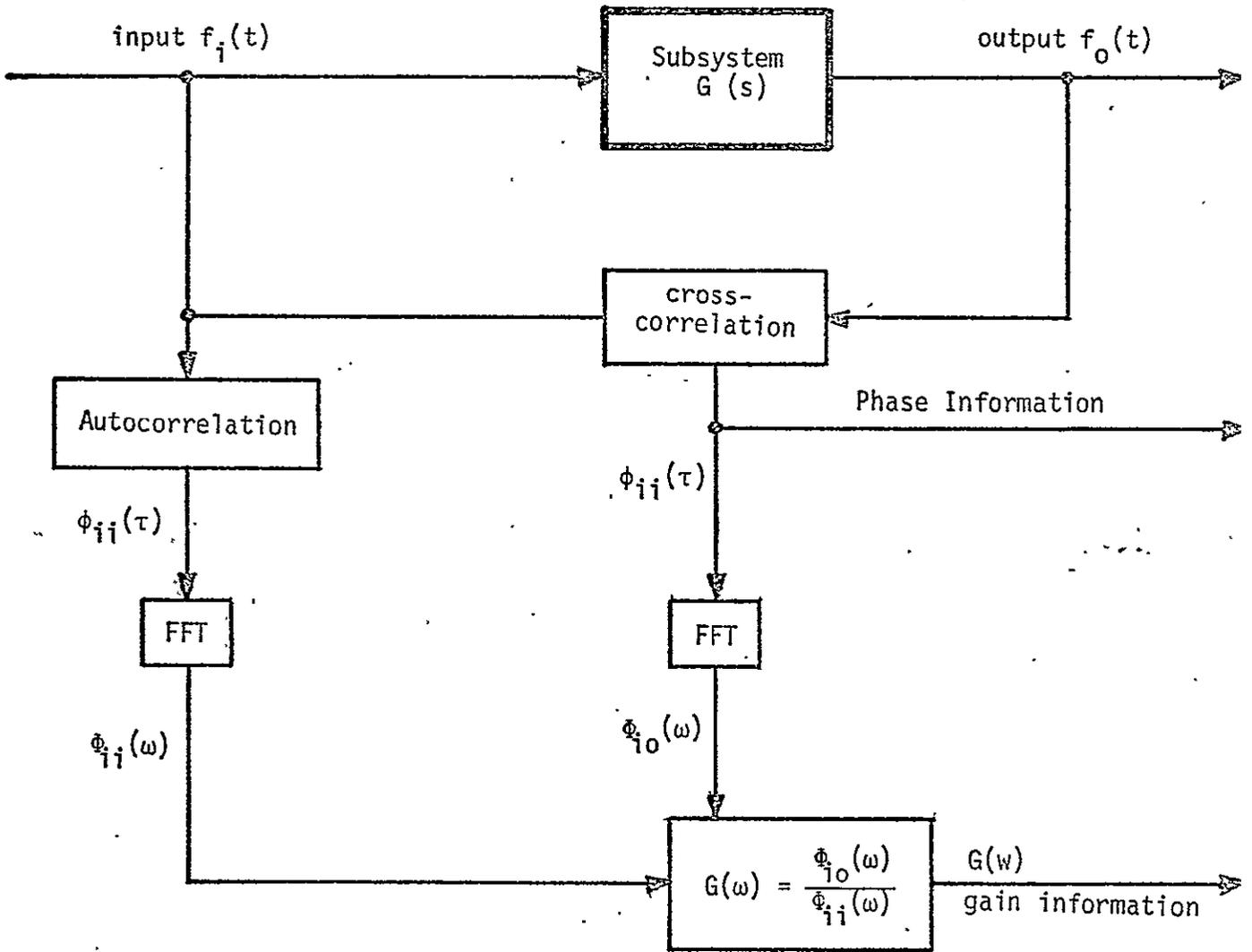


FIGURE 7.3-7 CORRELATION PROCESSING TO OBTAIN FREQUENCY RESPONSE

The difficulty, of course, comes in the process of solving this set. Not only does the computational load increase rapidly with system complexity (proportional to the cube of the number of equations), but there is always a strong probability of ill-conditioning. Ill-conditioning would in turn lead to inaccurate answers even under ideal conditions; the presence of measurement noise or nonlinearities could make the results completely meaningless.

The advantages of this technique are the following:

- It can track system transfer functions continuously in real time.
- No special test signals are required.

#### 7.3.2.2 Noise Injection Techniques

Injection of a signal outside the operational frequency band of a system (e.g., spikes, pseudo-random noise), followed by extraction of the system's response signature, is a method tried some years ago for implementation of adaptive autopilots. Overall flow of this method is shown in Figure 7.3-1 after Garzia Reference 7.3-6. High-frequency pseudo-random noise, generated by digital means, is passed through an A/D converter and added to the basic input  $u(t)$  at a summing junction; the composite signal is input to the subsystem under test. The output  $x(t)$  will then be the subsystem's response to the total applied signal. If the subsystem is linear (superposition is valid), the output of the A/D converter/filter will consist of the subsystem's response to the noise signal alone. Cross-correlation of the noise input and the filtered output (a rather involved calculation) will then yield the high-frequency portion of the subsystem's transfer function. Low-frequency response cannot be detected by this technique.

#### 7.3.2.3 Correlation Processing

As an alternative to providing special test signals, system frequency response can be estimated by processing of actual operational input and output data. The computing load will of course be higher, and the accuracy of the estimated system characteristics will be lower; however, some testing time will be saved.

Figure 7.3-7 shows the computational flow for a determination of the input/output gain and phase components of a subsystem transfer function  $G(s)$  from monitored operational data. Denoting the input as  $f_i(t)$  and the output as  $f_o(t)$ , first compute  $\phi_{ii}(\tau)$ , the autocorrelation of the input, and  $\phi_{io}(\tau)$ , the input/output cross-correlation, using

$$\phi_{ii}(\tau) = \frac{1}{2T} \int_{-T}^T f_i(t) f_i(\tau-t) dt$$

$$\phi_{io}(\tau) = \frac{1}{2T} \int_{-T}^T f_i(t) f_o(\tau-t) dt$$

The FFT of the input autocorrelation is  $\Phi_{ii}(\omega)$ , the input power-spectral-density. The cross-correlation, which provides phase information directly, is also transformed by the FFT into  $\Phi_{io}(\omega)$ , the input/output cross-power-spectral-density. These two spectral densities then provide an estimate of the gain component of the subsystem frequency response, by

$$G(\omega) = \frac{\Phi_{io}(\omega)}{\Phi_{ii}(\omega)}$$

## SECTION 8

### DEFINITION OF HARDWARE AND SOFTWARE TECHNIQUES FOR SIMULATOR CHECKOUT

This section presents the results of Subtask 1.3 which addressed the definition of hardware and software techniques for implementing self test for the various simulator subsystems specifically in terms of their anticipated configuration and scope for future NASA training simulators. The subsystems addressed individually consist of the ancillary digital computers and their interfaces, Section 8.1; the data conversion equipment, Section 8.2; the controls and displays for the crew stations and the instructor operator stations, Section 8.3; the visual simulation subsystems, Section 8.4; the motion base equipment, Section 8.5; and miscellaneous equipment consisting of the aural cue system, the power supplies and the external clocks in Section 8.6.

The ancillary digital computers of primary concern in section 8.1 are the flight computers and the flight computer interface devices that are anticipated for this equipment. The testing proposed for this equipment consists primarily of functional testing to insure proper operation as part of a readiness check. The isolation of faults for this equipment should be addressed by maintenance software provided by the vendors since the effort required for development of such software for one specific application is exorbitant.

Self test of the data conversion equipment is discussed in Section 8.2. The self test of the data conversion equipment has been addressed in the past at the Johnson Space Center and the status of the current capability in this area is reported in Section 6. The primary requirement for self test of this equipment is test software and automatic switching for implementing loop closures for both end to end readiness testing as well as for fault isolation tests.

The self test techniques applicable to the controls and displays in the crew stations and the instructor operator stations are discussed in Section 8.3. The requirements for implementing controls tests are distinctly different from the requirements for displays. Consequently, overviews of the displays testing problem may be found in Section 8.3.1 and of the controls problem in Section 8.3.8.

Self test techniques for the visual simulation subsystems are discussed in Section 8.4. The equipment in the visual simulation subsystem has been divided into three groups each of which have relatively unique or distinctive test requirements. These three groups consist of the image transmission system, the display equipment and the scene generation equipment. The image transmission equipment analysis addresses the test techniques applicable to the TV system electronics and the TV camera electro-optical interface. The motion problems associated with the camera drive system and the probe are deferred to the scene generation equipment tests, Section 8.4.3. Section 8.4.2 addresses self test techniques for the display equipment which presents unique problems because of the marginal optical/observer interface.

The self test techniques applicable to the motion base equipment are presented in Section 8.5. Self test techniques for various several miscellaneous items are presented in Section 8.6.

The integration of the testing for the equipment addressed in Section 8 is the subject of Section 9.

## 8.1 ANCILLARY COMPUTERS AND INTERFACE EQUIPMENT

The variety and complexity of simulator equipment that exchanges digital data with the host computer (HOST) place a heavy demand on input/output processing. This demand is satisfied in the Shuttle Training Simulators, as defined in the Reference Configuration of Section 5, by minicomputers that perform the data formatting and time buffering required to handle simulator data. The most demanding interface in terms of type and rate of data transferred is the HOST to Flight Computer (FC) link. A special purpose Flight Hardware Interface Device (FHID) will be required to handle this input/output link. This FHID has both processing and storage capabilities peculiar to the HOST/FC interface characteristics.

The FHID verification problem is analyzed first and an approach developed in Section 8.1.1. The FC is then analyzed at a higher level than the FHID. Section 8.1.2 covers this analysis not so much with the objective of developing checkout techniques for each LRU in the FC, but rather to identify the various diagnostics that the manufacturer is likely to develop and to establish a checkout approach that makes maximum use of these diagnostics. Section 8.1.3 covers the analysis of minicomputers, their interfaces with the HOST, and the various techniques already documented in 8.1.1 and 8.1.2 that may be applicable to verification of this equipment.

### 8.1.1 Flight Hardware Interface Device

The Reference Simulator Configuration includes three flight computers. The FC's are used to achieve a high degree of fidelity both in the type of processing performed as well as the amount of time used to carry out this processing. In the actual Orbiter, the FC's interface with elements of various Shuttle systems. In the Simulator these elements or their functions are simulated by the HOST and it is this variety and complexity of functions that place a heavy demand in rate and data handling versatility at the interface between the HOST and the FC's. Figure 8.1-1 is a block diagram that illustrates the FHID interfaces.

There are some elements of the Avionics System, such as guidance elements, that generate data at such a rate as to possibly overload the HOST/FC link. If a controllable hardware generator of these data is used external to the HOST, then this potential saturation of the link can be prevented.

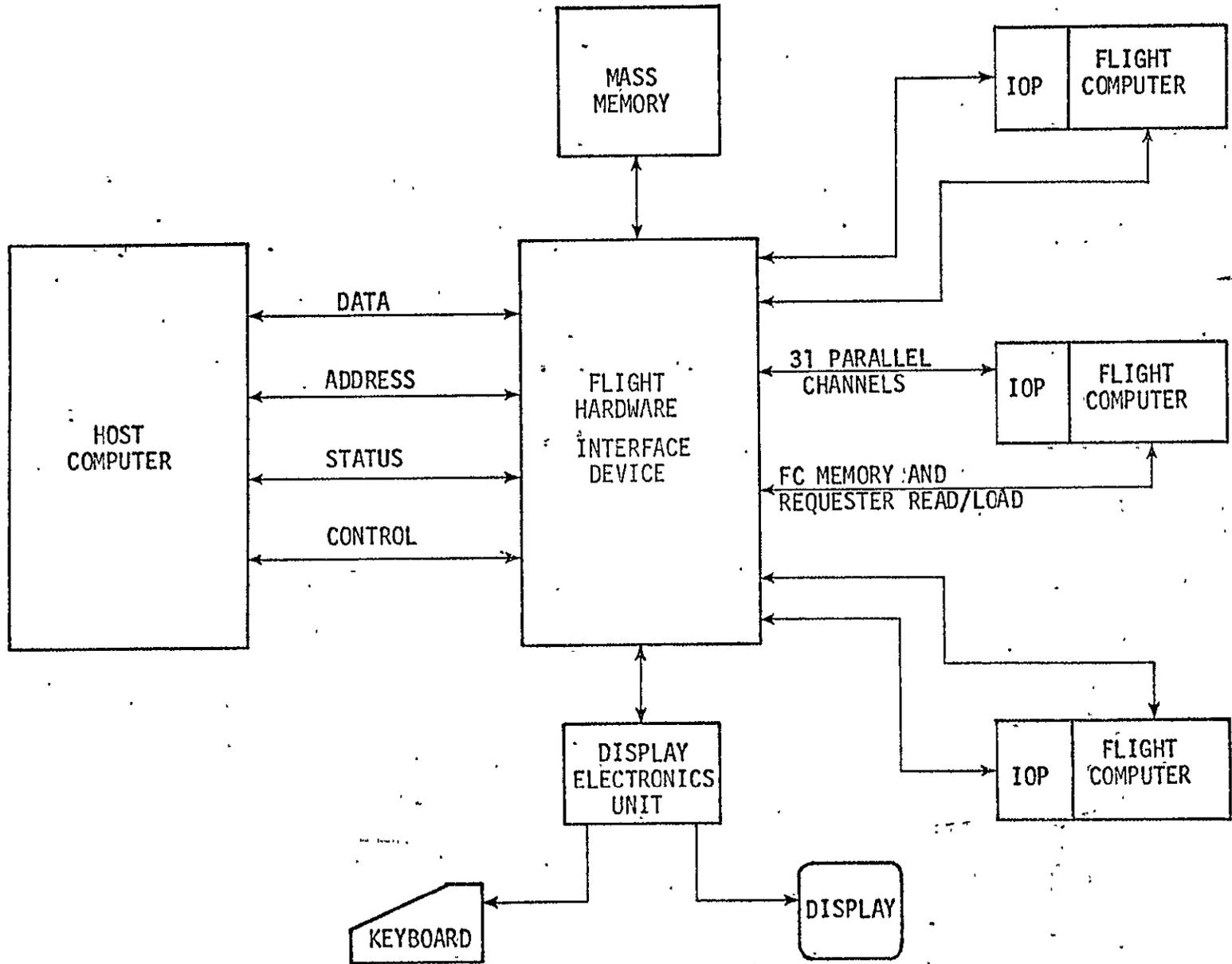


FIGURE 8.-1-1 FIELD EXTERNAL INTERFACES

The FHID provides all the interface services to permit direct communication between the HOST and the FC's. This device includes enough hardware modeling, buffering, and control to allow the flight program to run in real time. Hardware modeling in the FHID is particularly used to generate rate variable data and present time variable data to the FC. This modeling function contributes greatly to decreasing the amount of data transfers involving both the HOST and FC concurrently.

Interface between the FC and the Display Electronics Unit is provided within the FHID. This configuration permits the HOST to monitor and, if necessary, to alter the data flowing between the Flight Computer and the Keyboard and Display Units (KDU) at the crew station.

#### 8.1.1.1 FHID Description

The major functional components of the FHID are the HOST Interface, the various processors, a central buffer storage, the DEU interface, mass memory interface, a simulated Avionics Master Timing Unit, and the FC (IOP) and FC (AGE) interfaces. Figure 8.1-2 is a block diagram of the FHID illustrating the general information flow between each of these functional elements, and Table 8.1-1 lists the line replaceable units as defined for the FHID. Characteristic parameters for each of these LRU's are listed in Table 8.1-2. The following text further describes the function and interrelation of the FHID elements.

#### HOST Interface

This unit provides the necessary functions to permit orderly interchange of data between the HOST and the FHID. These functions are described below. It must be remembered that many actual quantitative details, such as number of bits per transfer, are dependent on the manufacturer and model number of the computer selected as HOST for the given simulator.

Logic Level Conversion - This function is performed by the drivers and receivers used to convert the computer cable signal levels to internal FHID logic levels which, for consistency, are defined as being equal to those in the DCE that is:

True, or Logic 1	+4.0	$\pm 1.5$ vdc
False, or Logic 0	+0.4	$\pm 0.4$ vdc

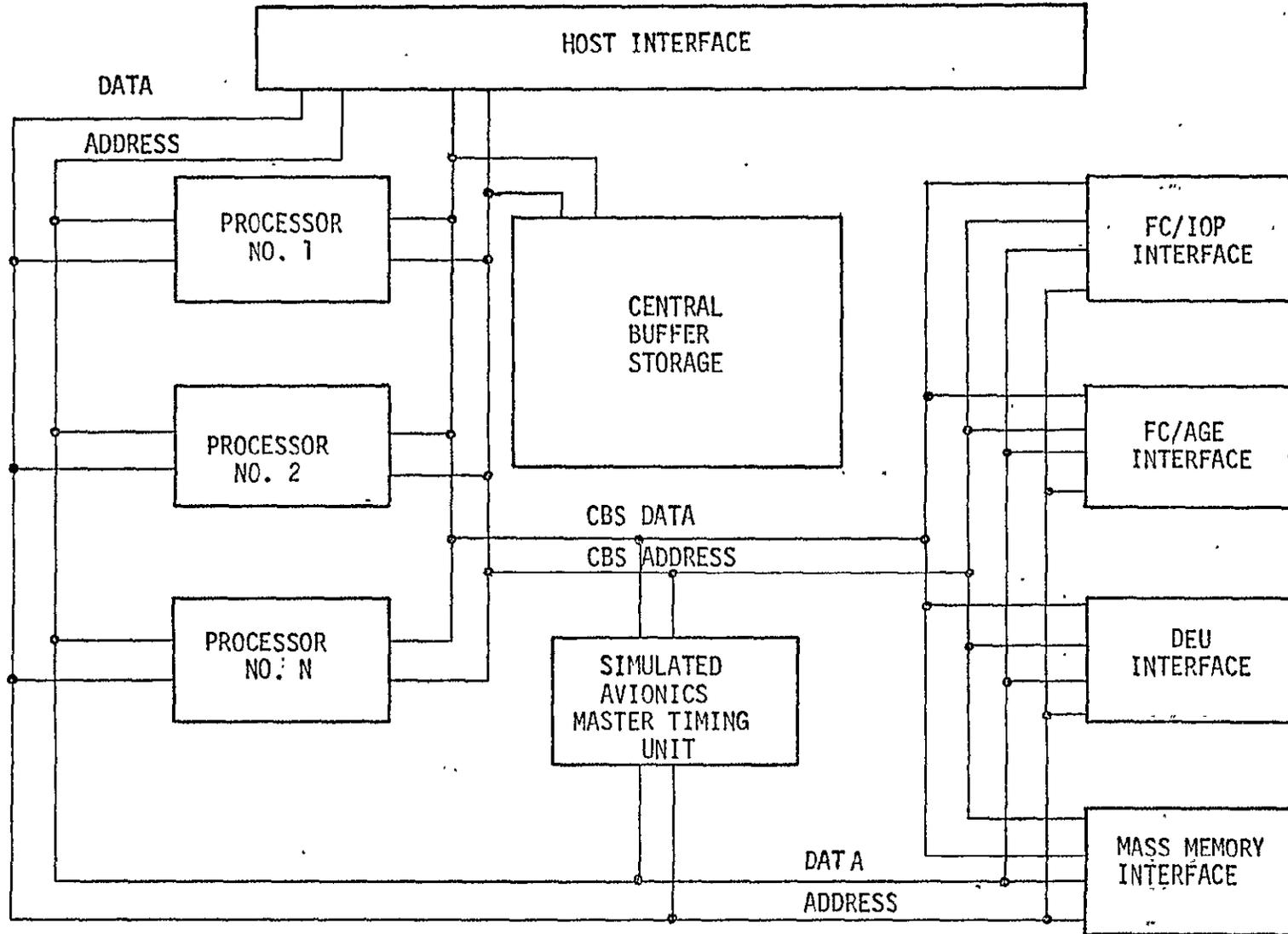


FIGURE 8.1-2 FHID INTERNAL INTERFACES

TABLE 8.1-1. LIST OF FHID LINE REPLACEABLE UNITS

LINE REPLACEABLE UNIT	FUNCTION	INTERFACES WITH
Host Interface (HI)	Data formatting, sequencing, and synchronizing for transfers between HOST and FHID.	HOST, P's, CBS, DEUI, MMI, SAMTU, FC/II, FC/AI
Processors (P's)	Data processing for interdevice communications. Computations for time and rate variable data. Fault simulation for FHID and FC self-test.	HI, other P's, CBS, DEUI, MMI, SAMTU, FC/II, FC/AI
Central Buffer Storage (CBS)	Temporary storage for flight program parameters. Temporary storage for block transfers between HOST, FC's, DEU, and FHID processors.	HI, P's, DEUI, MMI, FC/II, FC/AI
DEU Interface (DEUI)	Data formatting, sequencing, and synchronizing for transfers between DEU and FHID.	HI, P's, CBS, MMI, FC/II, DEU
Mass Memory Interface (MMI)	Data formatting, sequencing, and synchronizing for transfers between MM and FHID.	HI, P's, CBS, DEUI, FC/II, FC/AI, MM
Simulated Avionics Master Timing Unit (SAMTU)	Generate GMT and MET as indicated by HOST commands. Issue time data to devices requesting this information.	HI, P's, FC/II
FC (IOP) Interface (FC/II)	Data formatting, sequencing, and synchronizing for transfer of operational data between FC IOP and FHID.	HI, P's, CBS, DEUI, MMI, SAMTU, FC/IOP
FC (AGE) Interface (FC/AI)	Data formatting, sequencing, and synchronizing for transfer of flight program data between FC and FHID. Provide means for FC program execution control.	HI, P's, CBS, MMI, FC/AGE

8-7

LINE REPLACEABLE UNIT	FUNCTION	INTERFACES WITH
Host Interface (HI)	Data formatting, sequencing, and synchronizing for transfers between HOST and FHID.	HOST, P's, CBS, DEUI, MMI, SAMTU, FC/II, FC/AI
Processors (P's)	Data processing for interdevice communications. Computations for time and rate variable data. Fault simulation for FHID and FC self-test.	HI, other P's, CBS, DEUI, MMI, SAMTU, FC/II, FC/AI
Central Buffer Storage (CBS)	Temporary storage for flight program parameters. Temporary storage for block transfers between HOST, FC's, DEU, and FHID processors.	HI, P's, DEUI, MMI, FC/II, FC/AI
DEU Interface (DEUI)	Data formatting, sequencing, and synchronizing for transfers between DEU and FHID.	HI, P's, CBS, MMI, FC/II, DEU
Mass Memory Interface (MMI)	Data formatting, sequencing, and synchronizing for transfers between MM and FHID.	HI, P's, CBS, DEUI, FC/II, FC/AI, MM
Simulated Avionics Master Timing Unit (SAMTU)	Generate GMT and MET as indicated by HOST commands. Issue time data to devices requesting this information.	HI, P's, FC/II
FC (IOP) Interface (FC/II)	Data formatting, sequencing, and synchronizing for transfer of operational data between FC IOP and FHID.	HI, P's, CBS, DEUI, MMI, SAMTU, FC/IOP
FC (AGE) Interface (FC/AI)	Data formatting, sequencing, and synchronizing for transfer of flight program data between FC and FHID. Provide means for FC program execution control.	HI, P's, CBS, MMI, FC/AGE

TABLE 8.1-1 LIST OF FHID LINE REPLACEABLE UNITS

Data Formatting - The data transferred to and from the HOST are grouped in 8, 16, 32, 64 or any other number of bits depending on manufacturer and model number of the HOST. The internal word length of the FHID has been defined for purposes of this study as 16 bits. The formatting function comprises the packing and unpacking of FHID words into groups compatible with the HOST interface. For example, if this interface uses 64 bit transfers, each of these transfers is divided into, or assembled from, 4 different FHID words.

Timing - This function covers synchronization and time buffering of data transfers in order to overcome variations in time dependent differences between the HOST and FHID hardware. It also includes time tagging of certain data.

Sequencing Control - Data transfers involve the generation of and response to control signals used to achieve device selection and identification, and to determine when and how much data may be transferred.

Data Routing and Acquisition - The HOST Interface Section routes data from the HOST to the proper processor in the FHID to Central Buffer Storage, to the FC or EDU interface, or to any of the other sections of the FHID.

### Processors

There are several (the precise number to be determined during actual design) processors in the FHID to perform computations for rate and time variable data generation, as well as for processing, monitoring, and controlling other data routed through the FHID. These processors have their own memory, index and process control registers, data bus and I/O channel.

Each processor may accept from or issue data to other processors, the HOST, the FC, or other devices within the FHID.

### Central Buffer Storage

This is a large random access memory that stores simulation parameter data either permanently or during block transfers between digital elements of the simulator connected to the FHID. The capacity of this buffer will be determined during actual design but is currently estimated at no more than 40,960 eighteen bit words.

### Display Electronics Unit Interface

This section of the FHID provides the interface circuitry for data transfers between the FHID and the DEU. Data from the crew station digital command keyboard goes to the DEU and is routed to the HOST, and in some cases to the FC, through the FHID. Also, data intended for the CRT displays on the crew station, whether generated by the FC or HOST are routed through this section of the FHID.

The DEU interface performs the data formatting and routing functions for transfers involving the DEU, HOST, FC FHID processors, FHID CBS, and other devices within the FHID as these data pertain to the crew station keyboard or display unit.

### Mass Memory Interface

The inclusion of this interface assumes that a simulated or actual mass memory -- magnetic tape, disk, other -- would be a part of the simulator. The mass memory contains flight program data such as used for mission phase related overlays, and the onboard display formats. The mass memory function in the simulator could also be performed by the HOST computer complex, in which case this interface would not be necessary.

The mass memory interface section of the FHID handles data transfers between the mass memory and the FC, the HOST, or the DEU. Within the FHID, this section communicates with processors for data formatting and possible alteration under HOST control, with the FC/IOP interface, with the FC/AGE interface to simulate ground loads, and with the DEU for display information transfer.

### Simulated Avionics Master Timing Unit

Because of the functionally central location of the FHID in the digital data processing network of the simulator, the SAMTU is included in this device. The flight MTU provides, on request only, status information, Greenwich Mean Time, and Mission Elapsed Time. The FHID will perform this function in the simulator.

The SAMTU receives initializing time loads from the HOST, and responds to time data requests from the HOST, the FC, or any of the FHID processors involved with processing rate or time variable parameters for the simulation.

TABLE 8.1-2 CHARACTERISTIC PARAMETERS OF FHID LRU'S

LRU	INPUTS	OUTPUTS	PERFORMANCE
HOST Interface	-Data from HOST (1) -Address (1) -Interface Controls (1) -FHID Data (2) -FHID Device Status (3) -FHID Device Ready (3)	-Data to HOST (1) -Address (1) -Interface Controls (1) -Data to FHID devices (2) -FHID Device Select (3)	-Thruput 90,000 words/sec -Peak Thruput 9,000 words in 40 msec
Processors	-Data from FHID Devices (2) -Interface Controls (3) -Interrupts (3) 4 Priority Levels 16 Sublevels -Device Address (4) -Device Status (4)	-Data to FHID Devices (2) -Interface Controls (3) -Processor Address (4) -Processor Status (4)	-No. of Instructions 40 -Memory words (2) 4096 -Memory Cycle Time 500 nsec -Accumulators 4 -Index Registers 8 -Interval Timers (16-bit) 2
Central Buffer Storage	-Data from FHID Devices (2) -Interface Controls (3) -Storage Address (2)	-Stored Data (2) -Interface Controls (3) -Status Data (3)	-Capacity in Words 40,960 -Cycle Time 500 nsec

- (1) Characteristics depend on HOST Computer Selection
- (2) 16-bit words + 2 bits parity . Logic Levels
- (3) Logic Levels (See 8.1.1.1)
- (4) Number of lines and function depend on detailed FHID design

TABLE 8.1-2 CHARACTERISTIC PARAMETERS (CONTINUED)

LRU	INPUTS	OUTPUTS	PERFORMANCE
DEU Interface	-Data from DEU (5) -Interface Controls (5) -Data from FHID Devices (2)	-Data to DEU (5) -Interface Controls (5) -Data to FHID Devices (2)	
Mass Memory Interface	-Data from MM (6) -Interface Controls (6) -Data from FHID Devices (2)	-Data to MM (6) -Interface Controls (6) -Data to FHID Devices (2)	
Simulated AMTU	-Data from HI (2) -Interface Controls (3) -Time Data Request (3)	-Time Data (2) -Status Data (3) -Interface Controls (3)	-GMT Resolution .125 msec -Met Resolution .125 msec
FC/IOP Interface	-Data from FC (7) -Interface Controls (3) -Status Data (3) -Sync OK (3)	-Data to FC (7) -Interface Controls (3) -Status Data (3) -Sync OK (3)	-Number of Parallel Channels 31 -Data Rate 1 M bit/sec
FC/AGE	-Data from FC (8)  -Status Data (3) -Data from FHID Devices (2)	-Data to FC (8) -Address to FC 18-bits (3) -Interface Controls (3) -Data to FHID Devices (2) -FC Execution Controls (3)	

- (5) Logic Levels. Format compatible with DEU design (6) Dependent on detailed MM design  
 (7) 28-bit serial word, includes 3-bit sync, 1 parity bit, 24-bit data  
 (8) 36-bit parallel word, includes two 16-bit half-words with 2 bits each for parity and memory protect

### Flight Computer (Input Output Processor) Interface

All operational Avionics data to and from the FC during flight activity simulation is routed through the IOP. The IOP is directly connected to the FHID FC/IOP section where FC output data is routed to the HOST, the DEU interface, and various FHID processors. Control and request commands from FC are routed to the Mass Memory, the SAMTU; the HOST and FHID processors where other onboard systems are being simulated.

The FC receives through the FC/IOP section of the FHID flight and control data from the HOST, the DEU (crew keyboard inputs), Mass Memory, SAMTU, and various FHID processors.

Data transfers take place with format, sequence, and timing characteristics that faithfully reproduce the flight subsystems and permit the FC programs to execute in real time as they would during an actual Shuttle mission.

### Flight Computer (AGE) Interface

During checkout and prelaunch activities, the FC is connected to the AGE by a special connector that permits access to internal functions of the computer, and provides a means for ground controlled program loads. The FHID FC/AGE interface provides the means for data transfers between the HOST or the Mass Memory and the FC memory. In addition to these data links, control of program execution can be exercised at this interface. Control functions include; Reset Start, Stop, Single Step Advance, Compare Stop, Register Read, and Register Load.

#### 8.1.1.2 FHID Failure Mode Analyses

The FHID components are digital in nature. Failures of these components manifest themselves in one of three different modes: (1) a constant logic 1 output, unable to switch to a logic 0; (2) a constant logic 0 output, unable to switch to a logic 1; and (3) an output which may or may not represent the inputs and therefore is intermittently erroneous.

A failure may occur in data storage or gating circuitry, in which case erroneous data would affect the accuracy of flight data calculations, and therefore the fidelity and overall performance of the simulator. The Subsystem affected is

determined by the type of data involved and the area of the FHID hardware where the fault occurs. On the other hand, a failure may develop in address generation or recognition circuitry. This type of error causes data or commands to be routed to improper devices, or not to be transferred at all. The first case may not be immediately recognized, but instead may be processed having the same effect on the system as that caused by presenting wrong data to the right device. In the latter case, the I/O control circuitry may either lock up or, if so designed, time out after a predetermined period and reset the interface control circuitry after generating a time out error indication to the processor involved in the I/O operation.

Table 8.1-3 lists the various failure modes for each of the LRU's and the possible sources of these failures. Automatic self-test of the FHID should test overall performance capability of the device, and then test for the presence of any one of the problems listed in this table. Not included are overall failures caused by problems in the power circuits, or the logic synchronizing clock oscillator, divider, and gating circuitry. These type of failures are catastrophic enough to prevent self-testing, and are usually obvious enough not to need automatic checkout.

#### 8.1.1.3 FHID Test Techniques

The FHID is a piece of equipment that would be procured independent of other elements of the simulator not only because of its relatively large physical and cost magnitude, but also because it would necessarily be a special purpose device designed to match the interface characteristics of the various devices it interconnects. This fact of separate procurement implies that provisions for self-test of the FHID would be a part of FHID design and, to a large extent, independent of the self-test facilities of other elements of the simulator.

In many respects, and specifically in testing philosophy, the FHID can be treated as a computer. The FHID is a complex device whose function depends primarily on solid state, electronic, digital components. It has data processing and computations capability and is able to store data as input, as well as the computed or reformatted results to be output to any of various other elements

TABLE 8.1-3 FAILURE MODES OF FHID LRU'S

LRU	FAILURE	POSSIBLE SOURCE OF FAILURE
<p>HOST INTERFACE</p>	<p>1. Unable to communicate with HOST</p> <p>2. Unable to communicate with FHID device</p> <p>3. Errors in data transferred</p>	<p>1. a. Line drivers or receivers b. Interface control logic c. Address decode or generation circuits d. Status generation or recognition circuitry</p> <p>2. a. FHID device interface control logic b. FHID device address circuitry c. Interface logic within FHID device d. Status generation or recognition circuitry</p> <p>3. a. Line drivers or receivers b. Interface buffer registers c. Data gating circuitry d. Faulty cableing, internal or external to FHID</p>
<p>PROCESSORS</p>	<p>1. Unable to accept or issue data</p> <p>2. Errors in data outputs</p>	<p>1. a. Processor I/O channel circuitry b. Processor stopped c. Program error</p> <p>2. a. Processor I/O channel circuitry b. Arithmetic/Logic Unit c. Accumulator, Index, or other registers</p>

TABLE 8.1-3 FAILURE MODES OF FHID LRU's (Continued)

LRU	FAILURE	POSSIBLE SOURCE OF FAILURE
CENTRAL BUFFER STORAGE	<ol style="list-style-type: none"> <li>1. Unable to communicate with FHID device</li> <li>2. Errors in data read out</li> </ol>	<ol style="list-style-type: none"> <li>1. <ol style="list-style-type: none"> <li>a. FHID device interface control logic</li> <li>b. FHID device address circuitry</li> <li>c. Interface logic within device</li> <li>d. Status generation or recognition circuitry</li> </ol> </li> <li>2. <ol style="list-style-type: none"> <li>a. Memory Data interface register</li> <li>b. Data gating circuitry</li> <li>c. Memory elements in CBS</li> <li>d. Memory address selection logic</li> </ol> </li> </ol>
DISPLAY ELECTRONICS UNIT INTERFACE	<ol style="list-style-type: none"> <li>1. Unable to communicate with DEU</li> <li>2. Unable to communicate with FHID device</li> <li>3. Errors in data transferred</li> </ol>	<ol style="list-style-type: none"> <li>1. <ol style="list-style-type: none"> <li>a. Line drivers or receivers</li> <li>b. Interface control logic</li> <li>c. Address decode or generation circuitry</li> <li>d. Status generation or recognition circuitry</li> </ol> </li> <li>2. <ol style="list-style-type: none"> <li>a. FHID device interface control logic</li> <li>b. FHID device address circuitry</li> <li>c. Interface logic within device</li> <li>d. Status generation or recognition circuitry</li> </ol> </li> <li>3. <ol style="list-style-type: none"> <li>a. Line drivers or receivers</li> <li>b. Interface buffer registers</li> <li>c. Data gating circuitry</li> <li>d. Faulty cableing</li> </ol> </li> </ol>

TABLE 8.1-3 FAILURE MODES OF FHID LRU'S (Continued)

LRU	FAILURE	POSSIBLE SOURCE OF FAILURE
<p>MASS MEMORY INTERFACE</p>	<ol style="list-style-type: none"> <li>1. Unable to communicate with Mass Memory</li> <li>2. Unable to communicate with FHID device</li> <li>3. Errors in data transferred</li> </ol>	<ol style="list-style-type: none"> <li>1.             <ol style="list-style-type: none"> <li>a. Line drivers or receivers</li> <li>b. Interface control logic</li> <li>c. Address decode or generation circuitry</li> <li>d. Status generation or recognition circuitry</li> </ol> </li> <li>2.             <ol style="list-style-type: none"> <li>a. FHID device interface control logic</li> <li>b. FHID device address circuitry</li> <li>c. Interface logic within device</li> <li>d. Status generation or recognition circuitry</li> </ol> </li> <li>3.             <ol style="list-style-type: none"> <li>a. Line drivers or receivers</li> <li>b. Interface buffer registers</li> <li>c. Data gating circuitry</li> <li>d. Faulty cabling</li> </ol> </li> </ol>
<p>SIMULATED AVIONICS MASTER TIMING UNIT</p>	<ol style="list-style-type: none"> <li>1. Unable to accept time load from HOST or respond to time requests</li> <li>2. Errors in time data output</li> </ol>	<ol style="list-style-type: none"> <li>1.             <ol style="list-style-type: none"> <li>a. Interface control logic</li> <li>b. Address generation for decode circuitry</li> <li>c. Time data gating logic</li> </ol> </li> <li>2.             <ol style="list-style-type: none"> <li>a. Time computation processor</li> <li>b. Time counter advance circuits</li> <li>c. Data gating circuitry</li> <li>d. Address generation or decode circuitry</li> </ol> </li> </ol>

TABLE 8.1-3 FAILURE MODES OF FHID LRU'S (Continued)

LRU	FAILURE	POSSIBLE SOURCE OF FAILURE
FLIGHT COMPUTER IOP INTERFACE	<ol style="list-style-type: none"> <li>1. Unable to communicate with FC</li> <li>2. Errors in data transferred</li> <li>3. Unable to communicate with FHID device</li> </ol>	<ol style="list-style-type: none"> <li>1.                         <ol style="list-style-type: none"> <li>a. Line drivers or receivers</li> <li>b. Interface control logic</li> <li>c. FC address generation circuitry</li> <li>d. Address decode circuitry in FHID</li> <li>e. Status generation or recognition circuitry</li> <li>f. Faulty cableing</li> </ol> </li> <li>2.                         <ol style="list-style-type: none"> <li>a. Line drivers or receivers</li> <li>b. Interface buffer registers</li> <li>c. Data gating circuitry</li> <li>d. Faulty cableing</li> </ol> </li> <li>3.                         <ol style="list-style-type: none"> <li>a. FHID device interface control logic</li> <li>b. FHID device address circuitry</li> <li>c. Interface logic within device</li> <li>d. Status generation or recognition circuitry</li> </ol> </li> </ol>
FLIGHT COMPUTER AGE INTERFACE	<ol style="list-style-type: none"> <li>1. Unable to transfer data or exercise control of FC</li> <li>2. Errors in data transferred</li> </ol>	<ol style="list-style-type: none"> <li>1.                         <ol style="list-style-type: none"> <li>a. Line drivers or receivers</li> <li>b. Data gating or synchronizing circuitry</li> <li>c. Faulty cableing</li> <li>d. Interface control circuitry</li> <li>e. Address generation or decode circuitry</li> </ol> </li> <li>2.                         <ol style="list-style-type: none"> <li>a. Line drivers or receivers</li> <li>b. Interface buffer registers</li> <li>c. Data gating circuitry</li> <li>d. Faulty cableing</li> </ol> </li> </ol>

MCDONNELL DOUGLAS AERONAUTICS COMPANY - EAST  
 8-18

1 November 1974

TABLE 8.1-3 FAILURE MODES OF FHID LRU'S (Continued)

LRU	FAILURE	POSSIBLE SOURCE OF FAILURE
FLIGHT COMPUTER AGE INTERFACE (cont'd.)	3. Unable to communicate with FHID device	3. a. FHID device interface control logic b. FHID device address circuitry c. Interface logic within device d. Status generation or recognition circuitry

of the system. And finally, like a computer, the FHID is delivered and maintained with special purpose programmed diagnostic routines. The use of these "existing" routines, and dependence on "existing" built in test (BIT) hardware, is the basis for developing a self test approach to the hardware verification of the FHID. This section concentrates on identifying the magnitude and functional capability of both the test hardware and the test software that should be made a part of the FHID procurement, design, and delivery.

#### BIT Hardware

No special BIT hardware is required to be added to the FHID mainly because there would normally be enough lines and circuits in the basic design to permit access to components. This access is usually provided through the manual test panel included in computers or large pieces of digital equipment such as the FHID.

For purposes of this study emphasis is placed on automatic self-test performed on a daily basis in an integrated system manner. Manual operations at the test panel would require a high level of training in test personnel and extend the period of testing from seconds to minutes and even hours. However, if the connections available to the test panel are made accessible to the processor performing the automatic self-test, it will be possible to quite thoroughly check out the FHID in minimal time, even to the point of accomplishing a detailed level of fault isolation.

The only hardware, then, that can be considered BIT hardware is the relatively small amount of gating and cabling required to give the HOST access to test panel functions. The study is based on the assumption that no other BIT hardware is included in the FHID design.

#### Self-Test Software

The Self-Test operations of major concern to the study are those performed on a periodic basis after the simulator has become operational in order to verify the functional readiness of the system. It is, therefore, safe to assume that the HOST computer is connected to the FHID and ready to conduct checkout of the interface device. The overall self-test philosophy for the FHID is based on this assumption, thereby relegating primary control of FHID self-test to the HOST

computer. The software required for FHID self-test can be divided in two distinct groups:

1. Self-Test Software resident in HOST
2. Self-Test Software resident in FHID processors

The first group includes programs to supervise and control overall execution of the self-test. Also included are the LRU peculiar test sequences and pattern generators to verify FHID internal and external interfaces, as well as FHID device functional integrity. Figure 8.1-3 illustrates the overall structure and inter-component relationship of the HOST executed test software.

The second group covers the processor executed diagnostic software which, under HOST command and control, is used to verify the computational, storage, and interfacing performance of the processors. There are two sets of programs in this group related to two different hierarchical levels of test. One set includes the computational and interactive routines used to verify the link between the HOST and the processors, as well as the general functional readiness of the processors during FHID overall test. The other set includes the processor peculiar diagnostics used to verify every register, storage location, interrupt function, and instruction execution performance of the processor. Figure 8.1-4 shows the structure and components of the processor resident test software.

The primary functions of each of the self test software elements are described below. The actual sequence of test execution, and flow diagram of the test software follows the software element description.

FHID Test Supervisor - Supervisory functions for FHID self test are contained in the software element which is executed by the HOST computer. These functions are:

- Interrupt handling
- I/O Processing
- System parameter data base access
- Data time tagging and verification
- Test software loading and overlay control

Overall FHID Test - This program controls the sequence of test execution from the the host computer verifies results, and performs fault isolation to the LRU level.

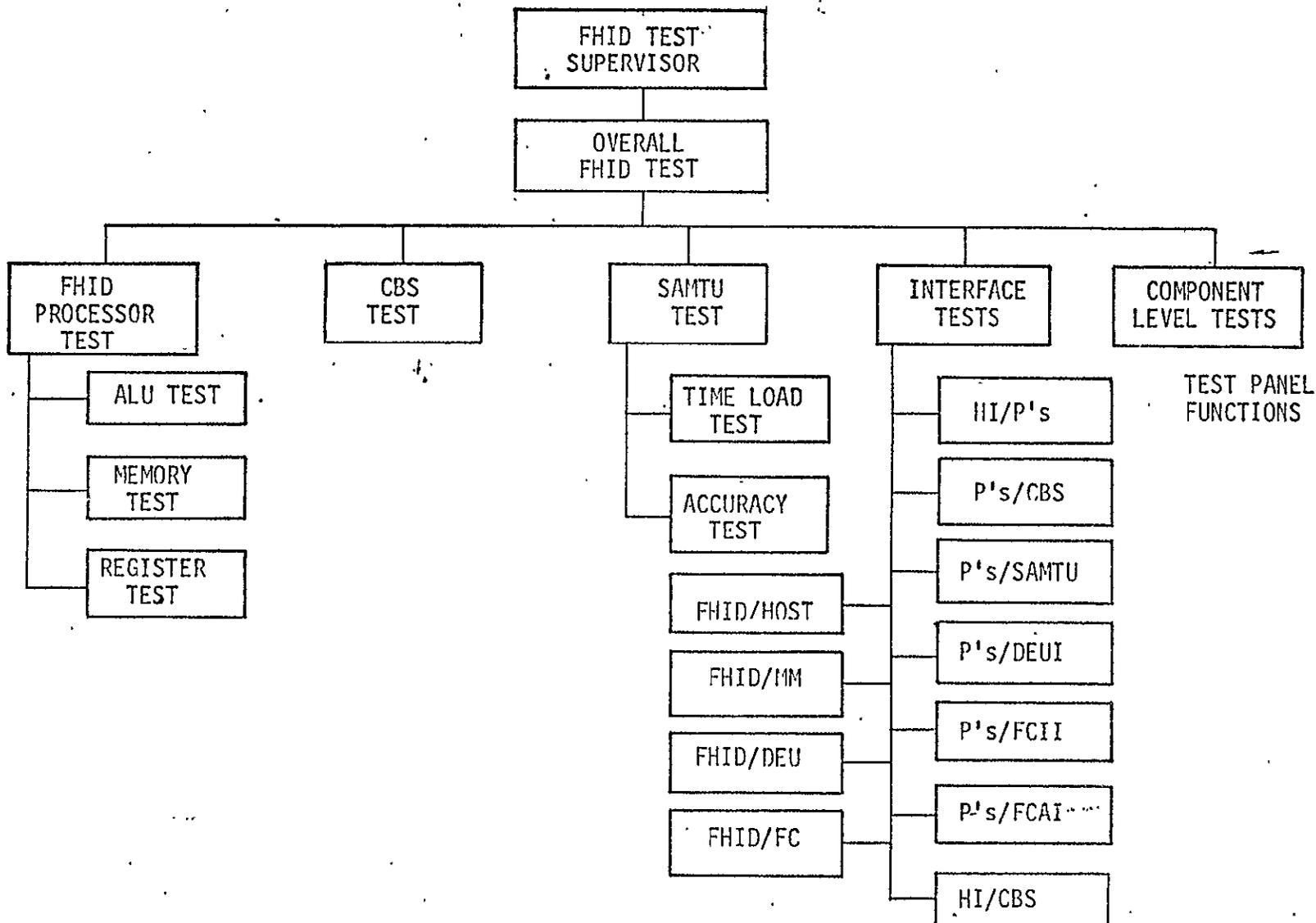


FIGURE 8.1-3 FHID SELF TEST SOFTWARE - HOST EXECUTED

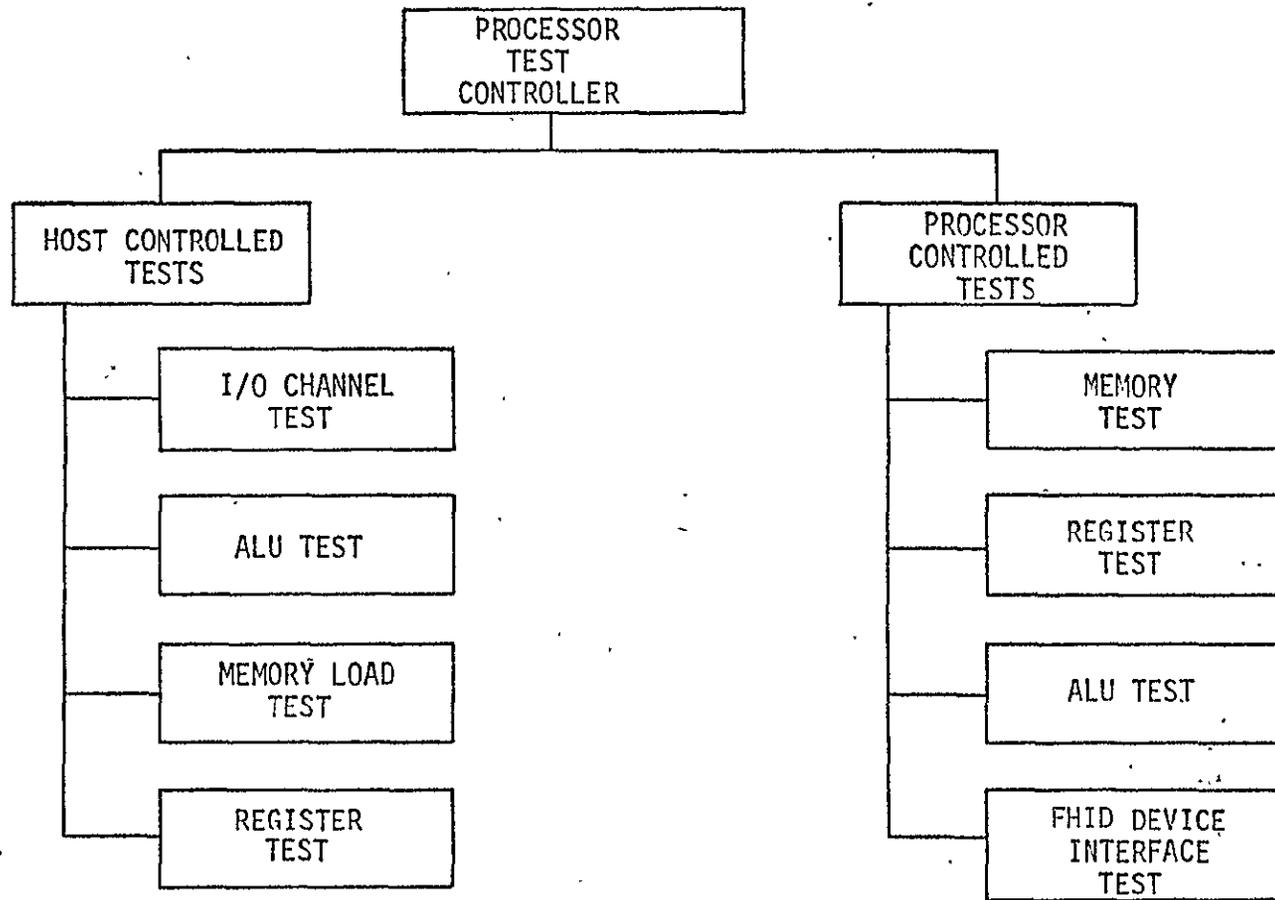


FIGURE 8.1-4 FHID SELF TEST SOFTWARE-PROCESSOR EXECUTED

The tests are performed in the following sequence:

- Interface Test
- FHID Processor Test
- CBS Test
- SAMTU Test

After these tests are performed, the operational readings of the FHID can be ascertained with a high level of confidence--greater than 98%. If a higher level of confidence is desired, or fault isolation beyond the defined LRU'S is to be performed, then the Test Panel functions can be exercised by the HOST with the Component Level Test.

Interface Tests - These tests verify the ability to establish and carry out digital communications between digital devices in a prompt and accurate matter. Interfaces between FHID and external units such as the HOST and FC are checked, as well as interfaces between internal FHID devices. Generally, these tests consist of the following steps:

1. Issue an I/O command that requires data to be transferred across the interface in question.
2. Transfer a data word consisting of an all one's pattern.
3. Verify accuracy of transfer.
4. Transfer data in an all zero's pattern.
5. Verify accuracy of transfer.
6. Check for parity error indication

The length of the data word and method for verifying accuracy of transfer are dependent on which interface is being tested. The interfaces to be checked for the FHID self test are:

- FHID/HOST
- FHID/MM
- FHID/DEU
- FHID/FC-(IOP)
- FHID/FC-(AGE)
- HI/P's
- P's/CBS
- P's/SAMTU

P's/DEUI  
P's/FCII  
P's/FCAI

Processor Tests - Each of the FHID processors can be thoroughly tested using processor diagnostic routines loaded from the HOST. The Processor Test software executed in the HOST consists of the subroutines required to perform this loading, and then to initiate, monitor, and control execution of processor diagnostics. This software also include test result evaluation and error reporting routines used by the HOST to establish processor integrity and determine test progress and success. In addition to Processor Test control software, executed by the HOST, there is a block of FHID of processor machine language program data. These data contain the self test diagnostics executed by the processor itself and consist of three main tests. The ALU Test can be performed with little intervention by the HOST. However, the memory and register test require the HOST to monitor test points brought out from the processor to verify correct loading and clearing of each memory device.

1. ALU (Arithmetic Logic Unit) Test - Verifies execution of all instructions in the processor repertoire. It performs the following:
  - Exercise all instructions using all available addressing formats (direct, indirect, relative, etc.)
  - Force occurrence of individual and simultaneous interrupts by commanding interrupt generation registers in the FHID verify handling according to pre-established priorities
  - Output results to host computer.
2. Memory Test - Verifies integrity of processor dedicated storage, by means of the following steps:
  - Load all storage locations with all ones and read back using direct memory access
  - Verify the load
  - Load all storage locations with all zeros and verify that the locations have been cleared
  - Simulate interrupts and verify performance of operations peculiar to specific storage locations

3. Register Test - Checks register operation by the following steps:

- Load all ones in each register and verify
- Load all zeros in each register and verify
- Load each register with a unique pattern and verify

Central Buffer Storage Test - Every storage location of the CBS is checked by this test. The steps involved are:

- Load all memory locations of the CBS with all ones. The load may be performed from the HOST or by using a HOST commanded digital pattern generator in the FHID
- Read all CBS locations from the HOST and verify an all ones load
- Load all memory locations of the CBS with all zeros. This load may also be performed by test hardware in the FHID, under HOST command
- Read all CBS locations from the HOST and verify that all locations have been cleared

SAMTU Test - The Simulated Avionics Timing unit in the FHID can be tested simply by verifying that an initial time load can be performed, and then checking the accuracy of the time advance calculations. The two tests, Time Load and Count Accuracy, can be accomplished with the following steps:

- Issue a Time Load command to the GMT timer
- Load a maximum GMT time (all ones)
- Read GMT and verify all ones
- Issue a start GMT Advance command
- Wait x.x seconds (to be determined by resolution and accuracy requirements)
- Read GMT and verify proper time count (    seconds plus I/o processing time)
- Repeat the above steps for MET timer

Component Level Tests - The availability of FHID test panel functions to the HOST permit fairly detailed automatic trouble shooting to be conducted. Through these test connections, the HOST can load and read any one of the registers, accumulators, buffers, or memory locations in the FHID circuitry. It is possible, also, to verify step by step signal or data processing within the FHID by giving the HOST control of the clock pulse generation circuitry in the FHID. This level of checkout and fault isolation, however, is beyond the defined LRU level and actual step by step implementation of the tests is dependent on the detailed characteristics of the

design, and the degree to which automatic troubleshooting is desired. Component level testing is mentioned here only to point out the possibility of including this capability in the initial design of the FHID and its associated Self Test system.

FHID Processor Test Controller - This software element is the supervisor, monitor, or sequencer that controls other software which, like the controller, is executed by the FHID processors. The Test Controller handles I/O operations, interrupt requests, and sequences overall processor software execution during testing operations. As previously stated, there are two types of tests that require FHID processor execution of test software: 1) those tests controlled by the HOST to establish overall readiness of the processor, and 2) those tests controlled by the processor to verify correct operation of the processor ALU, memory, and registers.

The HOST controlled tests require the processor to respond to individual queries from the HOST. These queries may involve:

- Execution of I/O operations to check overall performance of the processor I/O channel.
- Processing of logic and mathematical operations to verify overall performance of the ALU
- Execution of memory readout and data relocation subroutines to verify operational capabilities of FHID processor memory and registers

The processor controlled tests, although initiated and sequenced by HOST commands, are mainly executed by the processor and encompass the type of diagnostic self test operations generally performed by a stand alone, full-fledged computer. For the FHID processor, these diagnostics include:

- An Instruction Execution, or ALU Test
- A Register Test
- A Memory Test
- A FHID Device Interface Test

The Device Interface Test is of particular importance to FHID checkout. This test verifies the processor's ability to communicate with other FHID devices, such as the CBS or DEUI, using the hardware and software resources available to the processor.

during normal operations. The FHID device interface tests are performed following the same basic sequence of steps outlined above, under HOST Controlled Interface Tests.

#### 8.1.1.4 FHID Test Sequencing

Several of the references used in this study base their test sequence development on the "bootstrap", "building block," or "start small" philosophy. This is an approach to diagnostic testing which seeks solid hardware failures by first establishing that a portion of the hardware is operating properly, and then using that hardware to check additional hardware whose condition is unknown. Inherent here is the concept that each new area is relatively small and that those areas previously tested form the foundation needed to check the new area.

The exact order and scope of each building block should be depend on the best use of the hardware in the building block in performing the next test.

In selecting the first area to be checked, the concept of a "hardcore" is taken into consideration. Hardcore, as related to automatic self test, is that part of a system that must be checked manually prior to automatic analysis of other sections. This concept is relevant in FHID testing only in so far as it is required to verify that proper power is being applied to the hardware, and that the proper test software has been loaded in the HOST. Testing then can proceed in the following sequence

1. Verify HOST/FHID interface
2. Verify HI/CBS interface
3. Verify CBS
4. Verify HI/Processor I interface
5. Load FHID Processor I with Processor Self Test Software
6. Verify Processor I memory
7. Test Processor I instruction execution
8. Test Processor I register operation not already verified by 5, 6, and 7 above. This includes interrupt processing hardware Test.
9. Repeat for Processor 2 through N.
10. Load SAMTU related software on FHID processor assigned to Master Timing Unit computations.

11. Verify interface between each FHID Processor and the FHID devices assigned to be serviced by that Processor.
12. Verify FHID/MM interface
13. Verify FHID/DEU interface
14. Verify FHID/FC-AGE interface
15. Verify FHID/FC-IOP interface

At this point overall operational readiness of the FHID can be established. If a failure has occurred, further detailed testing can be performed using the component Level Tests possible by accessing the FHID Test Panel or control console functions. It is also possible to accelerate testing and verify parallel processing capabilities of the FHID processors by loading diagnostic tests in each processor. Execution of these tests laid out in Figure 8.1-5, can then be initiated simultaneously in all processors, and the results communicated to the HOST.

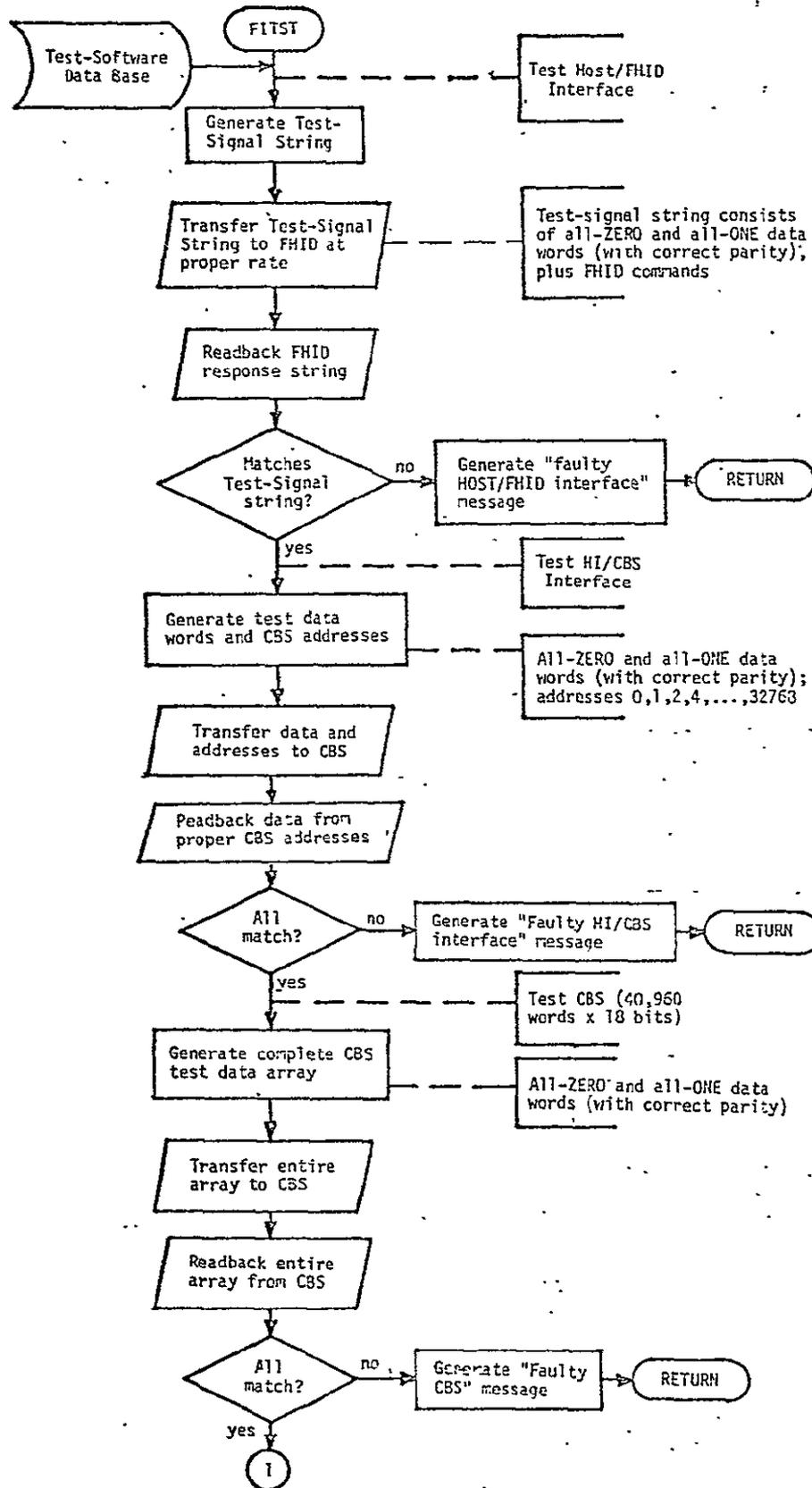
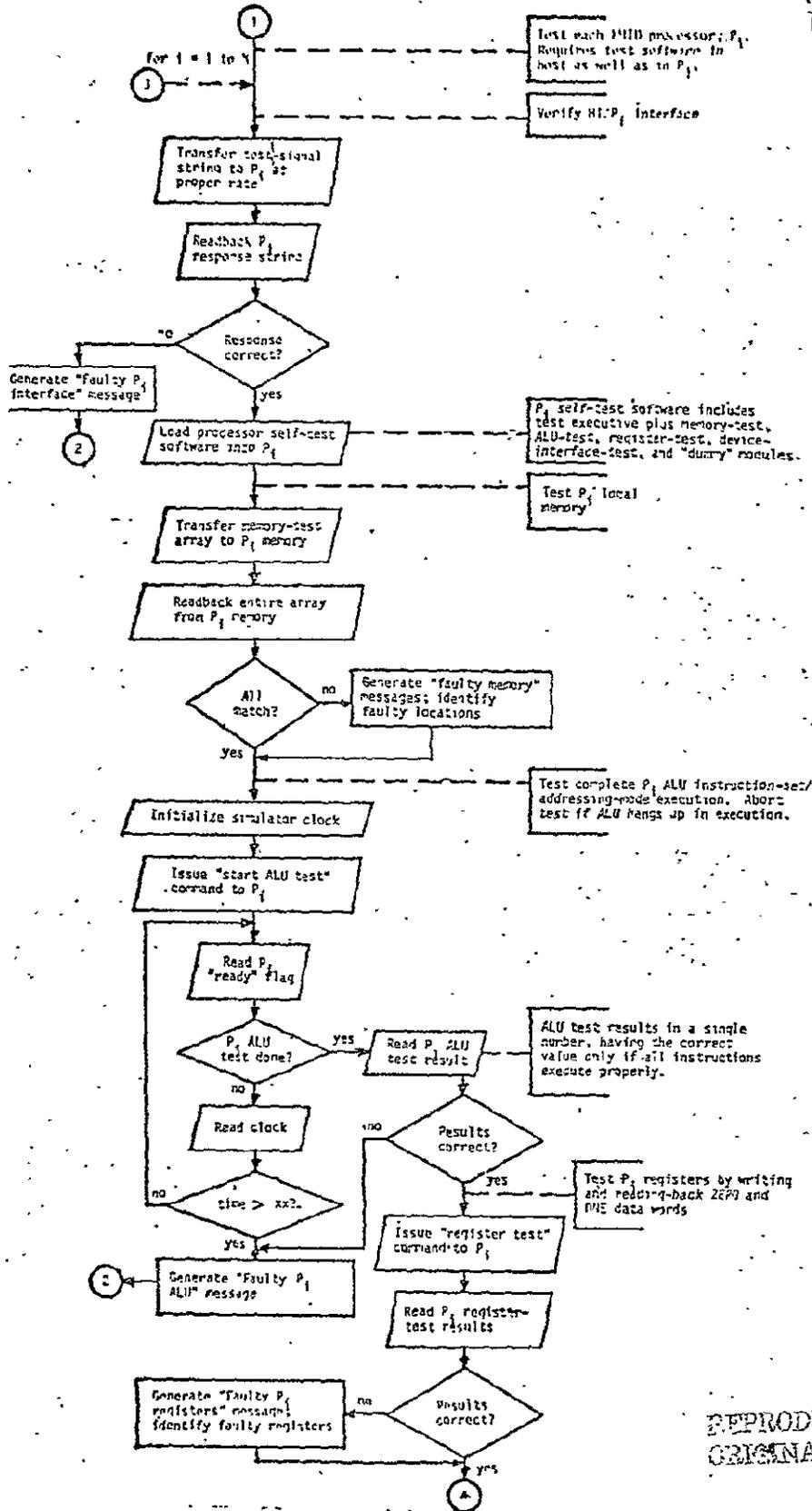


FIGURE 8.1-5 FLIGHT HARDWARE INTERFACE TEST FLOW (FITST)



REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

FIGURE 8.1-5 FLIGHT HARDWARE INTERFACE TEST FLOW (FITST) (Continued)

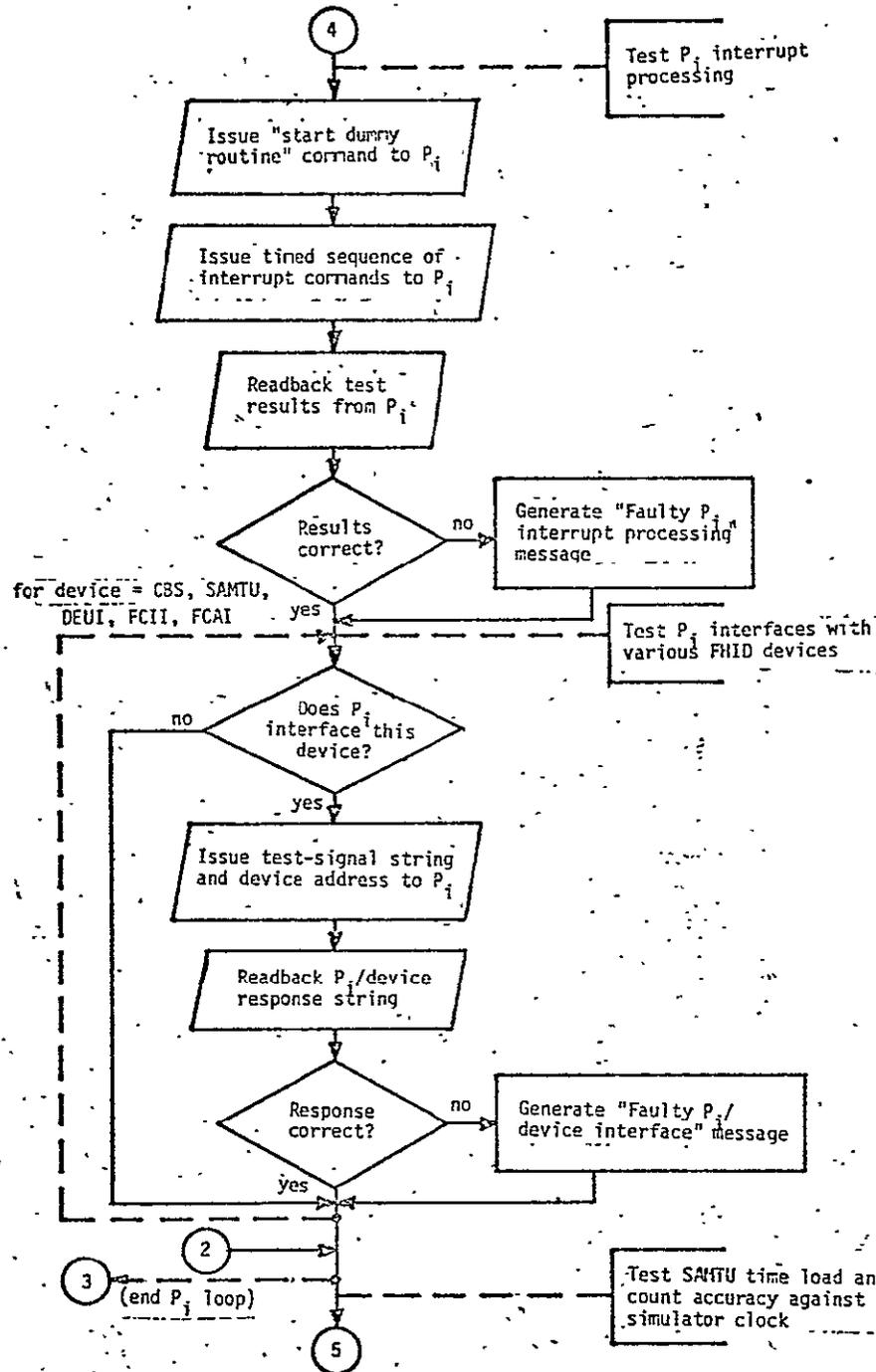


FIGURE 8.1-5 FLIGHT HARDWARE INTERFACE TEST FLOW (FITST) (Continued)

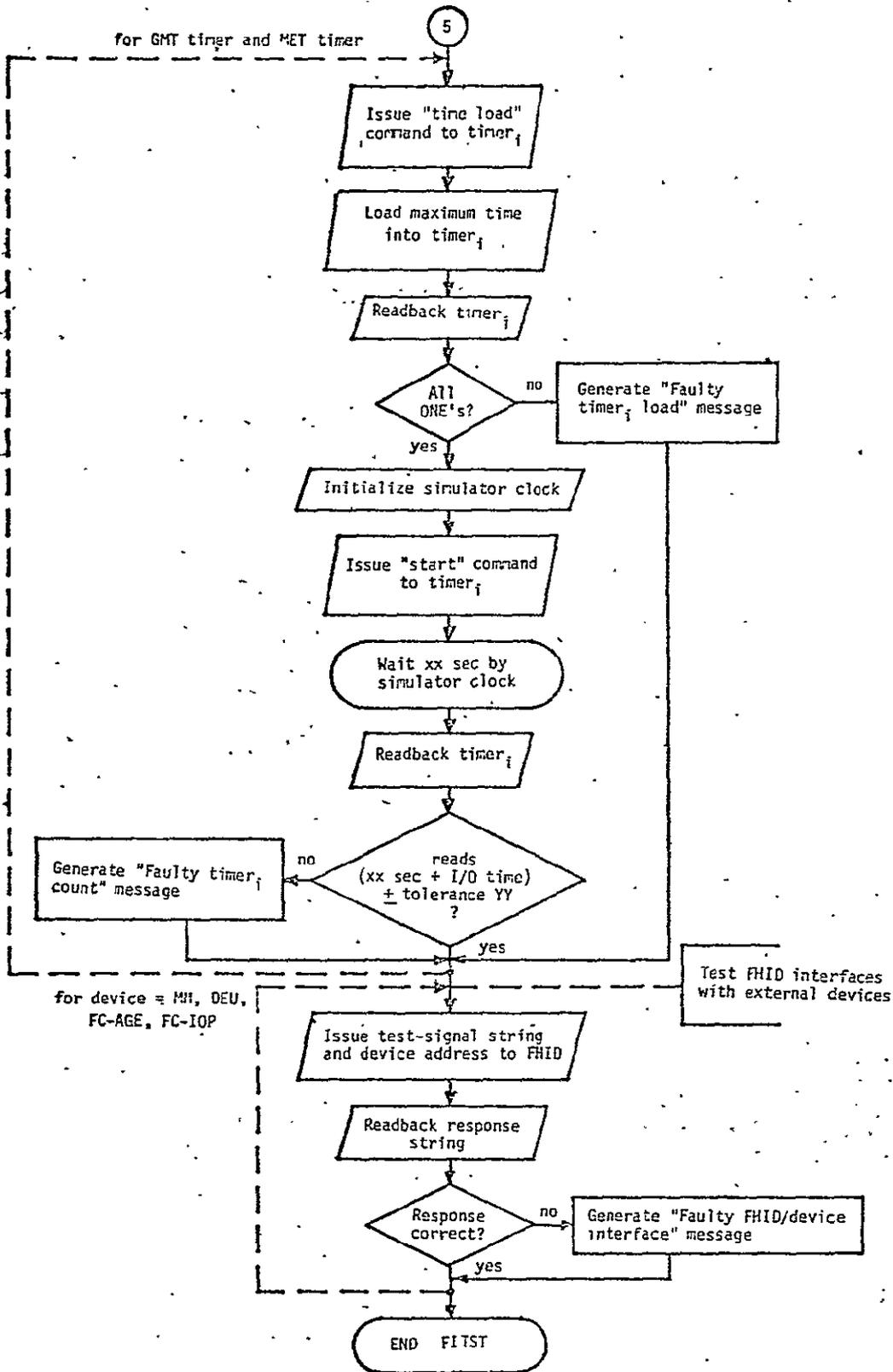


FIGURE 8.1-5 FLIGHT HARDWARE INTERFACE TEST FLOW (FITST) (Continued)

### 8.1.2 Flight Computer

The Flight Computers (FC's) provide the following data processing capability in the orbiter: computation for guidance, navigation and control, payload manipulator subsystem control, payload management, and system performance monitoring for orbiter and orbiter dependent payloads. There are five Flight Computers in the operational Orbiter. Shuttle simulators have three or five, depending on simulator configuration.

The machine selected by NASA for the FC job is an IBM AP-101 computer, of the 4 Pi family. The FC verification techniques survey that follows depends entirely on the use of vendor supplied diagnostic routines, as currently documented.

#### 8.1.2.1 Flight Computer Description

The FC consists of two major processing elements: The Central Processor Unit and the Input/Output Processor. Both processors have access to a common memory. Figure 8.1-6 shows the principal functional elements of the FC and their operational interrelation.

The AP-101 machine includes several features that make it a unique and advanced performance flight computer. Expanded storage capacity, microprogramed control, and a sophisticated multiplex input/output architecture are key features in the FC that give it a high level of performance. Of special interest to this study, is the inclusion of Built In Test Equipment (BITE) in the original design of the FC.

Standard memory capacity is 65,536 32-bit words, expandable in 8,192-word modules up to 131,072 words. The microprogramming of control increases the instruction execution efficiency of the computer, and facilitates adaptability and response of the machine performance to changes in vehicle or mission design.

Input/Output operations are performed by a processor capable of conducting 31 operations concurrently using 31 different data buses. Each data bus operates at one megahertz and can service up to 31 separate subsystems. Control of data bus transfers is maintained by a Bus Control Element (BCE). The IOP permits cross

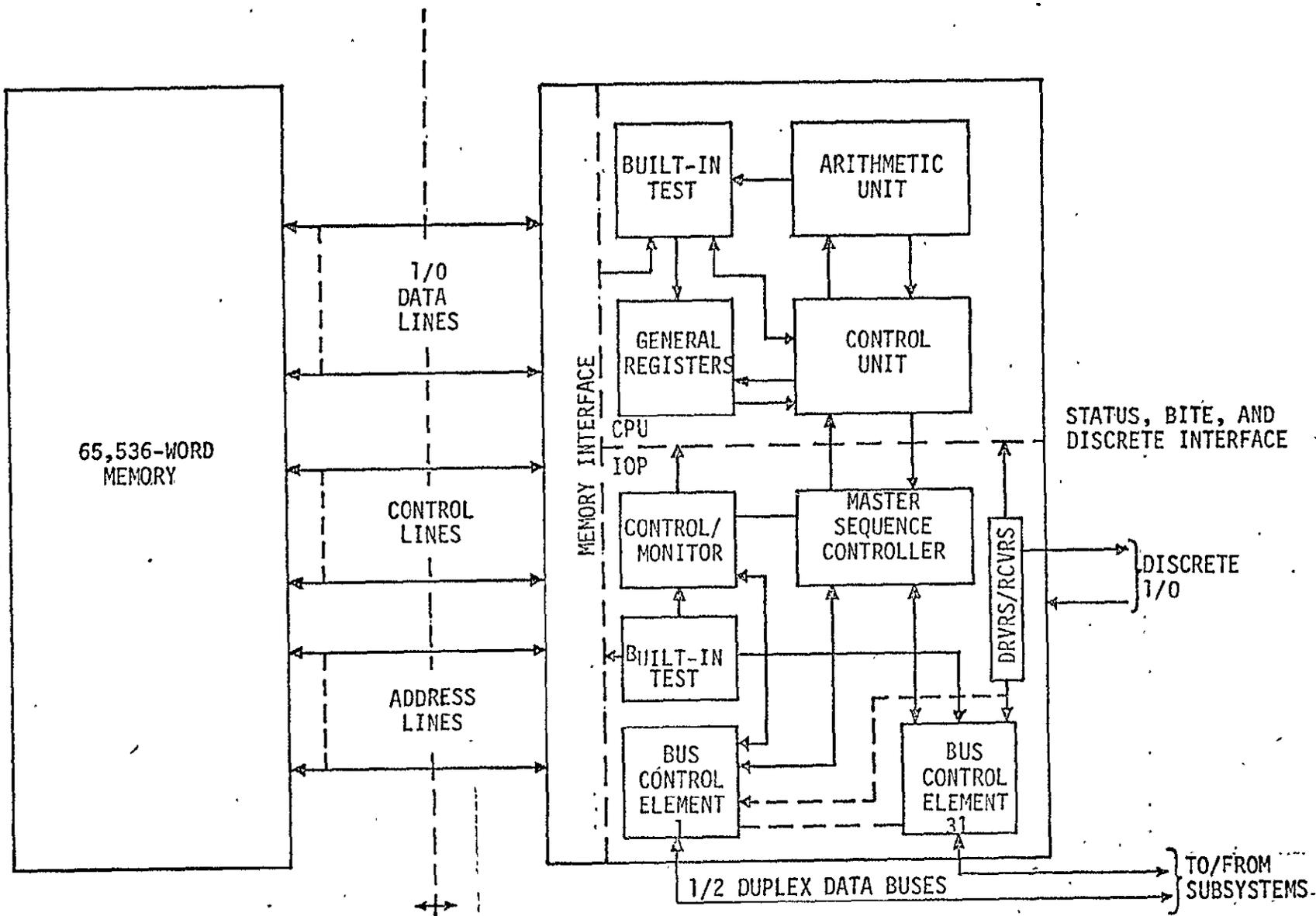


FIGURE 8.1-6 FLIGHT COMPUTER BLOCK DIAGRAM

coupling BCE's in pairs in order to allow data transfers directly from one sub-system to another without going through CPU hardware, or IOP local memory. I/O rates vary depending on the mode of transfer from 250,000 to 833,333 full words per second.

The built-in test capability includes sufficient built-in test equipment (BITE) to provide rapid (on line) and off-line self detection of malfunctions. This hardware, together with self test software and the use of micro-diagnostics provide the FC with an extensive capability for verifying the operational status of the machine.

#### 8.1.2.2 FC Built In Test Equipment

There are two sets of BITE hardware in the FC: one set in the IOP, and another in the CPU. The IOP BITE includes the following:

1. A watchdog timer. (Times out if not reloaded by program within time loaded--768 microsec to 3.1 seconds)
2. Redundancy Management voter logic
3. Parity check on IOP microinstructions
4. Parity check on memory to IOP bus
5. Timer/clock oscillator stop detector
6. Power loss for longer than 400 microseconds
7. Programmable device response timer

If any of the above hardware detect a malfunction, the FC is notified by means of an external interrupt.

The CPU BITE monitor activity between the CPU and memory and between CPU and IOP. This equipment includes the following:

1. CPU/IOP response timer (about 8 microseconds)
2. Parity check on CPU microinstructions
3. Parity check on CPU/IOP bus
4. Parity check on CPU/memory bus
5. Store protect key compare logic

Detection of a malfunction by the first four result in a machine check interrupt, an attempt to access protected storage by a program with the wrong key results in generation of a program check interrupt.

All of the above BITE is on line actively checking for malfunction during normal operation of the FC. Self test capability includes the means to verify that all BITE hardware is operational, to a .99 probability.

#### 8.1.2.3 FC Self Test Description

The AP-101 self test hardware and software ascertain the operational readiness of all computer functions to a .96 level of confidence. Self test software include the means to checkout both the CPU and IOP functions.

The CPU test program checks execution of the FC instruction set. Instructions are exercised using all registers and verifying the direct, indirect, relative and other memory addressing formats. Machine check errors are forced in order to test the fault detection capabilities of the BITE. Program interrupts are also caused, and response of the computer to these interrupts is checked. CPU tests make use of special microdiagnostics that permit quick testing of the hardware and firmware by exercising normal paths of data transfer, and also test paths that bypass BITE when it is desired to force an error, as in the case of generating erroneous parity to verify parity detection circuitry. Some of these microdiagnostics can be called for on line testing during normal FC operations.

The memory test program verifies the ability to store and read exhaustive test data patterns in all operationally accessible core locations of main storage. This test also verifies interrupt handling and related operations peculiar to specific core locations.

The IOP test program verifies correct performance of IOP elements in executing IOP microinstructions. IOP self tests also check local buffer storage operation and verify bit accuracy and timing characteristics of the Master Sequence Controller and the Bus Control Element.

#### 8.1.2.4 FC Self Test Operations

Daily Self test of the FC during operational period of the training simulator can be initiated and controlled from the HOST. The HOST can either load all self test software through the FHID, either from the HOST memory or from the Mass Memory. The total self test program to be loaded on the FC has been estimated by IBM to be 1500 words in length.

The HOST, through the FHID and the FC AGE interface, monitors the progress of the self test and detects the generation of fault indications as forced during execution of the various tests. Successful or unsuccessful completion of self test can be communicated to the HOST by the FC, or the HOST can detect the outcome by monitoring the various test indications available at the GSE connections of the FC.

#### 8.1.3 Other Minicomputers

The Simulator Reference Configuration of Section 5 includes auxiliary minicomputers to process I/O transfers between the HOST and other elements of the simulator. Minicomputers perform the formatting, data buffering, and time related adjustments in transfers to and from the Data Conversion Equipment, and also to and from the Motion Base monitor and control hardware.

These minicomputers are expected to be relatively general purpose, off-the-shelf machines supplied to NASA with support and self test software. The actual capability of this software, the types of tests run, and the level of confidence achieved by these tests vary from model to model. In either case, the testing approach discussed above for the Interface Device and its processors and also for the Flight Computers can be applied to auxiliary interface minicomputers.

The basic self test approach for ancillary computers and flight equipment interface hardware can be summarized as follows:

1. Self test of this equipment should be based on manufacturer supplied self test software and built-in test equipment.
2. Evaluate adequacy of manufacturer's self test software and hardware. Specify additional capability if required.

3. Determine method of loading self test diagnostic programs on machine to be tested. If possible, it is better to use operational loading path instead of loading through the HOST. This method of loading verifies some of the operational interfaces and reduce the amount of processing required of the HOST.
  
4. Design self test system to be controlled by the HOST in order to maximize utility of common test control, test pattern generation, and pattern compare routines. Another advantage of maintaining control at the HOST is to provide one central place where the operator can initialize total system or individual subsystem testing and monitor the results of each of the tests run.

## DATA CONVERSION EQUIPMENT

This section discusses techniques for self test of the data conversion equipment. The primary requirement for DCE self test is to establish operational readiness of the equipment using automatic techniques. The secondary requirement on the selected approach for self test is the ability to detect faults and identify the faulty LRU using automatic means for fault isolation. The most effective method for automatic test is to output data to the output devices, monitor these outputs with the input devices, and perform comparisons and fault isolation using the resultant data. This section will concentrate on techniques and methods for acquiring this data, detecting faults and isolating to an LRU.

### 8.2.1 Data Conversion Equipment Description and Characteristic Parameters

The Data Conversion Equipment is that part of the simulator which interfaces the computer with Crew Station and IOS display and control elements, as well as the Motion Base components and the Visual Simulation System. The DCE provides digital/analog/digital conversion as well as data formatting, synchronizing, storage and drive where required for the subsystems mentioned above. Figure 8.2-1 shows the major portions of the DCE involved in this signal conversion and distribution task.

#### 8.2.1.1 Input/Output Control and Address

The input/output control logic performs the control signal sequencing required to perform an orderly transmission of digital data between the minicomputer and the rest of the DCE. It is this logic that generates I/O interrupts to initiate, alter, or terminate I/O operations. Signals from this section of the logic start and advance both the input and the output address counters.

The output address counter is reset to channel zero and an upward count started by the I/O control logic with the receipt of a WRITE command at the beginning of a block transfer output. The counter is advanced to the next channel each time a word is received. The word received is routed to a specific digital-to-analog converter (DAC) or to the group of discrete drivers as indicated by the value of the address counter. The block labeled "digital output routing and address decoder" contains the logic hardware that determines which output channel to receive the data transmitted from the minicomputer.

8-41

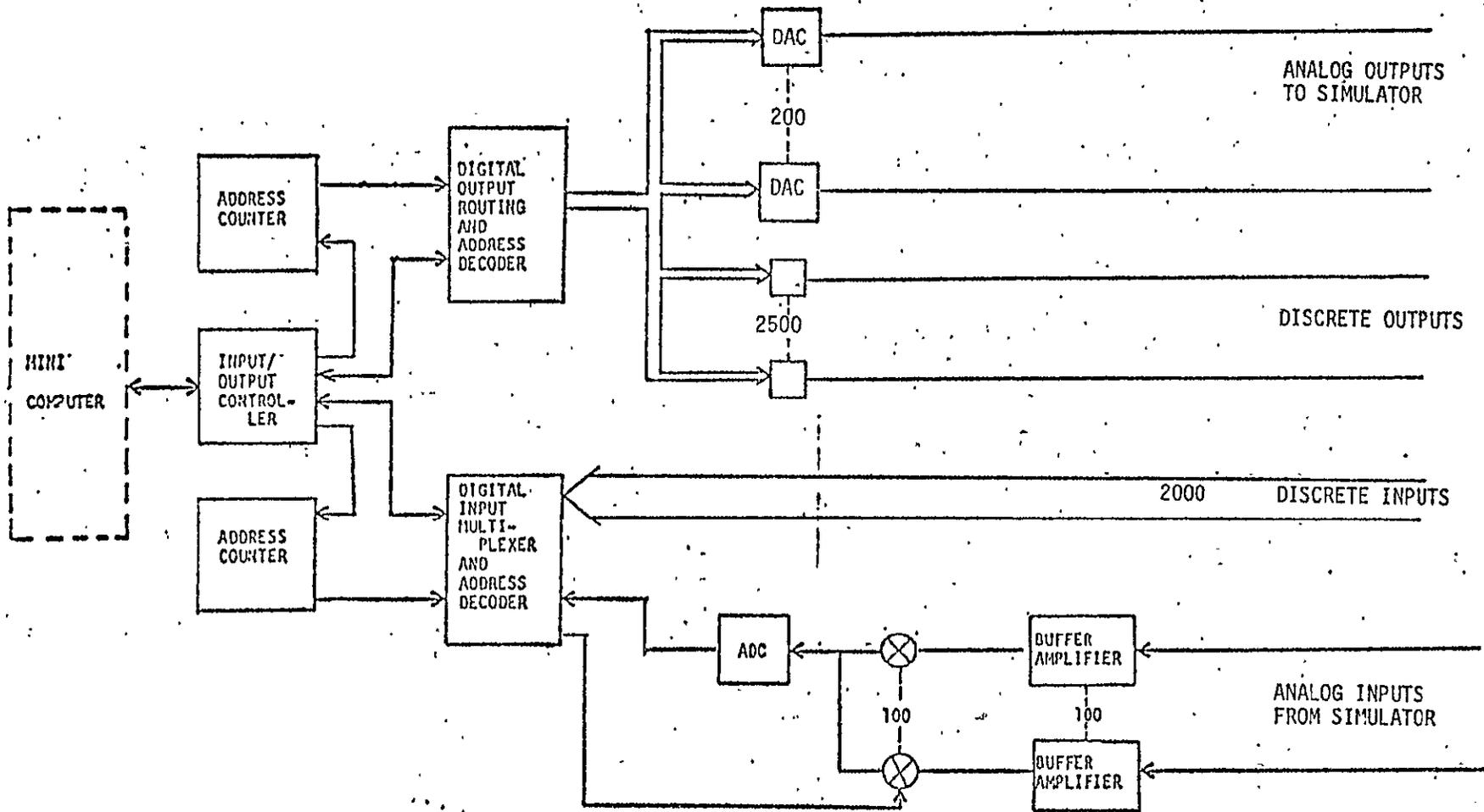


FIGURE 8.2-1 DATA CONVERSION EQUIPMENT CONFIGURATION

Data from the simulator components are also read in by the minicomputer in block form. A READ command resets the input address counter and starts a count which is advanced by the transfer of each word. Both discrete and analog information is transmitted in and out of the minicomputer in each block transfer.

### 8.1.2 Analog Signal Paths

There are 200 analog output channels, each of them with its corresponding ADC. Only one ADC is used to digitize analog data from the 100 input channels. One channel at a time is connected to the ADC by means of multiplexing solid state switches.

Buffer amplifiers are used between the input lines from the simulator equipment and the multiplexing switches in order to provide a low impedance source to the ADC, while presenting a high impedance to the analog sensors. The input impedance varies from 500 to 10,000 ohms depending on the specific design. Most of the analog sensors have impedances greater than 500 ohms and loading substantially affects the accuracy of the reading obtained.

The open impedance ( $R_{so}$ ) of the multiplexing switches may be as low as megohms. Figure 8.2-2 shows the equivalent circuit for a group of these switches feeding into the ADC. The source, or buffer amplifier output impedance ( $R_s$ ) must be much less than the input impedance of the ADC in order to assure less than 1% loading or error in the signal fed to the ADC.

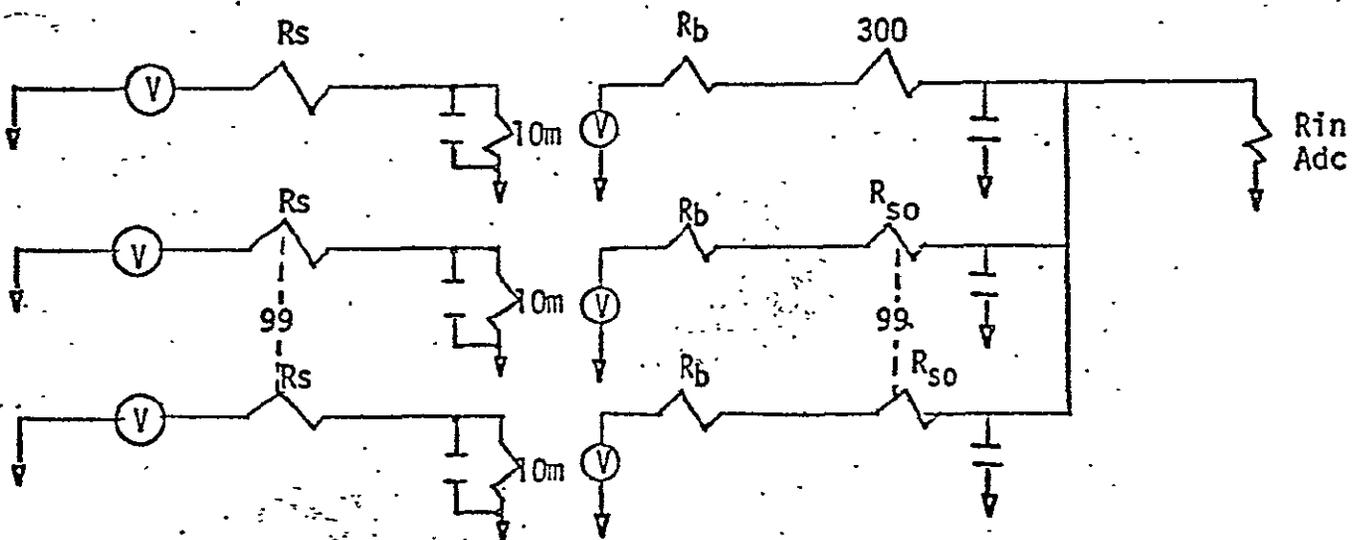


FIGURE 8.2-2 EQUIVALENT CIRCUIT WITH A BUFFER AMPLIFIER PER INPUT

### 2.1.3 Discrete Signal Paths

The 16 bits in a minicomputer input/output word can contain status/command information for a group of 16 discrete channels. Output commands are routed, according to address decode logic, to 16 bit registers which serve as the memory location for the various discrete signal drivers. These drivers provide a ground or open signal to bilevel devices in the simulator such as indicator lights or electrical valve actuators. There are 2500 of these drivers in the reference configuration.

Input discrete information is contained in the 2000 lines from simulator components to the DCE. These lines are strobed onto the minicomputer I/O bus, in groups of 16, during an input block transfer. The group routed to the bus is determined by the digital input multiplex logic.

### 2.1.4 Line Replaceable Unit Characteristic Parameters

This section summarizes the characteristic parameters of the components in a group of components, within the DCE, which will be treated as line replaceable units. It is to this level that checkout and fault isolation analysis will be carried out for automatic checkout considerations.

Table 8.2-1 lists the characteristic parameters for each of the DCE LRU's. These are the parameters of concern in the self test process, in that they are measured to determine the operational status and readiness of the corresponding LRU. The paragraphs that follow further discuss each LRU as to its function and electrical configuration. These discussions list characteristic parameters and other numerical and qualitative data that will be helpful in defining the Simulator Self Test System.

TABLE 8.2-1 DCE LRU CHARACTERISTIC PARAMETERS

LRU	CHARACTERISTIC PARAMETER	DIMENSIONS
CONTROLLER	Data	16 Bits at LL*
	Device Address	16 Bits at LL*
	Interface Controls	12 Lines at LL*
ADDRESS COUNTER	Channel Address	10 Bits at LL*
DIGITAL OUTPUT ROUTING AND ADDRESS DECODER	Data	16 Bits at LL*
	Analog Channel Select	200 Lines at LL*
	Discrete Group Select	156 Lines at LL*
DIGITAL TO ANALOG CONVERTER	Analog Output Voltage	-10 to +10 Vdc
	Output Current	Up to 10 ma
	Accuracy	0.05 to 0.025% of full scale +1/2 value of Least Significant Bit
	Settling Time	10 usec to 0.1% full scale
DISCRETE SIGNAL DRIVER	ON-Output Impedance	Less than 5 ohms
	OFF-Output Impedance	Greater than 5 Megohms
DIGITAL TO DIGITAL CONVERTER	Data	15 Bits at LL*
	Accuracy	0.02 to 0.025% full scale
	Resolution	1 part in 32,768
	Conversion Time	100 usec
ANALOG AMPLIFIER	Analog Output	-10 to +10 Vdc
	Accuracy	0-1% full scale
	Settling Time	10 usec to 0.1% final value
ANALOG CHANNEL SELECT SWITCHES	Output Signal	-10 to +10 Vdc
	Switching Time	2 usec
DIGITAL INPUT MULTIPLEXER AND ADDRESS DECODER	Data	16 Bits at LL*
	Discrete Group Select	125 Lines at LL*
	Analog Channel Select	100 Lines at LL*
	Start Conversion Command	1 Line at LL*

LL is Logic Levels which are defined as: True:  $+4.0 \pm 1.5$  Vdc  
False:  $+0.4 \pm 0.4$  Vdc

### Input/Output Controller

The input/output controller provides the signal conditioning and control between the minicomputer and the DCE. The characteristic parameters associated with signal conditioning are easily defined in physically quantitative terms; however, the control and sequencing operational characteristics require timing diagram descriptions which are usually peculiar to the minicomputer being used. This analysis will limit itself to the following parameters:

Interface signals are at logic levels which are:

True +4.0  $\pm$  1.5 Vdc

False +0.4  $\pm$  0.4 Vdc

Data Format: 16 Bits parallel

Reliability: 50,000 hours MTBF

### Address Counters

These are two counters - one for input, and one for output. Counting is done in binary form starting at zero and incrementing the count by one each time a word is transmitted. The counter specifies the address of the data word to be transmitted. The following are the pertinent characteristics:

All signals are at logic levels:

Inputs: Control timing.

Advance count command

Outputs: 10 bit binary address

Reliability: 500,000 hours MTBF

Quantity: 2

### Digital Output Routing and Address Decoder

This unit receives a binary address from the address counter, decodes the address and, upon command from the I/O Controller, transfers data words to the addressed DAC or group of discrete drivers. The following are the pertinent characteristics:

All signals are at logic levels

- Inputs: 16-bit data word
- 10-bit address
- Transfer output word command
- Control timing
- Outputs: 16-bit data words
- Storage select
- Reliability: 50,000 hours MTBF

#### Digital to Analog Converter

The digital-to-analog converters receive digital words and convert these to analog signals. The DAC receives a digital word as parallel bits accompanied by a strobe signal. This data is strobed into a digital storage register in the DAC. The digital output of the register activates analog current paths proportioned to the bit weight. These current paths are summed and converted to an analog voltage output. The pertinent characteristics are listed below:

- Inputs: All inputs are logic levels
- 8-15 bit digital value
- Outputs: -10 to +10 Vdc, at 10 ma
- Accuracy: 0.05% - 0.25% full scale  $\pm$  1/2 Value of the least significant bit (LSB) of digital output
- Settling Time: 100 microseconds to 0.1% of final value
- Reliability: 200,000 hours MTBF
- Quantity: 200

#### Discrete Signal Drivers

These drivers respond to logic level input pulses by storing data and supplying an open circuit or ground return path to bilevel indicators such as lights or flags. These drivers may be either packaged at the crew station or within the same cabinet containing the minicomputer. In either case, they are considered to be a part of the data conversion equipment. The pertinent characteristics are listed below:

Input: Logic level signals  
Outputs: High impedance, open circuit  
          Low impedance to ground  
Quantity: 2,500  
Reliability: 2,000,000 hours MTBF

#### Analog to Digital Converter

These converters take analog signal levels and generate equivalent binary values in digital format. A convert command initiates a comparison operation between the incoming analog signal and an internally generated reference voltage. This voltage is generated using digital to analog conversion. When coincident comparison is achieved, the digital bits required to generate the reference voltage are output as the digital representation of the incoming analog value. Following are the pertinent characteristics:

Input: -10 to +10 Vdc  
          Convert command - logic levels  
Output: 12 to 15 bits of data - logic levels  
Accuracy: 0.02% to 0.25% full scale  $\pm$  1/2 LSB  
Resolution: Value of least significant bit  
Conversion Rate: 50,000 words/sec  
Quantity: 1  
Reliability: 100,000 hours MTBF

#### Buffer Amplifiers

The buffer amplifiers have high input impedance to prevent loading the analog sensor or parameter being measured, and a low output impedance in order to properly match the relatively low impedance of the ADC. This low impedance interface is required to eliminate any significant inaccuracies in the measurements caused by the presence of many analog channel select switches connected in parallel. The signal amplitude is not altered as it goes through the buffer amplifier. Following are the significant characteristics:

Input: -10 to +10 Vdc  
Output: -10 to +10 Vdc  
Accuracy: 0.1% full scale  
Input Impedance: 10 megohms minimum  
Output Impedance: Less than 10 ohms  
Settling Time: 10 microseconds to 0.1% of final value  
Quantity: 100  
Reliability: 200,000 hours MTBF

#### Analog Channel Select Switches

There is one of these switches between each buffer amplifier and the ADC. They are used to select which channel is to be digitized, connecting its corresponding buffer to the ADC, in accordance with channel select commands from the digital input multiplexer. The switches discussed here are MOS-FET semiconductor switches. Below are the pertinent characteristics:

Inputs: -10 to +10 Vdc analog signal  
Switch ON/OFF control - logic level  
Output: -10 to +10 Vdc analog signal  
ON Resistance: 300 ohms nominal  
OFF Resistance: 100 megohms nominal, 10 megohms minimum  
Switching Time: 2 Microseconds  
Operating Power: 25 milliwatts nominal  
Accuracy:  $\pm .001$  volt  
Reliability: 2,000,000 hours MTBF

#### Digital Input Multiplexer and Address Decoder

The function of this section of logic is to select which data are to be input to the minicomputer at each word transfer of a read operation. It is this logic that decodes the 10-bit input address to determine which group of 16 discrete lines will be transmitted from the simulator equipment, or which

analog input from the simulator is to be connected to the ADC for digitizing subsequent transfer to the minicomputer. The following significant characteristics are:

All signals are at logic levels

Inputs: 16-bit word from discrete select logic  
12-15 bit word from ADC  
10-bit address from input address counter  
Transfer control commands from I/O controller

Outputs: 16-bit data word  
125 discrete group select lines  
100 analog channel select lines  
Start conversion to ADC

Reliability: 100,000 hours MTBF

### 8.2.2 Failure Mode Analysis

Failure mode analysis for the data conversion equipment is performed at the functional LRU level. Except for the ADC's, DAC's and buffer amplifiers, which are generally packaged as integral units, the other LRU's in the DCE are defined functionally even though they are made up of several independently packaged, replaceable integrated circuit modules of digital logic.

Table 8.2-2 lists the various symptoms that may arise during faulty performance by the DCE. This table identifies the LRU which is the possible source of trouble, and the type of fault affecting operation of this LRU.

The following paragraphs discuss in more detail the failure modes summarized on the Table. In this discussion, the terms output and command are used synonymously to represent data issue from the interface minicomputer to the DCE; the terms input and response refer to data from the DCE to the minicomputer.

#### 8.2.2.1 Command Sent to Wrong Channel

Transfer of data at the minicomputer interface is done in block form, as described in Section 8.2.1. It is possible that an analog or discrete command intended for a given channel is routed to another channel; for example, an analog command to display a change in altimeter reading may instead cause a variation in the fuel tank pressure displayed, or a command to change the TO/FROM indication on the HSI may instead cause an APU FIRE indication to be turned

EXTERNAL SYMPTOM	I/O CONTROLLER	ADDRESS COUNTER	OUTPUT ROUTING	DAC	DISCRETE DRIVERS	DISCRETE SELECT LOGIC	ADC	BUFFER AMPLIFIERS	ANALOG CHANNEL SELECT SWITCHES	INPUT MULTIPLEXER
Command sent to wrong channel	-No output address counter advance	-Failed bit to one or zero	-Failed logic element in address decode							
Failure to issue analog command	-No data transfer strobe signal	-Failed in discrete channel range		-Failure to perform digital to analog conversion						
Failure to set or reset a discrete signal	-Unable to transfer data to or from minicomputer -Faulty data bus interface component	-Failed in analog channel range			-Failed discrete driver					
Wrong value output on analog channel		-Failed component in counter logic. Erroneous start address or nonsequential counter advance	-Logic element failure in data routing circuits	-Faulty conversion circuitry -Faulty reference supply						
Wrong value input to minicomputer from analog channel							-Faulty conversion circuitry	-Degraded or failed buffer amplifier	-Unable to switch correct buffer output to ADC	-Logic element failure selects wrong channel
Wrong analog channel input to minicomputer										
Unable to input analog data to minicomputer	-Faulty bus interface control logic component	-Failed in discrete channel range					-Failed ADC			-Unable to select analog channel
Unable to input discrete data to minicomputer	-Faulty line driver or receiver -Disconnected I/O cable	-Failed in analog channel range				-Failed logic element in address decode				-Logic element failure prevents discrete data select
Unable to transfer data to or from minicomputer										

TABLE 8.2-2 POSSIBLE LRU FAILURE AND INTERNAL SYMPTOM

on. This situation may be caused by faults in either the I/O Controller, input address counter, or output routing logic circuitry.

If the I/O Controller fails to issue the counter advance signal to the address counter, the output data will be routed to the channel identified by the previous count. This kind of failure would most likely manifest itself by a solid "zero" count, not affected by an output transfer of data. In this case no analog or discrete data would be routed to any channel since channel "zero" is not assigned a channel.

The address counter logic could fail in such a way that only channel numbers that are multiples of 2, 4, 8, etc. can or cannot be loaded with data. This kind of fault could be caused by failure of a flip-flop in the one or zero state, or by failure of a logic gate in the counter sequencing logic. Failures of this kind would cause many commands to be routed to channels for which they were not intended.

A special case of counter faults discussed above would lock the counter in the discrete channel range, thereby preventing analog data to be routed to the analog channels for which they were intended.

The output routing logic is the most likely area for developing a fault that would cause erroneous steering of data. This logic is dependent on proper operation of gating to determine which channel is to receive the information. Failure of a gate would cause misdirection of data to the wrong channel.

#### 8.2.2.2 Failure to Issue Analog Command

This type of failure generally becomes evident by absence of movement in instruments, visual element, or motion base component when the minicomputer has issued an analog command to cause such movement. This failure can be caused by any of the following conditions:

- I/O controller is unable to conduct communication sequences with the minicomputer, or does not transfer data on the minicomputer bus to the DCE section for which it is intended. This fault could be caused by faulty logic circuits in the controller.

- Output address counter is locked in the discrete channel range, thereby preventing routing of analog data to analog channels.
- The digital-to-analog converter for the selected channel has failed and is unable to generate analog signals from digital data sent by the minicomputer.

#### 8.2.2.3 Failure to Set or Reset a Discrete Signal

When this type of failure occurs, a bilevel indication such as a light or flag does not change states when commanded to do so by the minicomputer. The most likely sources of the problem are:

- I/O controller is unable to achieve a successful transfer of data from the minicomputer; either because of a failed logic component or line driver or receiver.
- Output address counter may be unable to advance or is locked in the discrete channel range.
- The driver associated with the particular discrete signal has failed.

#### 8.2.2.4 Wrong Value Output on an Analog Channel

Erroneous analog outputs, except for excessive deviations from expected values, present probably the most subtle and therefore difficult to detect symptom during operational usage of the simulator. For example, if a radio navigational aid has been commanded by the computer to indicate a heading of 351 degrees, but the instrument displays 322, the pilot very likely would accept this heading as correct, and initiate a 29 degree right turn maneuver for the entire vehicle. Unless there are redundant displays, or other supporting cues the man would be reacting to an erroneously displayed situation without being aware of any discrepancy. In this case the training personnel at the IOS, or the computer itself would point an accusing finger to the innocent trainee.

Automatic checkout of the analog output paths can detect any of the following sources of erroneous analog outputs:

- I/O controller is unable to achieve a successful transfer of data from the minicomputer. This situation could prevent the alteration of a

- previously issued analog value, or cause the transfer of erroneous data through the failure of a logic or bus line interface component.
- The output address counter may have a failed component that would cause nonsequential count or erroneous start address initiation. Either one of these faults would misdirect data so that the wrong value would be sent to the analog channel in question.
  - Output routing logic component failures could cause misdirect data so that the wrong value would be sent to the analog channel in question.
  - Output routing logic component failures could cause misdirection of data as described above. However, in this case, the problem is more likely to be localized to a particular channel or group of channels. In contrast, an address counter problem usually propagates to the entire set of output channels, possibly including both analog and discrete outputs.
  - The digital-to-analog converter is again, the most likely source of problems. Erroneous analog outputs can be caused by faulty performance of the conversion circuitry due to failure of any of the many components involved. A special case of faulty conversion is that caused by improper calibration or degraded performance of the analog voltage reference supply. This situation becomes evident by the consistently inaccurate operation of all open loop, analog driven elements in the simulator.

#### 8.2.2.5 Wrong Value Input to Minicomputer on Analog Channel

The existence of this problem is generally not easily detected by the crew or training personnel during normal operation. However this may not be the case if the problem occurs in a channel that causes immediate and obviously erroneous issuance of commands from the computer to simulator components being monitored at the time. The faults that most likely cause an erroneous analog input are:

- I/O controller faults, as described above for outputs, can also cause erroneous analog inputs.
- The analog-to-digital converter is the most likely culprit in the transfer of erroneous analog values to the minicomputer. A failure in the built-in reference voltage supply, or comparing digital-to-analog conversion circuitry could cause inaccurate analog to digital conversion of simulator signals.
- The buffer, if degraded, may have variations in the unity gain characteristic of this amplifier, thereby presenting an altered signal to the ADC.
- If the analog channel select switching circuitry is faulty, it is possible that the desired channel cannot be switched from the buffer on the ADC. It is possible, although not a likely mode of failure for semiconductor switches, that the ON resistance could be exceedingly high thereby reducing the level of the buffer output signal as fed to the ADC.
- A logic fault in the input multiplexer can also input erroneous values to the minicomputer by selecting the wrong channel to be connected to the input bus.

#### 8.2.2.6 Wrong Analog Channel Input to Minicomputer

The comments above regarding wrong values input are also applicable to the case where erroneous channels are being connected to the input bus. Again, this type of failure becomes evident to the operating crew only when the channel affected has significant impact on the human sensors, or when the fault is such that all channels are selected erroneously. Sources of wrong channel input selection are:

- Input address counter could have faulty logic circuitry affecting the starting address or count sequence. A fault in the counter would most likely misdirect all channels.
- Analog channel select switching, if faulty would not accomplish the correct channel connection from buffer to ADC.
- The input multiplexer is the most likely source of this kind of problem. Erroneous decoding of the input address count could cause erroneous selection of channel to be input to the minicomputer.

#### 8.2.2.7 Unable to Input Analog Data to Minicomputer

The development of this problem would very readily become evident to someone in the simulator as he would be unable to alter the course attitude, or power setting of the vehicle with the activation of hand controllers, levers, or throttles. Possible faulty areas are:

- I/O controller, as in the output case, could have logic or line interface circuit faults that would prevent a successful data transfer.
- If the input address counter should fail in the discrete channel range of the count, no analog data could be sent to the minicomputer.
- Failure of the analog-to-digital converter would affect data conversion for all analog channels, and therefore prevent successful transfer of analog simulator data to the minicomputer.
- The input multiplexer could be the possible source of this fault if unable to issue channel select signals to the analog channels in question.

#### 8.2.2.8 Unable to Input Discrete Data to Minicomputer

Again, as in the case of no analog inputs, failure to input bi-level data would soon be noticed by the crew as no control could be exercised by means of two position switches, or digitally encoded analog devices such as hand controllers. Sources of this kind of problem are:

- The I/O controller cannot accomplish a successful input operation because of any of the reasons discussed above.
- The input address counter could have a logic failure that would lock it in the analog channel range.
- The digital input multiplexer could have a failure in its input address decode circuitry, and thereby be unable to select a group of discrete channels. A special case of failure in this circuitry could cause erroneous selection of discrete channels. This problem would be generally not obvious unless it affects a channel of significant interest at the time.

#### 8.2.2.9 Unable to Transfer Data to or from the Minicomputer

This would be one of the most severe, and therefore obvious failures. The most likely sources of hardware failure are the I/O controller logic circuits in any of the modes described above, or a simple cable connection fault.

The information documented here regarding failure mode sources will be used in determining the automatic checkout and fault isolation sequences that follow.

#### 8.2.3. Checkout Techniques

In developing the checkout techniques for the DCE, the bulk of the analysis is centered around various loop closure techniques. Various methods of closing the loop are considered and analyzed for both the analog channels and the digital channels. Various hardware options for implementing each of these loop closure methods are detailed. In addition to loop closure, other techniques are also analyzed.

Loop closure techniques for the analog channels are discussed in subsection 8.2.3.1. Discrete channels are addressed in subsection 8.2.3.2. Other techniques considered for DCE checkout are discussed in subsection 8.2.3.3. Test software flows are presented and discussed in subsection 8.2.3.4.

##### 8.2.3.1 Loop Closure of Analog Channels

The recommended loop closure configuration for checkout of the analog signals is shown within the dotted lines in Figure 8.4-3. The following paragraphs discuss the configuration, the hardware options within the configuration, and the reasons for selection of the configuration.

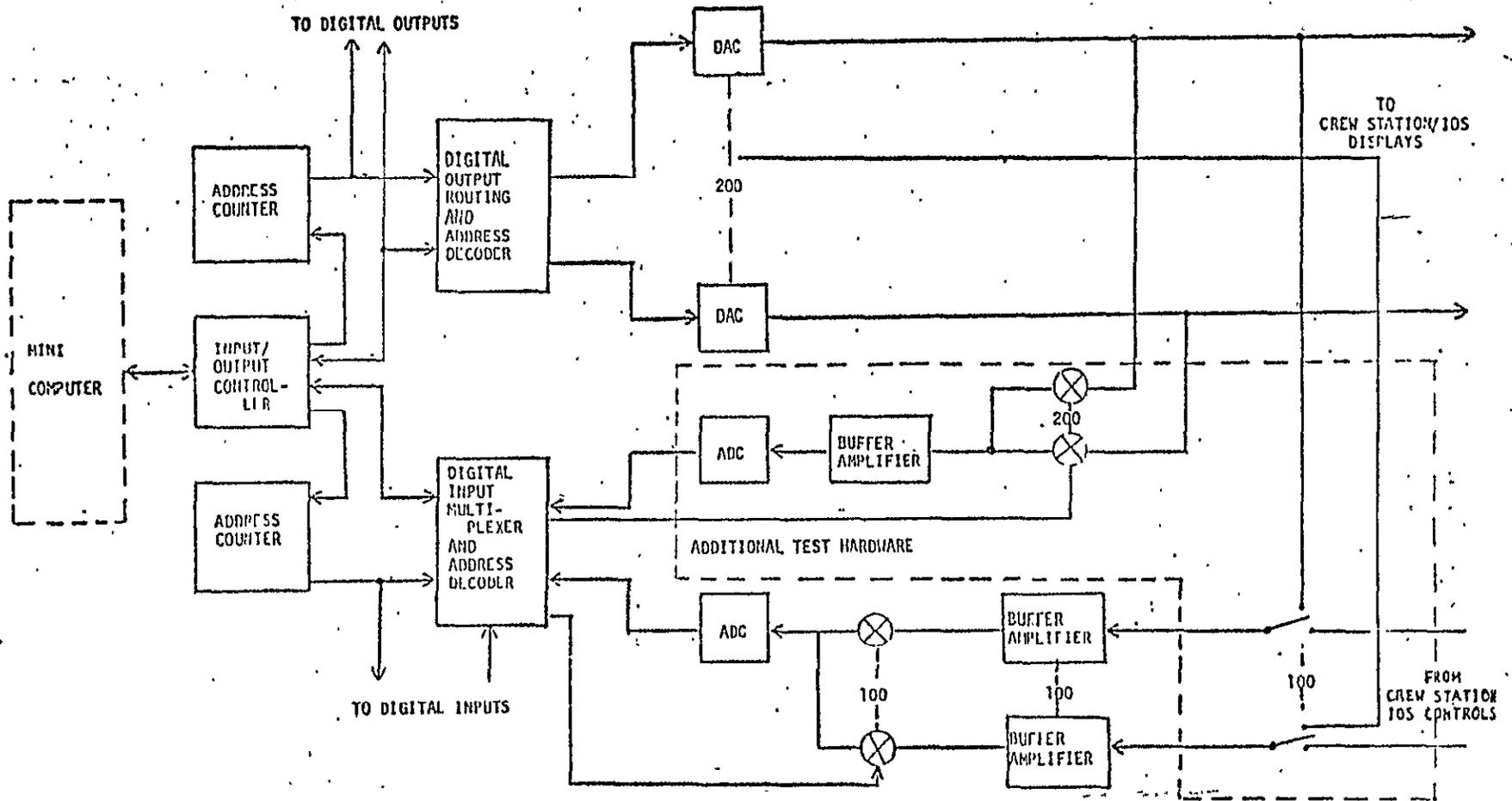


FIGURE 8.2-3 DCE ANALOG TEST CONFIGURATION

Recommended Loop Closure Method

The outputs of the digital-to-analog converters are tested through the test multiplexer and ADC, and the analog inputs are tested by applying signals from the DAC's through the single pole, double throw switches to the inputs. In the reference configuration there are 200 analog outputs and 100 analog inputs that must be switched. In performing the switching it is not necessary to disconnect the analog outputs from the simulator since the limits of the output voltages do not exceed the limits of the simulator equipment. However, it is required that the inputs be switched from the simulator to the DCE analog outputs in order to prevent interference of simulator signals with test signals.

By providing separate monitoring and conversion for test, there is no degradation to the simulator data paths. The alternative to providing separate conversion is to increase the quantity of multiplexer inputs to the ADC in the simulator data path. Figure 8.2-2 shows the equivalent circuit for inputs from the simulator with a buffer amplifier between each switch and the driving source. As can be observed from the equivalent circuit, additional switches have the effect of increasing the capacitive loading thus increasing the switching time. In addition, the loading to the driving source is increased. This increase in loading presents no problem if the buffer amplifier output impedance is low; however, inaccuracies result if the impedance is high or the amplifiers eliminated and the switches driven by higher impedance sources such as potentiometers, etc.

The equivalent circuit for the test inputs with the switches tied directly to the driving source and one buffer amplifier is shown in Figure 8.2-4. As can be readily observed from the equivalent circuit, if  $R_s$  is small the error through the multiplexer is negligible, i.e., 300 divided by 3.3M or 0.008% for 200 switches and a buffer amplifier with 10M input impedance. Since  $R_s$  is negligible for the DAC's, the error through the switches is small for the test

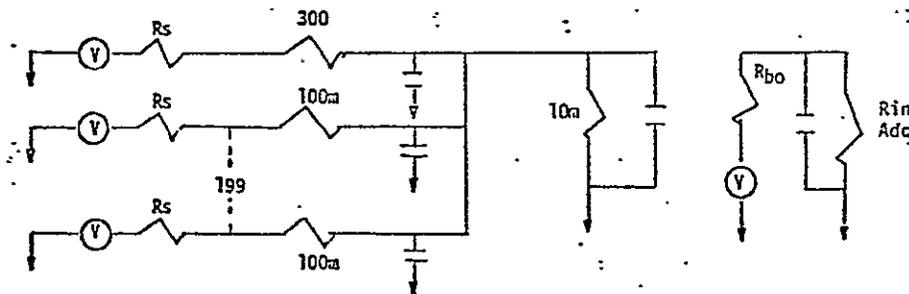


FIGURE 8.2-4 EQUIVALENT CIRCUIT WITH ONE BUFFER AMPLIFIER

loop; therefore, buffer amplifiers are not required in front of the switches as is the case for inputs from the simulator since some simulator driving sources such as potentiometers have high source impedances.

Additionally, having a separate test data path allows for more flexibility in differentiating between failures of analog outputs and analog inputs. Each of the first 100 analog outputs may be routed through either one of two analog inputs for test: the test input or the normal input. For example, data is output to DAC No. 1 and measured through test analog input No. 101; a failure is detected. The failure may be isolated to output or test input by routing DAC No. 1 through normal input No. 1.

For the test input, the multiplexer must have the capability to drive high impedance loads, shown in Figure 8.2-4 accept plus and minus inputs, and have fast switching capability to permit extensive multiplexer sequencing during test. The MOS-FET's described below meet these requirements. Relays tend to be eliminated by the switching rate requirement.

The double throw switches in the analog input loop may be all switched prior to test with the requirement that the switches be settled before test. Thus, relays or similar type switches may be used. MOS-FET's may also be used for this application; two MOS-FET's are required per switch.

MOS-FET semiconductor switches may be used for the multiplexer and the double throw switches. These MOS-FET switches are reliable and have high switching speeds; the "off" resistance is nominal. The operational characteristics for these switches are listed in paragraph 6.4.1.4.

The packaging is small for the MOS-FET's; two to four switches with drivers can be packaged in one dual-inline integrated circuit. The estimated cost of parts for the system of 200 switches is \$2,500.

Relays are reliable, provide low "on" resistance, high "off" resistance, and high current capability. Generally, the switching time is very low compared to other methods. Also, the power required for operation is relatively high and usually require additional driver circuits. Some characteristics of relays in the general class of interest are as follows:

Reliability - 100 million operations

On resistance - 200 milliohms

Switching time - 1.0 milliseconds

Off resistance - 10,000 megohms

Operating power - 50 milliwatts

Due to the low current requirements, the packaging is small: dual-inline packages or equivalent. The estimated cost for the system of 300 switches is \$2,500 for parts.

### Other Configurations

Four other configurations were considered for closing the loop; however, these did not appear adequate enough to be recommended for implementation. The following paragraphs discuss these configurations.

Crossbar Matrix - The crossbar provides the capability of switching inputs to outputs in a matrix configuration, as shown in Figure 8.2-5. The switching of the 100 inputs to the 300 outputs (200 DAC's and 100 simulator outputs) requires, as a minimum, two 10 x 30 x 6 crossbars. This configuration, however, is not a universal matrix but is limited to switching any group of six inputs to any group of six outputs. Additionally, connections may be made only inside each crossbar. This switching technique is more than adequate for isolating failures, however; since the capability of switching between only two separate paths for each signal is required for isolation.

Crossbars are reliable, provide low "on" resistance, high "off" resistance and high frequency range. The switching time is very slow and the power required for operation is high. Typical characteristics of the crossbar are as follows:

Reliability - 100 million operations

ON resistance - 400 milliohms

OFF resistance - 100,000 megohms

Switching time - 18 milliseconds

Operating power - 1.5 watts

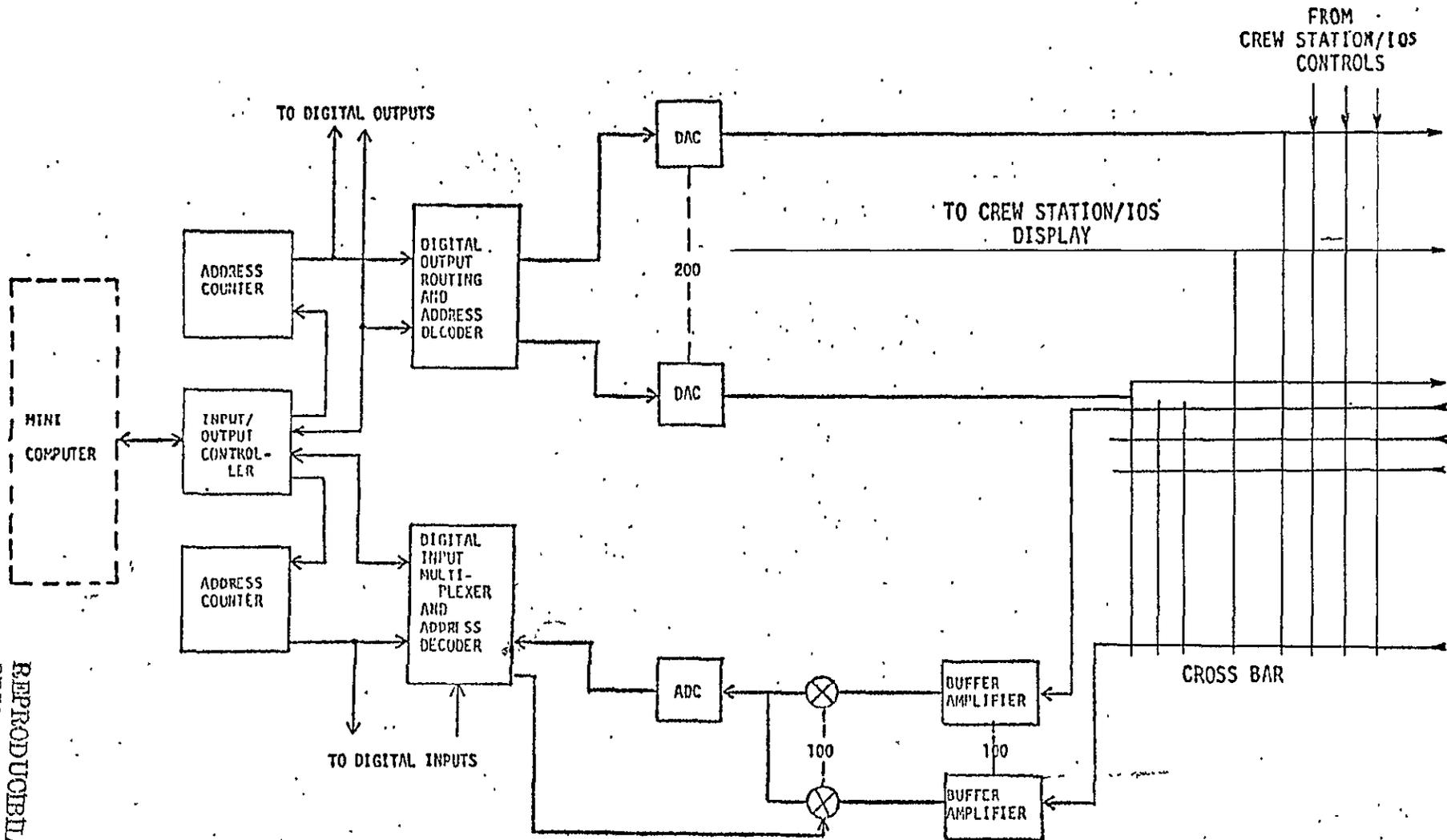


FIGURE 8.2-5 DCE CONFIGURATION WITH CROSSBAR SWITCHING

The packaging for the crossbar switches is larger than for small relays: on the order of 2 to 3 times. The estimated cost for the two 10 x 30 x 6 matrices is \$4,500 for parts.

The crossbar provides much more flexibility than is required for this application. The crossbar provides the capability of switching multiple inputs to multiple outputs; whereas, switching between two data paths is sufficient for fault isolation. The major drawback to the crossbar is the switching time of 18 milliseconds. Using two 10 x 30 x 6 matrices, the crossbar must be switched twice: before checkout starts and in the middle of checkout. An additional switching will occur if a failure is detected. The second switching may be eliminated by using four 10 x 30 x 6 matrices and increasing the input multiplexer from 100 to 200 inputs. Two additional negatives are the slightly higher costs and the larger space required.

Reed Relay Matrix - Reed relays may be configured in a matrix to switch combinations of inputs to combinations of outputs similar to the crossbar. The quantity of relays required are a function of the number of combinations in the matrix. The minimum number of relays required for switching the 100 inputs to the 300 outputs is 300; this provides two data paths for each DAC output. Figure 8.2-6 shows the switching configuration for the minimum number of relays. The maximum number for a universal 100 x 300 matrix is 30,000 relays. The minimum number is sufficient since only two separate data paths for each DAC is required for isolating faults. The major drawback to the reed relays is the switching time, 0.5 milliseconds minimum. With the minimum configuration the relays are switched twice for checkout and again if a failure is detected. The reed relays do, however, switch much faster than the crossbar.

#### 8.2.3.2 Loop Closure of Discrete Channels

The recommended loop closure configuration for checkout of the discrete signals is shown in Figure 8.2-7. This configuration and variations thereof are discussed in the following paragraphs. Also, using relays for loop closure are discussed.

8-63

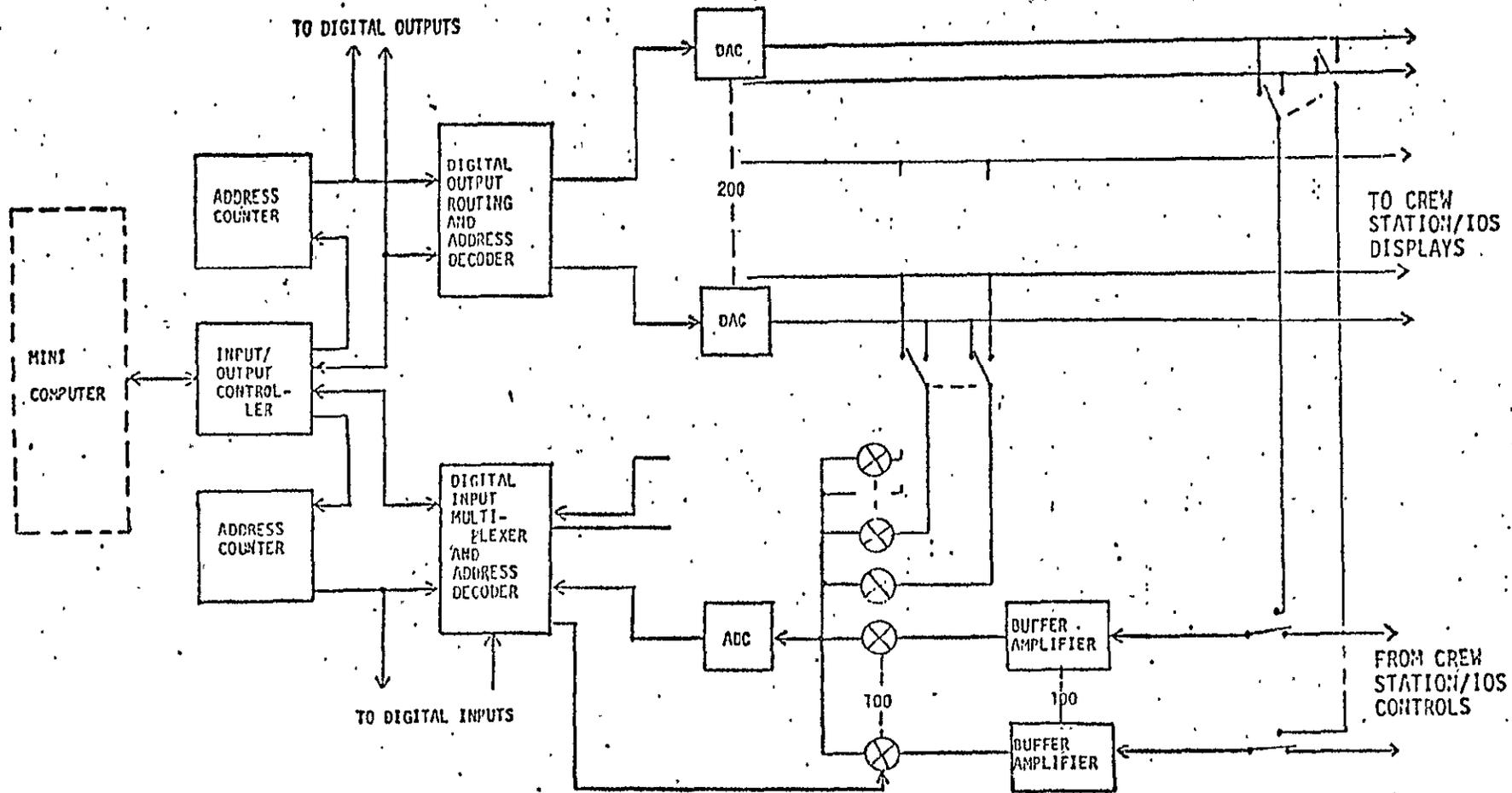


FIGURE 8.2-6 DCE CONFIGURATION WITH RELAY SWITCHING

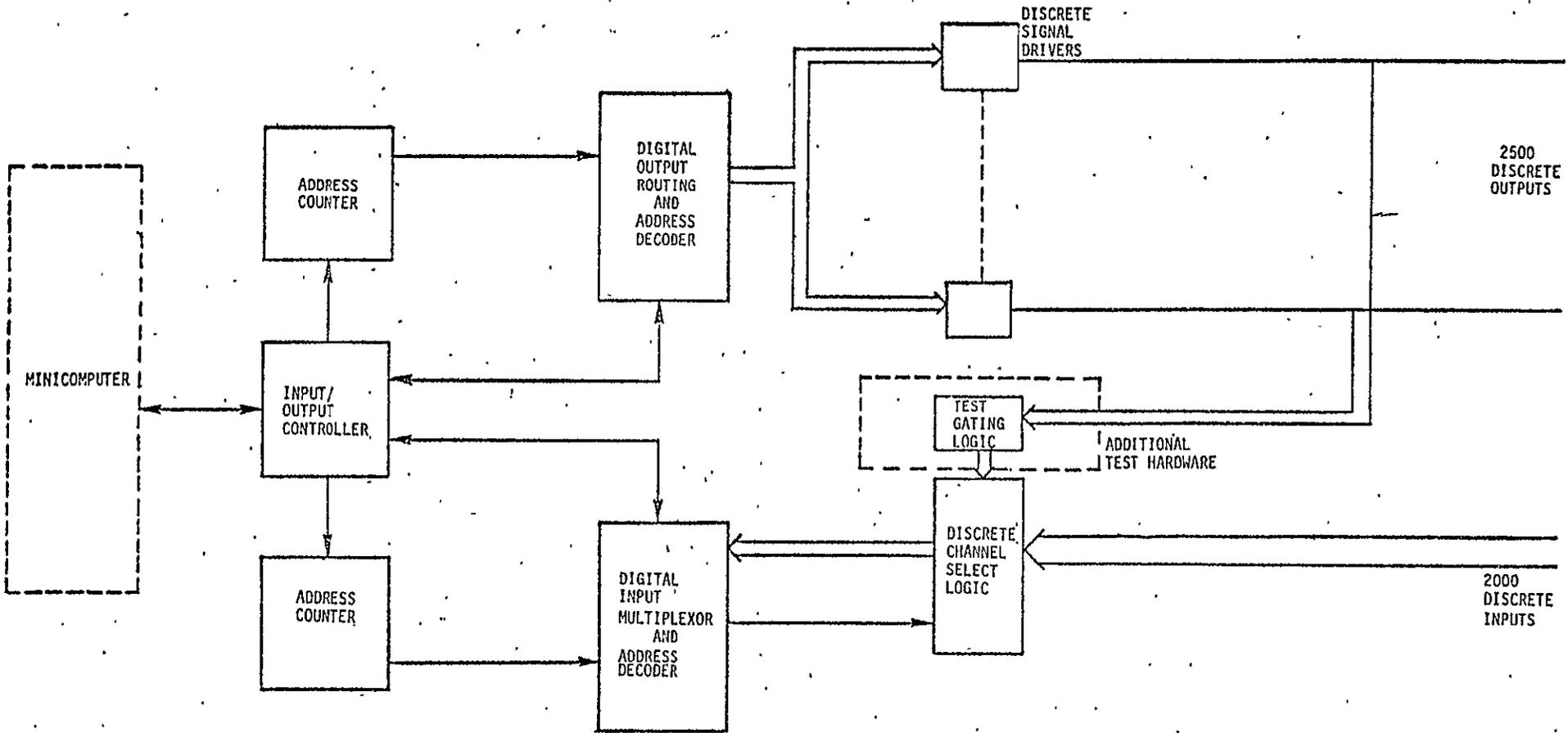


FIGURE 8.2-7 DISCRETE DCE TEST CONFIGURATION

### Recommended Loop Closure Method

The test configuration for checkout of the discrete inputs and outputs is shown in Figure 8.2-7. The loop is closed by increasing the input capability of the digital input multiplexer. The output discrettes are monitored through these inputs. The additional input circuitry is shown in Figure 8.2-8. The circuit within the dotted lines is that required for one bit of the 16-bit word transmitted to the minicomputer. Thus, approximately  $171 \times 16 = 2,736$  additional gates are required to monitor the 2500 discrete outputs. This additional input circuitry is a duplicate of the operating input circuitry of the multiplexer. The circuitry for both will have the same address with a test mode signal or run mode signal selecting which input. The outputs are OR'd at the output line drivers. The operating input gates between the line drivers and input signals are assumed to be operating properly. This assumption is reasonable since in automatic checkout some fraction of either the test or operating hardware must be assumed correct. For example, if relays are used to switch input signals, then the relay contacts to the inputs are assumed correct. Another reason that the assumption is valid is that the reliability for this group of input gates is approximately 2,000 hours mean-time-between-failure. The failure rate is low enough so that the failure is assumed elsewhere until all other possibilities are eliminated. The estimated cost for parts is \$2,000.

### Other Configurations

Two additional configurations were examined for closing the loop for the discrete signals. These are discussed in the following sections.

OR'd Input Gates - A variation of the logic diagram in Figure 8.2-8 was examined. This variation is shown in Figure 8.2-9.

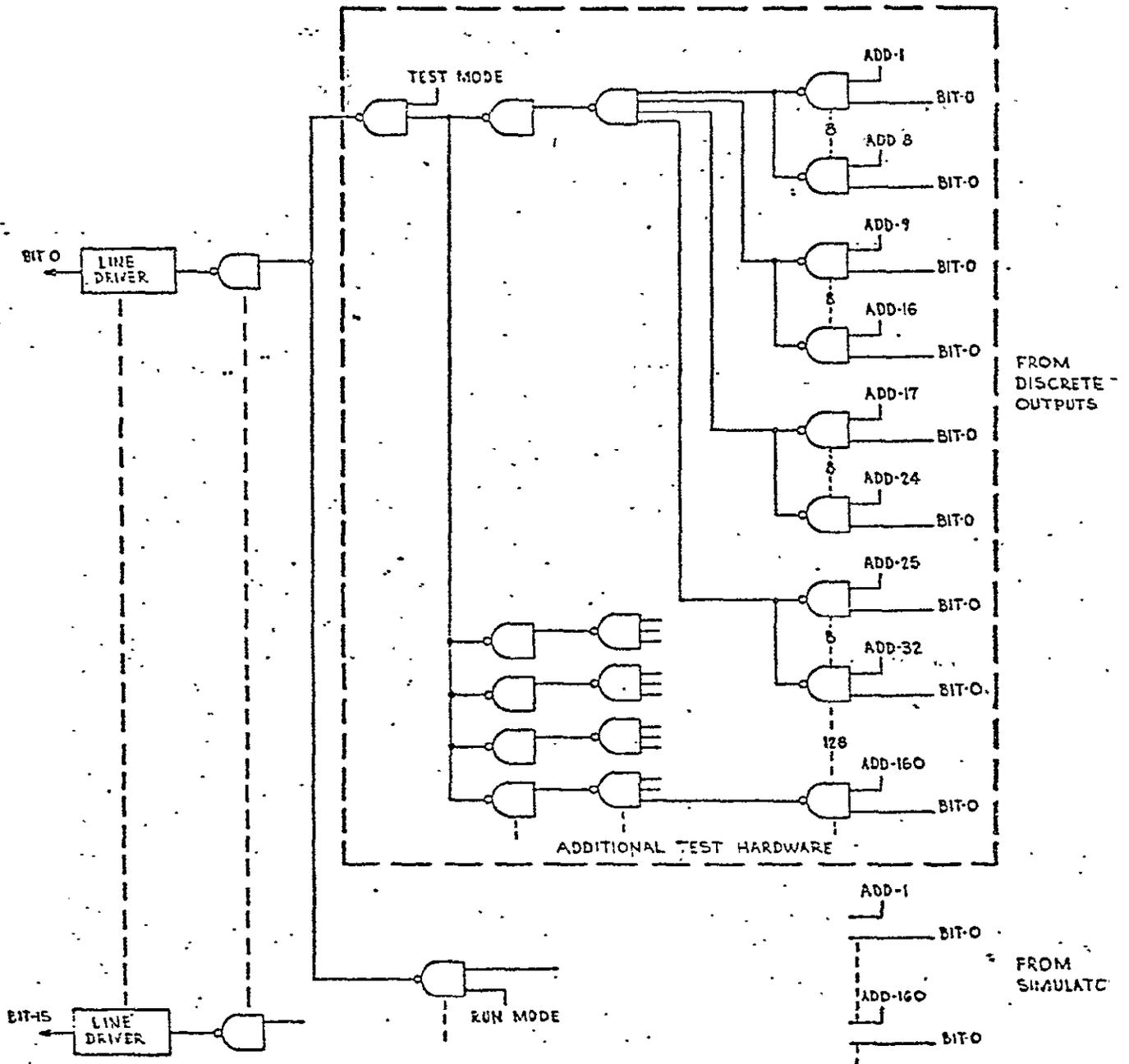


FIGURE 8.2-8 ADDITIONAL TEST HARDWARE FOR BIT-0  
FOR DISCRETE LOOP CLOSURE

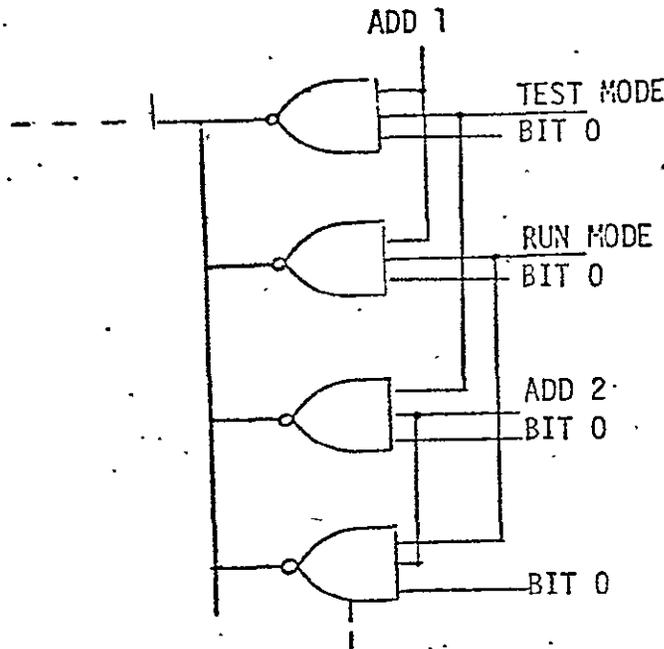


FIGURE 8.2-9 OR'D INPUT GATES

The test gates and operating gates are OR'd at the outputs of the first line of gates instead of at the inputs to the line drivers. Combining the test and operating signals at the first gate has the advantage of checking the additional eleven gates per bit or 176 gates between the operating input gates and the inputs to the line drivers. However, as seen in Figure 8.2-9, gates with three inputs are required instead of two at the inputs. There are approximately 2500 of these gates. The two-inputs gates are normally packaged four per integrated circuit, and the three-input gates are packaged three per integrated circuit. Therefore, approximately 200 additional IC packages are required for the configuration with three-input gates. Since the reliability is based upon the number of packages and not the number of gates, the logic configuration in Figure 8.2-8 will have less failures in the operating gates that are not being tested by the loop closure than this configuration.

Relays - Another configuration examined included the relay configuration described for the analog checkout in Section 8.2.3.1. The relays are packaged one per package; therefore approximately 2500 units are required or four time the number of packages required for implementing with logic circuits. The cost for parts for the relays is estimated \$15,000 or over seven times the cost of parts using integrated circuit logic.

### 8.2.3.3 Other Techniques Analyzed

Sections 8.2.3.1 and 8.2.3.2 discussed the recommended loop closure techniques for verification of the data conversion equipment. Five other techniques were analyzed as to their applicability and value to DCE checkout. Table 8.2-3 summarizes these techniques and the analysis performed.

#### Close Loop at Crew Station

Although this approach may be considered a packaging variation of the loop closure within the DCE, it does increase the level of confidence achieved by verifying that signal paths are intact all the way to the actual crew station.

This checkout configuration requires the installation of digital-analog-digital conversion elements directly at the crew station. The number of lines between the DCE and the crew station may be reduced by using multiplexing and decoding logic hardware. Data is sent in serial words from the cabinet containing the minicomputer to the conversion equipment in the crew station.

A disadvantage, however, is the need to install the conversion equipment at the motion base crew station. This installation imposes extra demands on the load carrying capability of the motion base; the additional hardware requires space that could be better used for crew station simulation equipment; and further, the additional power requirements increase the need for cooling equipment and for other environmental control facilities at the crew station.

#### Using Output Signal Measuring Equipment

This technique involves the addition of fairly independent channel monitoring and comparison hardware. This hardware has the capability to sequence from channel to channel reading and comparing its output to a value previously defined by a computer output to the test hardware. If a channel is found to be out-of-tolerance the hardware stops sequencing and the sequence counter value is sent to the computer as an identifier of the faulty channel. Sequencing then is restarted, and continued until all channels are checked.

TECHNIQUE	HARDWARE MODIFICATION	SOFTWARE REQUIREMENTS	RELATIVE COST	REMARKS
CLOSE LOOP WITHIN DCE	<ul style="list-style-type: none"> <li>- ADD 200 SEMICONDUCTOR SWITCHES FOR ANALOGS</li> <li>- ADD 1 ADC</li> <li>- ADD 1 BUFFER AMP</li> <li>- ADD 2500 GATES FOR SWITCHING DISCRETES</li> </ul> ALL ABOVE TO BE ADDED WITHIN DCE CABINETS	EXTENSIVE RECONFIGURATION, STIMULI AND RESPONSE READ COMMAND SEQUENCING, TOGETHER WITH RESPONSE ANALYSIS PROCESSING FOR FAULT ISOLATION REPORTING	MODERATE LOW	<ul style="list-style-type: none"> <li>- DOES NOT CHECK CABLING FROM DCE TO CREW STATION</li> <li>- DOES PROVIDE TEST CAPABILITY FOR ACTUAL DATA CONVERSION EQUIPMENT TOGETHER WITH ADDRESSING AND SELECTION LOGIC</li> </ul>
CLOSE LOOP AT CREW STATION	<ul style="list-style-type: none"> <li>- ADD ALL COMPONENTS LISTED ABOVE PLUS</li> <li>- ADD MULTIPLEXING HARDWARE IN CREW STATION TO REDUCE DCE TO CS LINES</li> </ul>	SAME AS ABOVE	MODERATE	<ul style="list-style-type: none"> <li>- CHECKS SIGNAL PATH ALL THE WAY TO THE ACTUAL INSTRUMENT OR ACTUATOR DRIVE SIGNAL</li> <li>- ADDS WEIGHT, SPACE, AND POWER REQUIREMENTS ON CREW STATION INCLUDING MBCS</li> </ul>
ADD OUTPUT SIGNAL MEASURING EQUIPMENT	<ul style="list-style-type: none"> <li>- ADD REDUNDANT SET OF INPUT DCE: I/O CONTROLLER, MULTIPLEXER, ADC, BUFFER AMP, ETC. PLUS SIGNAL COMPARING CIRCUITRY AND CHANNEL SEQUENCING LOGIC</li> </ul>	SAME AS ABOVE BUT LESS RESPONSE ANALYSIS PROCESSING <ul style="list-style-type: none"> <li>- ADD DEVICE HANDLER FOR TEST EQUIPMENT</li> </ul>	MODERATE HI ON HDWE MODERATE LOW ON SFWE	<ul style="list-style-type: none"> <li>- PERMITS ON-LINE MONITORING WITH QUICK FAULT DETECTION CAPABILITY</li> <li>- CHECKS ONLY OUTPUTS</li> </ul>
ADD INPUT SIGNAL SIMULATION EQUIPMENT	<ul style="list-style-type: none"> <li>- ADD ONE I/O CONTROLLER, ONE ADC, BUFFER AMP, SWITCHING LOGIC, AND CHANNEL SEQUENCING LOGIC</li> </ul>	SAME AS ABOVE BUT WITH EMPHASIS ON RESPONSE ANALYSIS	MODERATE	<ul style="list-style-type: none"> <li>- CHECKS ONLY INPUT PATHS</li> </ul>
ADD BOTH OUTPUT MONITOR AND INPUT SIMULATION EQUIPMENT	<ul style="list-style-type: none"> <li>- ADD ONE I/O CONTROLLER PLUS ALL OTHER HARDWARE FOR ABOVE</li> </ul> 2 CANDIDATES	SAME COMPLEXITY AS CLOSING LOOP BUT LESS INDIVIDUAL CHANNEL DATA BASE PROCESSING	HIGH	<ul style="list-style-type: none"> <li>- INCREASES COMPLEXITY AND QUANTITY OF HARDWARE ADDING POSSIBLE SOURCES OF FAULTS</li> </ul>
ADD ADDRESSABLE STATUS MONITOR REGISTERS	<ul style="list-style-type: none"> <li>- ADD 8000 LOGIC GATES TO SWITCH ANY LINE ON TO THE MINICOMPUTER</li> <li>- ADD TEST ADDRESS DECODE LOGIC</li> <li>- ADD TEST SEQUENCE CONTROL LOGIC</li> </ul>	<ul style="list-style-type: none"> <li>- SAME COMPLEXITY AS FOR CLOSING LOOP, EXCEPT NOT AS EXTENSIVE</li> <li>- COMPLEX FAULT ISOLATION ANALYSIS</li> </ul>	MODERATE ON HDWE HIGH ON SOFTWARE	<ul style="list-style-type: none"> <li>- PERMITS DETAILED FAULT ISOLATION BEYOND LRU AND DOWN TO COMPONENT LEVEL</li> </ul>

TABLE 8.2-3 CHARACTERISTICS OF CANDIDATE CHECKOUT TECHNIQUES

The test sequence is as follows:

1. The computer issues a reference bit pattern to the test hardware.
2. The same pattern is issued to each analog output channel.
3. The test sequence is started at the test hardware. Each analog channel is compared, after being digitized, with the reference bit pattern. Tolerances are defined and allowed for in the comparison operation.
4. Comparison out-of-tolerance is followed by an interrupt to the computer and the subsequent transfer of identifying address of faulty channel to computer.
5. Reference bit pattern is sent by the computer to groups of discrete signal drivers.
6. Test sequence then is initiated for discrete channels. No tolerances are allowed in comparison because deviation in any single bit indicates fault in a discrete channel.
7. Faulty comparison causes the generation of an interrupt, followed by transfer of group address and pattern read to identify faulty channel.
8. Other bit patterns, different from those issued above, are then used to verify that all bit positions in output channels are operating correctly.

The additional hardware is more complex than that required for connecting output lines to input channels. However, software is simplified in that only a few data patterns are stored for outputs of stimuli, and no comparison or channel identification software is needed as these functions are performed by the hardware.

A significant advantage of using output measuring hardware is the ability to perform on-line monitoring of output generation equipment performance. With some additions to the hardware and the DCE device handler software, every output can be routed both to the test hardware and the actual DCE. Comparison is performed while the output block transfer is being accomplished. If a faulty comparison occurs, the computer is notified by the hardware and an error recovery routine initiated to determine whether a fault in that particular

channel warrants an abort, a retry, or no action at all. A possible drawback to this type of on-line monitoring is that a failure in the test hardware could impair performance of the simulator when actually the operational hardware is healthy.

#### Using Input Signal Simulation Equipment

This technique is quite similar to the output measuring technique discussed above. However, the emphasis is on checking the integrity of input channels.

The philosophical argument for thoroughly checking output or input only is based on the increased confidence obtained by daily checking that half of the system that is the largest source of abortive failures. For the reference configuration used in this study, the output section is the one to be checked daily. This section is larger in component count and contains more digital/analog conversion equipment than the input section.

The section which does not get the daily verification is checked less frequently, say on a monthly basis. The lower frequency requirements for this checkout permits the use of less automated techniques which although take longer to perform, are used to detect incipient failures at an overall reduced cost for the life of the program.

#### Using Automatic Test Hardware Both on Input and Output Paths

This technique is an integration of the two configurations above. The presence of test hardware on both output and input has the advantage of higher confidence level in readiness status of the simulator, plus the additional benefits of on-line monitoring for output channels.

Major disadvantage is the added cost of test hardware in that the hardware added represents a nearly 80% increase in total component count. Clever design in switching and multiplexing circuitry could reduce this estimate somewhat, but even then the magnitude of added complexity may tend to reduce reliability of the system rather than enhance it as intended.

### Add Addressable Status Monitor Registers

This technique requires addition of gating logic to connect various operational bi-level component outputs to the minicomputer input bus. Additional registers are required to store status information, as well as the occurrence of predefined events or sequence of events. These registers have information regarding existence or loss of synchronization either in timing or serial transfer operations, as well as a record of past successful or erroneous activities such as the number of channels activated and whether a checksum or parity error occurred during a data transfer.

The major advantage of this technique is the extensive automatic fault isolation capability available. In effect, the DCE has test points at all the important lines that contain digital information regarding the operational capabilities of the equipment. These test points permit identification of the faulty flip-flop, or even gate, whenever a fault is found to exist.

The major disadvantage of this technique is that no actual analog outputs are verified, and therefore the performance of the digital-to-analog converters is not checked. Only those paths containing digital data are instrumented. On the input side, no check of buffer amplifiers or ADC is achieved.

Software complexity for this technique is excessive in that the software must maintain a virtual logic diagram of the DCE in the data base. This information is needed in the process of isolating the faulty component in the event of a failure.

#### 8.2.3.4 Test Software For Closed Loop DCE Checkout

The flow chart for verifying the data paths is shown in Figure 8.2-10. Values are transmitted to each digital-to-analog converter from FS (full scale) to -FS. The values selected are +FS, +1/2FS, +0, -0, -1/2FS, -FS. Using these values all bits are exercised; the linearity is checked; the full scale reference is verified; and the zero offset is checked. These values are set in memory in sequence from +FS, to -FS; i.e., address n contains +FS, n+1, 1/2FS, n+2, +0, etc. Each DAC receives one value during a block output. Because of the location of data in memory, each adjacent DAC has a different value



with each value repeating every seventh DAC. By sequencing the output in this manner channel selection and crosstalk are verified. Additionally, the same values are output to the discrete channels. All bits are tested and sufficient values are output to verify crosstalk and channel selection.

Other data values considered but rejected for checking the data paths are: to output from -FS to +FS in increments of one binary bit or to output 5252 and 2525. Outputs in steps of one binary bit is the most comprehensive test that can be made since every possible combination of bits is output. However, outputs of every combination require considerably more time and memory than the limited quantity of values proposed above since each value must be transmitted and stored in a memory location. A limited quantity of outputs is adequate to verify that the equipment is functioning properly since it is assumed that the equipment has passed an initial acceptance test verifying that the equipment at one time functioned properly, was wired correctly, etc.

Outputs of 5252 and 2525 exercises all bits since this combination represents alternate bit patterns. A disadvantage of this combination is that full scale and zero offset are verified only by extrapolation. For instance, the loop may be within tolerance with a 5252 value, but the output of an analog unit in the loop may saturate with a full scale input because of power or component conditions before full scale output is reached thereby causing full scale output to be out of tolerance. Also, crosstalk verification is limited since only two values are output. Additionally, because only two values are output, verification that the correct channel is responding is limited since there is a good chance that the correct value will be input even though the incorrect channel is responding. This deficiency may be corrected by outputs to only one channel at a time with all other channels zero or full scale. Outputs to one channel with all others full scale also adequately verifies crosstalk. Another disadvantage is that outputs in this manner are very time consuming because a block of data is output for each value resulting in two blocks per channel.

After the range of data is output to all channels and input thru the input channels, inputs are compared to outputs. If all channels are within tolerance, the system is functioning properly. If any channels are out-of-tolerance, these channel addresses are noted and the program branches to the fault isolation routine.

#### 8.2.3.5 Fault Isolation

The first step in the fault isolation process is to examine the faulty channel addresses to determine if any discrete channels are failed. If there are discrete channel failures, the program is branched to a fault isolation routine. The first step of the discrete fault isolation routine is to determine the quantity of bad channels.

If all the channels are defective and the input data is not changing with different outputs, then the failure is likely the address counter or the control signals in the input/output controller. If all the channels are defective and one or more bits remain constantly a one or zero, then the failure is in the transmission of data through the digital output routing, digital input mux, I/O controller, line drivers or line receivers.

If multiple channels are failed, further tests are performed. All one's are output to the faulty channels in turn with zeros to all other channels. If the data is returned thru the wrong channel, then the address is incorrect. This indicates a failure in the address counter, address decoder, or line drivers or receivers for transmitting the address.

If the channel addresses are correct, the bits are examined in each channel to determine if the same bit is failing in each channel. This type failure indicates a failure in data transmission through the digital output, digital input multiplexer, or the line drivers or receivers transmitting the data.

If one or more channels are failed with no correlation to other channels, then the fault is likely in the digital storage, address decoder or input mux.

After the discrete channels are checked, the failed channel addresses are examined to determine if any analog channels are failed. If there are analog failures, the next step in the fault isolation process is to determine the quantity of faulty channels. This quantity is grouped into three categories: all channels, one channel, or multiple channels bad. If all the channels are faulty, this indicates that the error is either in transmitting digital data to the analog devices or in receiving data from the analog devices. The nature of the failure may further pinpoint the faulty LRU. If one or more bits are consistently dropped, then the fault is in either the input/output controller, the output routing, or the digital input multiplexer; that is, the fault is occurring in the digital word as it is transmitted between the minicomputer and analog devices.

If other inputs occur, such as constant inputs or inputs with no apparent relationship to the outputs, then the control circuitry is likely at fault: either the input/output controller is not transmitting the correct timing signals, the control signals are not received by the digital output routing or input multiplexer, or an address counter is not functioning. An out-of-tolerance condition with no defective bits is likely caused by either the reference or power supplies out of tolerance.

If only one channel has failed, the failure is isolated to a single channel. To further isolate the failure the output may be switched to a different input channel. If the fault does not appear thru this input, then the first input channel is failed. If the fault is still present, the analog output is at fault. For the situation in the test configuration where 100 of the 200 channels cannot be switched to an alternate input, the test hardware must be assumed to be correct. This is a reasonable assumption because for the test hardware to fail, either the multiplexer switch or the switch channel select circuitry would have failed. From the reliability data for these components, only one failure occurs in the group of 100 switches every 20,000 hours. This failure rate is low enough not to warrant redundant input switches. When the choice is between the analog output and multiplexer switch, the failure will be assumed in the analog output path, and for those few instances when these units are checked to be good, then the input switch will be replaced.

### 8.3 CONTROLS AND DISPLAYS

The implementation of self test techniques for the controls and displays incorporated in the Crew Stations and the Instructor Operator Station, IOS, presents a variety of peculiar problems but at least provides a ready solution to one aspect of the problem, that is fault isolation. The controls and displays are a direct interface with the trainees and consequently an undetected performance fault is highly likely to produce a negative training impact on the trainee. This factor increases the desirability of self test features for this equipment. Provision of self test is relatively difficult, however, because both the input command for a control and the output of a display (for example, meter needle motion, indicator light illumination, etc.) are designed for interaction with the human operator. Consequently, self test of a control requires a calibrated driving mechanism that can supply the physical force and motion that are nominally supplied by a crew member. Similarly, self test of a display requires sensing of the motion of physically small components, such as a needle, without interference with their natural behavior. Fortunately, each of the instruments or displays, as well as the controls, is either readily replaceable or must be removed from the installation for servicing in the event of any major malfunction. Effectively, these devices are Least Replaceable Units and if self test facilities can detect the malfunction of any one device, then there is no further fault isolation to be performed.

In the following subsections we describe a variety of means for implementing self test for the various displays and controls, whether they are located in the Crew Station or the Instructor Operator Station. Some of these techniques may be judged too expensive for practical application. This is a judgment that eventually will have to be made for each specific simulator application. Our objective is primarily to conceive and propose these techniques and in so doing, establish some alternatives so that later judgment may be a matter of choice or cost effectiveness analysis rather than a decision by default.

In subsection 8.3.1, we review the basic problems involved in providing self test for displays in general and outline the scope of the displays which we considered for analysis. In the following subsections we then present our conclusions with respect to each of the display devices analysed.

Similarly, in subsection 8.3.8, we review the basic problems inherent in providing self test capability for the various control devices. In the subsequent sections we consider in greater detail the specific requirements of the various types of controls.

### 8.3.1 Displays Overview

A display is a device that transforms the information in an electrical signal into a visually observed effect which can then be viewed by the crew member (or trainee) at his convenience. Although the electrical signal is readily sampled and evaluated by a computer, a truly adequate test of a display must be one that not only verifies that the device is operable but assesses the information being displayed and verifies its accuracy. Consequently, the problem of implementing self test for the display devices is, for the most part, the problem of sensing the response of an electro-mechanical instrument or the luminance or luminance pattern of an illuminated device such as an indicator or cathode ray tube.

The specific displays considered for this study are described briefly in the following paragraph. Subsequent paragraphs address the basic sensing techniques that are available for this application.

#### 8.3.1.1 Display Devices Considered for Analysis

The display devices considered for analysis for this study consisted of the following:

- o Meters incorporating galvanometer movements
- o DC servo driven meters
- o Synchro/resolver driven meters
- o Lighted indicators
- o Bilevel electromechanical flags
- o CRT graphics display devices

The meters display continuously variable parameters whose value is indicated by the extent of displacement motion exercised. There is no signal content in any lumination which may be provided to the device. The lighted indicators and

the electromechanical flags display the status of discrete parameters. The flags have two mechanically possible positions and the lighted indicators have either the lighted or unlighted condition. The CRT presents continuously variable and/or discrete information entirely through programmed luminance patterns.

The available techniques for sampling the information contained in the above displays is discussed in the following paragraph.

#### 8.3.1.2 Basic Display Response Sensing Techniques

The basic techniques for sensing display performance or response may be divided into two categories. These consist of either continuous signal or response measurement devices or discrete position or condition measurement devices. Devices that have only discrete sensing capability, such as magnetic switches, for example, can however, be applied to sensing the response of continuously varying devices.

Discrete response sensing devices or techniques considered for the displays analyses consisted of the following:

- o Photo sensors
- o Bistable fluidic sensing elements
- o Electrical contacts
- o Snap action switches
- o Magnetically operated switches
- o Thermistors

Continuous response sensing devices considered consisted of the following:

- o Position potentiometers
- o Current sensors
- o Closed circuit TV camera

In the following paragraphs, we will review the basic capabilities, advantages and penalties associated with each of the above concepts. This information is intended as background information in support of the evaluations made for each of the displays discussed in the following subsections.

### Photo Sensors

Photo sensors are available with and without their own light sources and, with the proper circuitry, provide a light-on or light-off indication. This is very useful for sensing meter needle positions, since the needle shadow can be used to interrupt the light source and render the light-off condition. If the sensor has its own light emitter, then the light measured is that reflected by an adjacent object. Again, this could be the reflection from the back of a meter needle if properly painted, or it might be the reflection from a spot painted on the back of the tape in a tape meter. Some specifications for typical photo sensors are reproduced in Figure 8.3-1.

Photo sensors have several specific advantages for sensing display instrument position. Primary among these is the fact that there is total non interference with the operation of the equipment being instrumented. Secondly, they are relatively compact and consequently simple to install. In addition, they are relatively inexpensive and installation costs are likely to be a bigger factor than component costs. Finally, they are relatively reliable since they involve no moving parts and, since they operate without any electrical connection to the basic equipment, they are entirely fail safe. That is, failure of a photo sensor in an instrumentation application could in no way interfere with the proper operation of the original equipment.

The major disadvantage of the photo sensor is that it is only useful for checking discrete events. That is, there is no way to obtain a continuous reading of instrument position except by going to a larger number of discrete measuring photo sensors.

### Bistable Fluidic Sensing Elements

Bistable fluidic elements can be applied as discrete sensors similar to photo sensors. In this case, the motion of a needle or a perforation in a tape is detected by whether it covers a control port in a bistable fluidic element. The air flow through the element is diverted from one stable path to another depending on whether the control port is covered or not.

Again the advantage of a fluidic device such as this is relative non interference with the free motion of the device being monitored. Reliability is also a basic virtue of fluidic systems.

OPTICAL/ELECTRICAL CHARACTERISTICS (25°C)

PARAMETER	LIGHT CURRENT		DARK CURRENT	COLLECTOR BREAKDOWN	EMITTER BREAKDOWN	LIGHT CURRENT RISE TIME	SATURATION VOLTAGE	ANGULAR RESPONSE	SPECTRAL RESPONSE	MAXIMUM TEMPERATURE
TEST CONDITION	$V_{CE} = 5.0V$ H as shown		$V_{CE} = 30V$ H = 0	$I_C = 100 \mu A$	$I_E = 100 \mu A$	$R_L = 1000 \Omega$ $V_{CC} = 5V$ $I_L = 1.0 mA$	$I_C = 0.4 ma$ H as shown	NOTE 1	Figure 1	
SYMBOL	$I_L$		$I_D$	$BV_{CEO}$	$BV_{ECO}$	$t_r$	$V_{CE(sat)}$	$\theta$		
UNIT	mA		nA	VOLTS	VOLTS	$\mu sec$	VOLTS	degrees		°C
	MIN	H*	MAX	MIN	MIN	TYP	TYP	TYP		Operating Storage
SD-1440-1†	0.7	5	1000 @ 10V	> 10			0.3	12		-65 to +125
SD-1440-2†	1.5	5	25	50	7	4	0.3	12		
SD-1440-3†	3.0	5	25	50	7	8	0.3	12		
SD-1440-4†	6.0	5	25	50	7	16	0.3	12		
SD-1441-1	.5	20	1000 @ 10V	> 10		3	0.3	30		
SD-1441-2	2.0	20	100	50	7	12	0.3	30		
SD-1441-3	4.0	20	100	50	7	20	0.3	30		
SD-2440-1	0.5	20	1000 @ 10V	> 10			0.3	24		
SD-2440-2	2.0	20	25	50	7	4	0.3	24		
SD-2440-3	4.0	20	25	50	7	8	0.3	24		
SD-2440-4	7.0	20	25	50	7	14	0.3	24		
SD-2441-1	1.0	20	1000 @ 10V	> 10	7		0.3	18		
SD-2441-2	4.0	20	100	50	7	4	0.3	18		
SD-2441-3 (See Note 1)	8.0	20	100	50	7	8	0.3	18		
SD-2441-4 (Note 2)	12.0	20	100	50	7	12	0.3	18		
SD-3440-1	0.3	20	1000 @ 10V	> 10			0.3	45		
SD-3440-2	0.6	20	25	50	7	4	0.3	45		
SD-3440-3	1.2	20	25	50	7	8	0.3	45		
SD-3440-4	2.4	20	25	50	7	16	0.3	45		
SD-3442-1	0.5	1.0	100 TYP	10	5	40	0.15	45		
SD-3442-2 (See Note 3)	1.0	1.0	@	10	5	60	0.15	45		
SD-3442-3 (Note 3)	2.0	1.0	5V	10	5	80	0.15	45		
SD-5440-1	0.8	5	1000 @ 10V	> 10			0.3	3		
SD-5440-2	2.0	5	25	50	7	1.5	0.3	3		
SD-5440-3	4.0	5	25	50	7	3	0.3	3		
SD-5440-4	8.0	5	25	50	7	6	0.3	3		
SD-5442-1	2.0	1.0	100 TYP	10	5	40	0.15	10		
SD-5442-2 (See Note 4)	4.0	1.0	@	10	5	60	0.15	10		
SD-5442-3 (Note 4)	8.0	1.0	5V	10	5	80	0.15	10		

- \* Irradiance in mW/cm<sup>2</sup> from a tungsten source at a color temperature of 2870°K.
- † Offered with extra lead attached to case, Suffix "L" (SD-1440-2L).
- 1. The angle between incidence for peak response and incidence for 50% of peak response.
- 2. Larger aperture for higher sensitivity; reduced angular response minimizes crosstalk.
- 3. Very high sensitivity with wide field of view. Drives TTL directly, even at low irradiance levels.
- 4. Highest sensitivity available. Drives TTL directly, even at low irradiance levels.

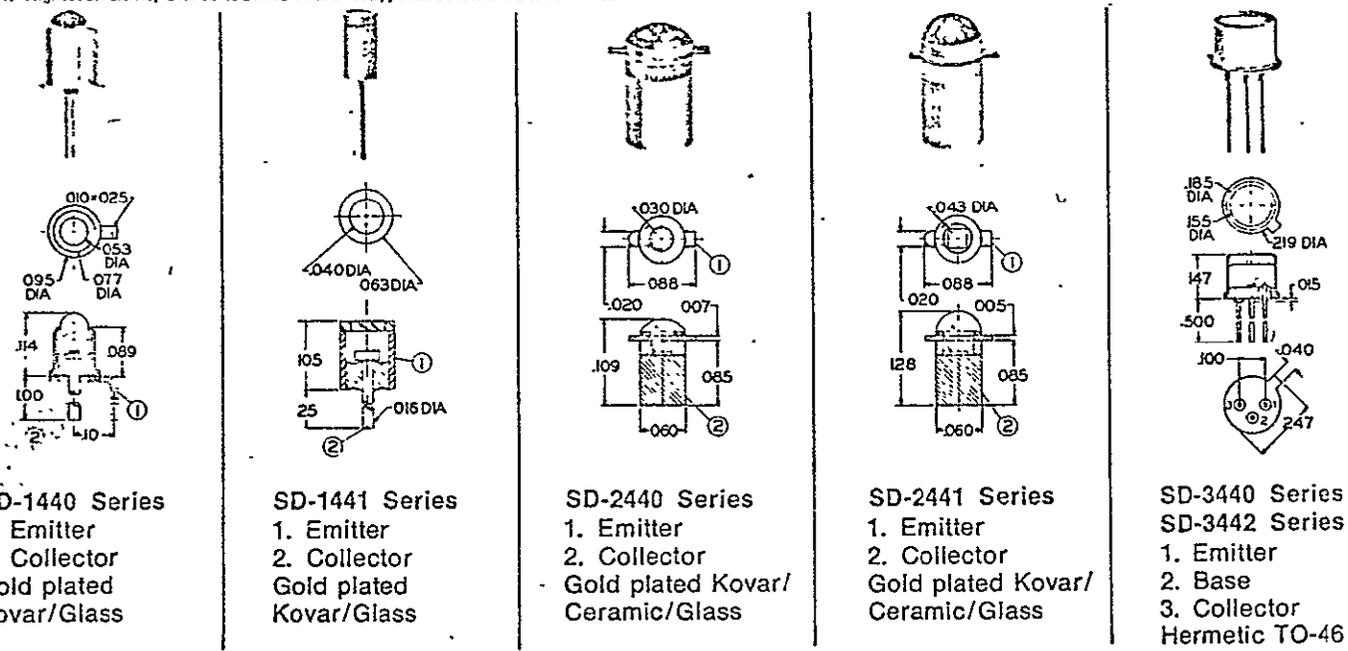
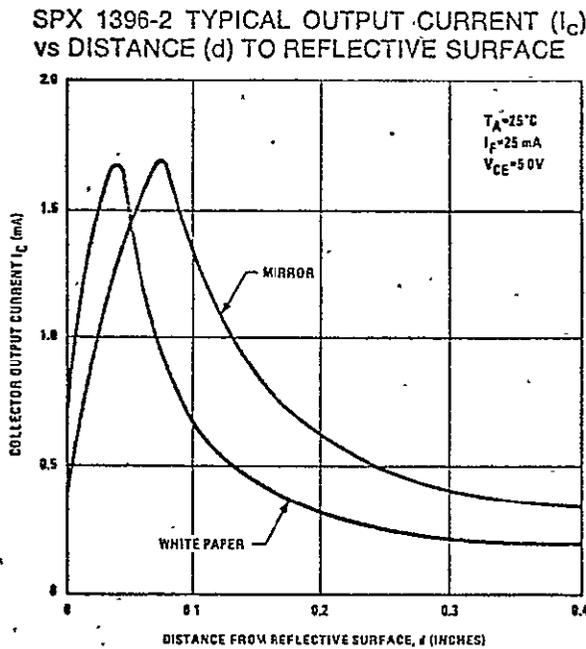


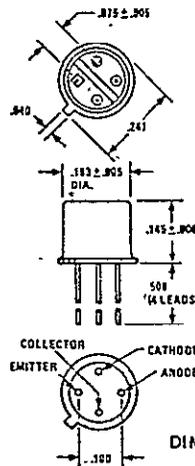
FIGURE 8.3-1 TYPICAL SILICON PHOTOTRANSISTORS AND REFLECTIVE SENSORS (SPECTRONICS INC.)

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

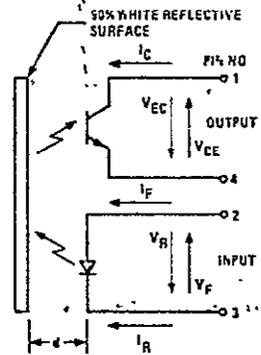
REFLECTIVE SENSOR ASSEMBLY\*



OUTLINE DIMENSIONS



SCHEMATIC



DIMENSIONS IN INCHES

ABSOLUTE MAXIMUM RATINGS (at Standard Atmospheric Conditions unless otherwise specified)

PARAMETER	SPX-1396	UNITS
$I_{FICONT}$	50	mA
$V_{RIL1}$	6	V
$V_F$	1.5	V
$PD_L$	65	mW
$V_{CEO}$	20	V
$V_{ECO}$	5	V
$I_C$	12	mA
$PD_S$	65	mW
$T_{Opr}$	-40 to +70	$^\circ\text{C}$
$T_{Stg}$	-40 to +70	$^\circ\text{C}$

1. Derate linearly from 25 $^\circ\text{C}$  at 1.4 mW/ $^\circ\text{C}$   
 2. Derate linearly from 25 $^\circ\text{C}$  at 0.6 mW/ $^\circ\text{C}$   
 Maximum lead soldering temperature for 10 sec: 260 $^\circ\text{C}$

\* This compact reflective assembly combines Spectronics' solution-grown Light-Emitting Diode and silicon NPN phototransistor in a metal, epoxy-sealed package. Designed to operate at an optimum distance of .050 inches, this assembly will generate a TTL-compatible signal when directed at a reflective surface.

FIGURE 8.3-1 TYPICAL SILICON PHOTOTRANSISTORS AND REFLECTIVE SENSORS (SPECTRONICS INC.) (continued)

Fluidic devices introduce several disadvantages, however. The information to position is contained in the direction of flow exiting the fluidic element. In order to introduce this information into the computer for processing, the signal must be converted to an electrical signal through a transducer. In addition, the use of fluidic sensors requires the provision of an air supply and the associated tubing for its distribution and return. For this application, these requirements render the fluidic system likely to be more costly, complex and cumbersome than some of the other alternatives.

### Electrical Contacts

The installation of contacts for sensing the position of a moving indicator at its extreme positions is a simple and direct way of obtaining discrete information. Reliability is at a maximum because there is, literally, no component to fail except for the contacts themselves.

The strongest advantages of the use of electrical contacts are the low cost for components, the high reliability and the accuracy of the indications.

The disadvantages of using contacts are that they restrict the free motion of the moving indicator either by completely stopping it at the extreme scale positions or by imposing a relatively high friction load. The restriction of the motion has the pursuant effect that loss of calibration on the one side cannot be detected. In addition, all of the moving indicator or some part of it must be conductive and the conductive part must be suitably insulated from the instrument case to avoid shorts. Also, conductivity of the contacts must be assured by the use of special metal plating or by periodic maintainance.

### Snap Action Switches

Snap action switches are inexpensive, compact and reliable.

Snap action switches don't offer any particular advantage for the present applications other than those cited above.

Disadvantages for snap action switches are that they restrict the free motion of the moving indicator and they do this less predictably than do the contact points. In addition, they apply this restriction before the moving indicator

reaches its scale limits unless there is a substantial amount of off scale motion available. Switches are more expensive and require more room than the contacts previously discussed.

#### Magnetically Operated Switches

Magnetically operated switches that are small and bounce-free are available from Microswitch Division of Honeywell as well as other sources. These switches require the article to be sensed to be magnetic or to have attached to it some magnetic material. Magnetic sensing reed relays must then be installed at the points at which the passing of the magnetic material must be detected. There is no direct physical contact between the moving element and the sensors.

Magnetic switches have the advantage that they do not interfere in any way with the existing limitations on the displacement motion of the moving indicator. Nor do they require any modifications to an instrument that would be conspicuous and degrade its fidelity as a simulation. Further advantages are they they do not require a light source or light reflection which may be difficult to provide in some instances and they do not interfere with the absolute physical motion limits of the device being sensed.

The disadvantages are that the magnetic material which is required on the moving component may change its dynamic response properties by substantially affecting its weight. The magnetic fields created may also interfere with the proper operation of the instrument.

#### Thermistors

Thermistors are simply resistors whose resistance is a monotonic function of temperature. When properly calibrated, thermistors are used as temperature indicators. They are inexpensive and can be used to detect whether a heat generating light source is on by virtue of the associated heat output.

The advantages of the thermistor are low price and simplicity. It is simply a passive material with a unique property.

The disadvantages of thermistors are the requirement for bridge circuitry that is normally used with them to measure temperature changes. Secondly, they

are limited to detecting heat which is a relatively minor application for the present study. Proximity requirements for proper heat detection could affect appearance of the display.

### Position Potentiometers

Position Potentiometers are simply potentiometers in a voltage divider circuit that produce a voltage that is proportional to positional displacement whether the displacement is angular or linear.

The advantages of position potentiometers are reliability, low cost, accuracy and availability. In fact, they are quite often used for generating feedback signals in servo driven systems. In this case, the voltage from the existing potentiometer may simply be sampled through a buffer amplifier.

The disadvantages of a position potentiometer are that if it is not already present as part of a servo loop, then it may impose a friction load on the device drive that alters its dynamic performance if not accuracy. Space limitations may also be a problem.

### Current Sensors

Current sensors detect very low current levels using a circuit such as shown in Figure 8.3-2. The sensing of very small currents is a corollary of operation without loading the circuit being measured.

The advantages of current sensing include greater reliability than optical or light sensing concepts. The result is directly available as a voltage for computer processing and, of course, the signal is continuous and can be used to measure a graduated or continuously variable parameter.

The disadvantage of using a current sensor for display instrumentation is that it doesn't really measure the final observable phenomena, whether it is the illumination of an indicator light or the displacement of a needle. It can only sense the applied current.

### Closed Circuit TV Systems

A black and white closed circuit television system can be used to monitor various types of display manifestations or observable events. The nature of a

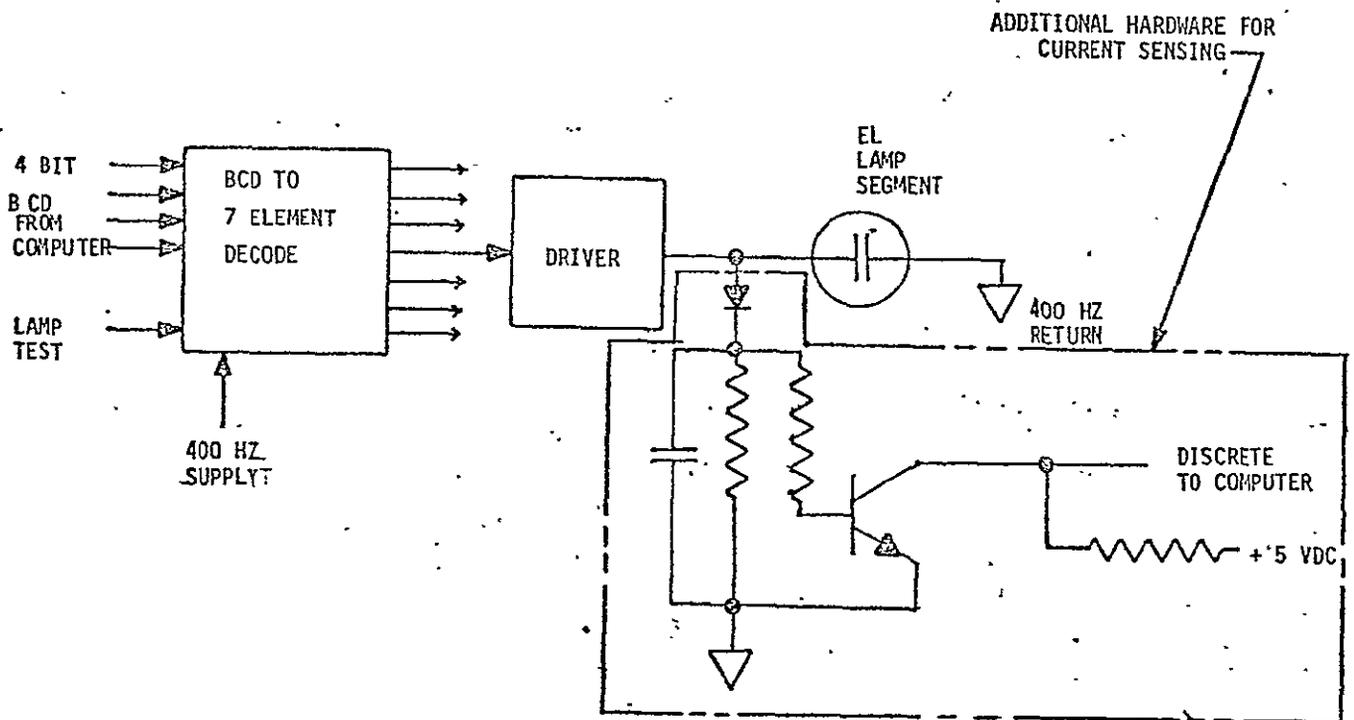


FIGURE 8.3-2 EXAMPLE OF A CURRENT SENSING CIRCUIT

horizontal scan line and its reflection of various image properties is discussed in subsection 8.4.2. If the direction in which the camera is pointed and its optical field of view are established accurately, then specific or selected lines in the image field can be monitored to detect display phenomena without requiring the excessively complex processing associated with pattern recognition techniques. This can be accomplished by digitizing a single line or even several lines in a manner similar to that described in Section 8.4.2. The identification of significant data in the scans then requires processing of only a one dimensional array of data for each scan. In addition, the area of the data array in which the information is expected to appear is fairly accurately predictable.

The advantages of a TV scan approach to display monitoring are substantial. The TV system can monitor either continuous phenomena or discrettes such as indicator lights on or off. The TV approach requires virtually no modification of the instruments in the crew station. This has the subsidiary advantage of permitting implementation of the self test system after the crew station or IOS is completely fabricated.

The disadvantages and problems created by the application of closed circuit TV are equally substantial. For one line of the TV image which might be processed for each instrument or for an array row of indicator lights, the system described in Section 8.4 produces 512 samples. In order to achieve a 1/60th second time resolution, alternate lines must be sampled and processed, assuming commercial scan rates and an interlaced scan pattern. This approach quickly produces a large amount of data for processing if a dynamic test is considered. If a camera is used to scan one simple meter, the processing may be minimized. However, this makes the cost of the instrumentation exorbitant. In order to scan more than one instrument, the camera field of view can be expanded, reducing the available resolution for monitoring each instrument as well as the signal strength of the actions being observed. Processing of multiple lines is required if a matrix of instruments is considered. However, the instruments on one horizontal scan line at a time could be tested and processed before proceeding to another row. Consideration of the use of a TV system rapidly expands into a design problem that is highly sensitive to the particular instrumentation being monitored and its configuration.

### Summary

The above review describes most of the sensing concepts that were considered for instrumenting one or more of the display devices of interest. The trade offs between these approaches and a few others that are uniquely applicable to specific displays are reviewed in the analyses of those specific displays starting in subsection 8.3.2 .

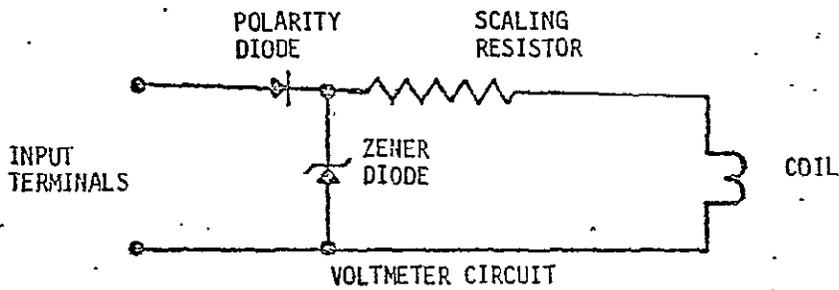
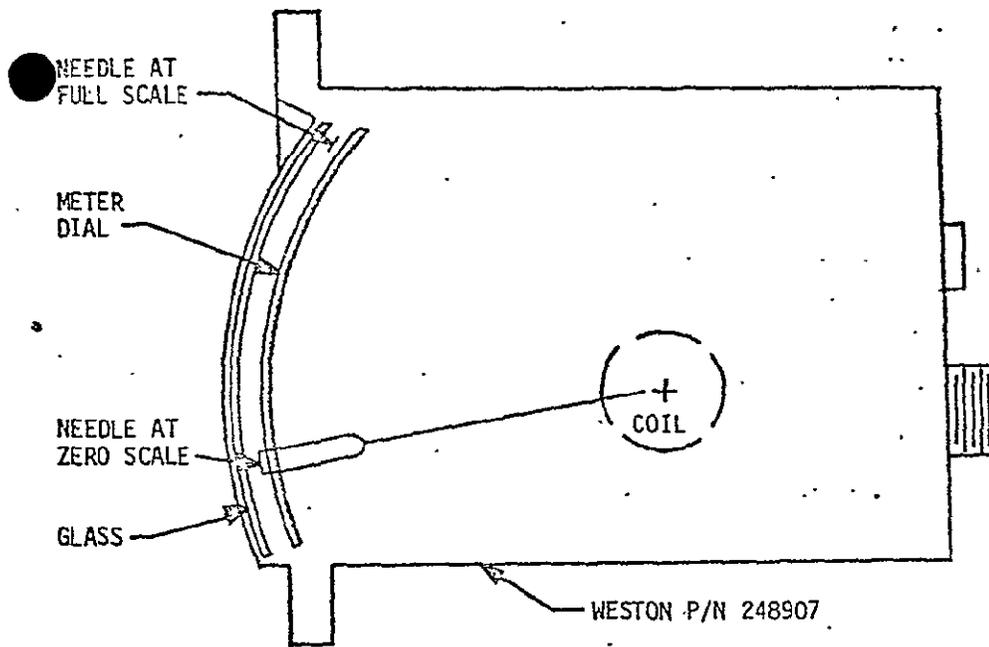
### 8.3.2 Galvanometers

There are approximately 70 galvanometers in the crew station and 35 in the IOS. Various techniques for checking the galvanometers have been examined, and one specific approach is recommended.

#### 8.3.2.1 Description and List of Displays Applicable

The meter has one or two galvanometer movements contained within a round, horizontal or vertical rectangular case. The dial face contains a scale that is marked in the units and range that are appropriate to the parameter that is being displayed on the meter; i.e., 0 to 5000 pounds of oxidizer. If a standard meter does not contain the desired scale, the simulator manufacturer may make his own scale on a decal and apply it to the meter face. Other alterations that may be performed by the simulator manufacturer is the addition of overload protection with a zener diode, or protection from voltage reversals by the addition of a polarity diode. These components seldom fail. The range of a meter may be changed by the substitution of shunt or series resistors. See Figure 8.3-3 for a typical case design, electrical circuit and equation of motion for the 70 meter movements.

In normal operation all of the parameters affecting motion, except the applied voltage, are constants to the simulator engineer. The applied voltage (instrument input) will vary according to the variation in the parameter that is being displayed; i.e., fuel quantity. The needle movement (instrument output) varies according to the ratio of applied voltage to full scale voltage. See Table 8.3-1.



Equation of motion

$$P \frac{d^2\theta}{dt^2} + (K + \frac{G^2}{R}) \frac{d\theta}{dt} + U\theta = \frac{GE}{R}$$

- P = moment of Inertia
- K = mechanical damping coefficient
- U = restoring torque constant
- G = the motor constant
- E = the applied voltage
- R = total circuit resistance

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

FIGURE 8.3-3 TYPICAL GALVANOMETER CASE, CIRCUIT AND EQUATION OF MOTION

TABLE 8.3-1 ANALOG METERS

INPUT: 0 to +3 volts ranging up to +100 volts DC

OUTPUT: needle movement, zero scale to full scale

ACCURACY: +2% full scale in crew station  
+3% full scale in IOS

RELIABILITY: 20,000 hr MTBF

REFERENCE CONFIGURATION QUANTITY: 58 meter movements

EXPECTED FAILURE MODES: short circuit  
 open circuit  
 no change, or non-linear change in output  
 due to evaporated lubricant  
 electrical overload may bend needle or  
 break a moveable part

CRITICAL MEASUREMENT POINTS: input terminals  
 needle position

TABLE 8.3-2 FAILURE MODES

MODE	SYMPTOMS	
	EXTERNAL	INTERNAL
OPEN CIRCUIT	NO NEEDLE MOVEMENT	NO CONTINUITY
SHORT CIRCUIT	NO NEEDLE MOVEMENT	LOWER THAN USUAL RESISTANCE
MECHANICAL FAILURE	NO NEEDLE MOVEMENT OR NON-LINEAR NEEDLE MOVEMENT	DRIED LUBRICANT OR PHYSICAL DAMAGE
MECHANICAL CALIBRATION	NEEDLE DOES NOT REST AT ZERO WITH ZERO VOLTS	ADJUSTMENT SCREW OR LEVER MOVED
POLARITY DIODE FAILURE	NO NEEDLE MOVEMENT	NO CONTINUITY
	CHANGE IN ELECTRICAL CALIBRATION	CONTINUITY BOTH DIRECTIONS
ZENER DIODE FAILURE	NO NEEDLE MOVEMENT	ZERO RESISTANCE ACROSS ZENER
	NONE	NO CONTINUITY ACROSS ZENER

These principles of operation apply to round, horizontal or vertical faced galvanometers. It is assumed that the correct test voltage will be applied to the terminals of each meter. This implies that the circuitry between the computer and meters have been checked out and calibrated.

Meters in the flight vehicle are usually illuminated internally, but only mission simulator meters are usually so illuminated.

#### 8.3.2.2 Failure Modes Analysis and Characteristic Parameters

Table 8.3-2 lists the failure modes and the related symptoms. Potential failures are open circuit, short circuit, mechanical failure, mechanical calibration, polarity diode or zener diode failure.

It can be concluded from Table 8.3-2 that, in addition to total inoperability, galvanometer faults can reveal themselves as calibration errors and as erratic movements of the meter. Detection of the possible faults using only discrete sensors at full scale and zero scale positions can be accomplished by assessment of the following characteristic parameters assuming an underdamped meter movement and full scale step input function:

- Time to first crossing of 100% value (rise time)
- Time to second crossing of 100% value
- Time to steady state reading (settling time)

If the meter movement is overdamped, the first two parameters must be replaced by a rise time which is typically the time to reach some percent of full scale.

#### 8.3.2.3 Hardware Requirements for Galvanometer Self Test

The alternative means for sensing the galvanometers response to test stimuli are summarized in Table 8.3-3. The use of photoelectric sensors is recommended on the basis of ease of installation, non-interference with normal meter operation, and cost. The choice between simply passive photo transistors and self stimulating reflective devices, as summarized in Figure 8.3-1, can and must be made during detail design.

CANDIDATE TECHNIQUE	RELATIVE COST	HARDWARE MODIFICATIONS		SOFTWARE REQUIREMENTS	REMARKS
		MODS TO INSTRUMENT	ADDITIONAL MODS EQUIPMENT		
PHOTOELECTRIC SENSORS FOR FULL SCALE AND ZERO SCALE	MOD	INSTALL 2 PHOTOSENSORS ON REINFORCED SCALE	LIGHT SOURCE FOR ILLUMINATION 2 DISCRETE INPUTS/METER POWER SUPPLY(?)	PROCESSING OF DISCRETE DATA	NO INTERFERENCE WITH NEEDLE SELECTED TECHNIQUE
NEEDLE CONTACT FOR CONTINUITY		INSTALL 2 CONTACTS	2 DISCRETE INPUTS/METER		CALIBRATION INTERFERENCE. OVER-RANGE NOT DETECTED.
FLUIDICS SENSING OF NEEDLE POSITION	MOD HIGH	VENT CASE. INSTALL 2 FLUIDIC SENSORS.	ADD AIR SUPPLY SYSTEM ADD TRANSDUCERS 2 DISCRETE INPUTS/METER	TEST CIRCUIT CHECK	EXTRA HARDWARE FOR PNEUMATICS, AND EXTRA MAINTENANCE OF PNEUMATIC SYSTEM
MAGNETIC SENSORS FOR FULL SCALE AND ZERO SCALE	MOD HIGH	INSTALL 2 MAGNETIC REED RELAYS; 2 MAGNETS OR A MAGNETIC NEEDLE.	2 DISCRETE INPUTS/METER		EXACT POSITION LESS CERTAIN. EXTRA WEIGHT ON NEEDLE REQUIRES COUNTER-BALANCING, AND STRONGER COIL
CURRENT SENSORS	HIGH	INSTALL AMPLIFIER INSIDE CASE	POWER SUPPLY(?) 1 ANALOG CHANNEL/METER	PROCESSING AND EVALUATION OF QUANTITATIVE PARAMETERS	INTERFERENCE DUE TO RADIATED AND CONDUCTED RFI. SIGNAL TO BE SENSED IS VERY SMALL. VARIATION IN THE SIGNAL IS STILL SMALLER. NEEDLE MOVEMENT (OUTPUT) IS UNCERTAIN.
TV SCAN	VERY HIGH	POSSIBLY PAINT NEEDLE	ILLUMINATION SOURCE TV CAMERA(S) CONTROL CIRCUITRY VIDEO PROCESSING HARDWARE	COMPLEX PROCESSING OF IMAGE DATA ARRAYS	HIGH COST. UNCERTAIN THAT NEEDLE DYNAMIC MOVEMENT CAN BE ADEQUATELY DETECTED WITH CONVENTIONAL SCAN RATES. LARGE DATA PROCESSING AND HANDLING REQUIREMENT

TABLE 8.3-3 CHARACTERISTICS OF CANDIDATE TECHNIQUES

The special hardware requirements for this test are two photosensors and associated circuits per meter movement, a power supply (if different from other simulator hardware), and two bilevel circuits per meter movement. A typical circuit for a photosensor is given in Figure 8.3-4.

To detect needle position, a sensor is placed at the location to be checked. When the needle is at the correct position, the sensor circuitry will produce a discrete voltage level that will be transmitted via a bilevel line to the computer interface. It is desirable to detect the needle position at two points. These two points should be at the extreme ends of the scale so that it can be verified that the needle has the required freedom of movement, and also to verify that the mechanical zero and full scale calibration is correct.

The full scale sensor should give an ON signal to the computer when, and only when, the full scale analog signal is applied. An OFF signal would indicate (1) mechanical breakage or failure, (2) meter out of mechanical calibration, (3) test circuit failure, or (4) electrical failure.

The zero scale sensor should give an ON signal to the computer when, and only when, the zero scale analog signal is applied, or if no signal is applied. An OFF signal would indicate (1) meter circuit failed ON or PARTLY ON, (2) meter out of mechanical calibration, (3) test circuit failure, or (4) mechanical failure.

Figure 8.3-4 shows a typical sensor installation at both extremes of a meter. In this case phototransistors are used as the sensors. The circuit shown indicates the component arrangement to generate the discrete signal required. This arrangement requires one bilevel line for each sensor. Otherwise, test circuit failures can give incorrect answers to the computer logic system. This design allows the software to determine whether there has been a failure in the test circuit, or in the meter circuit. With a suitable software program in the simulator computer, the meter, plus the test circuits, can be verified very quickly.

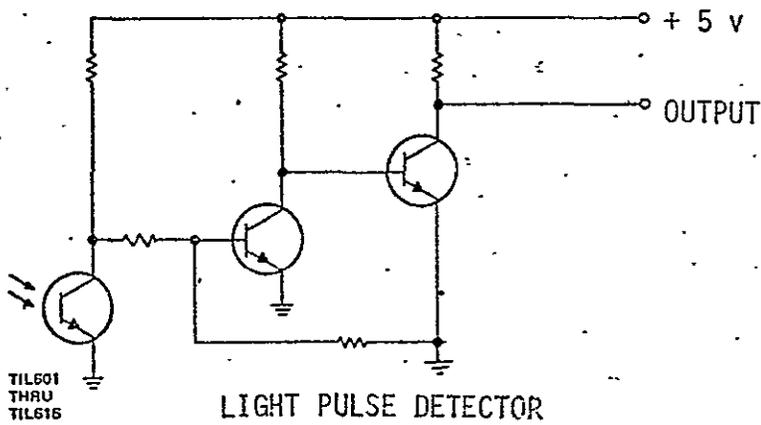
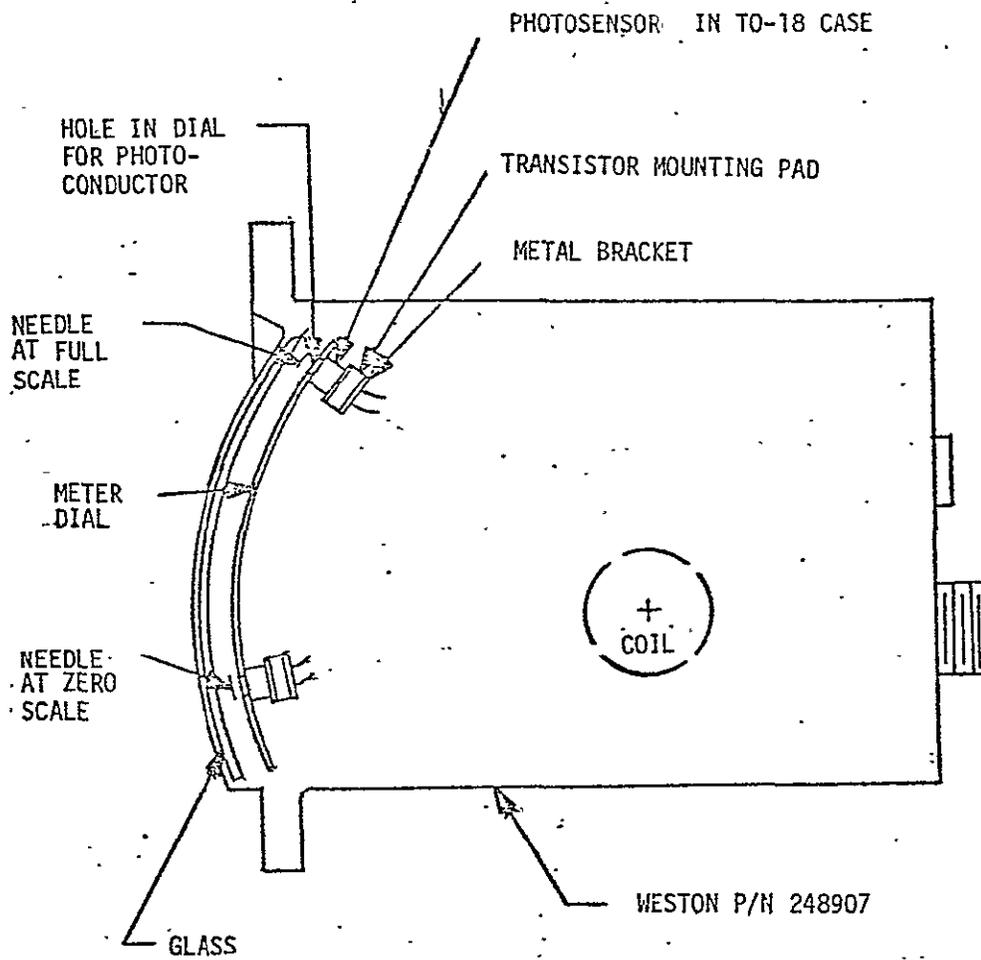


FIGURE 8.3-4 SAMPLE PHOTOCELL LAYOUT AND CIRCUIT

Signal line requirements for a simulator with 70 meter movements, using a two position measurement, would require an additional 140 bilevel input lines to the computer. Switching of bilevel lines that are normally dedicated to the simulation will be considered but the cost of dedicated test circuit discrete inputs will be weighed against the cost and reliability of the switching network.

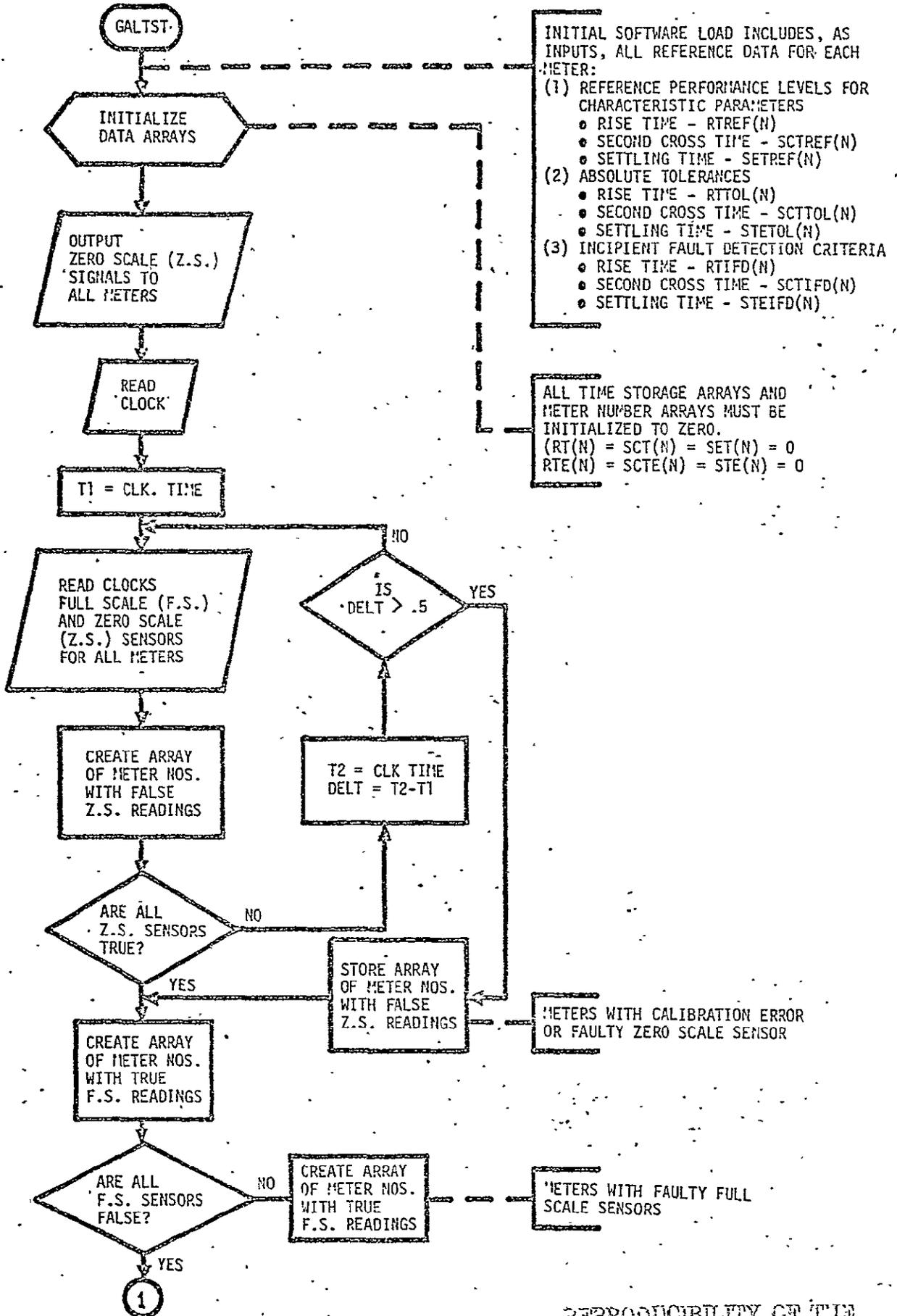
#### 8.3.2.4 Software Requirements and Data Base Impact

A software flow for checkout of an array of galvanometers is shown in Figure 8.3-5. This software flow assumes that the complete array of meters can be tested with the loops for sampling the meter sensors having a short enough time to provide adequate resolution time wise. If the software is executed in a slow computer or if other considerations prevent this possibility, then the full complement of meters will need to be subdivided into smaller arrays. This does not impact the basic concepts shown in the software flow.

The time reference for the data is obtained from an external clock which should be sampled every time the data on sensors is sampled. This assures a positive and accurate time reference for each sampling. There is no requirement for this particular test for the data samples to be at equal time intervals. Consequently, there is no provision for monitoring the clock for this purpose.

The input data which must be loaded from the data base consists entirely of characteristic time information. These arrays may be summarized as follows:

- Rise time
- Rise time tolerance
- Rise time marginal performance levels
- Second crossing time
- Second crossing time tolerance
- Second crossing marginal performance levels
- Settling time
- Settling time tolerance
- Settling time marginal performance levels



INITIAL SOFTWARE LOAD INCLUDES, AS INPUTS, ALL REFERENCE DATA FOR EACH METER:

- (1) REFERENCE PERFORMANCE LEVELS FOR CHARACTERISTIC PARAMETERS
  - RISE TIME - RTREF(N)
  - SECOND CROSS TIME - SCTREF(N)
  - SETTLING TIME - SETREF(N)
- (2) ABSOLUTE TOLERANCES
  - RISE TIME - RTTOL(N)
  - SECOND CROSS TIME - SCTTOL(N)
  - SETTLING TIME - STETOL(N)
- (3) INCIPIENT FAULT DETECTION CRITERIA
  - RISE TIME - RTIFD(N)
  - SECOND CROSS TIME - SCTIFD(N)
  - SETTLING TIME - STEIFD(N)

ALL TIME STORAGE ARRAYS AND METER NUMBER ARRAYS MUST BE INITIALIZED TO ZERO.  
 (RT(N) = SCT(N) = SET(N) = 0  
 RTE(N) = SCTE(N) = STE(N) = 0

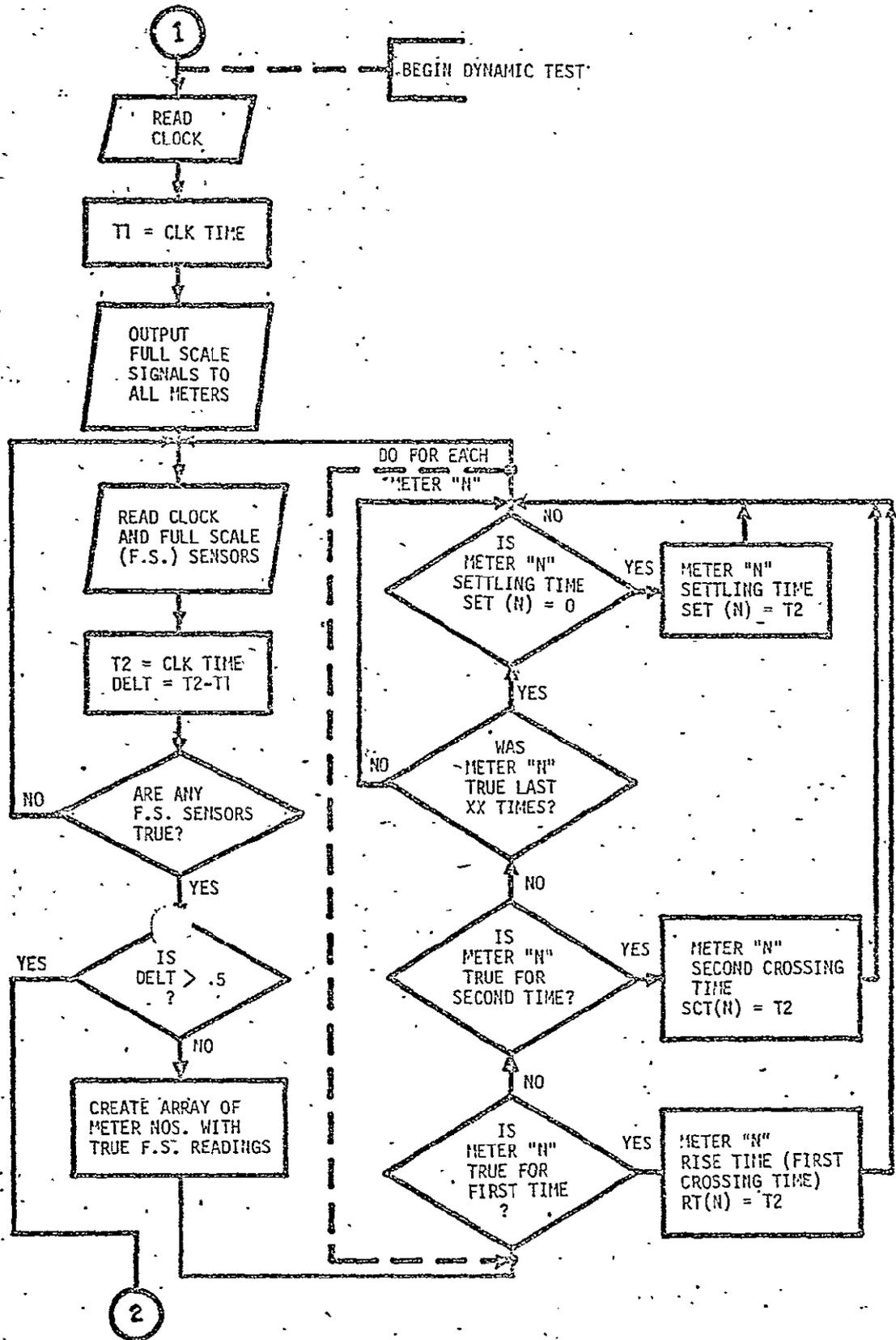
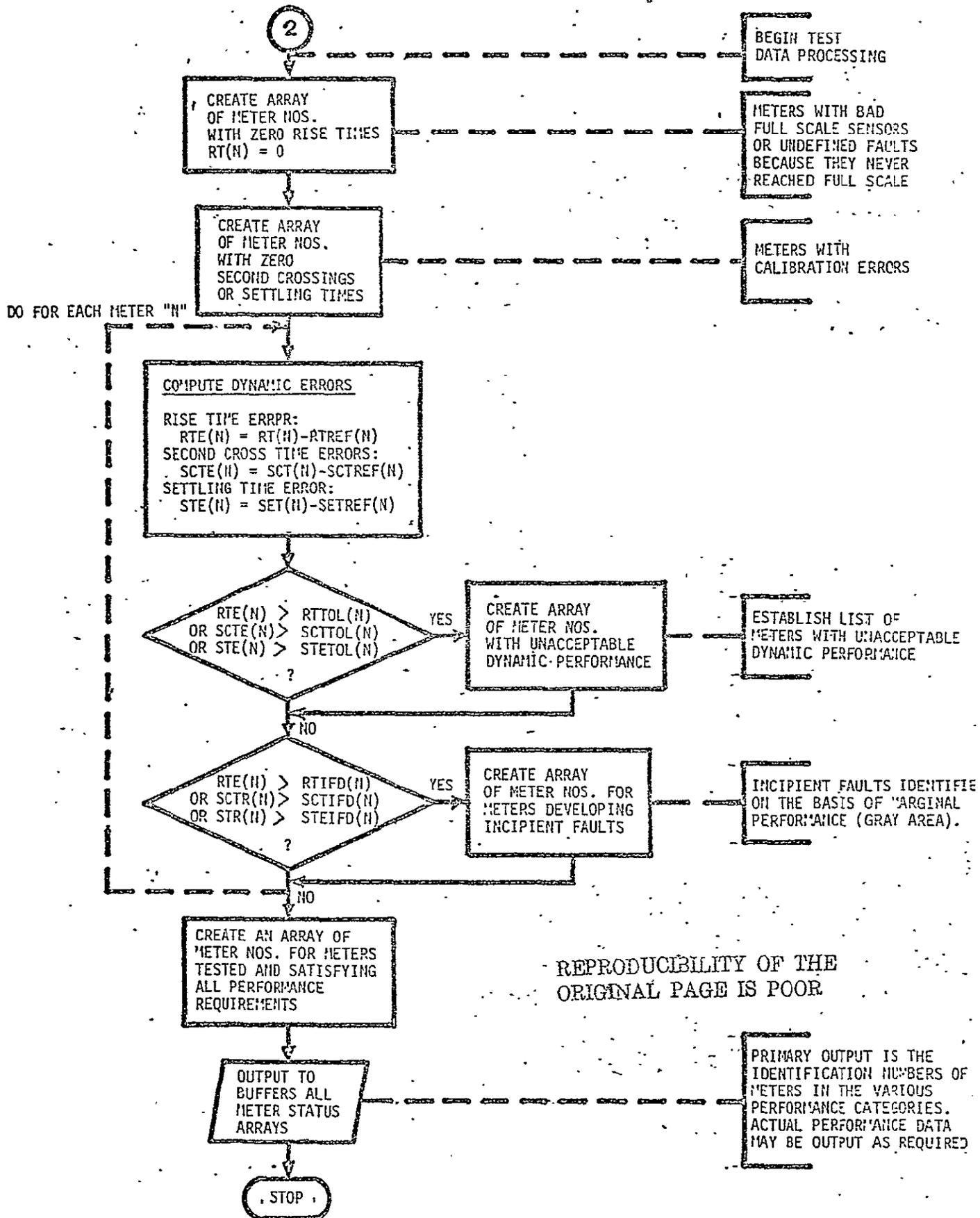


FIGURE 8.3-5 GALVANOMETER TEST SOFTWARE (continued)



The galvanometers in the IOS are likely to be duplicates of the meters in the crew station and would not require unique reference data. Likewise, many of the meters in the crew station are likely to have similar movements and not require unique data. Overlooking the latter aspect because convenience or configuration management aspects may make it desirable to have complete reference arrays, the storage of the above parameters would require 630 floating point words for a total of 70 galvanometers.

The number of instructions required for the test software can be expected to be fairly small; 300 instructions is probably conservative.

The results or outputs of the test are the status of the various meters by meter number, that is, the meter numbers of meters with undetermined faults; of meters with calibration errors; of meters with failed sensors, etc. In addition, the rise times or other parameters which may be of potential interest for follow-up action can also be displayed or printed as desired.

### 8.3.3 DC Servo Driven Meters

This section describes various techniques of checking tape meters and "normal" meters that have DC analog servo drives. There are approximately 36 meter movements that use a servo drive technique in the crew station. One technique is recommended as the best choice for accuracy, simplicity and flexibility.

#### 8.3.3.1 Description and List of Displays Applicable

The servo meter movement contains a DC analog servo system that moves a needle or pointer through an arc of 120 to 360 degrees full scale. The dial face contains a scale that is marked in the units and range that are appropriate to the parameter that is being displayed on the meter; e. g., Gs of acceleration. See Figures 8.3-6 and 8.3-7 for a typical case design and electrical circuit. Additional description is given in Table 8.3-4. The specific meters that are expected to have DC servo drives are the Attitude Director Indicator (error and rate needles) and the Normal/Longitudinal Accelerometer Indicator.

In normal operation, the DC signal from the computer goes through a summing amplifier together with a feedback signal from the position potentiometer. This summed signal goes to a torque motor which drives the needle and position potentiometer. See Figure 8.3-8. Some simulator instruments convert 26VAC, 400 hertz to supply DC power to the motor and regulated +10 VDC and -10 VDC to the position potentiometer and amplifier.

Tape meter movements contain a DC analog servo system that moves a tape past a window and a pointer. See Figure 8.3-9. The tape contains an extra long scale that is too long to be placed on the face of a servo meter of the same package size. Another version of the meter uses the tape as the pointer (with contrasting color areas) for display against a fixed scale. Additional description is given in Table 8.3-5.

In normal operation, the DC signal from the computer goes through a summing amplifier together with a feedback signal from the position potentiometer. This summed signal goes through a second summing amplifier together with a feedback

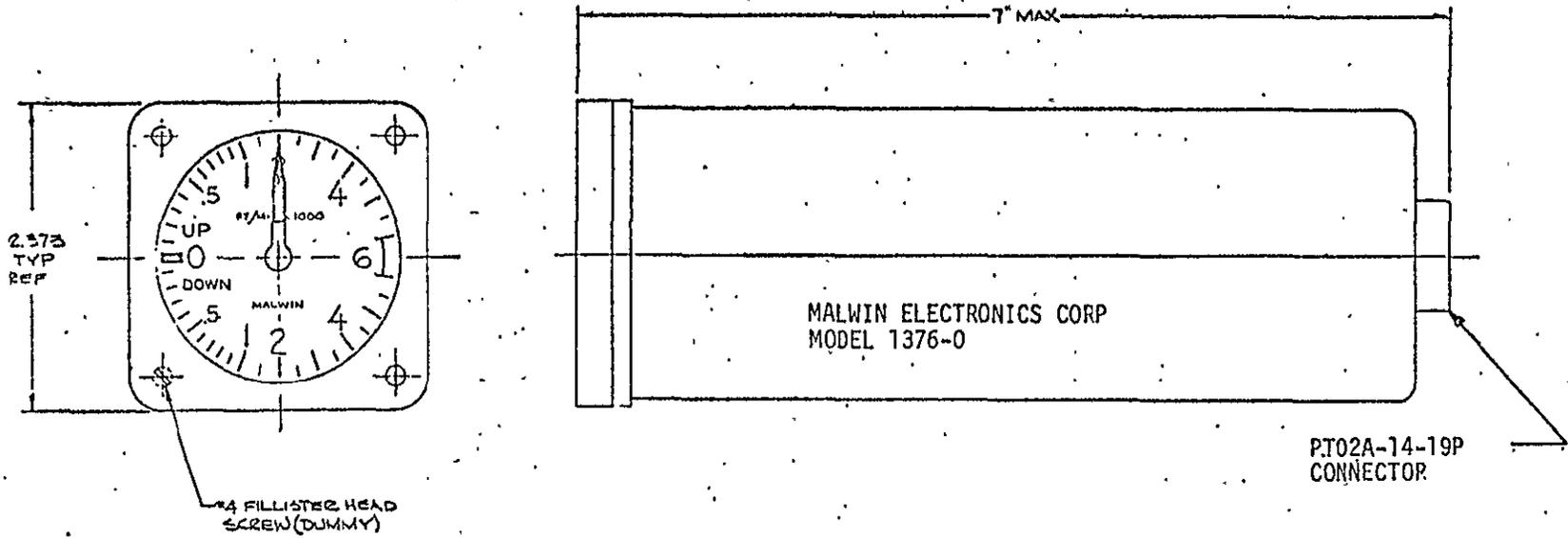


FIGURE 8.3-6 TYPICAL SERVO DRIVEN INSTRUMENT

8-102

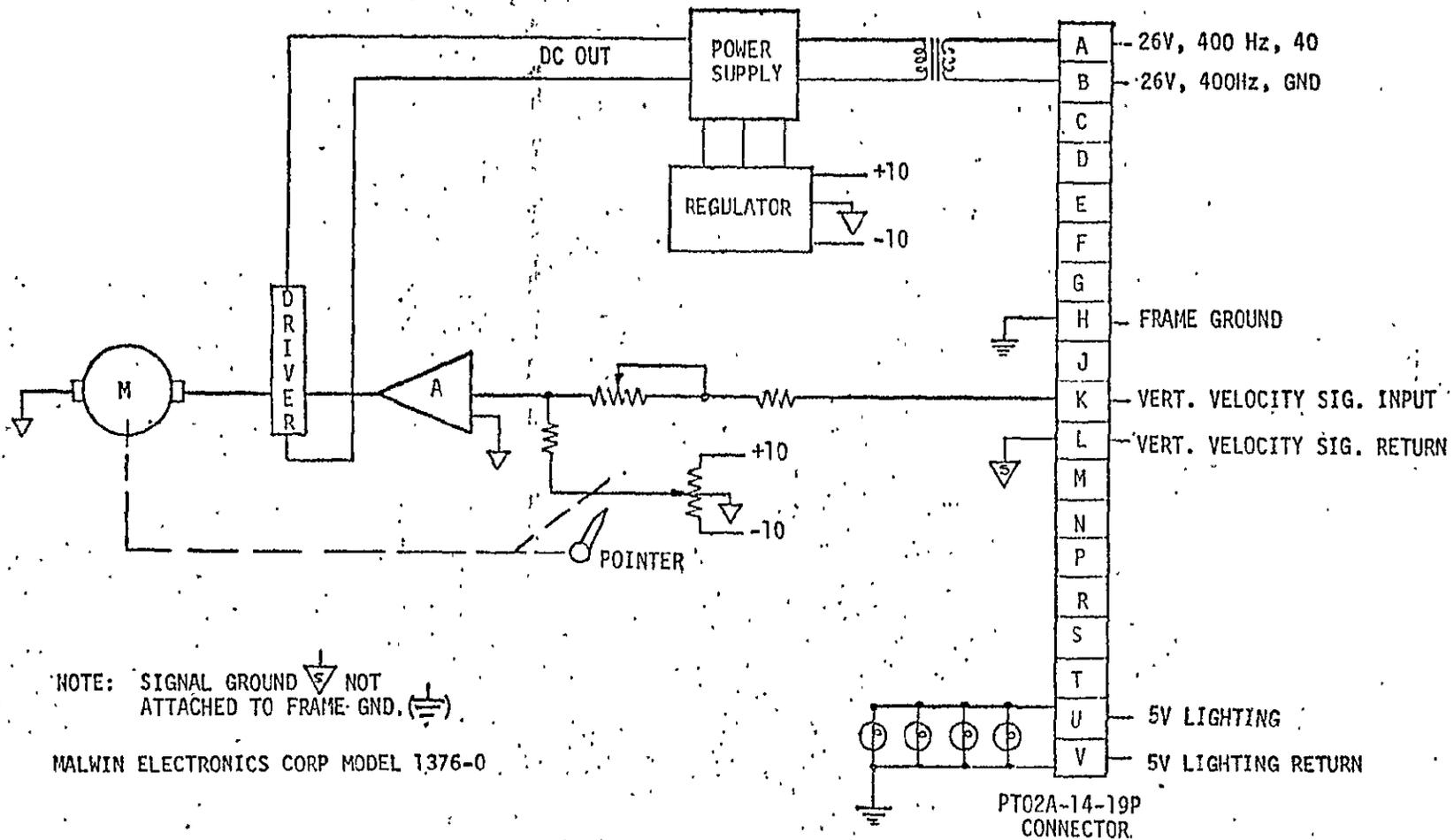


FIGURE 8.3-7 TYPICAL INSTRUMENT ELECTRICAL DIAGRAM

TABLE 8.3-4 DC SERVO DRIVEN METERS

INPUT: 26 VAC, 400 HZ (power)  
-10 to +10 volts DC (signal)

OUTPUT: Needle movement, zero scale to full scale

ACCURACY:  $\pm 2\%$  full scale

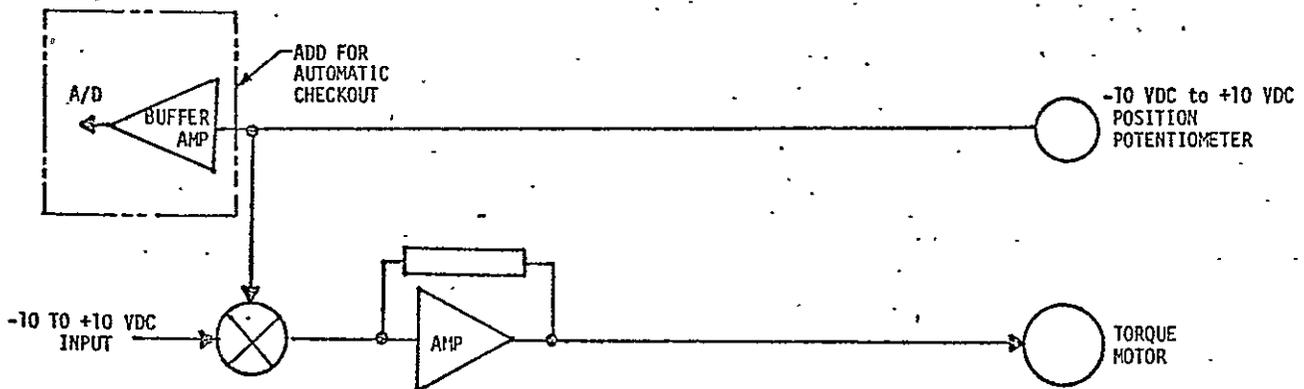
RELIABILITY: 7,500 hr MTBF

REFERENCE CONFIGURATION QUANTITY: 16 meter movements

EXPECTED FAILURE MODES: short circuit in internal connections  
open circuit in internal connections  
no movement, or non-linear movement due to evaporated  
lubricant, or degraded amplifier  
electrical overload may bend or break mechanical part(s)  
failure of internal electrical or mechanical components

CRITICAL MEASUREMENT POINTS: input terminals  
needle position  
position feedback signal

FIGURE 8.3-8 DC SERVO DRIVEN METER



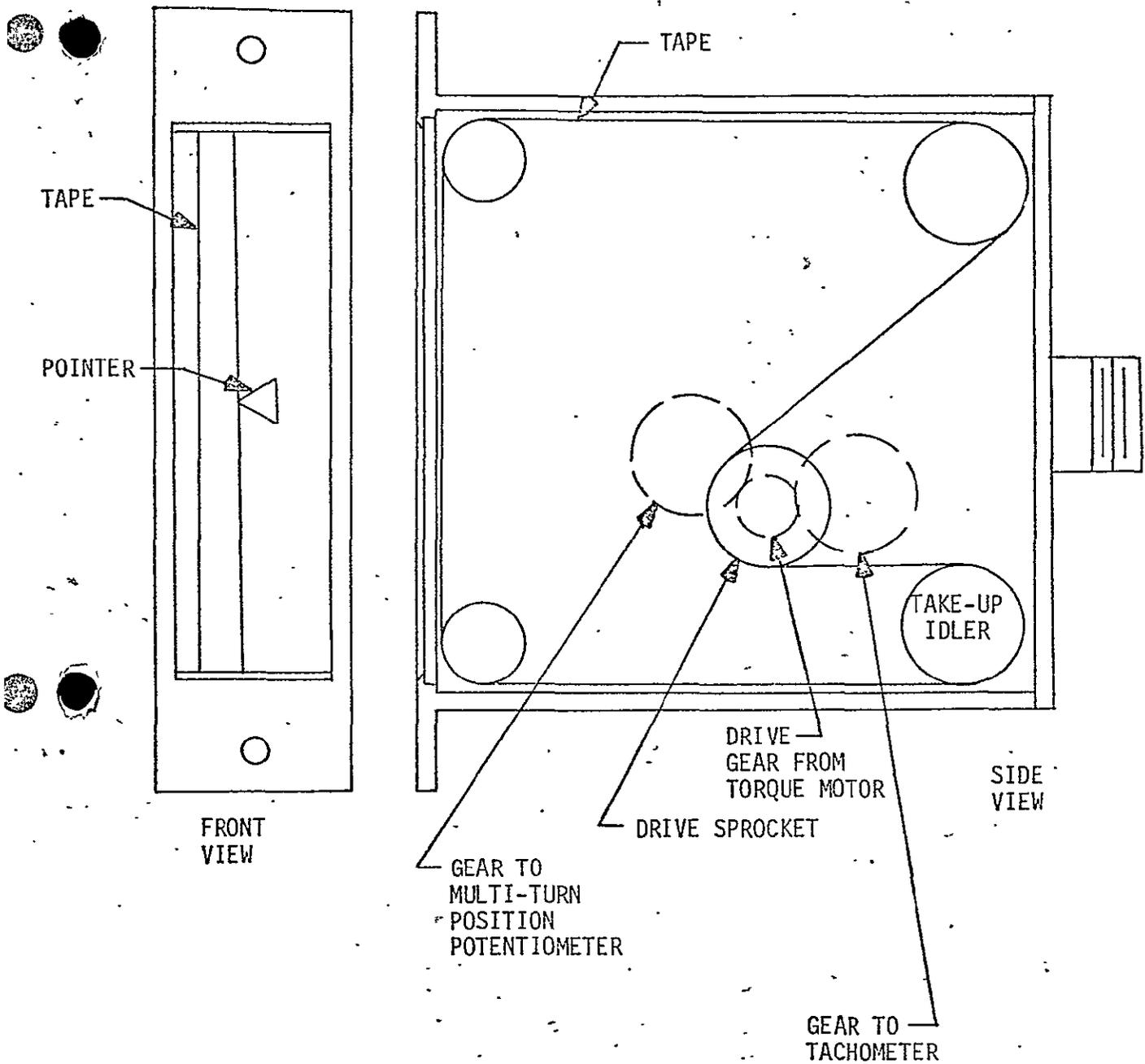


FIGURE 8.3-9 TAPE METER CONSTRUCTION

TABLE 8.3-5 TAPE METERS

INPUT: 0 to +10 volts DC

OUTPUT: Tape movement, zero scale to full scale

ACCURACY:  $\pm 2\%$  full scale

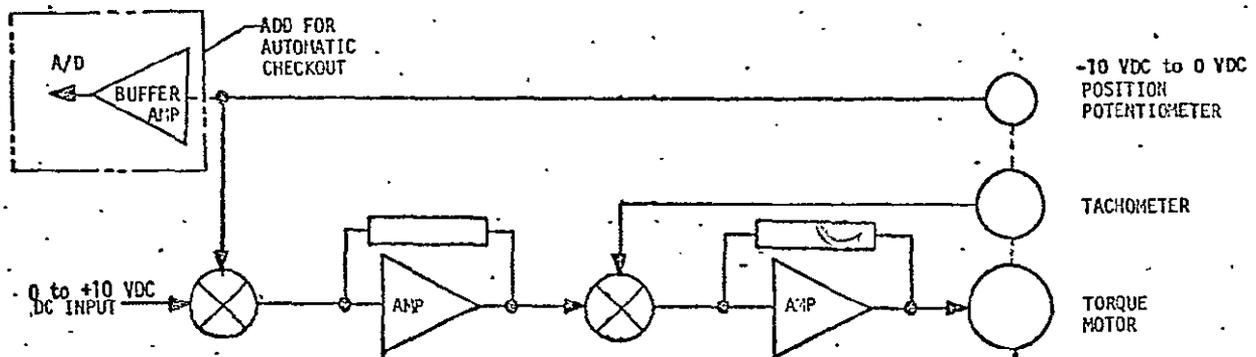
RELIABILITY: 3,000 hr MTBF

REFERENCE CONFIGURATION QUANTITY: 20 meter movements

EXPECTED FAILURE MODES: short circuit  
open circuit  
no change, or non-linear change in output due to  
evaporated lubricant, or degraded amplifier  
electrical overload may bend or break mechanical part(s)

CRITICAL MEASUREMENT POINTS: input terminals  
tape position/position feedback signal

FIGURE 8.3-10 TAPE METER ANALOG SERVO SYSTEM MODIFIED FOR AUTOMATIC CHECKOUT



REPRODUCIBILITY OF THE  
ORIGINAL PAGE IS POOR

signal from the tachometer. The resulting output goes to a torque motor which drives the tape sprocket, tachometer and position potentiometer. See Figure 8.3-10. Specific meters that are expected to have tape drives are the Air Speed/Mach Indicator, and Altimeter/Vertical Velocity Indicator.

The following analysis assumes that the correct voltage will be applied to the connector pins, and that the internal voltage regulator and power supplies have been checked and calibrated as a result of previously performed checkout and calibration on the rest of the simulator equipment.

### 8.3.3.2 Failure Mode Analysis and Characteristic Parameters

The failure modes of servo meters and tape meters are summarized in Tables 8.3-6 and 8.3-7, respectively. Since these devices are both basic servomechanisms, the normal performance criteria for simple control systems are applicable. In this instance, the evaluation of the meters response to a step input should be sufficient to identify the failures noted in the table. In each case, the characteristic parameters may be taken to be the following:

- Rise time
- Percent overshoot
- Settling time
- The time of peak overshoot may also be useful
- Static calibration check points

Hardware requirements  data acquisition, as well as software requirements are summarized below.

### 8.3.3.3 Self Test Hardware Requirements

Automatic checkout or self-test of DC servo driven meters or tape meters, in general, requires a means for sensing needle/tape movement in response to voltage input, and the generation of an electrical response signal indicative of needle/tape position, which can be communicated back to a computer for processing. The various techniques for sensing needle/tape position enable detection of discrete points of needle/tape position, such as full scale and zero scale or selected intermediate points, or the continuous monitoring of needle/tape position.

MODE	SYMPTOMS	
	EXTERNAL	INTERNAL
OPEN CIRCUIT	NO NEEDLE MOVEMENT	NO CONTINUITY
SHORT CIRCUIT	NO NEEDLE MOVEMENT	LOWER THAN USUAL RESISTANCE
MECHANICAL FAILURE	NO NEEDLE MOVEMENT OR NON-LINEAR NEEDLE MOVEMENT	DRIED LUBRICANT OR PHYSICAL DAMAGE
400 HZ POWER SUPPLY OR TRANSFORMER FAILURE	NO NEEDLE MOVEMENT	OPEN/SHORTED TRANSFORMER WINDING, NO DC OUTPUT FROM POWER SUPPLY
FAILURE OF REGULATOR FOR REFERENCE VOLTAGE	NO NEEDLE MOVEMENT. INACCURATE NEEDLE MOVEMENT	NO POWER TO AMPLIFIER, DRIVER, MOTOR, FEEDBACK POTENTIOMETER. DRIFT
MOTOR	NO NEEDLE MOVEMENT	OPEN/SHORTED WINDING
	ERRATIC NEEDLE MOVEMENT	BOUND BEARINGS BENT SHAFT
SIGNAL AMPLIFIER, DRIVER	NO NEEDLE MOVEMENT	OPEN CIRCUIT
	INCORRECT NEEDLE MOVEMENT	NON-LINEAR VOLTAGE OUTPUT
POSITION POTENTIOMETER	NEEDLE QUICKLY DRIVEN OFF SCALE	OPEN WIPER, COIL OR SERIES RESISTOR
	NEEDLE OPERATES ON HALF OF SCALE ONLY, DRIVEN AGAINST HARD-STOP	ONE VOLTAGE MISSING ON COIL
CALIBRATION POTENTIOMETER	STEP SHIFT IN NEEDLE POSITION: CALIBRATION REQUIRED	OPEN CIRCUIT IN BYPASS LINE
	NO NEEDLE MOVEMENT	OPEN COIL OR SERIES RESISTOR.

TABLE 8.3-6 FAILURE MODES OF SERVO METERS

MODE	SYMPTOMS	
	EXTERNAL	INTERNAL
OPEN CIRCUIT	NO TAPE MOVEMENT	NO CONTINUITY
SHORT CIRCUIT	NO TAPE MOVEMENT	LOWER THAN USUAL RESISTANCE
MECHANICAL FAILURE	NO TAPE MOVEMENT OR NON-LINEAR TAPE MOVEMENT	DRIED LUBRICANT OR PHYSICAL DAMAGE
DEGRADED AMPLIFIER	NON-LINEAR TAPE MOVEMENT	NON-LINEAR VOLTAGE OUTPUT
MOTOR	NO/JERKY TAPE MOVEMENT	OPEN/SHORTED WINDING. BOUND BEARINGS. BENT SHAFT.
POSITION POT	TAPE DRIVEN TO EXTREME ENDS ONLY	OPEN WIPER OR COIL
TACHOMETER	OVER/UNDER SPEED OF TAPE, JERKY MOVEMENT OF TAPE	NO/LOW OUTPUT VOLTAGE BROKEN SHAFT OR GEAR TOOTH. SLIPPING GEAR.
TAPE	NO TAPE MOVEMENT	TAPE OFF SPROCKET TEETH OR SLIPPED A TOOTH. BROKEN TAPE.

TABLE 8.3-7 FAILURE MODES OF TAPE METERS

Candidate techniques applicable to the DC servo driven meters are summarized in Table 8.3-8.

It is recommended that the built-in position feedback signal be utilized to send a position signal to the computer during checkout. This should be a minimum cost modification for addition of a buffer amplifier and afford excellent resolution and minimal software requirements. See Tables 8.3-8 and 8.3-9 for a summary of characteristics of the discussed techniques.

The wiper on the position potentiometer and the needle/tape are mechanically connected so that both move according to a specific ratio. According to Figures 8.3-7 and 8.3-9, the position potentiometer feeds an analog signal to a summing amplifier in proportion to the needle/tape movement. If this signal is also routed to an analog-to-digital channel for needle/tape position processing by the computer, the basic checkout criteria are met. To prevent impedance mis-matching at the summing amplifier, a buffer amplifier would be used in the analog-to-digital line to the computer. The computer software can "rectify" any undesired signal inversion that may take place.

The use of the feedback signal from the position pot enables dynamic testing of these meters with a continuous position signal available for computer processing. If experience with a specific meter should establish calibration to be a major or frequent problem, calibration checks can be implemented using the phototransistors as discrete position sensors as proposed for the galvanometers.

#### 8.3.3.4 Self Test Software and Data Base Impact

The software flow diagram for the self test software for the servo driven needle and tape meters is shown in Figure 8.3-11. The software takes advantage of a standard dynamic test module defined and described in Section 7.1.1.3. The meter test software consists of the limited number of operations required to initialize the test software and identify the arrays for all required input data for the test module as well as the arrays to store results.

CANDIDATE TECHNIQUE	RELATIVE COST	HARDWARE MODIFICATIONS		SOFTWARE REQUIREMENTS	REMARKS
		MODS TO INSTRUMENT	ADDITIONAL MODS EQUIPMENT		
PHOTOELECTRIC SENSORS FOR FULL SCALE AND ZERO SCALE	MOD	INSTALL 2 PHOTOSENSORS. PUNCH 2 HOLES IN TAPE.	LIGHT SOURCE FOR ILLUMINATION 2 DISCRETE INPUTS/METER POWER SUPPLY(?)	PROCESSING OF DISCRETE DATA	WINDOW OPENINGS VERY SMALL. TAPE STRENGTH DEGRADED.
SPECIAL CONTACT AREA ON TAPE		INSTALL 2 CONTACT SURFACES AND BRUSHES.	2 DISCRETE INPUTS/METER		SHORT TAPE LIFE.
FLUIDICS SENSING OF TAPE POSITION	MOD HIGH	VENT CASE. INSTALL 2 FLUIDIC SENSORS. ADD HOLES TO TAPE.	ADD AIR SUPPLY SYSTEM ADD TRANSDUCERS 2 DISCRETE INPUTS/METER.		TEST CIRCUIT CHECKED:
MAGNETIC SENSORS ON TAPE FOR FULL SCALE AND ZERO SCALE	MOD HIGH	INSTALL 2 MAGNETIC REED RELAYS. 2 MAGNETS ON TAPE.	2 DISCRETE INPUTS/METER		EXACT POSITION LESS CERTAIN. DIFFICULTY WITH MAGNET GOING AROUND SPROCKET.
FEEDBACK FROM POSITION POTENTIOMETER	LOW	INSTALL AMPLIFIER INSIDE CASE. TAP FEEDBACK SIGNAL	1 ANALOG CHANNEL/METER	PROCESSING AND EVALUATION OF QUANTITATIVE PARAMETERS.	POSITION SIGNAL EXISTS.
TV SCAN	VERY HIGH	POSSIBLY PAINT TAPE	ILLUMINATION SOURCE TV CAMERA(S) CONTROL CIRCUITRY VIDEO PROCESSING HARDWARE	COMPLEX PROCESSING OF IMAGE DATA	HIGH COST. UNCERTAIN THAT TAPE (SMALL AREA) MOVEMENT CAN BE ADEQUATELY DETECTED.

TABLE 8.3-8 CHARACTERISTICS OF CANDIDATE TECHNIQUES FOR TAPE METERS

8-110

CANDIDATE TECHNIQUE	RELATIVE COST	HARDWARE MODIFICATIONS		SOFTWARE REQUIREMENTS	REMARKS
		MODS TO INSTRUMENT	ADDITIONAL MODS EQUIPMENT.		
PHOTOELECTRIC SENSORS FOR FULL SCALE AND ZERO SCALE	MODERATE	INSTALL 2 PHOTOSENSORS ON REINFORCED SCALE	LIGHT SOURCE FOR ILLUMINATION 2 DISCRETE INPUTS/METER POWER SUPPLY(?)	PROCESSING OF DISCRETE DATA	NO INTERFERENCE WITH NEEDLE
NEEDLE CONTACT FOR CONTINUITY		INSTALL 2 CONTACTS	2 DISCRETE INPUTS/METER		CALIBRATION INTERFERENCE. OVER-RANGE NOT DETECTED.
FLUIDICS SENSING OF NEEDLE POSITION	MOD HIGH	VENT CASE. INSTALL 2 FLUIDIC SENSORS.	POWER SUPPLY(?) ADD AIR SUPPLY SYSTEM ADD TRANSDUCERS 2 DISCRETE INPUTS/METER	TEST CIRCUIT CHECKED.	EXTRA HARDWARE FOR PNEUMATICS, AND EXTRA MAINTENANCE OF PNEUMATIC SYSTEM.
TV SCAN	VERY HIGH	POSSIBLY PAINT NEEDLE	ILLUMINATION SOURCE TV CAMERA(S) CONTROL CIRCUITRY VIDEO PROCESSING HARDWARE	COMPLEX PROCESSING OF IMAGE DATA	HIGH COST. UNCERTAIN THAT NEEDLE (SMALL AREA) MOVEMENT CAN BE ADEQUATELY DETECTED.
FEEDBACK SIGNAL FROM POSITION POTENTIOMETER	LOW	ADD BUFFER AMPLIFIER. TAP FEEDBACK SIGNAL	ONE ANALOG CHANNEL PER METER.	PROCESSING AND EVALUATION OF QUANTITATIVE PARAMETERS	NO INTERFERENCE WITH NEEDLE. LOWEST COST. MOST VERSATILE IN SOFTWARE MODIFICATIONS. POSITION SIGNAL EXISTS.
SNAP ACTION SWITCHES	MODERATE	INSTALL 2 LONG LEVER SNAP ACTION SWITCHES	2 DISCRETE INPUTS/METER	PROCESSING OF DISCRETE DATA TEST CIRCUIT CHECKED	LOOSE TOLERANCE ON REPEATABILITY GIVES POOR TEST RESULTS. LOW FORCE REQUIREMENTS FROM NEEDLE.

TABLE 8.3-9 CHARACTERISTICS OF CANDIDATE TECHNIQUES FOR SERVO METERS

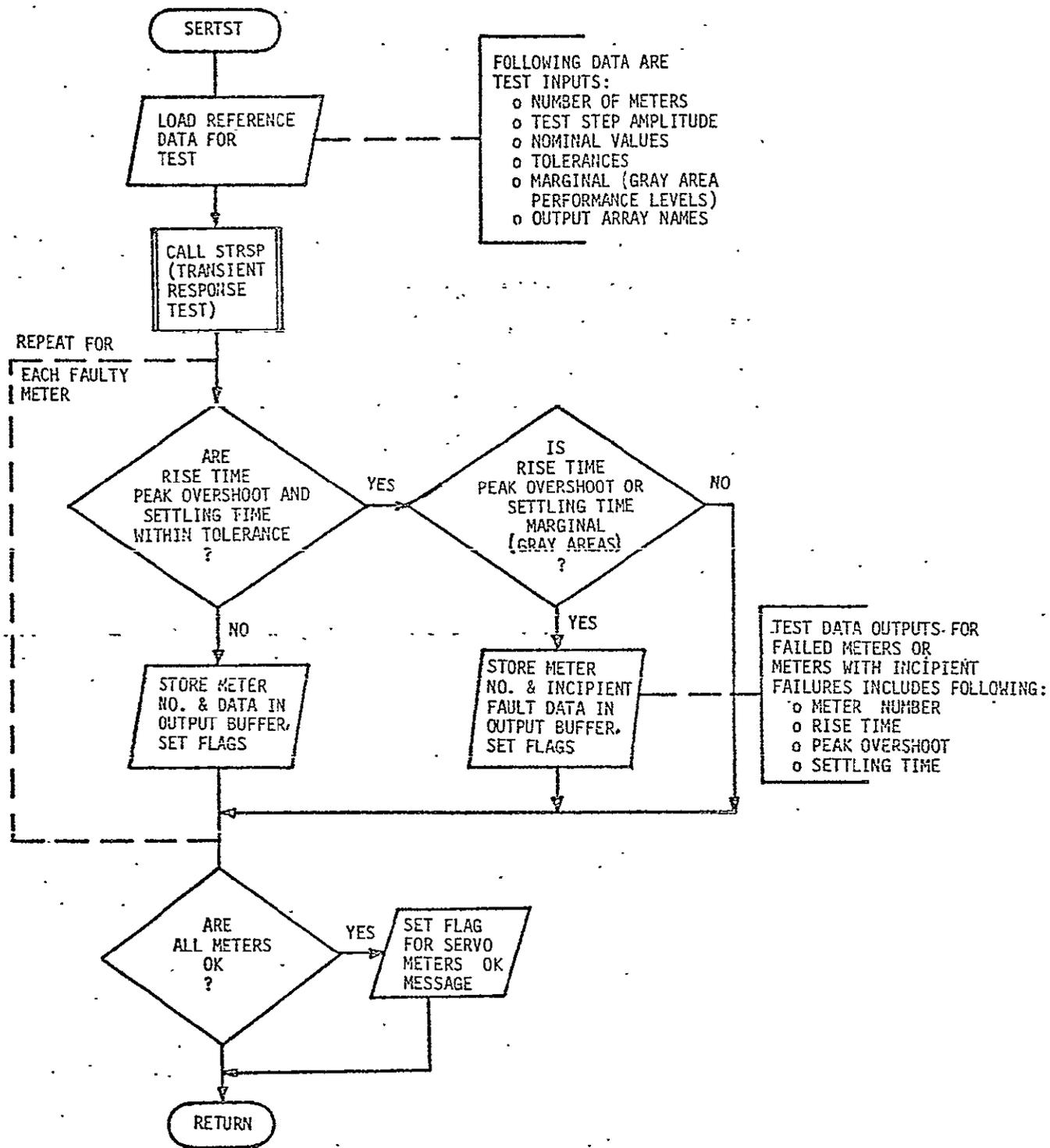


FIGURE 8.3-11 SERVO METER SELF TEST SOFTWARE FLOW, SERTST

The reference data arrays requiring data base storage consist of the following:

- Step input test signal amplitude or scale factors
- Rise time nominal values
- Rise time tolerances for absolute failure
- Rise time tolerances for gray area performance (incipient faults)
- Peak overshoot nominal values
- Peak overshoot tolerances for unacceptable performance
- Peak overshoot tolerances for incipient fault detection
- Settling time nominal values
- Settling time tolerances for absolute failure
- Settling time tolerances for incipient fault detection

The total data base impact for 36 meters in the crew stations is 9 X 36 or 324 floating point parameters. Instruction requirements peculiar to the meter test are insignificant. The dynamic test module is counted as a library routine. If phototransistors are added for calibration checks, the data base is not impacted significantly. The parameters acquired for that purpose are simply discretized.

1 November 1974

### 8.3.4 Synchro/Resolver Driven Meters

The Attitude Director Indicator (ADI) contains a resolver driven attitude ball and six servometer pointer movements. The Horizontal Situation Indicator is similar to the DC servo driven meters except for the TO/FROM Indicator, DME readouts, Heading Select Bug, and Selected Course Pointer. The crew station contains two each of these instruments and the IOS contains two each for a total of four ADI's in the simulator.

#### 8.3.4.1 Description and List of Displays Applicable

The Attitude Director Indicator has an attitude ball and six servometer pointer movements: three for rate indications and three for error indications. Figure 8.3-12 depicts the face of the ADI. One rate pointer and one error pointer are provided for each spacecraft axis: pitch, yaw and roll. The pointer is basically a motor driven pointer with a shaft extension that comprises the wiper of a position feedback potentiometer that is connected to +15VDC and -15VDC. See figure 8.3-13. This feedback voltage is fed as an error signal into an amplifier which drives the motor.

The attitude ball is a hollow sphere which is capable of continuous rotation in three axes to indicate the pitch, yaw and roll attitude of the spacecraft. Each axis is controlled by a 4-wire (sine/cosine) input network. Each axis also has a motor, a rate generator, and a resolver position feedback which operate from 400 hertz input. See Figure 8.3-14. When the ball is at the commanded position, the resolver feedback signal is at a null (minimum AC voltage output). The resolver feedback signal is fed into the amplifier which drives the motor.

The ADI, as used on a simulator, is normally a flight type instrument that has not been tested for airworthiness. It is usually sealed air tight, and may contain an inert gas. For more detailed mechanical information on a typical ADI, see the top assembly drawing, DJG264E, for the Apollo Flight Director Attitude Indicator (FDAI) produced by Honeywell, Inc., Military Products Group.

During the first five years of the Apollo and Skylab program, there were essentially no failures of the ADI. The only problems that developed were the

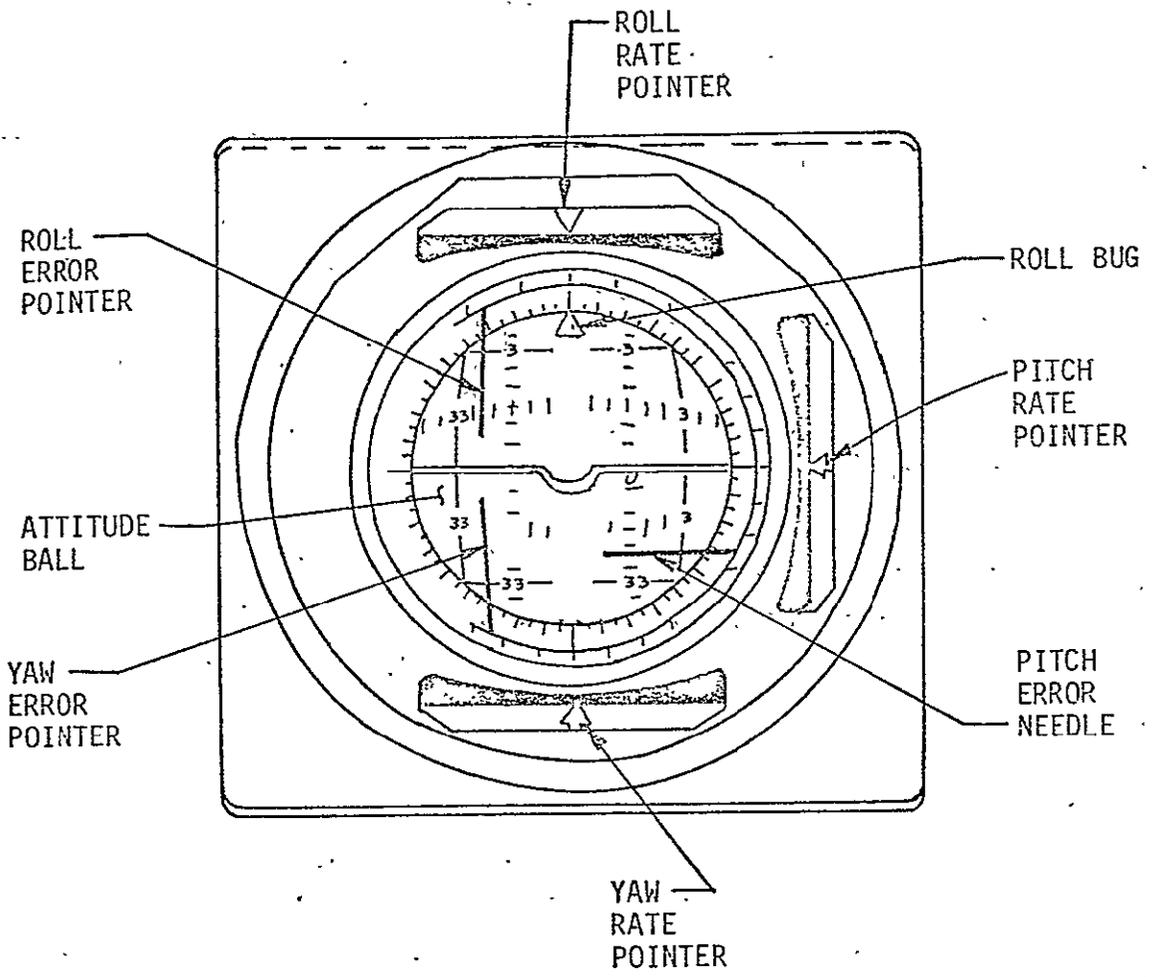


FIGURE 8.3-12 FACE OF ATTITUDE DIRECTOR INDICATOR

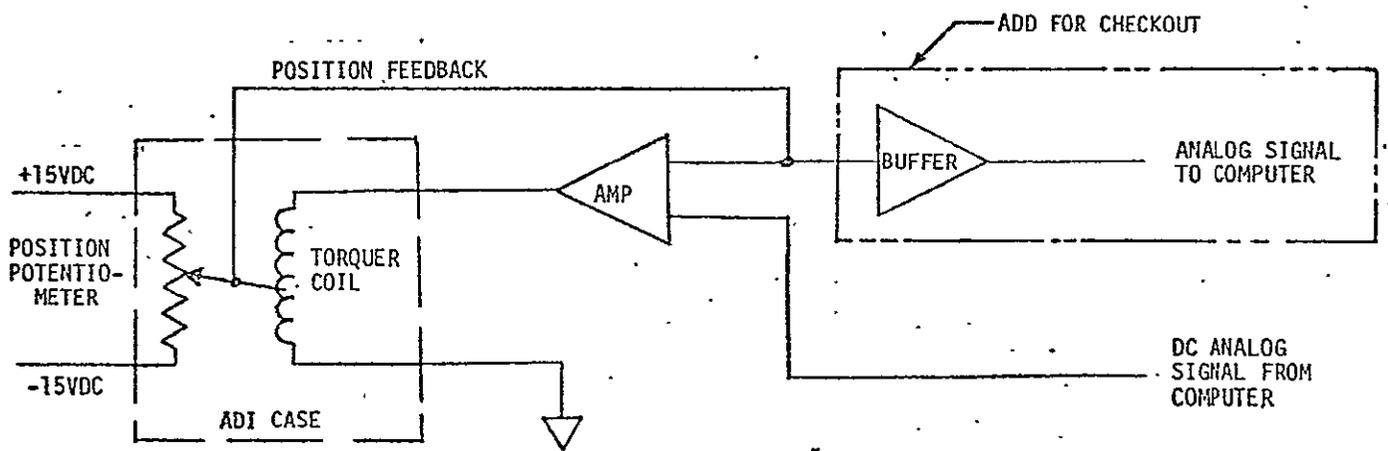


FIGURE 8.3-13 TYPICAL RATE/ERROR POINTER CIRCUIT

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

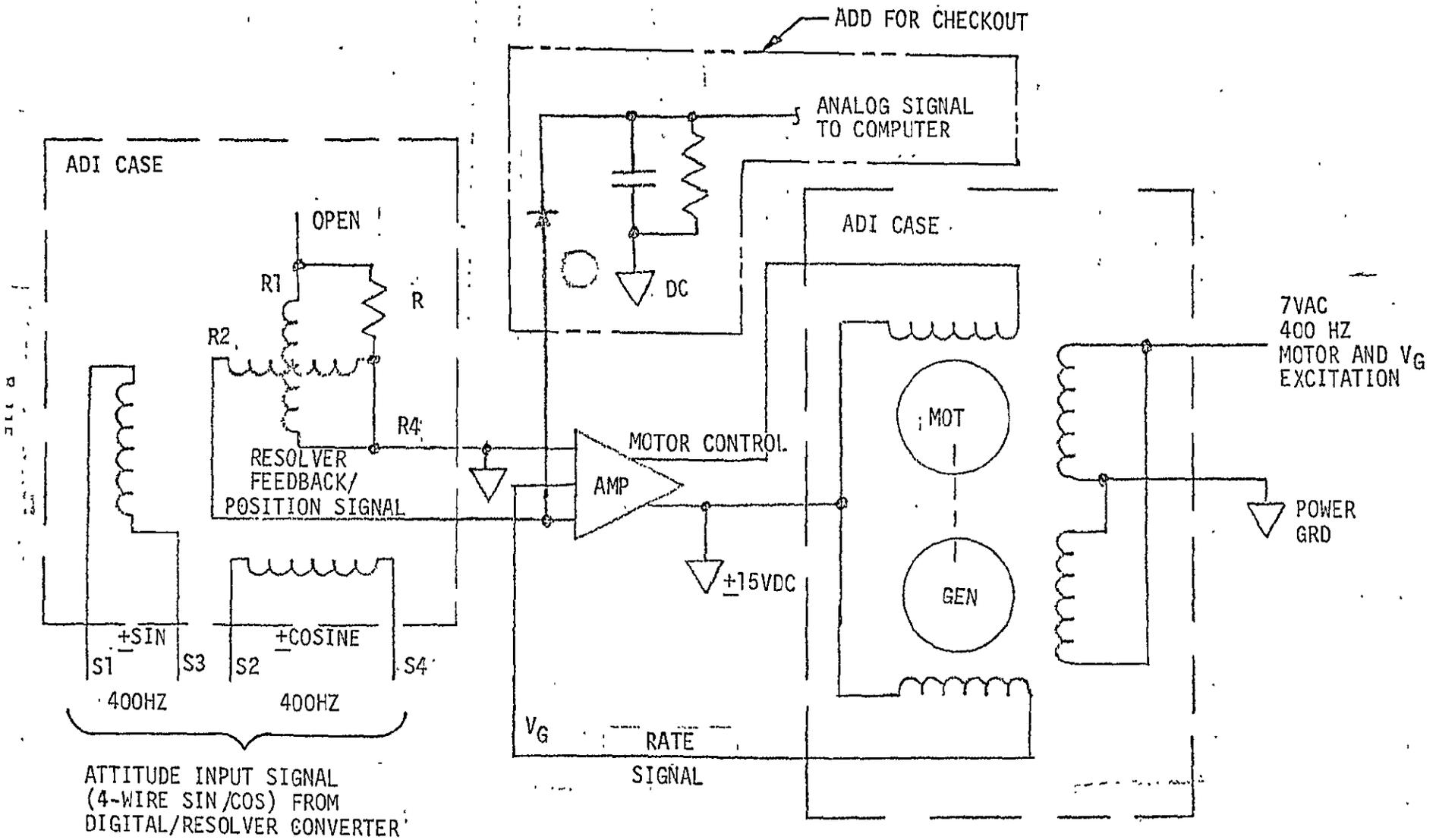


FIGURE 8.3-14 TYPICAL CIRCUIT OF ONE AXIS OF ATTITUDE BALL

1 November 1974

gradual degradation of the electroluminescent lighting and the cleaning and oiling of bearings. A few other problems developed during the next two years. See Table 8.3-10.

The Horizontal Situation Indicator (HSI) contains several components that display navigation information. The information displayed is: miles from a TACAN station (2 displays), magnetic heading, heading selected, course selected, angular deviation from course, whether approaching or leaving selected TACAN station, and angular position relative to the selected glideslope. A picture of the instrument is shown in Figure 8.3-15.

The components which display this information are listed in Table 8.3-11. Component description is also given in Table 8.3-11, and sample schematics are shown in Figures 8.3-16, -17, -18, and -19.

The analysis that follows assumes that the correct test voltages are applied to the connector pins of the instrument. This implies that the circuitry between the computer and the instrument has been checked out and calibrated.

#### 8.3.4.2 Failure Modes and Characteristic Parameters

The failure modes of the ADI and the HSI respectively are summarized in Tables 8.3-12 and 8.3-13 respectively.

The characteristic parameters for composite instruments of this type must necessarily be the characteristic parameters nominally used for assessing the performance of each of the basic type of instrument components.

The characteristic parameters required for the ADI consist of the following:

TABLE 8.3-10 ATTITUDE DIRECTOR INDICATOR

INPUTS: Motor control signal for each of 3 axis  
4 wire sin/cos signal for each of 3 axes  
400hz power  
6 DC signals for pointers  
+15VDC reference voltage

OUTPUTS: Movement of six pointers  
Attitude ball position in 3 axes  
3 pairs of 400hz 2 wire resolver feedback/position signal for ball position  
-15VDC to +15DC position feedback signals for each of 6 pointers

RELIABILITY: 7,000 hr MTBF

QUANTITY: 2 in crew station  
2 in instr./operator station

EXPECTED FAILURE MODES:

Short Circuit

Open Circuit

No change, or non-linear change in output due to dirt or evaporated  
lubricant (if instrument is not properly sealed)

Electrical overload may bend or break a moveable part.

CRITICAL MEASUREMENT POINTS: input and output terminals  
pointer position  
ball position

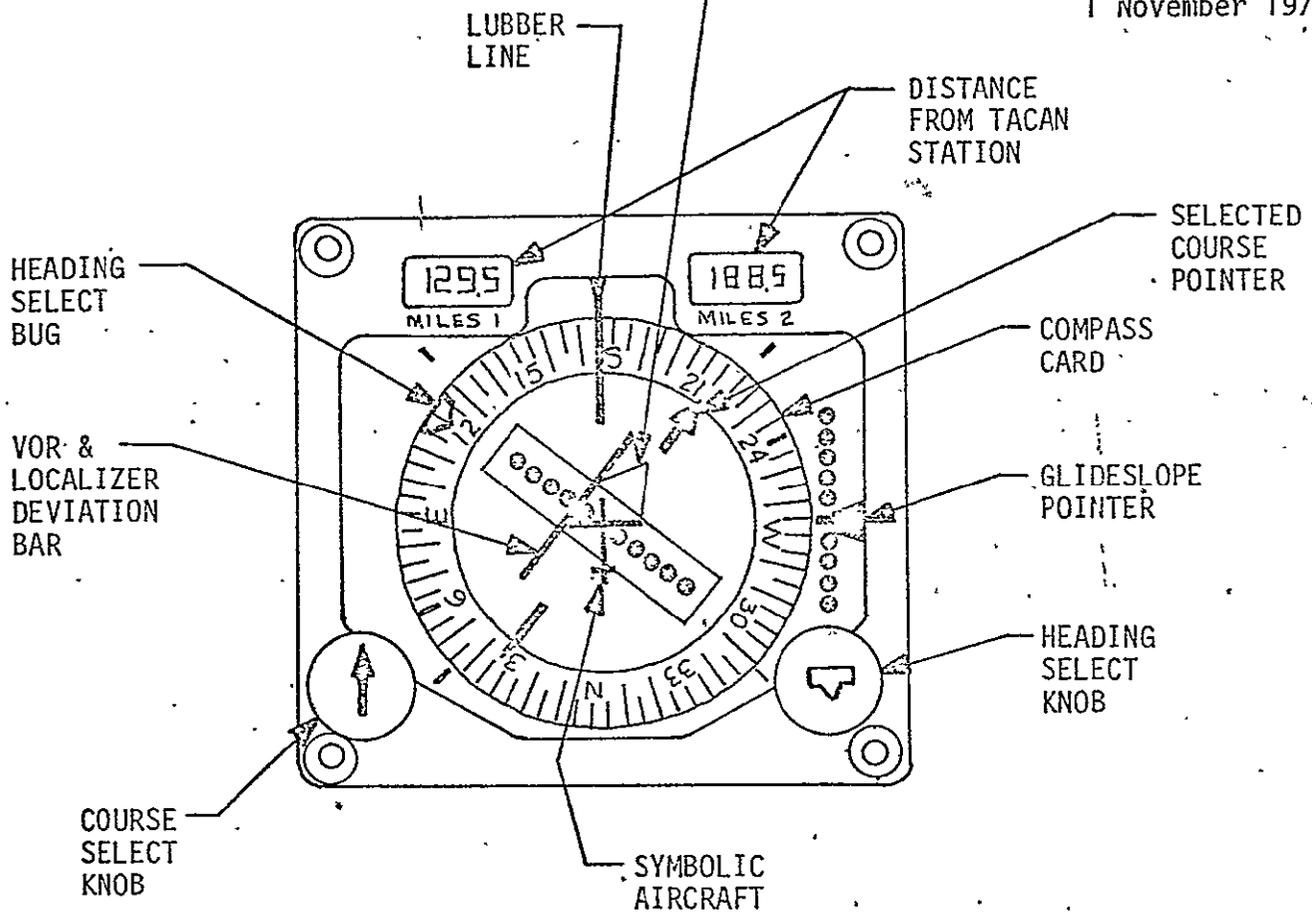


FIGURE 8.3-15 HORIZONTAL SITUATION INDICATOR.

INFORMATION DISPLAYED	COMPONENT COMMON NAME	INPUT	OUTPUT	TYPE OF DEVICE	REMARKS
1. MILES FROM TACAN	DISTANCE MEASURING EQUIPMENT (DME)	DIGITAL DC	ILLUMINATION OF SEGMENTS OF 7 ELEMENT DIGITS	7-ELEMENT READOUT, 4 DIGITS/WINDOW, 2 WINDOWS	ALTERNATE METHODS: DESIGN IS NOT FROZEN
		ANALOG DC	ODOMETER MOVEMENT	SERVO CONTROLLED ODOMETER	
2. MAGNETIC HEADING	COMPASS CARD AND LUBBER LINE	ANALOG DC	COMPASS "CARD" ROTATION LUBBER LINE IS FIXED.	SERVO	
3. HEADING SELECTED	HEADING SELECT BUG	HEADING SELECT KNOB	ANALOG DC	GEAR TRAIN AND POTENTIOMETER	
4. COURSE SELECTED	SELECTED COURSE POINTER	COURSE SELECT KNOB	ANALOG DC	GEAR TRAIN AND POTENTIOMETER	DEVIATION BAR AND "TAIL OF POINTER, ALSO "DOTS" AND TO/FROM INDICATOR ROTATE WITH POINTER MOVEMENT.
5. ANGULAR DEVIATION FROM COURSE	VOR AND LOCALIZER DEVIATION BAR	ANALOG DC	MOVEMENT OF DEVIATION BAR	GALVANOMETER*	
6. APPROACHING/LEAVING TACAN STATION	TO/FROM INDICATOR	DIGITAL DC	SEESAW MOVEMENT OF POINTER	SOLENOID	
7. ANGULAR POSITION RELATIVE TO GLIDESLOPE	GLIDESLOPE POINTER	ANALOG DC	MOVEMENT OF GLIDESLOPE POINTER	GALVANOMETER*	

\* KING RADIO CORP. MODEL K1525, PICTORIAL NAVIGATION INDICATOR. SHUTTLE INSTRUMENT IS EXPECTED TO BE SERVO DRIVEN.  
 QUANTITY: 2 IN CREW STATION, 2 IN INSTRUCTOR-OPERATOR STATION, FLIGHT INSTRUMENT, MODIFIED TO ACCEPT DC INPUTS.  
 RELIABILITY: NO DATA AVAILABLE.

EXPECTED FAILURE MODES: SHORT CIRCUIT  
 OPEN CIRCUIT  
 BREAKAGE OR SEPARATION OF PARTS FROM VIBRATION/SHOCK.  
 ELECTRICAL OVERLOAD MAY BEND OR BREAK A MOVEABLE PART.

NON-MOVING COMPONENTS: LUBBER LINE  
 SYMBOLIC AIRCRAFT

CRITICAL MEASUREMENT POINTS: INPUT AND OUTPUT TERMINALS  
 POINTER/INDICATOR POSITIONS

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

TABLE 8.3-11 FUNCTIONAL DESCRIPTION OF HSI ELEMENTS

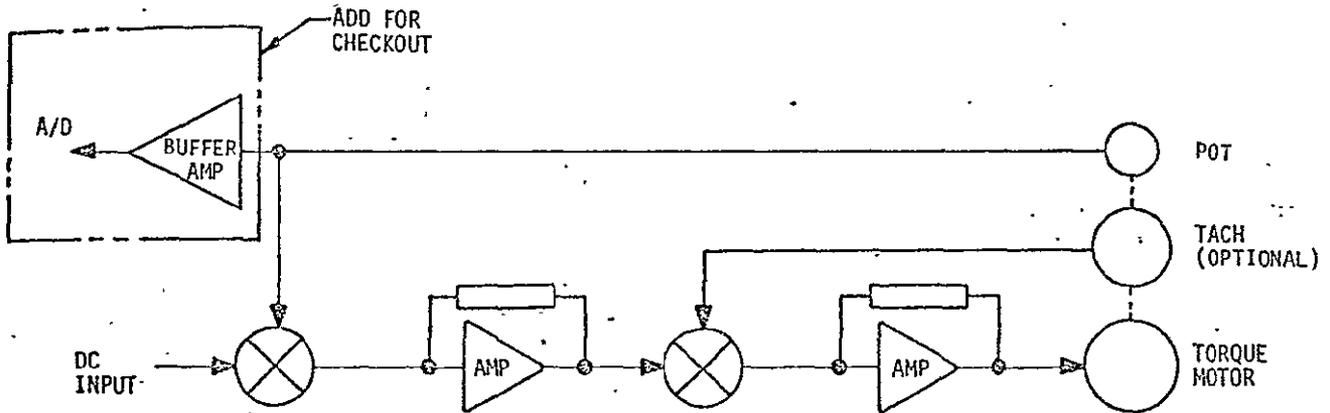


FIGURE 8.3-16 DC ANALOG SERVO SYSTEM AS USED FOR GLIDESLOPE,  
DEVIATION BAR, AND COMPASS CARD

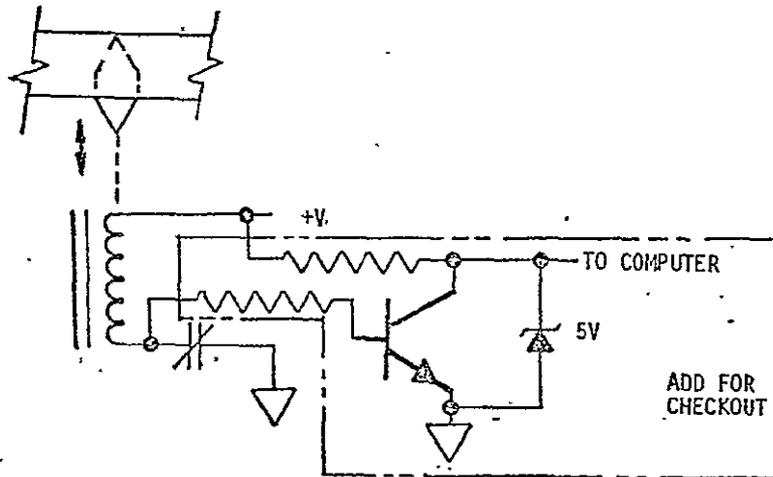


FIGURE 8.3-17 SOLENOID OPERATED TO/FROM INDICATOR

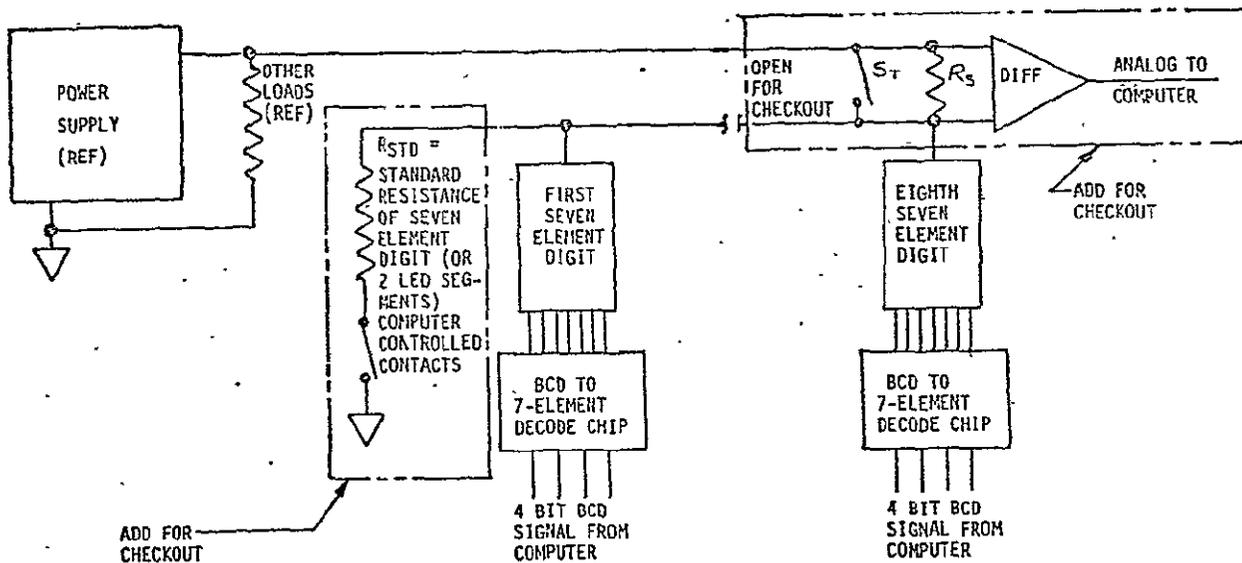


FIGURE 8.3-18. TECHNIQUE FOR MEASURING CURRENT THROUGH DME SEVEN-ELEMENT DIGITAL READOUT ASSEMBLIES

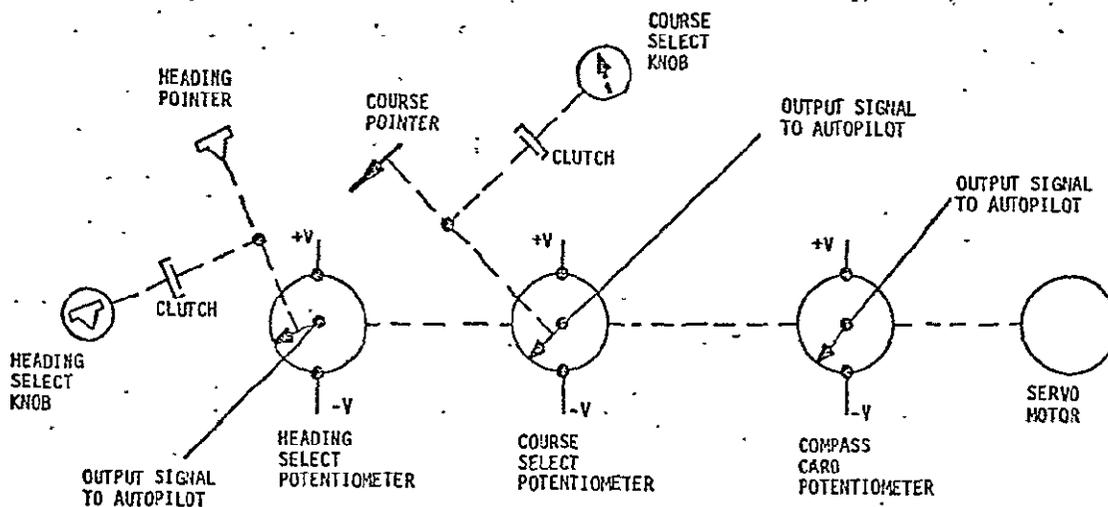


FIGURE 8.3-19 HEADING SELECT AND COURSE SELECT ELECTRICAL DIAGRAM

1 November 1974

Component	Characteristic Parameters
Attitude ball (servo driven)	Static position calibration parameters Dynamic response on each axis Rise time Peak overshoot Settling time Critical tolerances and marginal tolerances for the above parameters
Pointers (Galvanometers)	Static position checks Rise time Settling time Tolerances on the above

The characteristic parameters for the HSI must include the above for the servo driven and galvanometer movements as well as the following for the additional types of components noted:

Component	Characteristic Parameters
DME (7 element digital readout)	Discrete status for each element.
To/From Indicator (Solenoid)	Discrete status (two position)

The remaining functions of the HSI are either servo driven or galvanometers.

The combination of calibration check points, dynamic response for the continuously varying devices and condition checks for the discrete or bilevel devices should be sufficient to reveal any of the types of malfunctions noted in the Tables 8.3-12 and 8.3-13.

#### 8.3.4.3 Self Test Hardware Requirements

Automated checkout or self-test of pointer or ball instruments, in general, requires a means for sensing instrument movement in response to a voltage input, and the generation of an electrical response signal indicative of instrument position, which can be communicated back to a computer for processing. In the case of readouts which are composed of glowing numbers or segments of numbers,

MODE	SYMPTOMS	
	EXTERNAL	INTERNAL
By Failure of Component		
POINTER SERVO MOTOR	NO POINTER MOVEMENT	OPEN/SHORTED WINDING, WIPER
	ERRATIC POINTER MOVEMENT	BOUND BEARINGS, BENT SHAFT
POSITION POTENTIOMETER	POINTER DRIVEN TO EXTREME ENDS QUICKLY.	OPEN WIPER OR COIL
	POINTER OPERATES ON HALF OF SCALE ONLY, DRIVEN AGAINST HARDSTOP	ONE VOLTAGE MISSING ON COIL
TACHOMETER	OVER/UNDER SPEED OF BALL, JERKY MOVEMENT OF BALL	NO/LOW OUTPUT VOLTAGE, SLIPPING GEAR, OPEN/SHORTED WINDING IN 400HZ or V <sub>G</sub> COIL
ATTITUDE BALL RESOLVER MOTOR	NO/JERKY BALL MOVEMENT	BOUND BEARINGS, OPEN/SHORTED SECTION OF WINDING
RESOLVER NETWORK	BALL STOPS IN WRONG POSITION	OPEN/SHORTED WINDING OF ONE STATOR OR ONE ROTOR
	NO BALL MOVEMENT	OPEN/SHORTED BOTH 400 HERTZ STATOR COILS OR BOTH ROTOR WINDINGS
MECHANICAL FAILURE	IRREGULAR POINTER/BALL MOVEMENT	WORN BEARINGS, DRIED LUBE
	POINTER OUT OF CALIBRATION	BENT POINTER
	IRREGULAR POINTER/BALL MOVEMENT	DRIED LUBRICANT
OPEN CIRCUIT	NO POINTER MOVEMENT	NO CONTINUITY
SHORT CIRCUIT	NO POINTER MOVEMENT	LOWER THAN USUAL RESISTANCE

TABLE 8.3-12 ADI PECULIAR FAILURE MODES

MAJOR COMPONENT	MODE BY SUB-COMPONENT	SYMPTOMS	
		EXTERNAL	INTERNAL
GLIDESLOPE, DEVIATION BAR, COMPASS CARD	SERVO MOTOR	NO NEEDLE/CARD MOVEMENT	OPEN/SHORTED WINDING
		ERRATIC NEEDLE/CARD MOVEMENT	BOUND BEARINGS, BENT SHAFT
	POSITION POTENTIOMETER	NEEDLE QUICKLY DRIVEN OFF SCALE	OPEN WIPER, COIL OR SERIES RESISTOR
		NEEDLE/CARD OPERATES ON HALF OF SCALE ONLY. NEEDLE DRIVEN AGAINST HARDSTOP.	ONE VOLTAGE MISSION ON COIL
	SIGNAL AMPLIFIER, MOTOR DRIVER	NO NEEDLE/CARD MOVEMENT INCORRECT MOVEMENT	OPEN CIRCUIT NON-LINEAR VOLTAGE OUTPUT
CALIBRATION POTENTIOMETER	STEP SHIFT IN NEEDLE/CARD POSITION: CALIBRATION REQUIRED NO NEEDLE MOVEMENT	OPEN CIRCUIT IN BYPASS LINE OPEN COIL OR SERIES RESISTOR	
TACHOMETER	OVER/UNDER SPEED OF NEEDLE/CARD, JERKY MOVEMENT	NO/LOW OUTPUT VOLTAGE. BROKEN SHAFT OR GEAR TOOTH. SLIPPING GEAR	
HEADING SELECT BUG, COURSE SELECT POINTER OR KNOBS	POTENTIOMETER	INCORRECT/NO DC OUTPUT	OPEN/SHORT CIRCUIT
	GEAR TRAIN	INCORRECT DC OUTPUT	FAILED COMPONENT, SLIPPED GEAR OR SLIPPING CLUTCH
		NO MOVEMENT OF BUG/POINTER	SLIPPING GEAR TRAIN
TO/FROM INDICATOR	SOLENOID AND POINTER	INDICATOR DOES NOT MOVE WHEN TACAN/ VOR STATION IS PASSED	OPEN/SHORT CIRCUIT STUCK SOLENOID/POINTER
DME	DECODE-DRIVER MODULE	WRONG LIGHT SEGMENT(S) LIGHT UP OR FAIL TO EXTINGUISH	DECODE LOGIC MODULE FAILURE
	LIGHT SEGMENT(S)	LIGHT SEGMENTS DO NOT LIGHT UP	OPEN/SHORT CIRCUIT LIGHT SEGMENT BURNED OUT
	ODOMETER GEAR TRAIN	NO MOVEMENT OF DIGIT WHEEL	GEAR TRAIN LOCKED
	SERVO MOTOR, POSITION POTENTIOMETER, SIGNAL AMPLIFIER, MOTOR DRIVER, CALIBRATION POTENTIOMETER, TACHOMETER	SEE GLIDESLOPE, DEVIATION BAR AND COMPASS CARD. EXTERNAL SYMPTOMS APPLY TO DIGIT WHEEL(S) INSTEAD OF NEEDLE/CARD MOVEMENT.	

TABLE 8.3-13 FAILURE MODES OF THE HSI

CANDIDATE TECHNIQUE	RELATIVE COST	HARDWARE MODIFICATIONS		SOFTWARE REQUIREMENTS	REMARKS
		MODS TO INSTRUMENT	ADDITIONAL MODS EQUIPMENT		
FEEDBACK SIGNAL FROM POSITION POTENTIOMETER	LOW	ADD BUFFER AMPLIFIER. TAP FEEDBACK SIGNAL	ONE ANALOG CHANNEL PER METER	PROCESSING AND EVALUATION OF QUANTITATIVE PARAMETERS	NO INTERFERENCE WITH NEEDLE. LOWEST COST. MOST VERSATILE IN SOFTWARE MODIFICATIONS. POSITION SIGNAL EXISTS.
RESOLVER FEEDBACK	LOW	NONE	ADD RECTIFIER, FILTER AND ANALOG LINE		
SHAFT ENCODERS	HIGH	ADD DECIMAL SHAFT ENCODERS FOR EACH BALL AXIS	ADD BCD ENCODERS AND 24 DIGITAL LINES	PROCESSING OF DISCRETE DATA	MOST BULKY MOD TO INSTRUMENT AND REQUIRES MOST DIGITAL LINES TO COMPUTER
SINE-COSINE POTENTIOMETERS	MED HIGH	ADD SINE-COSINE POT PER BALL AXIS	ADD 6 ANALOG LINES TO COMPUTER	PROCESSING AND EVALUATION OF QUANTITATIVE PARAMETERS	LOWER ACCURACY. GOOD SOFTWARE VERSATILITY, WITH CONTINUOUS POSITION MONITORING.
CODE PINS ON BALL	MED HIGH	ADD BINARY CODED PINS TO EACH AXIS OF BALL, 45° INCREMENT	ADD 12 DIGITAL LINES TO COMPUTER	PROCESSING OF DISCRETE DATA	GOOD ACCURACY. MINIMAL FLEXIBILITY.

TABLE 8.3-14 CHARACTERISTICS OF CANDIDATE TECHNIQUES FOR ADI CHECKOUT

CANDIDATE TECHNIQUE	APPLIES TO INSTRUMENT MOVEMENT	HARDWARE MODIFICATIONS		SOFTWARE REQUIREMENT	RELATIVE COST	REMARKS
		MODIFICATIONS TO INSTRUMENT	ADDITIONAL MODIFICATIONS TO EQUIPMENT			
CURRENT SENSOR	DME 7 ELEMENT READOUTS	ADD DIFFERENTIAL AMP, SWITCH, SHUNT RESISTOR & STD. RESISTOR	ADD ANALOG LINE TO COMPUTER. ADD DISCRETE OUT OF COMPUTER	PROCESSING AND EVALUATION OF QUANTITATIVE PARA.	LOW	MORE RELIABLE IN SENSING THAN PHOTO-ELECTRIC SENSORS OR THERMISTORS
DC FEEDBACK	COMPASS CARD, COURSE SELECT POINTER, HEADING SELECT BUG, GLIDESLOPE, VOR DEVIATION BAR	ADD ONE WIRE PER CHANNEL	ADD BUFFER AMPLIFIER PER CHANNEL. ONE ANALOG CHANNEL PER METER	PROCESSING AND EVALUATION OF QUANTITATIVE PARAMETERS	LOW	MINIMAL INSTRUMENT MODIFICATION. DIRECT DC FEEDBACK EXISTS WITHOUT MODIFICATION FOR HEADING SELECT BUG AND FOR COURSE SELECT POINTER
FLUIDICS	TO/FROM INDICATOR, OPTIONAL WARNING FLAGS	VENT CASE. INSTALL 2 FLUIDIC SENSORS PER INDICATOR	ADD AIR SUPPLY SYSTEM, ADD TRANSDUCERS, ADD 2 DISCRETE INPUTS/METER, (ADD POWER SUPPLY?)	PROCESSING OF DISCRETE DATA	MOD HIGH	LIMITED VOLUME AVAILABLE. SENSES INDICATOR DISPLACEMENT.
SNAP ACTION SWITCHES OR MAGNETIC SWITCHES		ADD 1 SWITCH PER ACTION	ADD 1 DISCRETE LINE PER SWITCH		MOD	LIMITED VOLUME AVAILABLE. CALIBRATION INTERFERENCE DUE TO ADDED FORCE REQUIREMENT
PHOTOELECTRIC SENSORS		2 SENSORS PER INDICATOR	ADD 1 AMPLIFIER & DISCRETE LINE PER SENSOR. ILLUMINATION SOURCE (POWER SUPPLY?)		TEST CIRCUIT CHECKED. PROCESSING OF DISCRETE DATA	MOD

TABLE 8.3-15 CHARACTERISTICS OF CANDIDATE TECHNIQUES FOR HSI CHECKOUT

1 November 1974

a means of sensing light directly or indirectly is required, coupled with the generation of an "on-off" electrical signal to the computer.

Various techniques were evaluated for self test for the ADI and the HSI. Many of these require substantial modifications of the instruments. Modifications make the instrument a "special," require extra drawings, risk damage by the act of modification, may upset physical balance, and involve requirements for unknown clearances which may not be available. The recommended checkout instrumentation requires minimum modification to the instrument. The techniques considered for the ADI are summarized in Table 8.3-14. The following additional comments on these techniques may be of interest:

- A. Shaft encoders for each ball axis: Each encoder requires a large volume (three times) inside the instrument. Up to eight signal wires per axis (24 total) for BCD encoding, plus a common wire, would be added for resolution to one degree. This number of wires would almost certainly require one or more additional connectors. However, less precise resolution would require fewer wires. The encoders would require additional torque from the motors, which may then become overloaded.
- B. Sine-Cosine potentiometers attached to each ball axis: This method requires much less volume, and also requires fewer wires: a total of eight. The resolution is less precise than the shaft encoders unless gearing, and extra potentiometers and wires are added. In the usual arrangement two continuous-ring potentiometers are attached to each axis. Equal voltages of opposite polarity are applied 180° apart to each potentiometer ring, but the two potentiometers on the same axis are shifted 90° to each other. Thus, one potentiometer gives the sine signal: the other gives the cosine signal.
- C. Code pins or special contacts every 45° in each axis of the ball: By using fewer test points than plan A (shaft encoders), the number of wires and the physical volume are reduced. Binary coded

1 November 1974

contacts every 45° in each axis requires a maximum of four contacts per check point, and 12 extra wires at the connector. Unless the binary coded contacts can be hidden like those for the roll axis yoke, the ball would be unsightly and astronaut training value would be degraded that much.

The techniques considered for the HSI are summarized in Table 8.3-15.

The Horizontal Situation Indicator and Attitude Director Indicator are multi-component instruments. The recommended checkout technique for each component is discussed below.

It is recommended that the feedback loop signal from each servometer pointer movement be buffered and sent to the computer for analysis. The resolver feedback/position signal of the ADI should be rectified and also sent as an analog voltage to the computer for analysis. The correct resolver signal should be a null voltage, whereas the servo feedback voltage will vary from -15VDC to +15VDC.

The Horizontal Situation Indicator is similar to the DC servo driven meters except the TO/FROM Indicator, DME Readout, Heading Select Bug, and Selected Course Pointer.

Since there are several types of instrument movements within the HSI, each movement is treated separately in the checkout. The recommendations for each component are listed below and the associated hardware requirements are described.

#### ADI Self Test Hardware

The ADI has a sensor for detecting pointer movement and a proportionate electrical response signal (the position potentiometer) for each of the six pointers. These position signals can be tapped externally and sent to the computer via analog-to-digital channels, one for each pointer. See Figure 8.3-13. This idea is discussed in more detail in paragraph 8.3.3.3.

1 November 1974

Similarly, each axis of the attitude ball has a resolver feedback signal that can be tapped externally, rectified from AC to DC, filtered, and sent to the computer via an analog-to-digital channel. When the signal is at a null (zero volts) the ball has arrived at the commanded position. See Figure 8.3-14.

Final instrument design may necessitate one or more impedance buffers for computer interface circuit compatibility. The computer software will check the pointer movements similarly to the DC servo driven meters described in paragraph 8.3.3.3. The attitude ball can be checked by incrementing each axis  $45^{\circ}$  until a full revolution is completed, and reading the null voltage at the completion of each command.

The feedback signals also permit dynamic testing to be performed by applying step response analysis techniques, that can detect degraded performance or incipient faults.

#### HSI Self Test Hardware

The recommended techniques for self test of the HSI components may be summarized as follows:

<u>Component</u>	<u>Checkout Technique</u>
DME Readouts	Current sensors
To/From Indicator	Current sensors
VOR and Localizer Deviation Bar	Potentiometer position*
Course Select Pointer	DC position signal
Glideslope Pointer	Potentiometer position*
Heading Select Bug	DC position signal
Compass Card	Potentiometer position

\*A servo motor movement is assumed on the Shuttle. The King Radio Corp. KI525 Pictorial Navigation Indicator has a galvanometer. If a galvanometer is used on this instrument, reference is made to paragraph 8.3.2 for checkout techniques.

1 November 1974

The current sensor for the seven-element DME readout assemblies is shown in Figure 8.3-18. Its principal component is a differential amplifier that measures the IR drop (current x resistance = voltage) across a shunt resistor,  $R_S$ , during the checkout. During normal simulator operations  $R_S$  is effectively out of the circuit by the closure of the nearby switch,  $S_T$ . The power supply may vary as much as 5 to 10%, the differential amplifier by 2% and the incandescent bulbs by 5% (LEDs by 50%). Therefore, a maximum of one digit (seven incandescent elements) may be tested at one time. This quantity of seven segments means that all tolerances may accumulate to about 14% maximum (9 segments are restricted to 12%, and 58 segments are limited to less than 2%). Beyond that tolerance, detection of a burned out segment is not assured. The variations in power supply voltage and other circuit conditions are minimized by comparison of each digit (or group of segments) to the reading obtained from the "standard resistance" resistor in parallel to the seven element digits. Therefore, incandescent readouts may be tested by illuminating one digit at a time and measuring current flow with the differential amplifier. Only two or three LED segments may be tested at a time. Each segment(s) or digit would be turned ON/OFF by computer control.

The current sensor for monitoring the TO/FROM indicator is a transistor amplifier that is mounted externally. The only internal addition to the instrument is one more wire, which is brought through the instrument case to the base of the transistor. There is also one discrete channel to the computer.

Where servo movements are used, the potentiometer position feedback signal can be buffered and sent to the computer for analysis. See paragraph 8.3.3 and Figure 8.3-16 for more details.

The course select pointer and heading select bug send DC position signals external to the instrument. These signals can be processed similarly to the servo feedback signals. See Figure 8.3-19.

REPRODUCIBILITY OF THE  
ORIGINAL PAGE IS POOR

1 November 1974

#### 8.3.4.4 Self Test Software and Data Base Impact

The software flow logic for executing the calibration checks for the ADI is shown in Figure 8.3-20. Although error messages are shown in this flow chart, these messages could be issued by the executive software as part of the summary results of the total crew station test results. Dynamic test flows for the ADI are shown in Figure 8.3-21. Software requirements in addition to the servo and galvanometer test flow charted in the previous section are presented in Figure 8.3-22 for the Horizontal Situation Indicator.

Incipient fault detection requirements for these devices may be considered minimal because of their high reliability in past applications. However, for maintenance insurance in a high utilization, heavy training schedules condition, it is expected that the concept of marginal or gray area performance may be applied to detect degradation to unsatisfactory levels of component performance in these instruments. These components would be in the servo driven and the galvanometer movements.

The data base impact for these devices is very limited since we are addressing only two specific instruments. Each servo drive requires three nominal values and two sets of tolerances for each. That is a total of nine parameters stored for each servo. Galvanometers require basically two parameters with two sets of tolerances for each, for a total of six parameters. All discrete indicators are tested using computed references. The total floating point parameters for which data will require storage are as follows:

ADI	81 parameters
HSI	21-30 parameters

These parameters would be stored as floating point data.



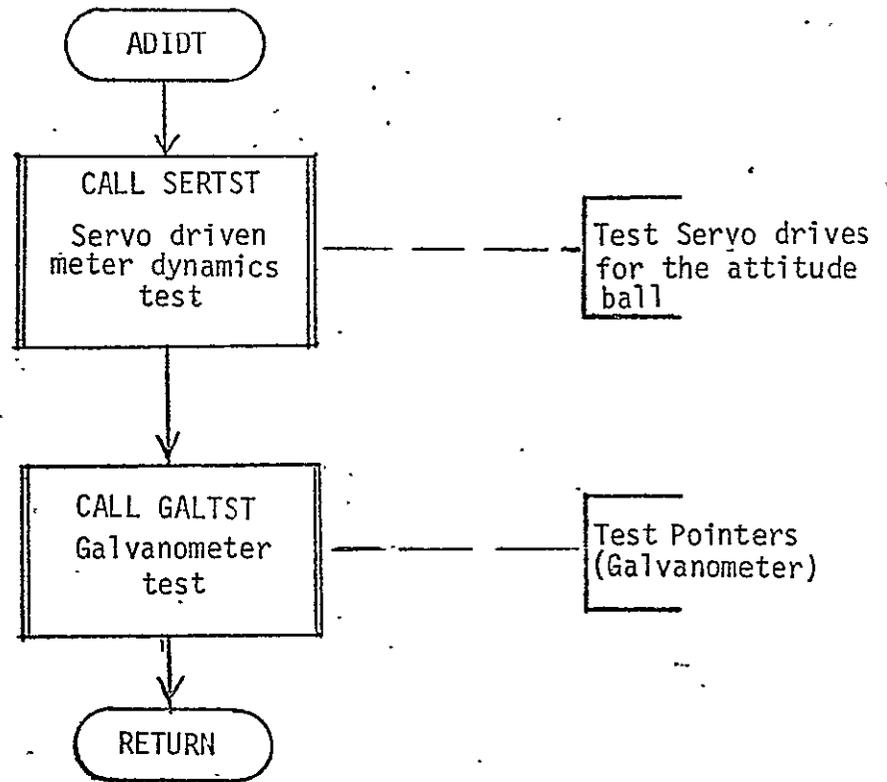


FIGURE 8.3-21 ADI DYNAMIC TEST SOFTWARE FLOW, ADIDT

8-133

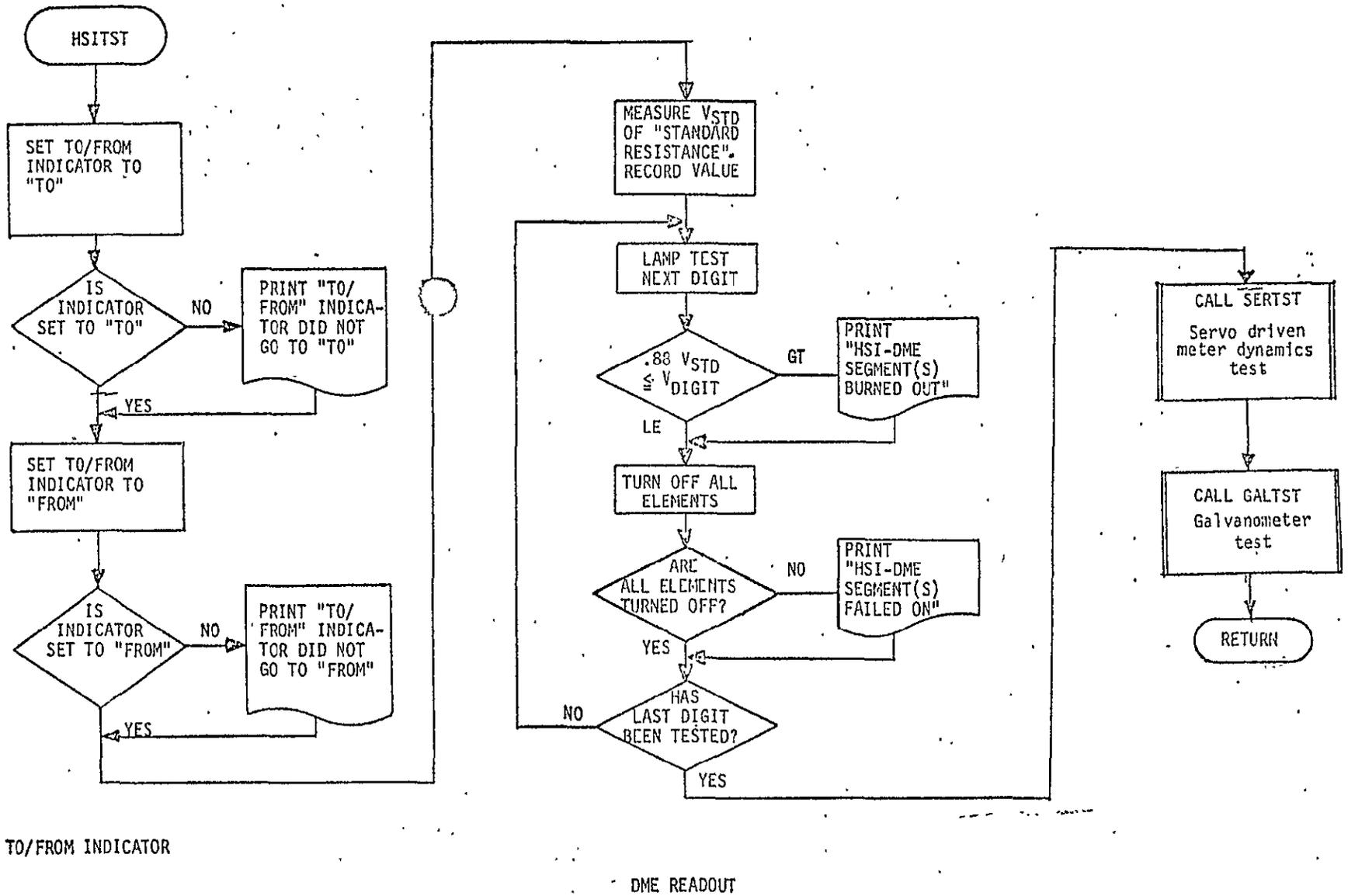


FIGURE 8.3-22 HSI TEST SOFTWARE FLOW, HSITST

### 8.3.5 Lighted Indicators

There are 700 lighted indicators in the Shuttle Mission Simulator that need to be checked for correct operation. This section reviews various techniques to verify their proper operation and recommends one specific approach for use in Shuttle simulators.

#### 8.3.5.1 Description and List of Displays Applicable

The lighted indicator is an individual indicator, an assembly of indicators, a component of a push button switch, or an individual segment of a digital readout. Most indicators are DC, but a few are AC powered. See Figure 8.3-23 for typical assemblies, and Figures 8.3-24, -25, and -26 for typical circuits.

Most indicators are of the incandescent type or of the light emitting diode (LED) type. With an overvoltage application, more light is radiated, but failure occurs much sooner. Incandescent bulb life varies as the ratio of (rated voltage/applied voltage)<sup>12</sup>. Bulb life can be greatly extended by the application of a slightly lower voltage. As shown in Figure 8.3-27, the application of 95% of the rated voltage will extend the life to about 1.85 times the normal life while reducing the candlepower to about 84% of the nominal brilliance.

Electroluminescent (EL) lights are constructed similarly to a capacitor whose plates remain flat. An electrical short may occur where the edges are cut and not properly dressed. Otherwise failure by gradual dimming occurs with age - with the application of a constant voltage and frequency. A general description of lighted indicators is given in Table 8.3-16.

This analysis assumes that at the start of the checkout test, the computer checks the lighted push button switches to verify that they are all in the position which turns on the associated light. Refer to Figure 8.3-26. If any are OFF, the computer controlled manipulator is commanded to turn the proper switches ON. If any such switch does not change state after the man-

# LED DISPLAY DEVICES

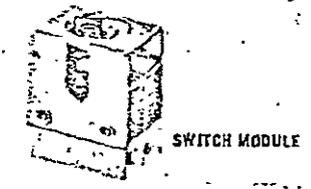
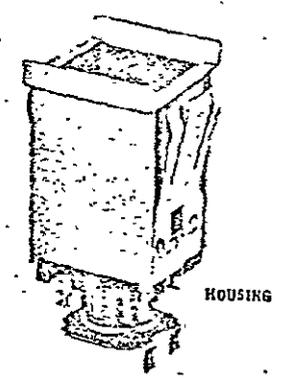
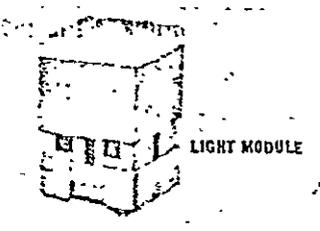
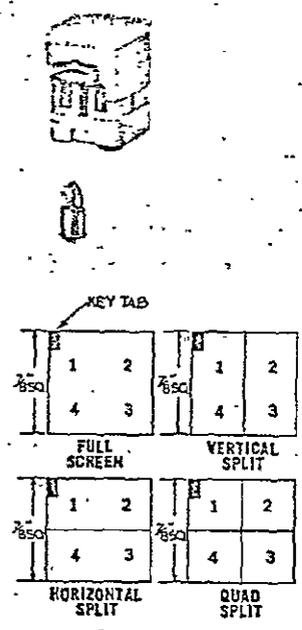
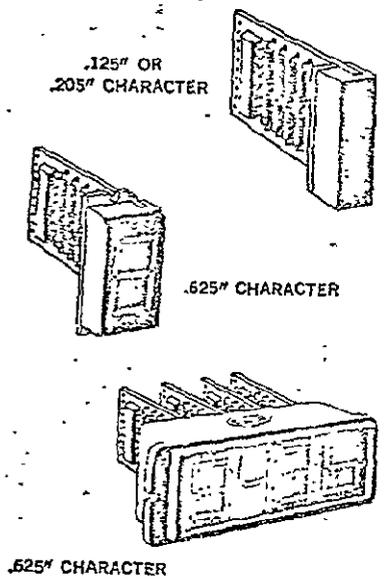


FIGURE 8.3-23. LIGHTED INDICATORS AND PUSH BUTTON SWITCHES

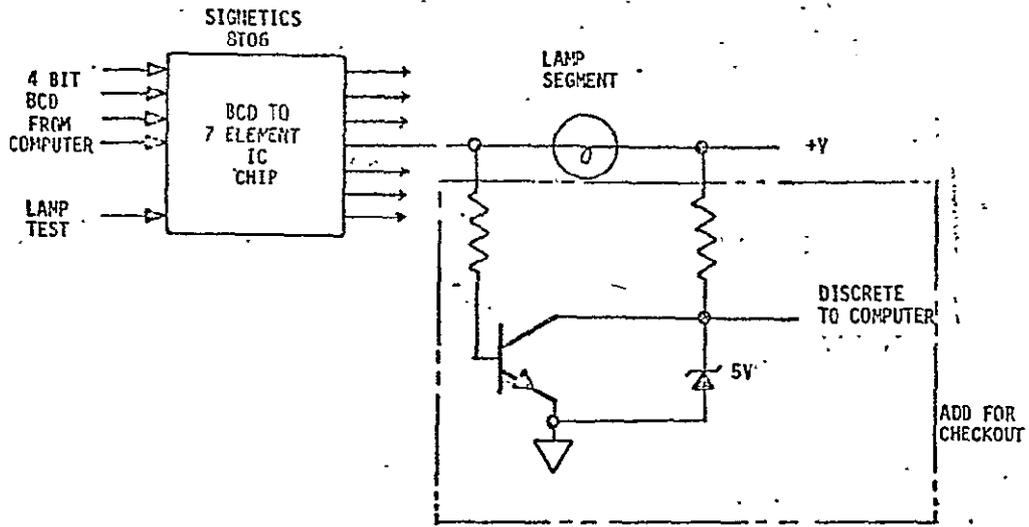


FIGURE 8.3-24 CURRENT SENSING DIAGRAM FOR DC CIRCUITS

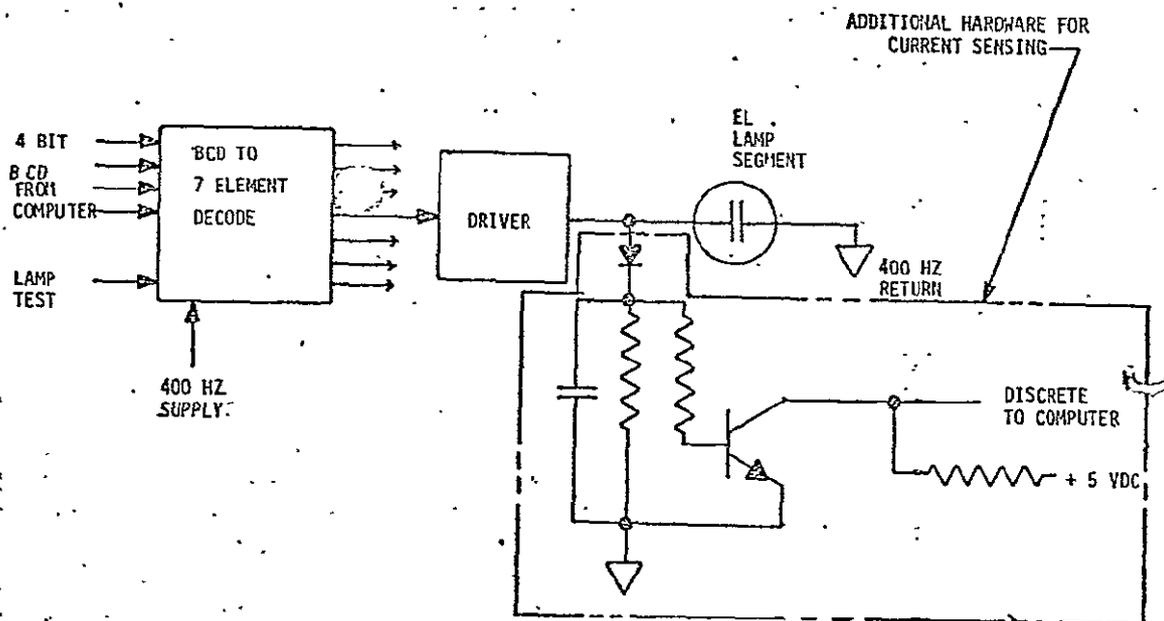


FIGURE 8.3-25 CURRENT SENSING DIAGRAM FOR AC CIRCUITS

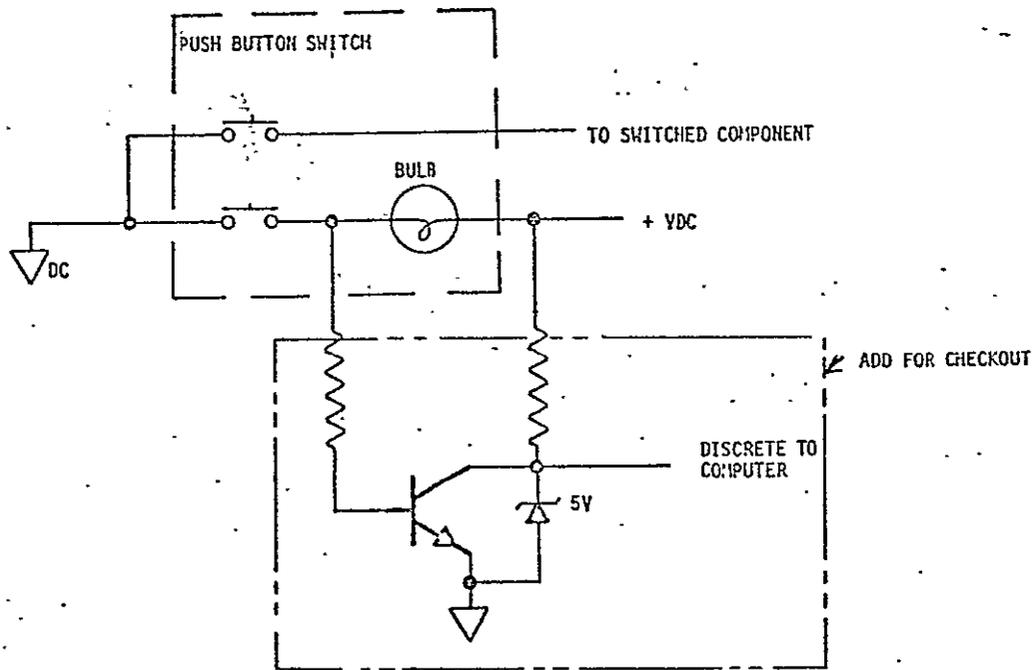
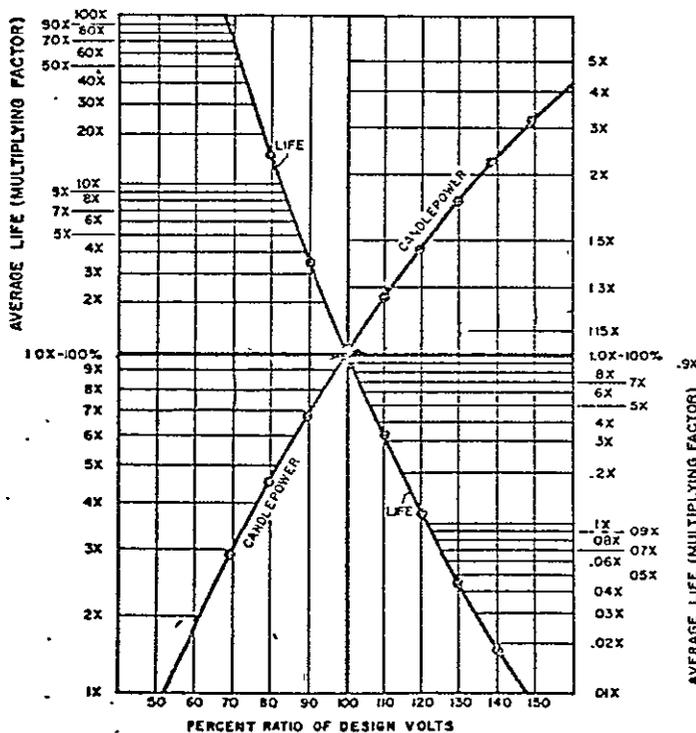


FIGURE 8.3-26 CURRENT SENSING DIAGRAM FOR PUSH BUTTON SWITCH



The graph illustrates the manner in which current, candlepower and life performance varies with percent changes in applied voltage. These values are typical for CM miniature and subminiature lamps.

DERATED M.S.C.P. =

$$\left(\frac{V}{V_1}\right)^{3.5} \times \text{M.S.C.P. AT DESIGN VOLTS}$$

DERATED LIFE =

$$\left(\frac{V_1}{V}\right)^{12} \times \text{LIFE AT DESIGN VOLTS}$$

V = APPLICATION VOLTAGE

V<sub>1</sub> = DESIGN VOLTAGE

FIGURE 8.3-27 LAMP LIFE AND CANDLEPOWER VS. APPLIED VOLTAGE FROM CHICAGO MINIATURE LAMPWORKS

TABLE 8.3-16

LIGHTED INDICATORS

INPUT: + VDC or - VDC  
GROUND (CONTROL SIGNAL)  
0 - 110 VAC, 400 HZ )  
0 - 110 VAC, 60 HZ ) FOR ELECTROLUMINESCENT LIGHTS

OUTPUT: LIGHT                      ULTRAVIOLET RADIATION  
HEAT                                INFRARED RADIATION

LIFE EXPECTANCY:

SWITCHES: 50,000 HR AT RATED LOAD  
INCANDESCENT: 150 TO 100,00 HR

QUANTITY: 400 IN CREW STATION  
300 IN IOS

EXPECTED FAILURE MODES:

OPEN CIRCUIT  
SHORT CIRCUIT  
ELEMENT OF BULB BURNED OUT  
LOOSE BULB OR CONNECTOR  
EL TYPE ONLY - SHORT BETWEEN THE TWO ELEMENTS AT ANY EDGE.  
GRADUAL DIMMING WITH AGE.

CRITICAL MEASUREMENT POINTS:

SOCKET/CONNECTOR/TERMINAL BOARD TERMINALS  
LIGHT ON/OFF

Manipulator turns it ON, a second attempt is made to turn the switch ON. If the second attempt fails, the switch is assumed to have failed and the appropriate message is printed by the computer.

#### 8.3.5.2 Characteristic Parameters

LEDs normally fail in an open circuit mode. Even if the fault is a short circuit, it soon burns into an open circuit - either in the diode or its driver circuit.

Characteristic parameters are simply the light status, ON or OFF.

#### 8.3.5.3 Hardware and Software Requirements

Automated checkout or self test of lighted indicators requires a means for sensing light, heat, current flow or other related physical characteristic in response to a voltage input; and the generation of an electrical response signal indicative of light output, which can be communicated back to a computer for processing. The recommended technique is current sensing.

Figures 8.3-24, -25, and -26 are schematics of how a current flow through various indicators can be sensed. Figure 8.3-28 gives the software flow diagram. Current flows through the transistor when a voltage appears at its base; this voltage appears when the light is turned off and a very small current flows through the bulb and limiting resistor.

The advantages of this recommended technique is that an immediate response and a positive indication are given. It is not affected by stray/adjacent lighting or adjacent heat generators. No warm up or cool down time is required. The circuit checks both the light segment and the driver/supply voltage.

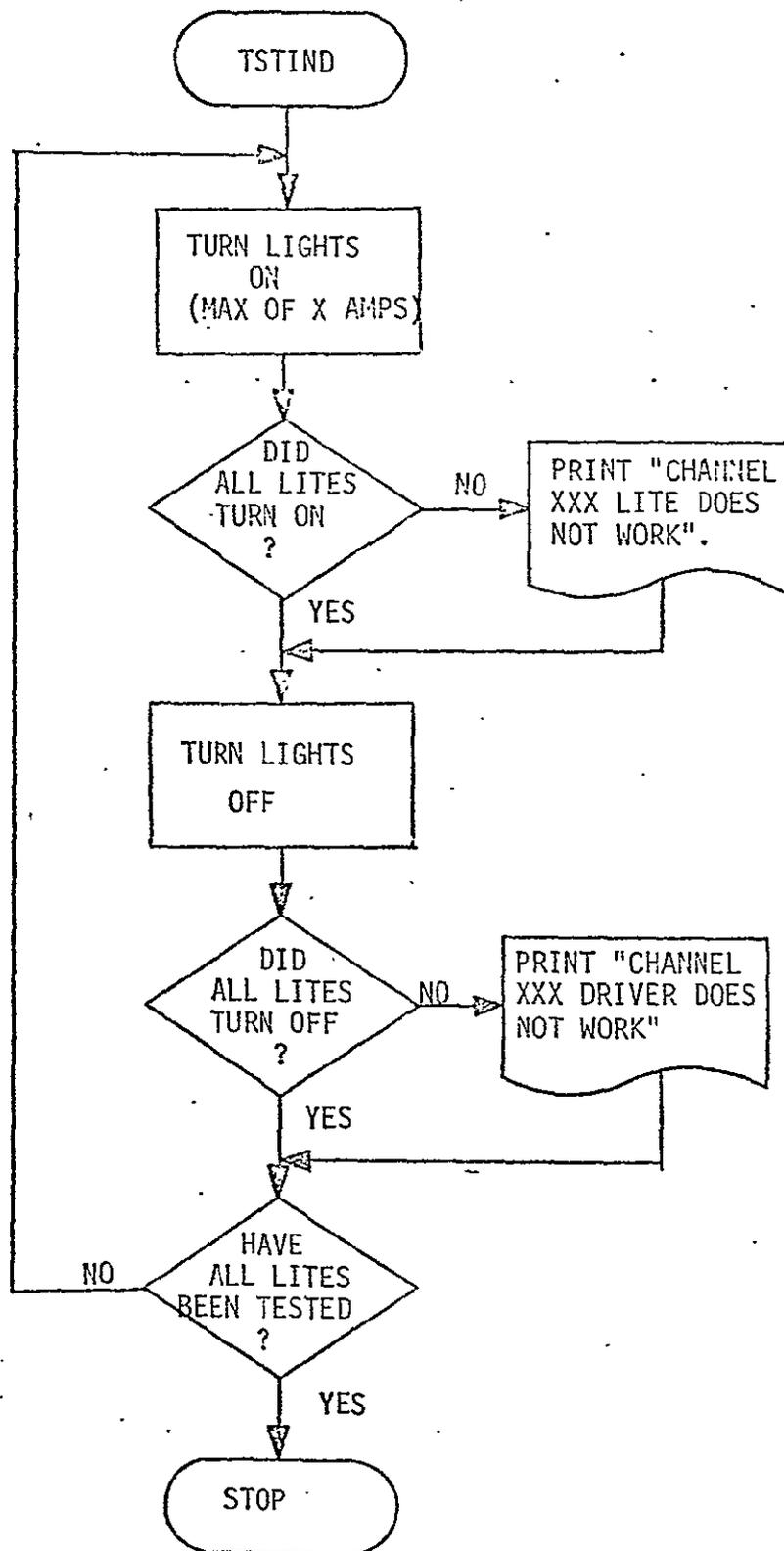


FIGURE 8.3-28 SOFTWARE FUNCTIONAL FLOW DIAGRAM

### 8.3.6 Bilevel Electromechanical Flags

This section describes various techniques for checking the 400 bilevel electromechanical flags in the Shuttle Mission Simulator or other Shuttle simulators. One specific technique is recommended, based upon cost and proof of operation.

#### 8.3.6.1 Description and List of Displays Applicable

Each flag assembly is a two-position indicator, electromechanically actuated. One position shows a grey panel, the other position shows a black and white barber pole pattern. With the application of rated voltage, the flag (drum) is moved to the alternate position. When the voltage is removed, a spring returns the flag to the original position. See Figure 8.3-29 and Table 8.3-17 for physical and electrical characteristics. Internal diodes permit the addition of test circuitry in the Shuttle simulator. Figure 8.3-30 shows the circuit of a typical flag that was used in the Apollo Command Module Procedures Simulator (CMPS).

Flags have a higher reliability than lamps in general. Therefore, they are used as "fail-safe" valve position displays for propulsion systems such as the Reaction Control System.

Some flags have a different construction than shown in Figure 8.3-29. They have a grey panel that drops in front of the barber pole. The grey panel is moved by a solenoid operated slider mechanism. This type has more failures than the drum type.

The following analysis assumes that the correct voltage is applied to the terminals as a result of previously performed checkout and calibration on the rest of the simulator equipment.

#### 8.3.6.2 Failure Modes and Characteristic Parameters

Table 8.3-18 lists the failure modes and related symptoms. The most failures have occurred in the type that utilize a slider mechanism. All listed

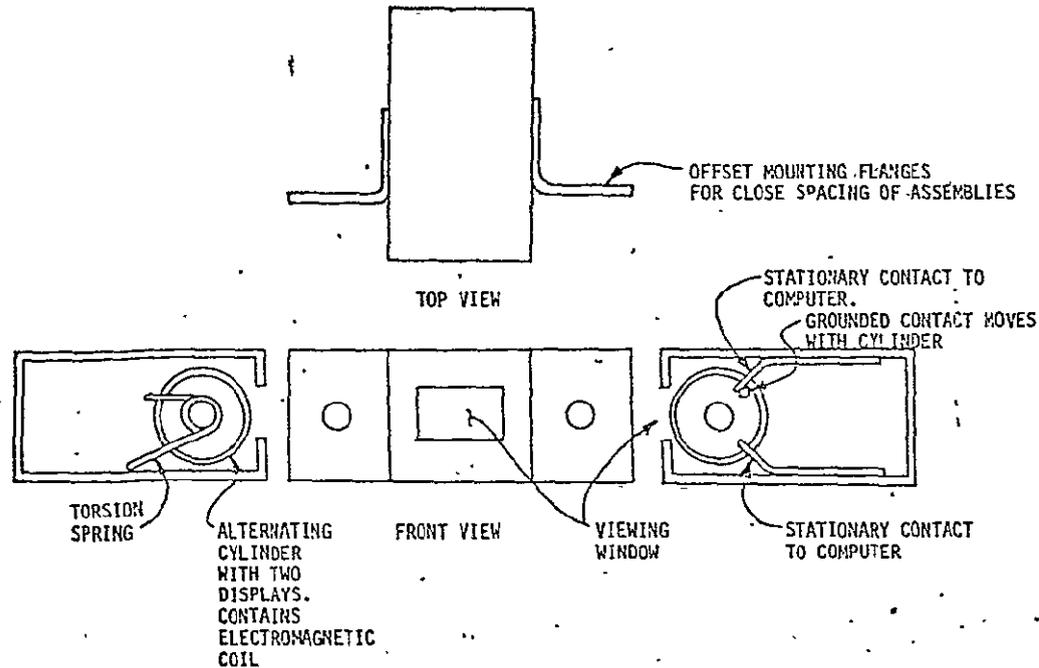


FIGURE 8.3-29 ASSEMBLY OF BILEVEL, ELECTROMECHANICAL FLAG

TABLE 8.3-17 BILEVEL, ELECTROMECHANICAL FLAGS

INPUT: 0 and +28 Vdc

OUTPUT: Flag Movement, Two Positions

REFERENCE CONFIGURATION QUANTITY: 200 in Crew Station  
200 in Instructor/Operator Station

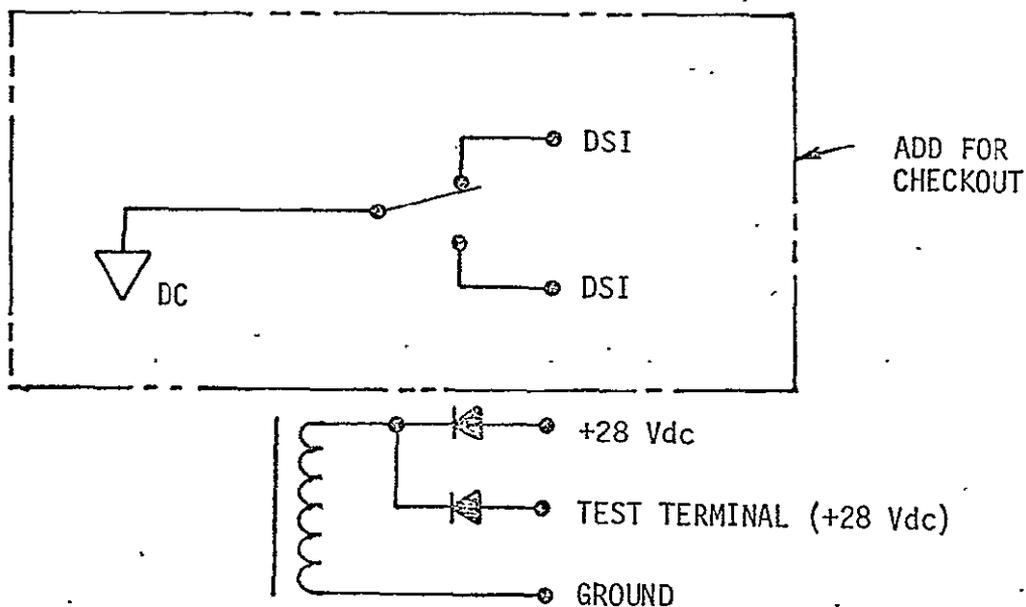
EXPECTED FAILURE MODES:

- Short Circuit
- Open Circuit
- No change due to dried lubricant or broken spring
- Electrical overload may bend or break mechanical part(s).
- Open/Shorted Diode
- Open/Shorted Solenoid Coil

CRITICAL MEASUREMENT POINTS: Input Terminals  
Flag Position

TABLE 8.3-18 FAILURE MODES

MODE BY FAILURE OF COMPONENT	SYMPTOMS	
	EXTERNAL	INTERNAL
DIODE	NO FLAG MOVEMENT	OPEN DIODE
	FLAG MOVEMENT IS UNPREDICTABLE DUE TO UNKNOWN IN EXTERNAL CIRCUIT	SHORTED DIODE - CONTINUITY IN BOTH DIRECTIONS
SOLENOID COIL	NO FLAG MOVEMENT	OPEN COIL
	FLAG MOVEMENT IS UNPREDICTABLE	SHORTED COIL
OPEN CIRCUIT	NO FLAG MOVEMENT	NO CONTINUITY
SHORT CIRCUIT	NO FLAG MOVEMENT	LOWER THAN USUAL RESISTANCE
SPRING	NONE - IF GRAVITY AND SPRING RETURN FLAG TO SAME POSITION. FLAG MIGHT "FLOAT" IF POWER IS OFF AND CREW STATION MAKES STRONG PITCH MOVEMENT.	BROKEN SPRING
SLIDER (ONE TYPE)	NO MOVEMENT	DRIED LUBRICANT, LOOSE PIN



GRIMES 65-0167-1

FIGURE 8.3-30 SCHEMATIC OF ELECTROMECHANICAL FLAG

failure modes and checkout techniques also apply to it, but the drum type is the one that is expected to be used in the simulator. Therefore, the text is directed to the drum type of flag.

#### 8.3.6.3 Hardware and Software Requirements

Automatic checkout or self test of a flag assembly basically requires a means for sensing flag movement in response to voltage input, and the generation of an electrical response signal indicative of flag position, which can be communicated back to a computer for processing.

Techniques evaluated and summarized in Table 8.3-19 included contacts for continuity checks and position potentiometers.

##### Electrical Contacts for Continuity

The flag position is sensed by two continuity pins: one at each of the alternate positions. If the moveable, grounded contact touches the pin at the extreme roll of the drum, a ground signal is sent to the computer. The pins are flexible to reduce shock and to provide tolerance for misalignment. The flexibility also permits wiping action that is necessary to reduce the buildup of high resistance material. Newer alloys have been developed to replace gold as contact plating material. Characteristics or properties are nearly as good as gold. This checkout technique gives reliable position information at a reasonable cost. See Figure 8.3-30.

##### Feedback Signal from a Position Potentiometer

A potentiometer can be mounted inside the case so that the wiper is moved in a fixed ratio to the drum rotation. The voltage from the wiper would be sent to the computer via an analog to digital line. The voltage applied to the potentiometer end terminals,  $+V_{REF}$  and ground or  $-V_{REF}$ , is dictated by the simulator computer analog system and the system designer's choice.

Since the flag is a bilevel device, there is no need to check intermediate flag positions. However, use of the feedback signal as a flag position indication enables implementation of simple steady state testing or of recording of the dynamic response to a step input.

CANDIDATE TECHNIQUES	RELATIVE COST	HARDWARE MODIFICATIONS		SOFTWARE REQUIREMENTS	REMARKS
		MODS TO INDICATORS	ADDITIONAL MODS TO EQUIPMENT		
CURRENT SENSOR	LOW	ADD 2 CONNECTIONS TO EXTERNAL AMPLIFIER	ADD TRANSISTOR AMPLIFIER AND ONE DISCRETE LINE TO COMPUTER FOR EACH INDICATOR	PROCESSING OF DISCRETE DATA	DOES NOT PROVE MOVEMENT
ELECTRICAL CONTACTS	LOW	ADD MOVEABLE CONTACT AND 2 TERMINALS	ADD 2 DISCRETE LINES TO COMPUTER PER INDICATOR		PREFERRED METHOD
POTENTIOMETER	MED	ADD POTENTIOMETER	ADD ONE ANALOG CHANNEL PER INDICATOR	PROCESSING AND EVALUATION OF QUANTITATIVE PARAMETERS	COMPUTER ANALOG INTERFACE MORE COSTLY THAN DISCRETE INTERFACE
TV SCAN	VERY HIGH	NONE	SERVO DRIVE FOR CAMERA(S), TV CAMERA(S), CONTROL CIRCUITRY, VIDEO PROCESSING HARDWARE, ANALOG LINES TO COMPUTER	COMPLEX PROCESSING OF IMAGE DATA	MULTI-INSTRUMENT USAGE LOWERS RELATIVE COST
FLUIDICS	MED	ADD FLUIDIC PORTS, RELIEF	ADD FLIP-FLOPS, TRANSISTORS, AIR SUPPLY SYSTEM, 2 DISCRETES PER FLAG	PROCESSING OF DISCRETE DATA	PACKAGE SIZE QUESTIONABLE. EXTRA MAINTENANCE AND PNEUMATIC HARDWARE REQUIRED.

TABLE 8.3-19 CHARACTERISTICS OF CANDIDATE TECHNIQUES

3) The cost of modifying the flags for either discrete contacts or for potentiometers is about the same. However, the cost of computer analog interface for the potentiometers is considerably higher. Therefore, the preferred method is to modify the flags to add electrical contacts.

The software requirements for flag testing are identical to those for lighted indicators since both are two position or condition devices.

### 8.3.7 CRT Graphics Display

There are five CRT displays on the IOS for providing interactive communications between the HOST and the instructors, and for giving the instructors the capability to monitor visual displays presented to crew members. These CRT's, together with associated alphanumeric and special function keyboard are used by the instructors to insert simulated malfunctions for training purposes and to control the progress and execution of the simulation task.

One instructor has two CRT's to completely monitor performance of the Commander trainee, while the other instructor has two CRT's to monitor Pilot performance. A fifth CRT, centrally located between the two instructors, provides a display of Event Time Monitoring functions. In addition to these five displays, there will be video monitors on the IOS which duplicate the flight graphics displays on the Crew Station panels.

Primary usage of the CRT displays is in presenting alphanumeric data to the instructors. Graphic display information will include bar type meters and a few circular meters. None of these applications require color presentation, and, therefore, a monochromatic system is used for this analysis.

#### 8.3.7.1 Hardware Description

Typical of these types of CRT Display Systems is the ADAGE/400 system, which is being used by Goodyear Aerospace Corporation in the F-15 simulator and by General Dynamics in the F-111 simulator. This display system is a low cost system that provides full general purpose graphics and alphanumeric display capability, using a random draw, stroke writing technique rather than a raster scan technique. Figure 8.3-31 presents a block diagram of the CRT Display and Keyboard System for the IOS using the ADAGE elements. Besides the CRT displays, there are four major elements in the system. They are:

1. The minicomputer Interface. This unit provides timing and formatting services for display data transferred between the display system and the minicomputer. Refresh of display data can be performed directly from the minicomputer through this interface, or the interface can store display information in a buffer memory and provide the necessary control to refresh display 30 or 40 times per second or as desired.

8-148

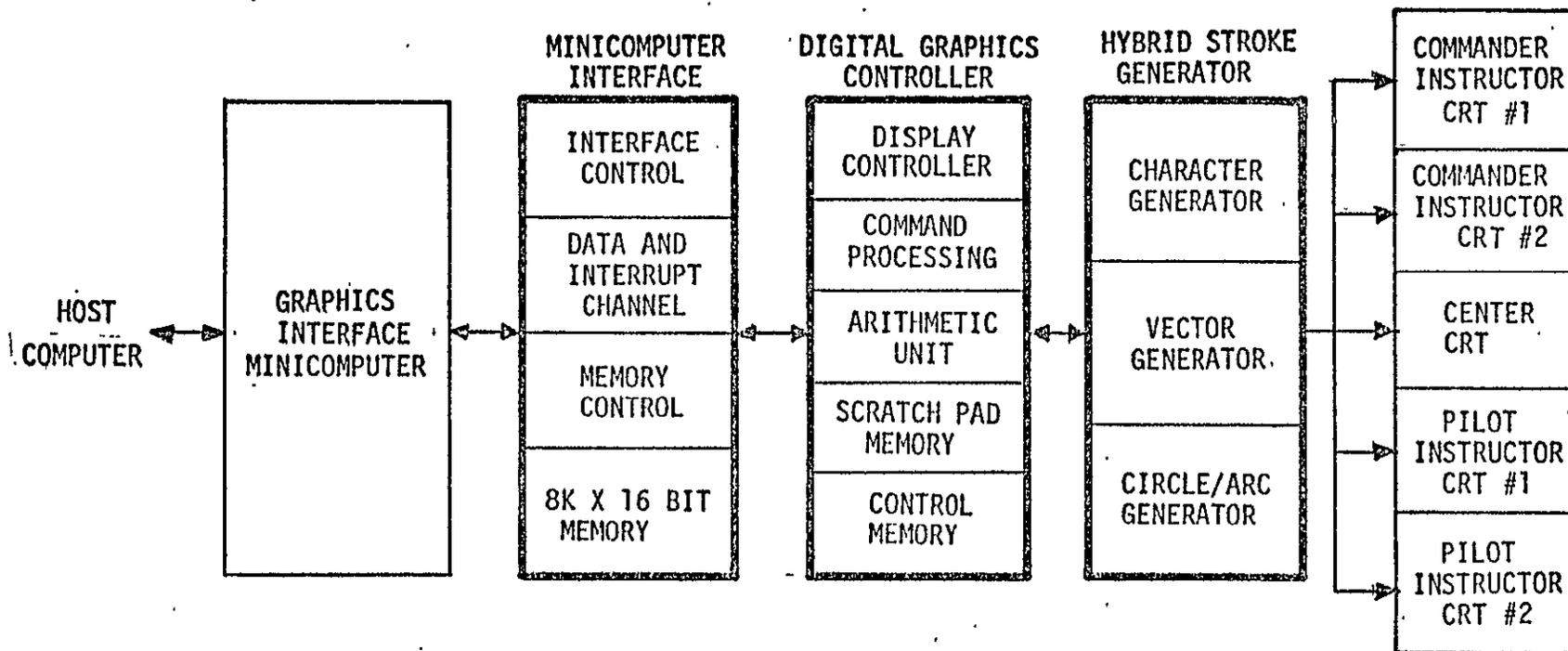


FIGURE 8.3-31 DISPLAY SYSTEM FUNCTIONAL BLOCK DIAGRAM (ADAGE/400)

Including the refresh task in the Minicomputer Interface cuts down the amount of data traffic on the minicomputer I/O ports and therefore frees the minicomputer for communication with other devices.

2. The Digital Graphics Controller. Some of the most advanced technology features of the system are included in this unit in that it is this circuitry that performs hardware transformation for image rotation computations and other perspective related processing.
3. The Hybrid Stroke Generator. This unit contains the necessary hardware to format alphanumeric characters in the desired font. Vector generation as well as circle and arc generation capabilities are included in this element.
4. The Remote Console. This console contains the alphanumeric keyboard, light pen, joy stick, control switches and indicators, and serves as a control panel for manual control and checkout of the display system.

Up to sixteen terminals can be operated simultaneously and independently from a graphics central controller. This controller, which could be any of various minicomputers in the market - such as the PDP-11, in turn is connected to the HOST computer that ultimately determines the contents and format of the data displayed.

#### 8.3.7.2 CRT Graphics Test Techniques

The diagnostic programs supplied by the vendor are intended to determine the overall readiness of the system and, in the event of a failure, to isolate the fault to a group of three or four integrated circuit modules in the display electronics circuitry. Maintainability features of the hardware, together with this diagnostic software capability can achieve a Mean Time To Repair for the display system in the area of ten to fifteen minutes.

The basic self test diagnostic process consists of calling up various test patterns that exercise the character, vector, and circle/arc generation capabilities of the system. Verification of displays obtained is performed by a human observer. The patterns are so designed as to point out subtle faults in the display system components. For example, a long diagonal line is generated by multiple short vectors. A problem in the vector computation or generation circuitry can be isolated to the small group of components involved with the length, orientation, or flicker characteristics of an out of line segment.

The Digital Graphics Controller microprocessor can store in firmware the desired test patterns for display system self test. The same patterns can be issued from the graphics minicomputer in order to test not only the displays, the stroke generators and the microprocessor, but also the refresh circuitry and other elements in the Minicomputer Interface. This capability to generate test patterns at different points in the display string permits the fault isolation process to be conducted efficiently using the resources for automatic test control contained in the basic system.

The test software and hardware described below are offered by some display system manufacturers as part of their basic system configuration, and are additional features offered by other manufacturers. Procurement of interactive CRT display systems for Shuttle simulators should emphasize the inclusion of self test hardware and software that achieve the now realistic goal of ten to fifteen minutes MTTR.

### 8.3.7.3 CRT Graphics Test Software Requirements

Typically, using the ADAGE/400 system for reference, the following elements of test software are found to make up the self test software package. It should be noted that this software is intended to isolate a fault to a removable circuit board in the card chassis. There are seven boards, and seven corresponding groups of tests as follows:

1. Minicomputer Interface Test. This package of test software verifies that the minicomputer is capable of establishing a communications link with the display system, and that address, command, and display data can be transferred correctly in both directions.

2. Digital Graphics Controller display and command processing test. This software verifies correct interpretation of display and command data by the logic circuits in the host board of the Digital Graphics Controller. Verification is performed by the minicomputer based on messages returned from the DGC.
3. DGC Arithmetic Unit test. The computing capability of the DGC micro-processor is tested by this program by commanding execution of a sequence of instructions that verify this capability thoroughly. A faulty result would point to a fault within the DGC Arithmetic Unit board.
4. DGC Memory Test. All the memory locations in the DGC scratch pad memory are written into and read out to verify bit by bit integrity. The Read Only Memory of the microprocessor used for special test pattern generation is also verified from the minicomputer by this test.
5. Character Generator Test. This software verifies the capabilities of the character generation circuitry, by commanding displays of alphanumeric information and verifying responses from the Hybrid Stroke Generator. It should be pointed out, however, that this and the following two tests intended to check the HSG cannot perform a complete automatic self test. Human observation of character, vector, and circle/arc displays is required to fully verify this HSG hardware.
6. Vector Generator and Circle/Arc Generator Test. This test causes random drawing of vectors, circles, and arcs and checks for specific responses from the HSG, and from the human observer prior to advancing the test or issuing an error message.
7. Stroke Generator Conversion Board Test. These tests verify adequacy of digital/analog conversion circuitry in the HSG.

The above tests are included in the ADAGE supplied test software package which is made up of the HCIT1 programs that perform test number 1 above; the DGCRT programs that include tests 2, 3, and 4; and DIXLD that perform tests 5, 6, and 7. This package does include the capability to isolate to a group of IC's within a board,

However, if a board is defined as an LRU, that level of isolation becomes more stringent than required to meet the basic fault isolation objectives stated for the study.

It is reasonable to specify that a CRT graphics system obtained from competitive suppliers include the self test capability described above.

### 8.3.8 Controls Overview

) The function of a control device is to transform a mechanical input such as position, motion, or force into an electrical signal. The mechanical input can come from a controlling system or from a human operator; in the case of the simulator, this input comes from the trainee. The electrical signal output is routed to the computer as an input to simulation sequence control, or math model processing. The type of controls of concern to the study are discrete devices such as toggle and pushbutton switches, or analog devices such as potentiometers or synchro/resolver control networks connected to hand or foot activated controls such as hand controllers or rudder pedals.

#### 8.3.8.1 Control Device Self Test Considerations

Using the stimulus/response approach to self test of controllers, it becomes evident that some automatic means of control activation and control signal monitoring are required to test these control devices. The monitoring function poses absolutely no problem as the electrical signals output from control devices are routed to the computer for normal operation of the simulator. Where the real problem arises is in implementing an automatically controlled source of mechanical motion to be applied to the device being tested. It should be remembered that control devices are the command link between man and computer. The automatic actuator, then, has to take the role of the human operator and do so without interfering with normal operation when the human is at the controls. Fidelity of simulation is of primary concern in training simulators and the introduction of automatic control actuators can impair this fidelity, and therefore reduce the operational effectiveness of the system. Therefore, any method used to automatically actuate simulator controls should meet the following criteria:

1. Automatic control actuation hardware should not affect the feel of the control device, or change the amount of force that the human operator must use in actuating that control.
2. Installation of automatic control actuation hardware should not impair the aesthetic fidelity of the crew station. This means that no levers,

linkages, cams, hinges, or bulges should be visible to the crewmen if these levers, linkages, etc. have been added to the simulator for automatic control actuation purposes and are not part of the operational vehicle crew station.

#### 8.3.8.2 Control Device Self Test Implementation

The main concern in implementing the control device self test capability then is to assure that the method used for control actuation does not interfere with the positive training aspects of the simulator. As mentioned above, there are discrete output and analog output control devices. The method of actuation for each type can be radically different.

In the discrete output case, all that is needed is some non-interfering means to move the control handle, lever, or button to its extreme positions, without regard as to the amount of force needed or the behavior of the control in between extremes. Toggle and pushbutton switches, as well as two position control handles and levers, can be actuated by individual electromechanical actuators, such as solenoids, or by a single actuator that drives multiple control devices while the actuator is driven by a single motor or electromechanical driver. A technique for multiple control device activation from a single actuator is illustrated by the movable panels or tumblers shown for toggle switch actuation in Section 8.3.9 .

Analog output control devices pose a calibration test problem in that the output must be proportional to the mechanical action induced, and this proportionality must be maintained within design specifications of the corresponding flight component. In order to assure proper calibration, some standard must be used in applying the amount of force or displacement desired on the control device. This standard can be a properly designed, tested, and calibrated servo driven actuator. Commands from the test control computer to this servo actuator can cause the actuator to apply a predetermined amount of force or displacement to the control device. The resulting output signal can then be measured and compared with the calibration data for that control device. The entire range of the control device can then be tested, not only as to the ability of the device to respond to the input motion, but also to verify that the resulting signal is of the proper magnitude and electrical characteristics.

### 8.3.9 Toggle Switches

) This section describes various techniques for checking the 600 toggle switches in the Shuttle Mission Simulator or other Shuttle simulators. One specific technique is recommended, based upon cost and proof of functional operation.

#### 8.3.9.1 Description and Type of Functions Applicable

The toggle switches are two-pole Microswitch 2TL1, two or three position switches. The reference configuration switches have one OFF position. One pole provides the ON position(s) or data bit input(s) (DBI): i.e., ground or open circuit to the computer. The other pole, if used, provides switching for IOS panel components or peripheral circuits.

Typical spacecraft functions that are switched by the toggles are solenoid operated fuel/oxidizer valves, power bus, audio and navigation signals, tank heaters and fans.

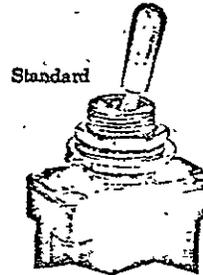
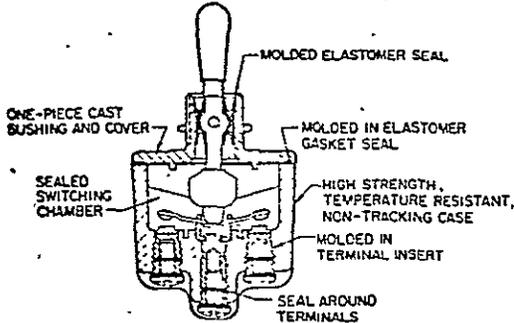
The switch contacts are silver cadmium oxide, step terminal construction, non-shorting, snap action, either maintained or momentary position, toggle action, and sealed. See Figure 8.3-32 for mechanical and electrical information that has been excerpted from Microswitch catalogs. Table 8.3-20 contains simulator related information.

#### 8.3.9.2 Failure Mode Analysis

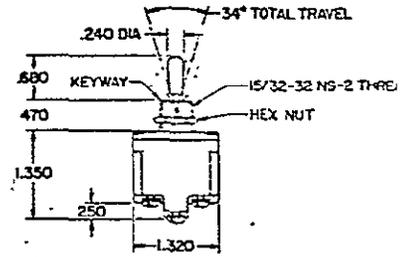
Switch life rating is normally stated for switching 28 VDC resistance loads at sea level, at 50,000 feet altitude, and also for "dry" switching. The current rating for a given switch increases as DC voltage decreases. The switch capability for DC "lamp" loads is lower than for "resistive" loads because of the high inrush currents when the incandescent bulb is cool. Inductive loads tend to burn the contacts by prolonging the arc when the circuit is broken by the switch. Typical DC ratings are 20 amps resistive load, 15 amps inductive load. Altitude (reduced air pressure) in non-sealed switches reduces the density of the air which is an insulator. This reduced insulation increases the burning of the contacts for inductive loads. Typical ratings at equal voltages are 3 amps inductive at sea level, 2-1/2 amps at 50,000 feet.

# TYPE "TL" LEVER ACTUATED TOGGLE SWITCHES

## CONSTRUCTION



## STANDARD TOGGLE LEVER



*Standard* toggle lever operates on a direct action spring loaded toggle mechanism to provide excellent tactile feedback in both the momentary and maintained toggle positions. The toggle lever is approximately 5/8-inch long and has a non-glare matte nickel plated finish.

- ENVIRONMENT-PROOF SEALING
- 1-, 2-, AND 4-POLE CIRCUITRY
- STANDARD AND PULL-TO-UNLOCK TOGGLE LEVERS
- 2- AND 3-POSITION, MAINTAINED AND MOMENTARY TOGGLE ACTION

Type "TL" toggle switches are the most versatile line of toggle switches offered by MICRO SWITCH to the military aerospace and commercial aircraft industries. Versatility is accomplished by combining a choice of toggle levers with two- and three-position maintained and momentary toggle action, and 1-, 2- and 4-pole circuitry. The high strength, temperature resistant, non-tracking case material, silver cadmium oxide contacts and the step terminal construction assure electrical stability and non-shorting wiring connections. Operating temperature range is from  $-85^{\circ}$  to  $+160^{\circ}$ F. TL toggle switches meet military Specification Number MIL-S-3950 (Immersion).

Weight: One Pole—1.44 oz.

Two Pole—1.92 oz.

FIGURE 8.3-32 MECHANICAL DATA FOR 2TL1 SWITCHES

TABLE 8.3-20 TOGGLE SWITCHES

Input: 0 VDC (ground)  
28 VDC

Output: Input voltage  
Open Circuit

Reference Configuration Quantity: 250 two position 2 pole  
350 three position 2 pole

Contact resistance: About .010 ohms, initially

Design Life: 25,000 cycles

Expected Life:	Dry	} Data Not Available
	Load	

Expected Failure Modes: excessive resistance  
open circuit (broken contact supports)  
closed circuit (welded contacts)  
mechanical (detent spring)

---

Critical Measurement Points: Input/Output Terminals

DC voltages shorten life more than AC voltages of equal magnitude. Typical 115 volt, resistive ratings are 3/4 amp DC and 15 amp AC. DC currents transfer the metal plating on the contacts in one direction only, whereas AC currents transfer the plating back and forth between contacts because of positive and negative voltages. The AC circuit is also broken at various voltage levels. The one-way DC transfer builds metal on one contact and causes pits on the other contact. This increases resistance, which increases the temperature, which in turn increases the rate of metal transfer. When the temperature of touching contacts becomes high enough, the contacts weld together. This welding temperature is usually lower than that of the pure element contact because an alloy is now present. This lower "eutectic" (alloy) temperature varies with the metal elements present and the ratio of each. In some cases the contacts could vaporize if they are separated while the contacts are hot.

In cases where only small currents flow, the heat produced ( $.239 I^2 R t$  calories, or .239 times current squared multiplied by the contact resistance, and time in seconds) may never be enough to cause welding. However, in computer bilevel circuits, the IR drop (voltage drop as a product of current and resistance) may be great enough that the computer will never recognize a change in switch position. When the switch switches between ground and open circuit the computer input line voltage should alternate between ground and  $V^+$ . If the voltage drop across the switch contacts raises the "ground" voltage above the computer's threshold voltage (about 1 volt), the boolean never changes state. At this point the switch is useless in a bilevel circuit.

Other factors that raise the voltage level in bilevel circuits are diodes for BCD encoding electromagnetic noise, the IR drop across connector pins, and wire resistance. Therefore, the number of ohms required to declare a switch "failed" is somewhat arbitrary. In this case eight ohms is used as the maximum number.

REPRODUCIBILITY OF THE  
ORIGINAL PAGE IS POOR

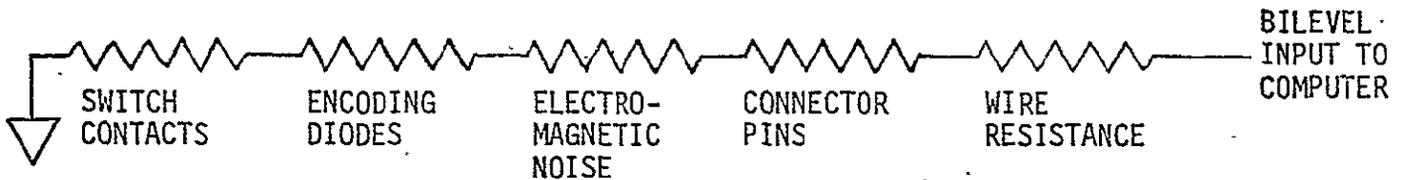


FIGURE 8.3-33 RESISTANCES OF BILEVEL INPUT LINE

When the contacts have changed from .010 ohms to 8 ohms the switch has had either long and/or hard usage. At TTL current levels (.00125 amps), and other normal circuit conditions, the switch is still useful, but it may be approaching the maximum (dry) life expectancy. In higher amperage circuits, heating of the contacts will be noticeable and the resistance will increase rapidly, therefore the life expectancy is nearing its end in either type of usage.

"Dry" switching is the exercise of the switch when no current is flowing. This action mechanically flexes and strains the springs and the metal supports of the contacts. Metallurgists call it "cold working". At failure, the metal cracks and breaks. A switch often will last longer under laboratory "dry switching" test conditions. However, when current is switched, so that temperature rises, an unfavorable change in the heat treatment condition may occur. Therefore the life rating nearest the actual load condition is the life span to expect.

See Table 8.3-21 for a summary of failure modes and related symptoms.

#### 8.3.9.3 Checkout Techniques and Hardware Requirements

Fully automatic checkout or self test of a toggle switch assembly requires the capability to sense open or closed switch contacts in response to

TABLE 8.3-21 FAILURE MODES

MODE	SYMPTOMS	
	EXTERNAL	INTERNAL
Excessive Resistance	"High" resistance across a pair of terminals	Burned and pitted contacts
Closed Circuit	Continuity between one terminal pair in all toggle positions	Welded contacts
Open Circuit	No continuity between terminals in any toggle position	Broken contact supports
Mechanical	Cannot change state	Broken spring or detent

switch toggle movement. However, there is no built-in switch mechanism in the Microswitch 2TL1 (reference configuration) to operate the toggle automatically, i.e., under computer control.

Such a switch would be equivalent to a relay with a mechanical toggle override: no such switch or relay is known to exist anywhere. In addition, no known relays of the shape and size of the 2TL1 switch have the three detent positions like the 350 three-position switches in the reference configuration. This approach is expanded in a later paragraph.

It is not likely that a simulator switch will fail in the "closed" (welded) position as described in paragraph two. Therefore, a possible compromise is the verification of switches in the closed position. As noted, this does not verify that a switch contact can be opened, but this type of failure is a very small probability. It also has the time and reliability disadvantages that a human operator is required to actuate the switches prior to the test. The first two test techniques are discussed with this paragraph's prerequisites and limitations in mind.

### Comparison of Contact Voltage Drop

As noted in paragraph two, there is an IR (voltage) drop across the switch contacts. Contact resistance (and voltage drop across the contacts) increases with usage. This larger resistance, in addition to the other resistance of Figure 8.3-33, raises the voltage at the computer interface toward the computer's threshold voltage of .8 to 2.0 volts for recognition of contact closure.

If a reference voltage, less than the value of the threshold voltage is fed into the  $V_{ref}$  terminal of a comparator such as Fairchild A734 (see Figure 8.3-34 below), and the switch signal is fed into the  $V_{in}$  terminal, early warning of switch failure can be detected by monitoring the output voltage at  $V_{out}$ . For automatic checkout, the switches can be multiplexed into the comparator by MOSFET switches, and  $V_{out}$  can be monitored by the computer. See software flow diagram in Figure 8.3-35.

If the switch has not been actuated to the correct position the comparator will give a "failed" output voltage.

The pole that controls components such as lights or relays may be handled similarly to Figure 8.3-36. However, this can be applied only where the maximum circuit (relay) voltage does not exceed the "highest input voltage range" and the "maximum differential input" voltage ratings of the comparator, unless protective resistors R1, R2, and R3 are added. Both limit ratings are near  $\pm 15V$  for fairchild Semiconductor comparators. The comparator works similarly to the above circuits, and as shown in Figure 8.3-34.

### Analog Measurement of Voltage Level

Figure 8.3-37 shows how a differential amplifier may be used to measure the IR drop across the switch contacts. One circuit is for a switch that controls a ground voltage, the other is for a component (e.g., relay coil) voltage. The component voltage is limited to the common mode voltage of the differential amplifier, which is normally  $\pm 15$  volts DC. Because the switch

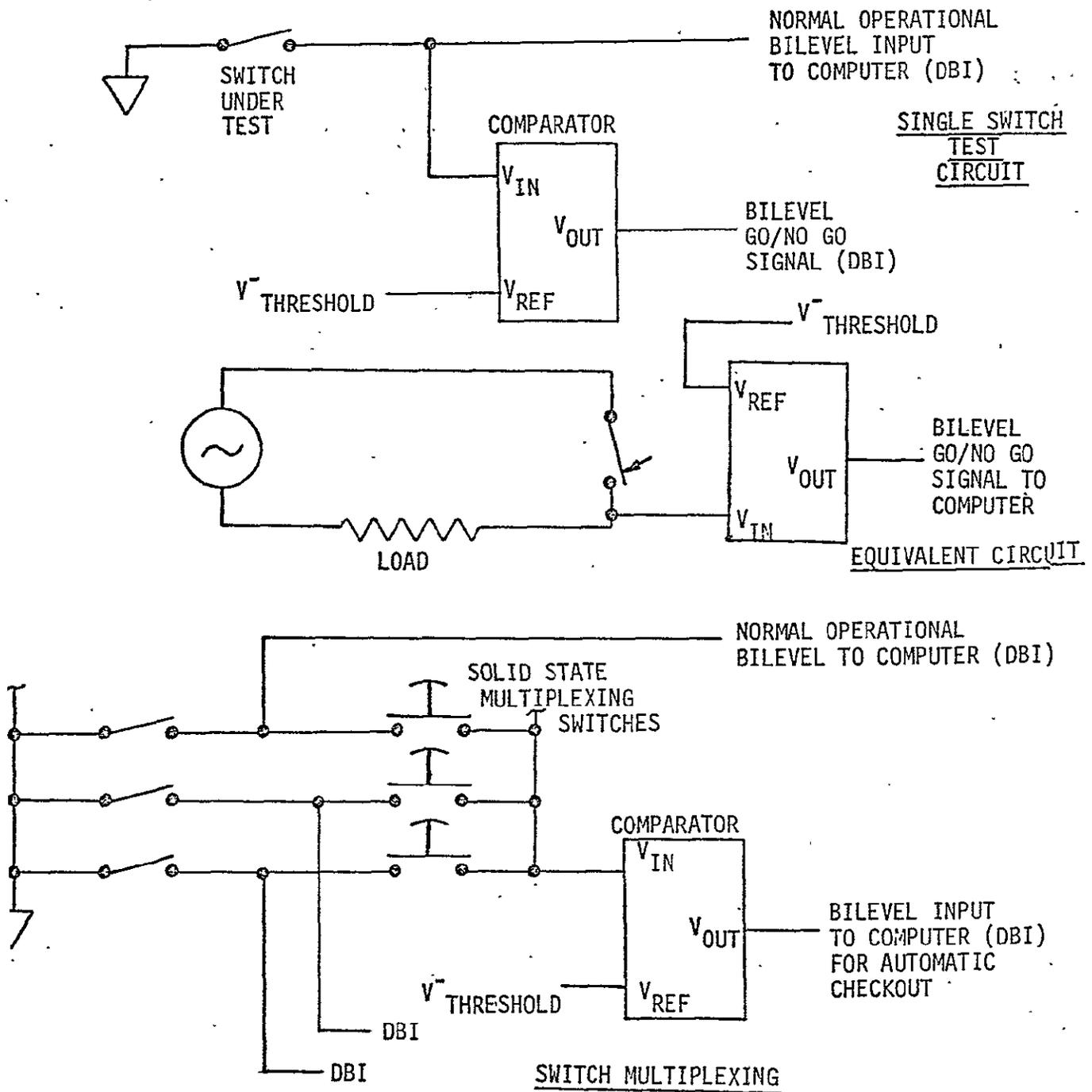


FIGURE 8.3-34 SWITCH CHECKOUT BY A COMPARATOR

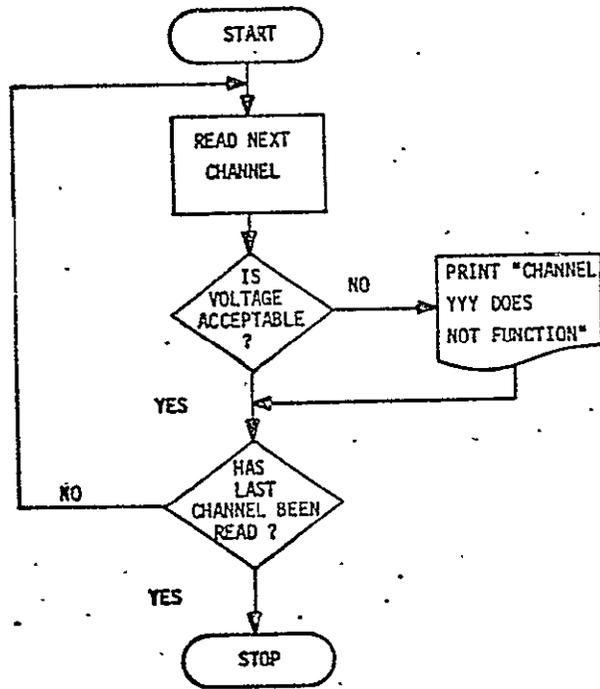
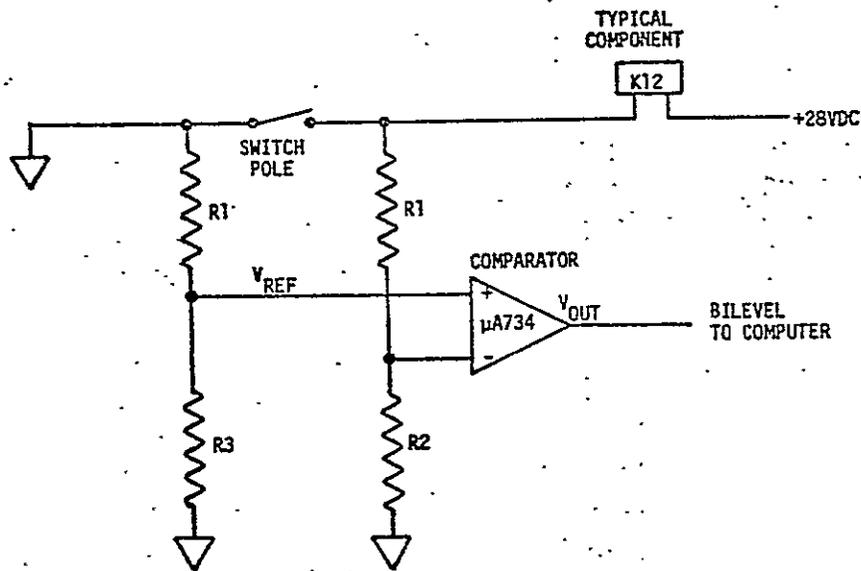


FIGURE 8.3-35 SOFTWARE FUNCTIONAL FLOW DIAGRAM



$R3 < R2$   
 IF SWITCH IS OK,  $V_{IN}^+ < V_{IN}^-$ ,  $V_{OUT} = "0"$  VOLTS  
 IF SWITCH RESISTANCE IS EXCESSIVE,  $V_{IN}^+ \geq V_{IN}^-$ ,  $V_{OUT}$  BECOMES A "1" ( $V^+$  VOLTS)

FIGURE 8.3-36 COMPARATOR CIRCUIT FOR COMPONENT SWITCHING POLE

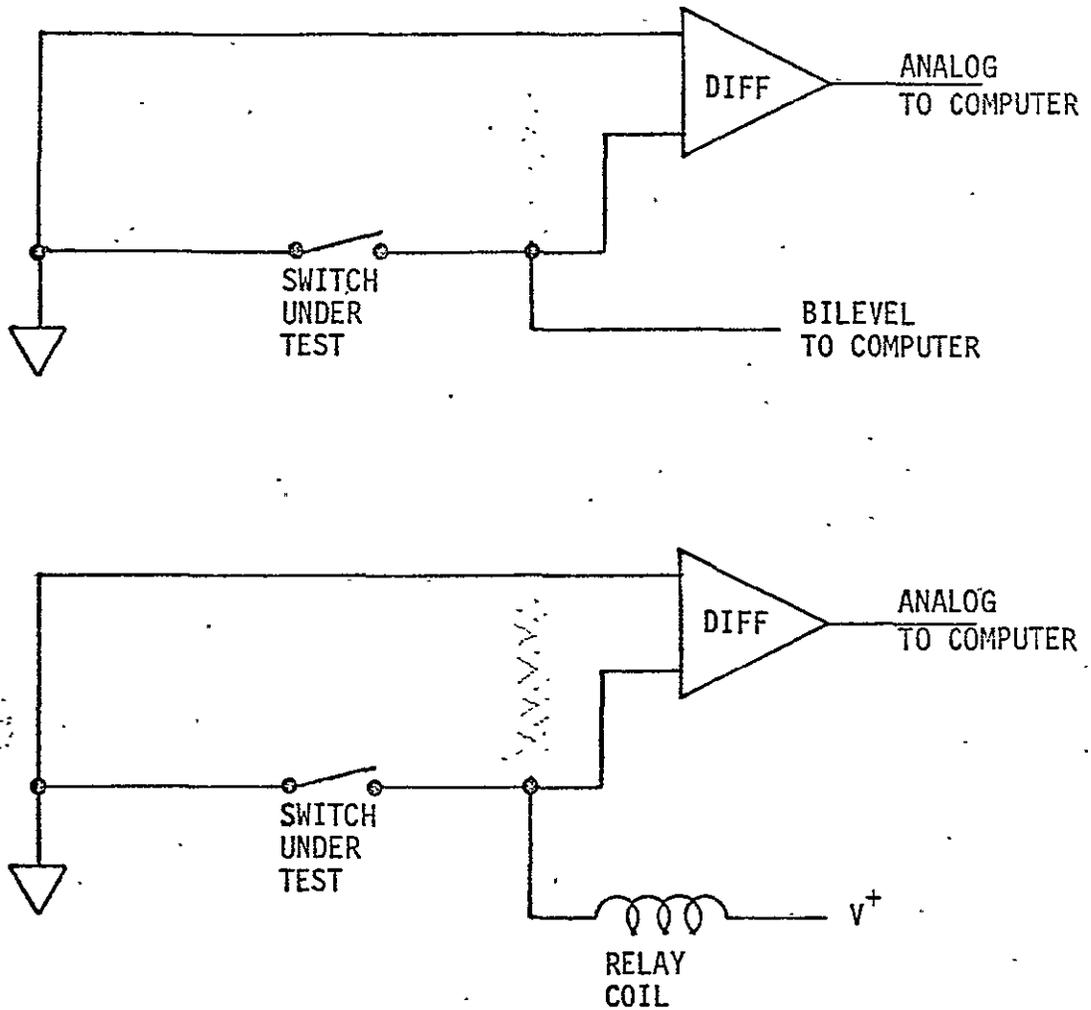


FIGURE 8.3-37 DIRECT MEASUREMENT OF VOLTAGE DROP ACROSS SWITCH CONTACTS

and relay coil may generate destructive voltage spikes, the component voltage may be limited to considerably less than the common mode voltage. However, protective resistors may be added as in Figure 8.3-36, so that +28VDC circuits may be monitored. A word of caution: if the differential amplifier is placed external to the panel on which the switch is mounted it will measure voltage drops of additional resistances as shown in Figure 8.3-33.

#### Switch Modification For Remote Operation

Theoretically, a toggle switch may be modified by adding solenoids to pull or push the contacts to the desired position. The action resembles a latching relay with a mechanical (toggle) override. However, the permanent magnets of a latching relay would not be required for checkout use. The coil can hold the contacts for the short test time required. Two coils per switch are required. The three position switch may have a center OFF detent if both coils repulse the contacts simultaneously. Neither configuration is commercially available, nor is it known whether a manufacturer is willing to try, or has ever tried, to design such a device.

Difficulties have arisen with DC relays that switched 120 VAC on previous simulators. This condition exists with about one percent of the crew station switches. AC voltage spikes were induced on the DC line when the AC line was switched at or near the peak of the AC waveform. These spikes have been falsely sensed as inputs to the computer and to the surrounding digital equipment. This condition could repeat on this type of switch if suitable precautions are not taken.

Care is required that the toggle actuation forces are not unduly altered by the addition of the coils. The modification of a 2TL1 switch to this degree is such a drastic percentage change that the part number would change. The reference configuration states that 2TL1 switches are used. Therefore, this modified switch, even if available and otherwise useable, does not conform to the constraints of the reference configuration and boundary of the study.

### Remote Switch Actuation

Some means of switch actuation by mechanical action are examined below. These techniques do not alter the 2T11 switch. Although preliminary panel layouts are available, many other pertinent details are lacking at this time. Some of these details are panel thickness, switch guards, recess depth of switches, incandescent light piping of nomenclature, spacing between switches, and grouping of two and three position switches. Therefore one cannot fully evaluate the validity of the proposed approaches that are given.

The first approach involves a modified cylinder of suitable length. See Figure 8.3-38. It encloses the switches and wiring, and protrudes through two panel slots per switch in the trough or recess. The "fingers" that push the toggle are stiffened with flutes. The rear of the cylinder is open for access to the switch terminals. The cylinder rotates to push all toggles down to the lower position. The cylinder reverses to move the toggle to the center position, pauses, then raises the toggle levers to the upper position. At each toggle position the computer monitors the corresponding bilevel input lines for sensing the correct switch function: i.e., open circuit or ground. Approximately 100 or more cylinders are required.

The other approach involves the movement of sections of the panel overlay [for lighted nomenclature] by a linkage. For fidelity reasons, the linkage and energizer are behind the panel as shown in Figure 8.3-39. The overlay pieces are reinforced for rigidity and minimum wear on the edges. The overlay would move the switch toggle while the computer monitors the switch output similarly to the modified cylinder as described above.

#### 8.3.9.4 Recommended Checkout Approach

The candidate techniques are compared and summarized in Table 8.3-22. Two techniques have much in common and are close in cost. They are (1) threshold voltage measurement by comparator and (2) contact voltage drop measurement by differential amplifier. The favored technique is the first one because of the simplicity of checking the output of the comparator.

If fully automatic checkout is required, then the "relay-switch" is recommended.

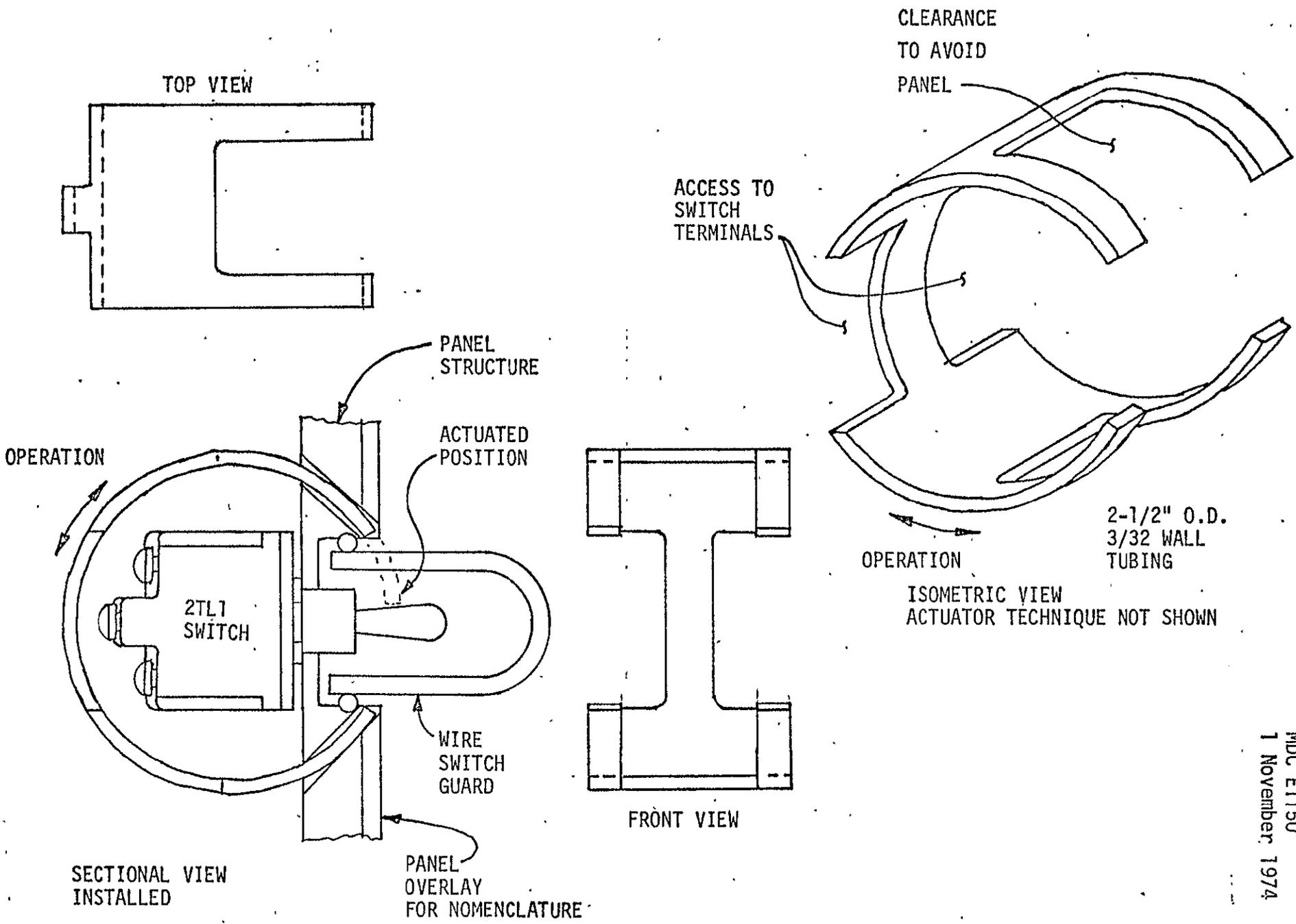


FIGURE 8.3-38 MODIFIED CYLINDER ACTUATOR FOR PAIR OF TOGGLE SWITCHES

The other approach involves the movement of sections of the panel overlay [for lighted nomenclature] by a linkage. For fidelity reasons, the linkage and energizer are behind the panel as shown in Figure 8.3-39. The overlay pieces are reinforced for rigidity and minimum wear on the edges. The overlay would move the switch toggle while the computer monitors the switch output similarly to the modified cylinder as described above.

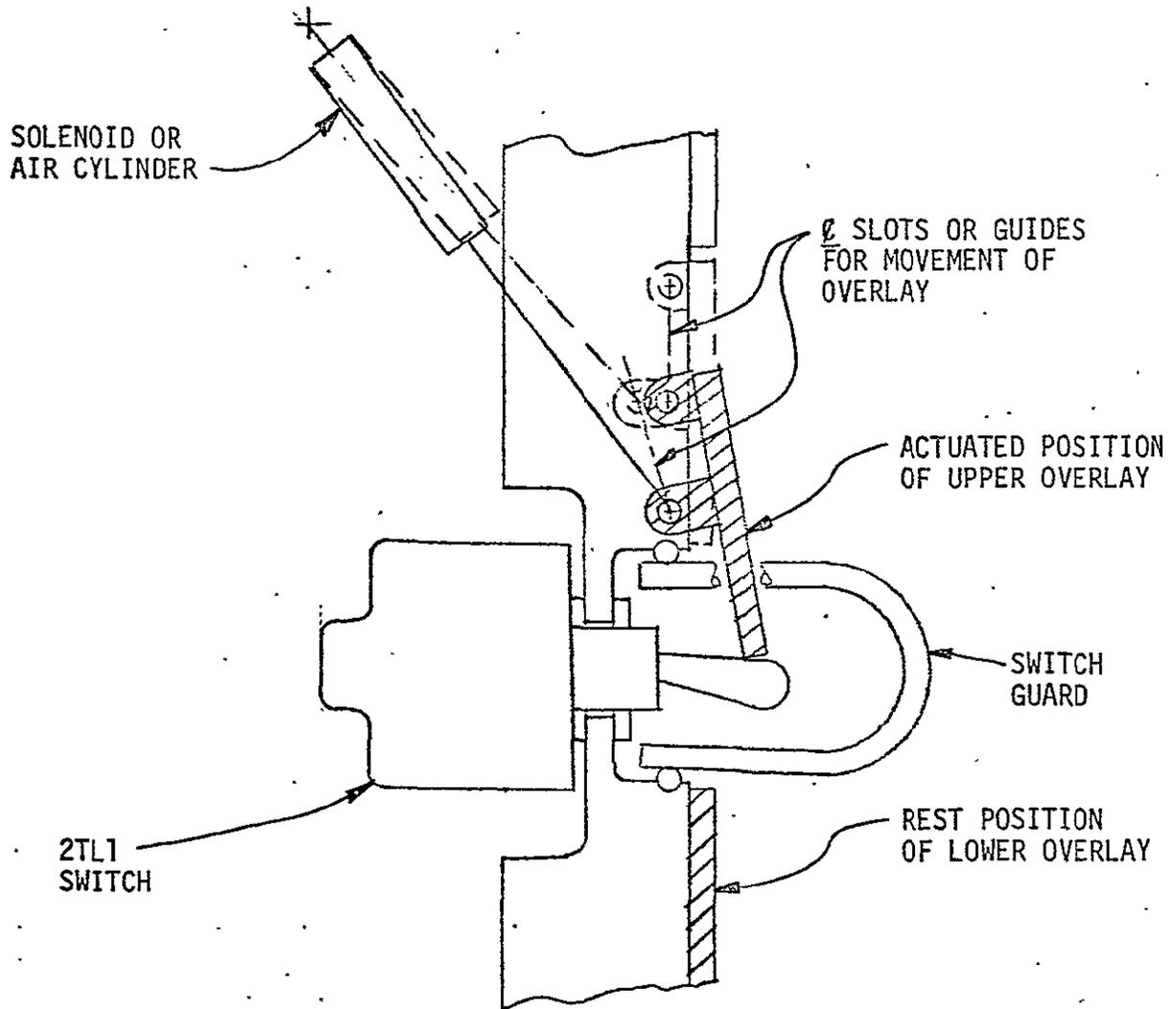


FIGURE 8.3-39 MOVEABLE OVERLAYS TO OPERATE TOGGLE

CANDIDATE TECHNIQUE	RELATIVE COST OF INSTALLATION	HARDWARE MODIFICATIONS		SOFTWARE REQUIREMENTS	REMARKS
		MODS TO SWITCH	ADDITIONAL MODS TO EQUIPMENT		
Voltage Comparator	Lowest	None	Add 1800 multiplexers, one comparator, and one DSI line.	Processing of discrete data	GO-NO GO checking external to computer. Limit changed by hardware adjustment. Human operator to manipulate switches.
Voltage Drop Measurement	Next Lowest	None	Add 1800 multiplexers, one differential amplifier, and analog to digital line.	Processing and evaluation of quantitative parameters.	NO-NO GO checking. Limit changed by programming. Human operator to manipulate switches.
Add two solenoids to switch assembly	Second Highest	Add 2 Solenoids to each model of switch.	Add discrete lines to solenoid coils and 950 multiplexers out of computer.	Processing of discrete data	Equivalent of relay with toggle override. Item not on market.
Actuate Switch Electro-mechanically	Highest	None	Add discrete lines to actuators, and 100 multiplexers out of computer. Design and install several sizes of actuators.		Detail design for clearance and fit for each location.

TABLE 8.3-22 CHARACTERISTICS OF CANDIDATE TECHNIQUES

### 8.3.10 Push Button Switches

There are 75 pushbutton switches in the crew station and another 60 on the Instructor Operator Station (IOS) of the Reference Configuration.

This section describes various techniques for checking these 135 push-button switches in the Shuttle Mission Simulator or other Shuttle simulators. One specific technique is recommended, based upon cost and proof of functional operation.

#### 8.3.10.1 Description and Type of Functions Applicable

The 75 pushbutton switches in the crew station are flight hardware. Those in the IOS are not defined as to the number of poles, accessories, or vendor; however, all switches have only one discrete output to the computer. See Table 8.3-23. Most lighted switches in the crew station (C/S) are alternate action (mechanical latchdown). The lighted section is checked by techniques described in paragraph 8.3.5.

Typical spacecraft functions that are switched by push button switches are computer keyboards, avionics fire protection, and push-to-talk.

#### 8.3.10.2 Failure Mode Analysis

Table 8.3-21 lists the failure modes and related symptoms. Section 8.3.9 discusses in detail the various modes of failure and means of detection. Briefly, the failure modes are excessive resistance, open circuit, welded contacts, and mechanical failures and are similar to toggle switch failures.

#### 8.3.10.3 Self Test Hardware and Software Requirements

Automated checkout or self-test of a pushbutton switch, in general, requires a means for sensing contact closure/opening in response to actuation of the pushbutton.

TABLE 8.3-23 PUSHBUTTON SWITCHES

Input: 0 Vdc (Ground)

Output: Input voltage  
 Open Circuit  
 One pole: one discrete

Reference Configuration Quantity: 75 in crew station, flight hardware  
 60 in IOS, vendor not defined

Contact Resistance (new): less than .025 ohms (Gold Contacts of "SM"  
 Subminiature switch modules)

Design Life: 100,000 or more operations for 2D1 module which includes  
 two "SM" switches

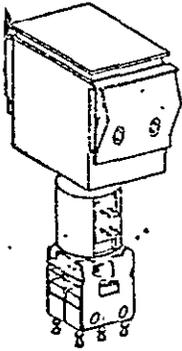
Expected Life: Dry - 10 million operations } 11SM1 module  
 Load - .1 million operations }

Expected Failure Modes: excessive resistance  
 open circuit (broken contact supports)  
 closed circuit (welded contacts)  
 mechanical

Critical Measurement Points: Input/Output Terminals

Parameters of Concern: contact resistance

A pull-in solenoid coil can be added to the pushbutton switch assembly and can be remotely operated to cause the switch to change state. See Figure 8.3-40 for Microswitch Series 2 commercial pushbutton switch with a pull-in coil. In the "relaxed" condition, the contacts of a momentary switch are open; in the "actuated" condition, the contacts are closed. Actuation of the solenoid of an alternate action switch merely causes the contacts to change to the other state. The regular switch circuits to the computer can be used to monitor the bilevel condition of the contacts. See Figure 8.3-41 for a software flow diagram. One possible limitation is that the physical diameter of the solenoid may exceed the switch width or height; for example, keyboard switches. In such a case, a smaller diameter, but longer coil is required to achieve the necessary actuation force.



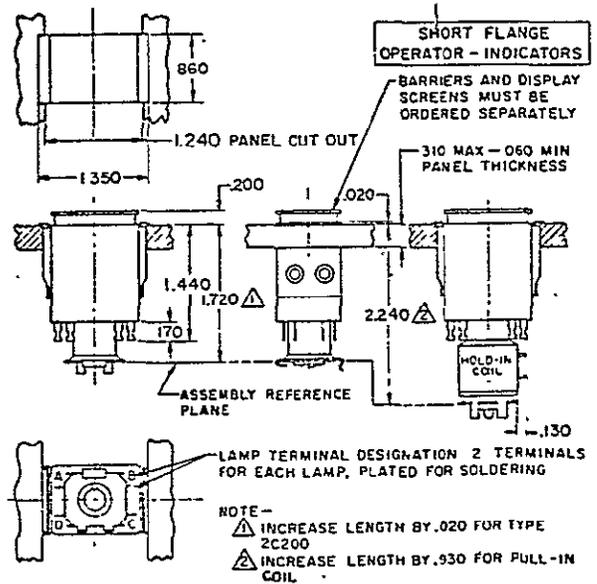
**HOUSINGS WITH FLANGES ON SHORT SIDES**

These housings have spring mounting clips ready-attached to the shorter sides.

**Catalog Listings**

Short Flange Operator-Indicator Housing Type	No of Sockets	Type 2C	Type 2C200
No Coil	2 (A & C)	2C10	2C210
	4 (A, B, C & D)	2C12	2C212
6 vdc Coil (Hold-in)	2 (A & C)	2C22	2C222
	4 (A, B, C & D)	2C24	2C224
28 vdc Coil (Hold-in)	2 (A & C)	2C34	2C234
	4 (A, B, C & D)	2C36	2C236
48 vdc Coil (Hold-in)	2 (A & C)	2C46	2C246
	4 (A, B, C & D)	2C48	2C248
28 vdc Coil* (Pull-in)	4 (A, B, C & D)	2C69	2C269

\*Also available with flanges on all four sides - 2C77 (Type 2C) and 2C277 (Type 2C200)



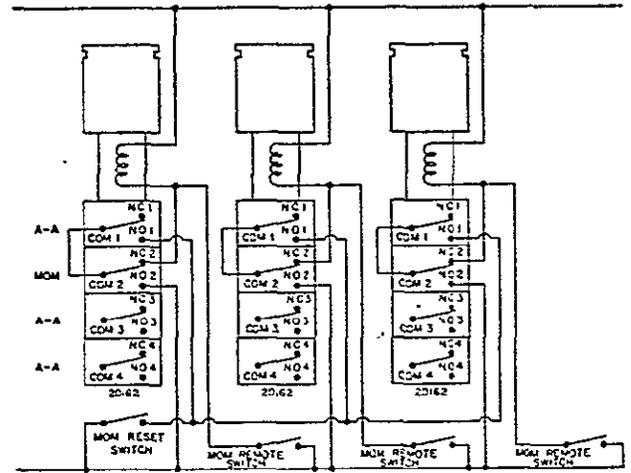
**PULL-IN COIL ELECTRICAL DATA**  
(available in Flange mount only)

Pull-in coil equipped operator-indicators enable electromagnetic actuated, maintained and released contacts. Energizing the coil pulls in the operating plunger, causing circuit transfer in the switching unit. At 28 vdc, 2.2 amps. (approx) is required during actuation (for about 20 MS or less) and only .11 amp. (approx) during hold-in. (Recommended for operation at 85 to 100% of rated voltage.)

Switch units that can be used are "TB", "PL" and all "SM" types except low force and 4-pole alternate-action designs. See Pages 12-13. Maximum ambient temperature limits are +120°F (1 ~.2 pole momentary-action), +110°F (3-pole momentary-action), +100°F (4-pole momentary-action) and +90°F (all alternate-action or momentary/alternate-action) switch units.

**ELECTRICAL BAIL**

The circuit at right shows how any number of Catalog Listing 2D162 can be operated in one-by-one sequence. Each actuation releases previously actuated switch.



Power is required only during the reset pulse. (Reset can also be accomplished by depressing previously actuated screen.)

**ELECTRICAL RATING**

**Silver Contacts**

30 vdc - 5 amps res. sea level or 50,000 ft., 3 amps ind. sea level, 2.5 amps ind. 50,000 ft., Max. inrush, 24 amps.

UL and CSA listing for basic switch:  
125 or 250 vac . . . . 5 amps

**Gold Contacts**

30 vdc - 0.5 amp ind., 1 amp res. sea level and 50,000 ft., 2 amps max. inrush.

FIGURE 8.3-40 SELECTED VENDOR DATA ON MICROSWITCH SERIES 2 PUSHBUTTON SWITCH

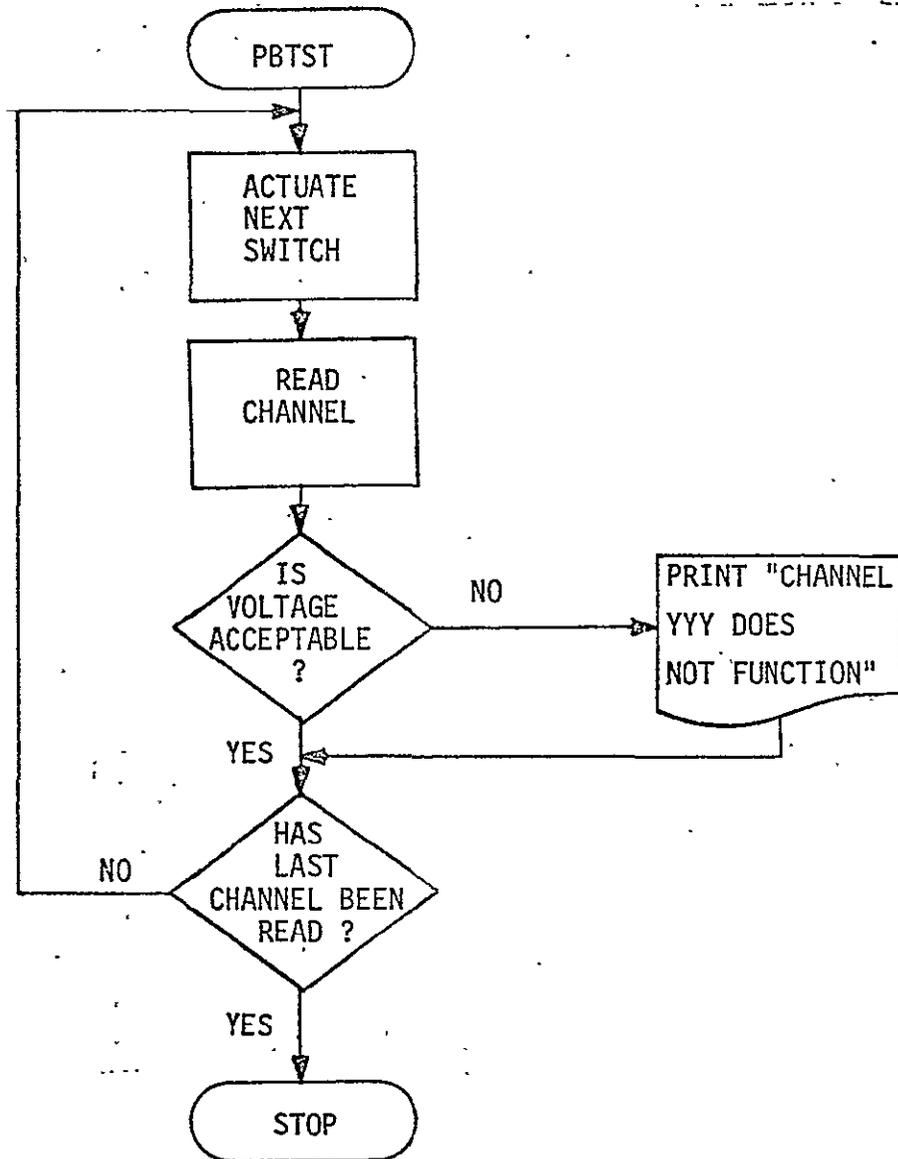


FIGURE 8.3-41 SOFTWARE FUNCTIONAL FLOW DIAGRAM

### 8.3.11 Rotary Switches and Potentiometers

There are 15 rotary switches and 55 potentiometers in the Crew Station and IOS of the Reference Simulator Configuration. This section discusses various techniques analyzed for checkout of these components. As stated in Section 8.3.9, the main consideration in defining self test techniques for control devices is the method of automatic mechanic actuation of the device. Rotary switches and potentiometers are treated here together because both of these types of components require circular, finger-compatible motion in order to produce the mechanized actuation required for self test, and therefore, definition of self test techniques applicable to one type of component is useful to the other type.

#### 8.3.11.1 Description and Applications

The rotary switch has three to nine positions and non-shorting contacts. See Figure 8.3-42 for typical commercial rotary switches and Table 8.3-24 for simulator related data. Data on the flight hardware rotary switches are not available but the commercial switches of Figure 8.3-43 have many of the expected features.

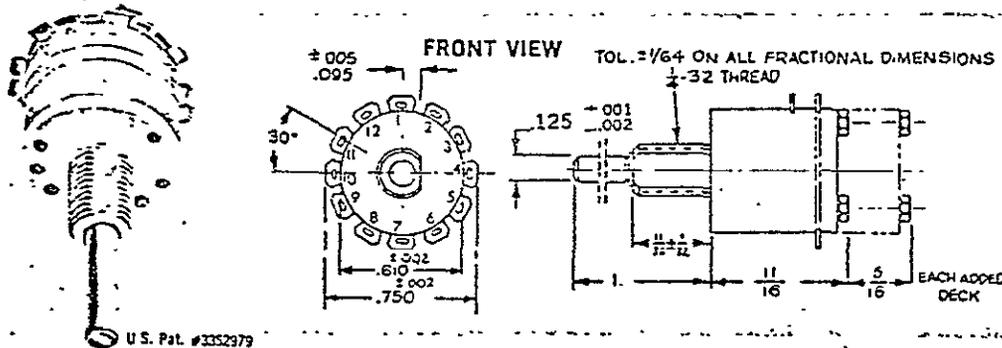
Typical spacecraft functions that are switched by rotary switches are electrical bus selections for ammeters and voltmeters, fire/smoke detectors, fuel/oxidizer pressures.

TABLE 8.3-24 ROTARY SWITCHES

<b>Input:</b>	Ground (0 Vdc)
<b>Output:</b>	Open Circuit Ground (0 Vdc)
<b>Poles:</b>	Same as flight hardware One pole per deck
<b>Reference Configuration Quantity:</b>	15 in crew station None in IOS
<b>Expected Failure Modes:</b>	excessive contact resistance welded contacts open circuit mechanical breakage
<b>Critical Measurement Points:</b>	Input/Output Terminals
<b>Positions:</b>	3 through 9 except 8
<b>Contacts:</b>	Non-shorting
<b>Life:</b>	50,000 plus mechanical operations
<b>Contact Resistance:</b>	.010 ohms, max., initial
<b>Dielectric Strength:</b>	800 VRMS between adjacent terminals 1500 V between active terminal and ground
<b>Insulation Resistance:</b>	$10^5$ megohms, at 25°C, 50% RH at sea level.

# miniature rotary switches

- Up to 12 active positions per deck shorting or non-shorting, with 30° spacing.
- Shorting AND non-shortng poles may be grouped on one deck in any combination.
- As many as 6 poles per deck.
- All individual deck parts are self-contained, and are permanently molded into place.
- Available with adjustable stops.

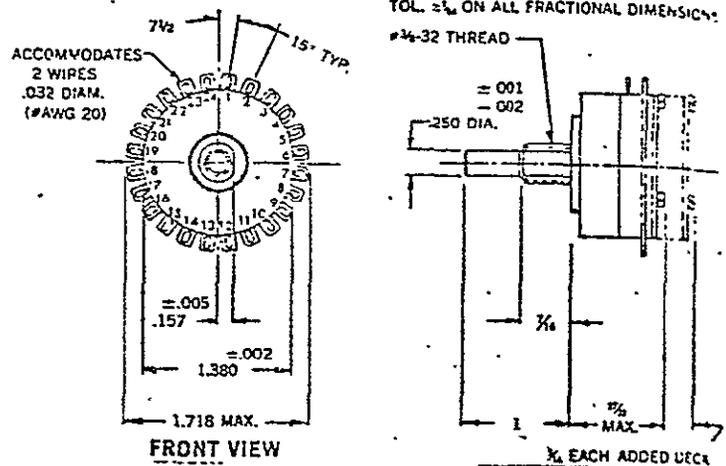


### ELECTRICAL FEATURES

1. Initial contact resistance — .010Ω max.
2. Current carrying capacity — 5.5 amps.
3. Current breaking capacities
  - ¼ amp — 115 V.A.C. resistive
  - ¼ amp — 28 V.D.C. resistive
  - ¼ amp — 28 V.D.C. inductive.
4. Dielectric strength (voltage breakdown) —
  - a. between any two adjacent positions — 800 v RMS (60 cycles).
  - b. between any active position to ground 1500 V (60 cycles).
5. Insulation resistance (measured at 25°C, 50% RH at sea level) — 10<sup>9</sup> megohms.
6. Thermal EMF — 1 microvolt/°C maximum.
7. Capacitance — between adjacent active contacts .5 MMFD Max.
8. Capacitance — between contact and common terminal .9 PFD Max.

### MECHANICAL FEATURES

1. Life expectancy — in excess of 50,000 mechanical operations.
2. Detent — positive stainless steel roller-type detent snaps from position to position. Detent housing made of special high-strength plastic, providing positive action and minimal wear.
3. Stop strength — Designed to withstand 10 inch-lbs. rotational force.
4. "Molded in" terminals give exceptional terminal strength.
5. Terminal locations clearly marked on rear cover plate.
6. On any one deck of a multi pole switch, a variety of combinations of shorting (BB) and non-shortng (BBN) actions may be obtained.
7. Materials and finishes:
  - Rotor — Lexan
  - Rotor base — Beryllium copper (gold plated)
  - Contacts & solder terminals — solid silver alloy
  - Housing — Diallyl Phthalate
  - Stator — Diallyl Phthalate
  - Shaft — stainless steel
  - All brass parts nickel plated
8. One piece contact terminal construction has holes large enough to accept two #22 AWG wires.
9. Shaft and panel seal at additional cost may be specified by using prefix "S" before switch type numbers.
10. Explosion proof per paragraph 4, p. 12 of MIL-S-3786



### SPECIAL MODIFICATIONS AVAILABLE:

All commercial version RCL switches are available with the following features at additional cost:

- Spring return feature (Series AC, CC and DC only)
  - "Push-to-turn" and "pull-to-turn" spring loaded action where an isolated position is required.
- Shaft and panel seats (specify by using prefix "S" when ordering).
  - Non-standard shaft and bushing lengths.
  - Specially coded outputs, including decimal to binary for digital applications.
    - ¼" diameter shaft and 3/8"-32 threaded bushing (Series CC, DC, HC, AC only)

CONTACT FACTORY FOR COMPLETE SPECIFICATIONS

FIGURE 8.3-42 TYPICAL COMMERCIAL ROTARY SWITCHES

### 8.3.11.2 FAILURE MODE ANALYSIS

) Failure modes of switch contacts are discussed in detail in paragraph 8.3.9.2 . Failure modes for rotary switches correspond to summary in Table 8.3-21 .

Failures of potentiometers occur primarily in worn wiper contacts or a loose wiper arm, or worn spots in the film resistor. Initial failures appear as intermittent high resistance or open circuit. At failure, volume of communications will decrease, or illumination of lights will decrease for the particular potentiometer.

One checkout technique is the addition of an amplifier and special analog circuit to the computer as shown in Figure 8.3-43. Actuation of the potentiometer is the same as the rotary switches.

### 8.3.11.3 Self Test Techniques

Fully automatic checkout or self test of a rotary switch requires the capability to sense open/closed contacts in response to switch shaft rotation and the ability to rotate the shaft automatically; i.e., by computer control. Switch manipulation can be accomplished by a motor attached to rear of switch. Measurement of contact parameters can be accomplished by the following techniques:

- Compare switched voltage to a reference voltage
- Measure voltage drop across contacts
- Direct sensing via computer operational bilevel channels

The switch manipulation technique will be discussed first, followed by the measurement techniques.

#### Motor Operated Switch

A small reversible geared motor can be mounted on the rear of a rotary switch or potentiometer. The switch can be driven to a starting position at one end of its rotation, and then through its full arc in the opposite direction. The starting position and other positions are sensed through the switch contacts.

The condition of the contacts can be sensed by techniques that will be discussed later.

REPRODUCIBILITY OF THE  
ORIGINAL PAGE IS POOR

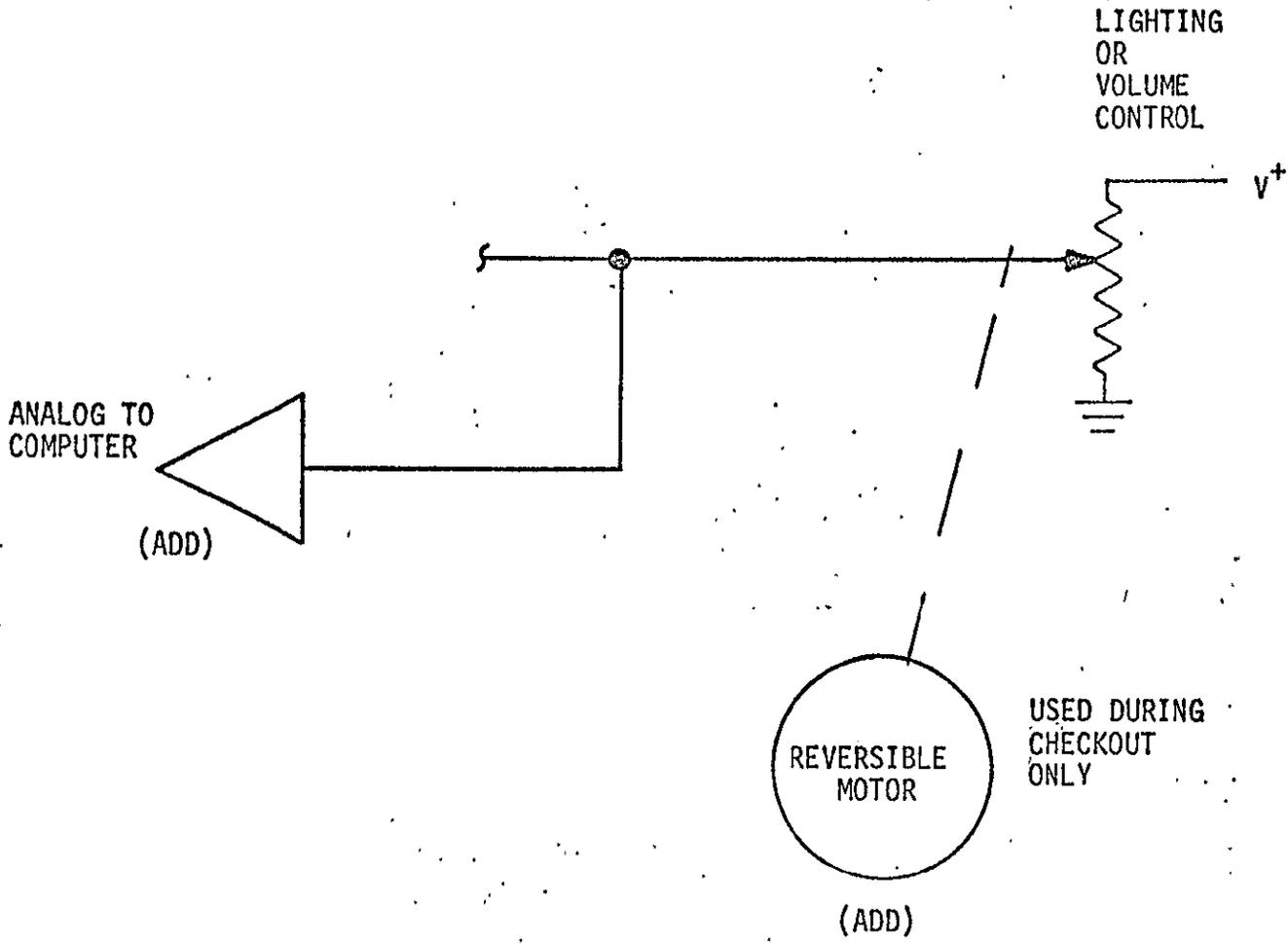


FIGURE 8.3-43 POTENTIOMETER CHECKOUT TECHNIQUE

### Sensing Techniques - Comparison of Contact Voltage Drop

As noted in paragraph 8.3.9.2, there is an IR (voltage) drop across the switch contacts. Contact resistance (and voltage drop across the contacts) increases with usage. This larger resistance, in addition to other circuit resistances, raises the voltage at the computer interface toward the computer's threshold voltage of .8 to 2.0 volts for recognition of contact closure.

If a reference voltage, less than the value of the threshold voltage, is fed into the  $V_{ref}$  terminal of a comparator such as Fairchild uA734 (see Figure 8.3-44), and the switch signal is fed into the  $V_{in}$  terminal, early warning of switch failure can be detected by monitoring the output voltage at  $V_{out}$ . For checkout, the switches and individual contacts can be sequenced into the comparator by one of the methods previously described and  $V_{out}$  can be monitored by the computer. See software flow diagram in Figure 8.3-45.

If the switch has not been actuated to the correct position, the comparator will give a "failed" output voltage.

A pole that controls components such as lights or relays may be handled similarly to Figure 8.3-46. However, this can be applied only where the maximum circuit (relay voltage does not exceed the "highest input voltage range" and the "maximum differential input" voltage ratings of the comparator, unless protective resistors R1, R2 and R3 are added. Both limit ratings are near  $\pm 15V$  for Fairchild Semiconductor comparators. The comparator works similarly to the above circuits, and as shown in Figure 8.3-44.

### Analog Measurement of Voltage Level

Figure 8.3-48 shows how a differential amplifier may be used to measure the IR drop across the switch contacts. One circuit is for a switch that controls a ground voltage, the other is for a component (e.g., relay coil) voltage. The component voltage is limited to the common mode voltage of the differential amplifier, which is normally  $\pm 15$  volts DC. Because the switch and relay coil may generate destructive voltage spikes, the component voltage may be limited to considerably less than the common mode voltage. However, protective resistors

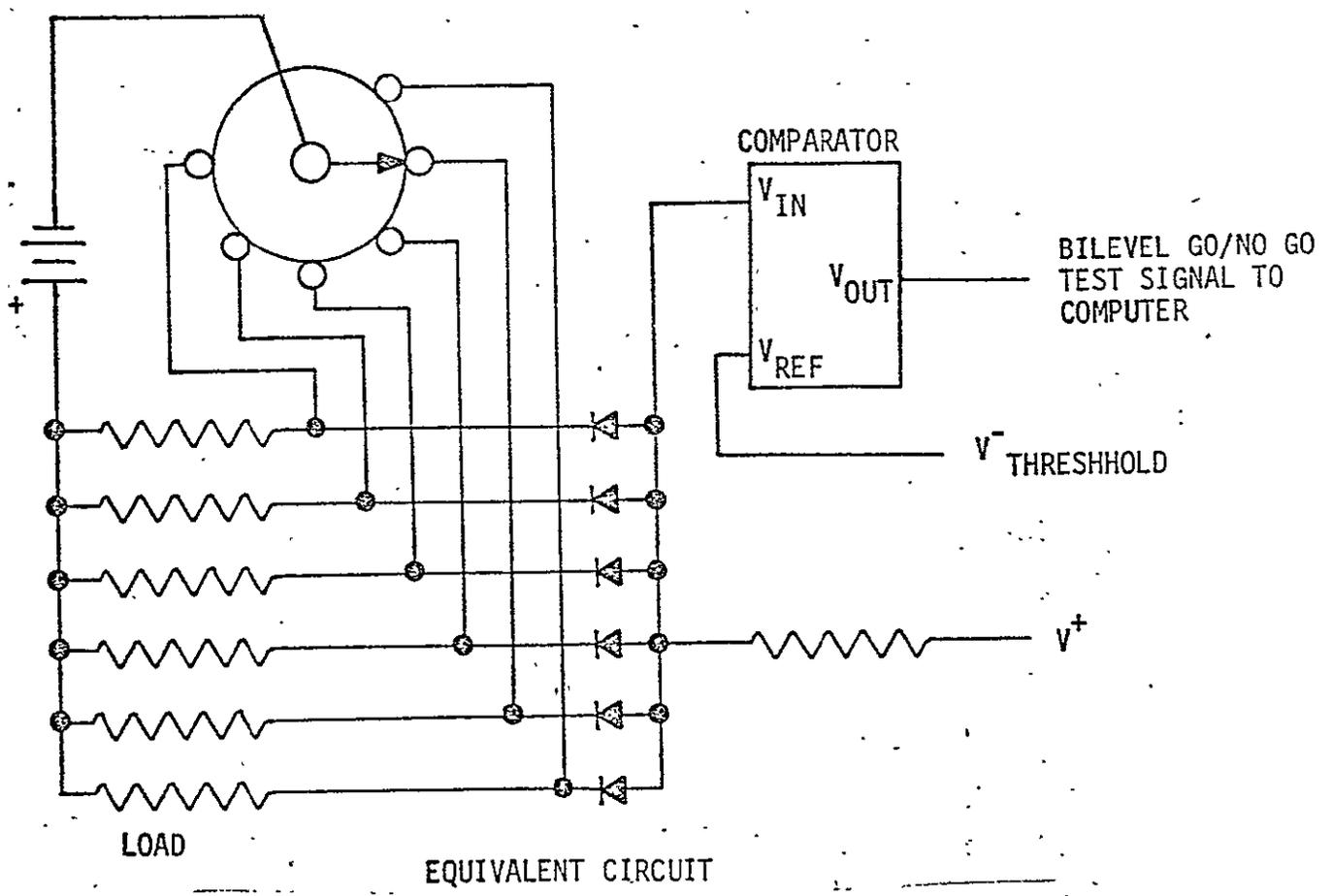
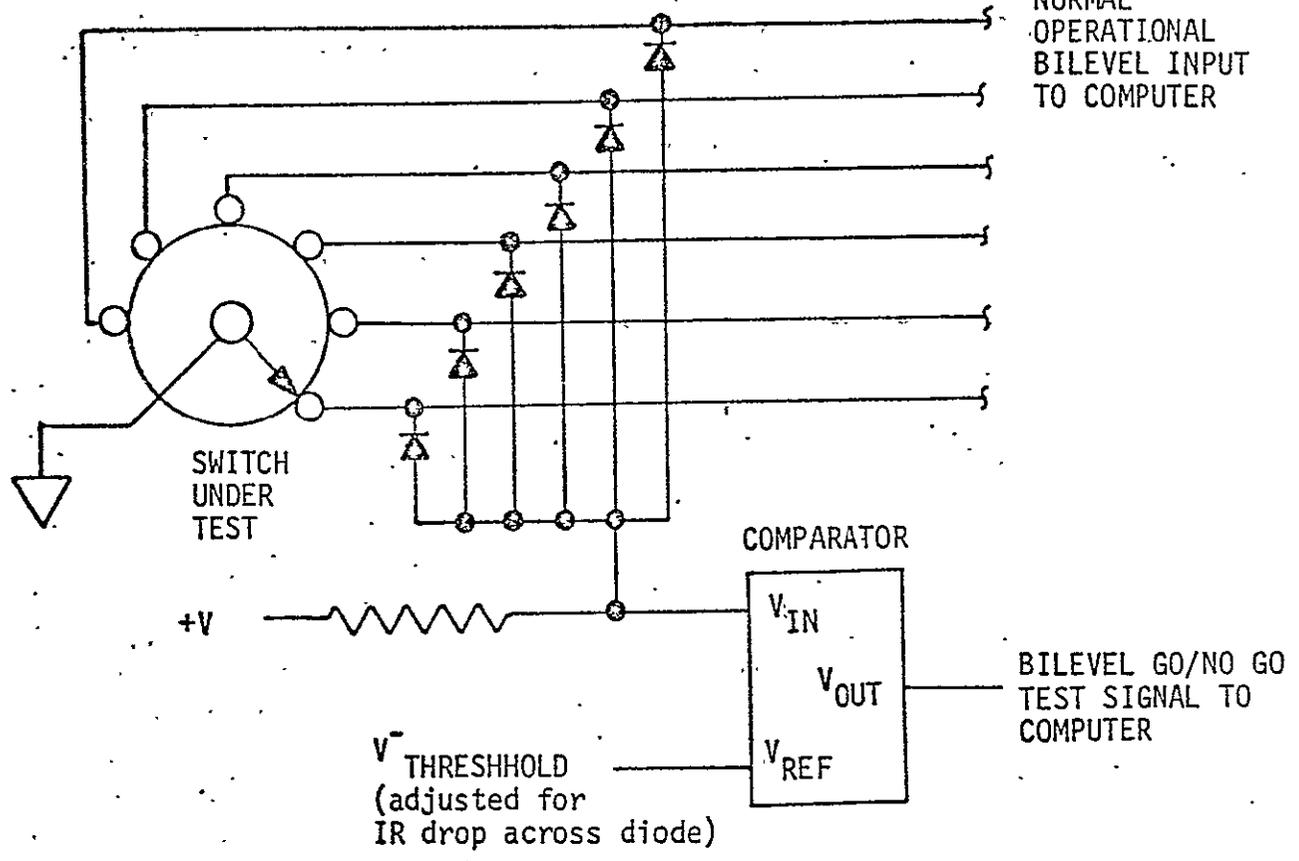


FIGURE 8.3-44 COMPARISON OF CONTACT VOLTAGE DROP

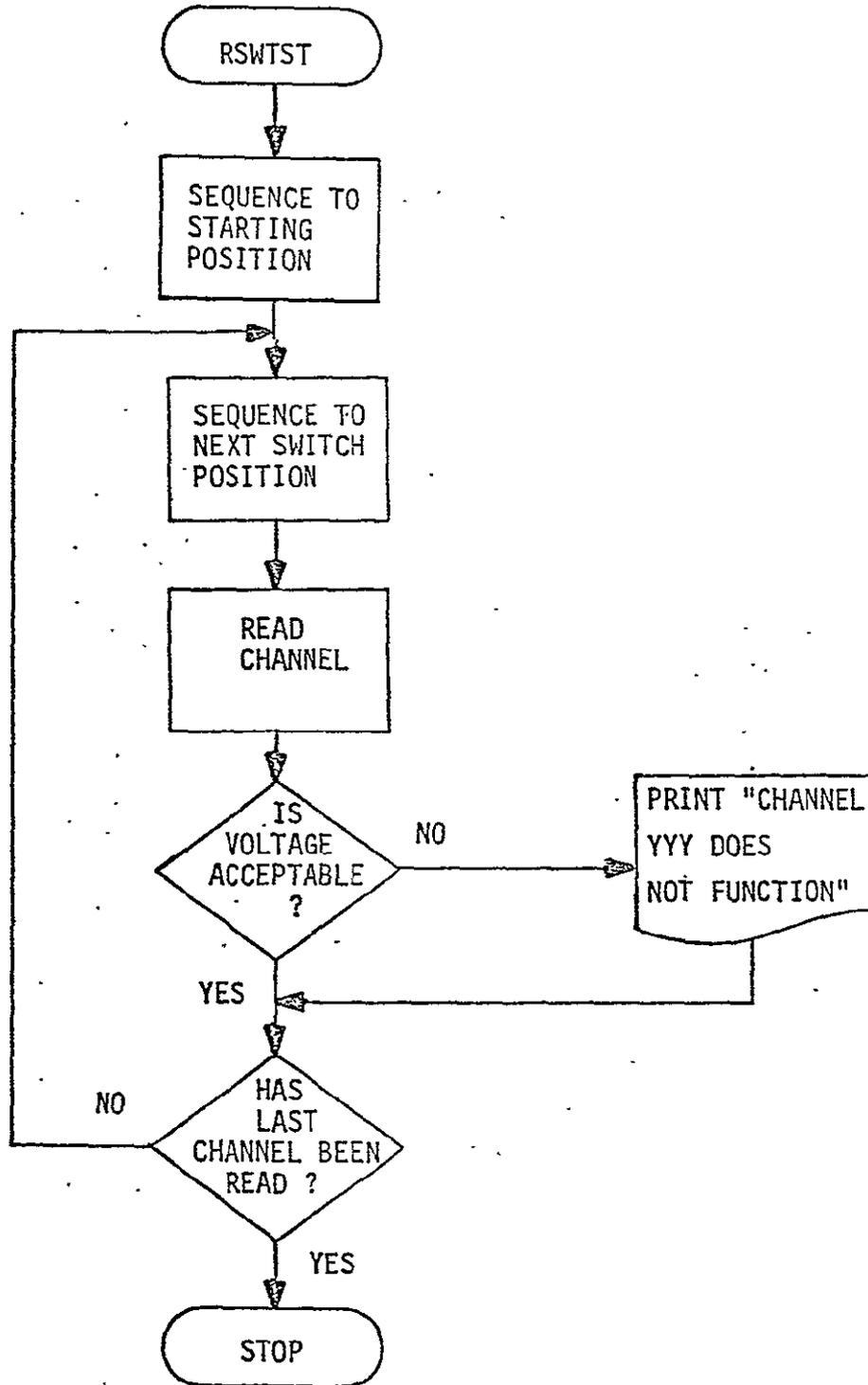


FIGURE 8.3-45 SOFTWARE FUNCTIONAL FLOW DIAGRAM

ly be added as in Figure 8.3-44 so that +28VDC circuits may be monitored. A word of caution: if the differential amplifier is placed outside the switch's panel, it will measure voltage drops of additional resistances as shown in Figure 8.3-33 in Section 8.3.9 .

#### Operational Computer Input Signals

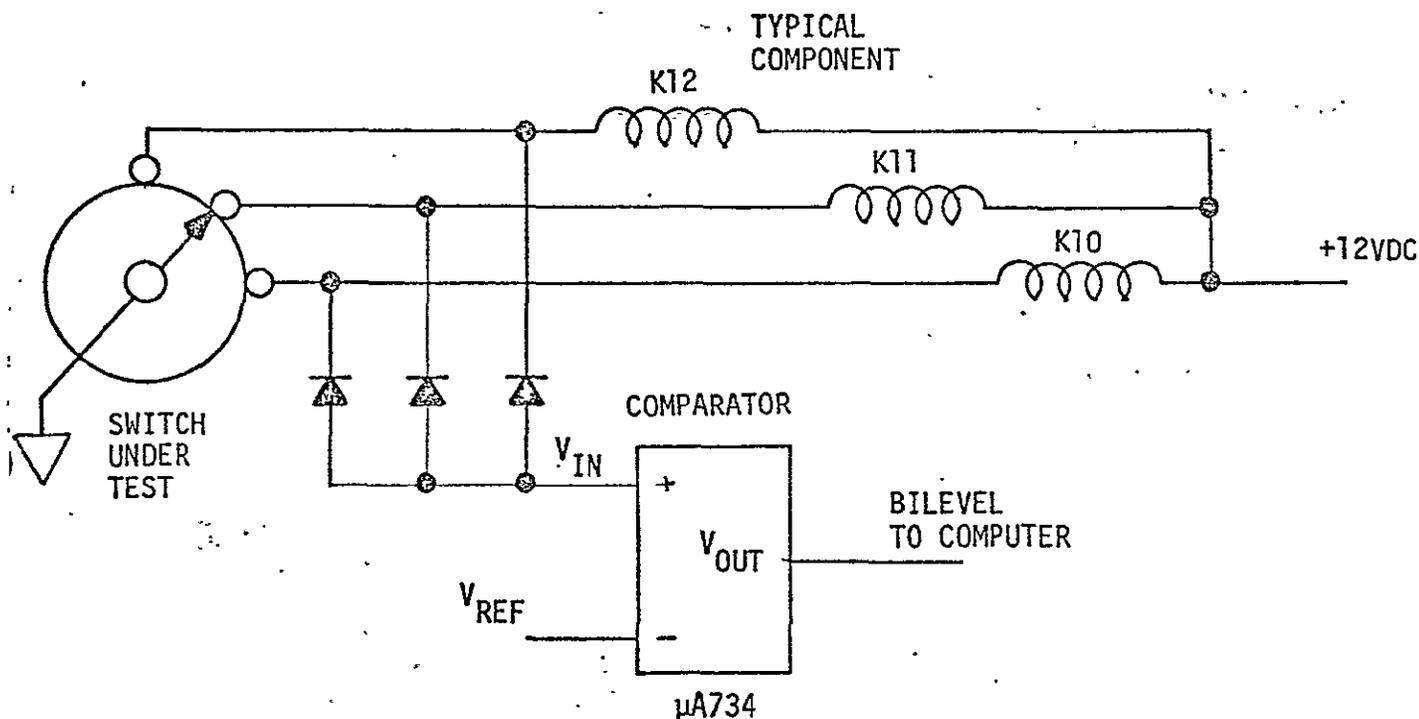
This technique adds no circuitry to the simulator. The program checks the regular bilevel computer inputs for functional operation. Since there are few switches and switch contacts, contact resistance and suspected failures can be economically checked by technicians on an "as required" basis when the computer senses a failure.

A comparison of these techniques is shown in Table 8.3-25.

#### 8.3.11.4 Hardware and Software Requirements

It is recommended that the switches be cycled by means of a small motor attached to the rear of the rotary switch.

It is further recommended that the contact condition be checked by the operational bilevel computer input signals.



IF SWITCH IS OK,  $V_{IN}^+ < V_{IN}^-$ ,  $V_{OUT} = "0"$  VOLTS

IF SWITCH RESISTANCE IS EXCESSIVE,  $V_{IN}^+ \geq V_{IN}^-$ ,  $V_{OUT}$  BECOMES A "1" ( $V^+$  VOLTS)

FIGURE 8.3-46 COMPARATOR CIRCUIT FOR COMPONENT SWITCHING POLE

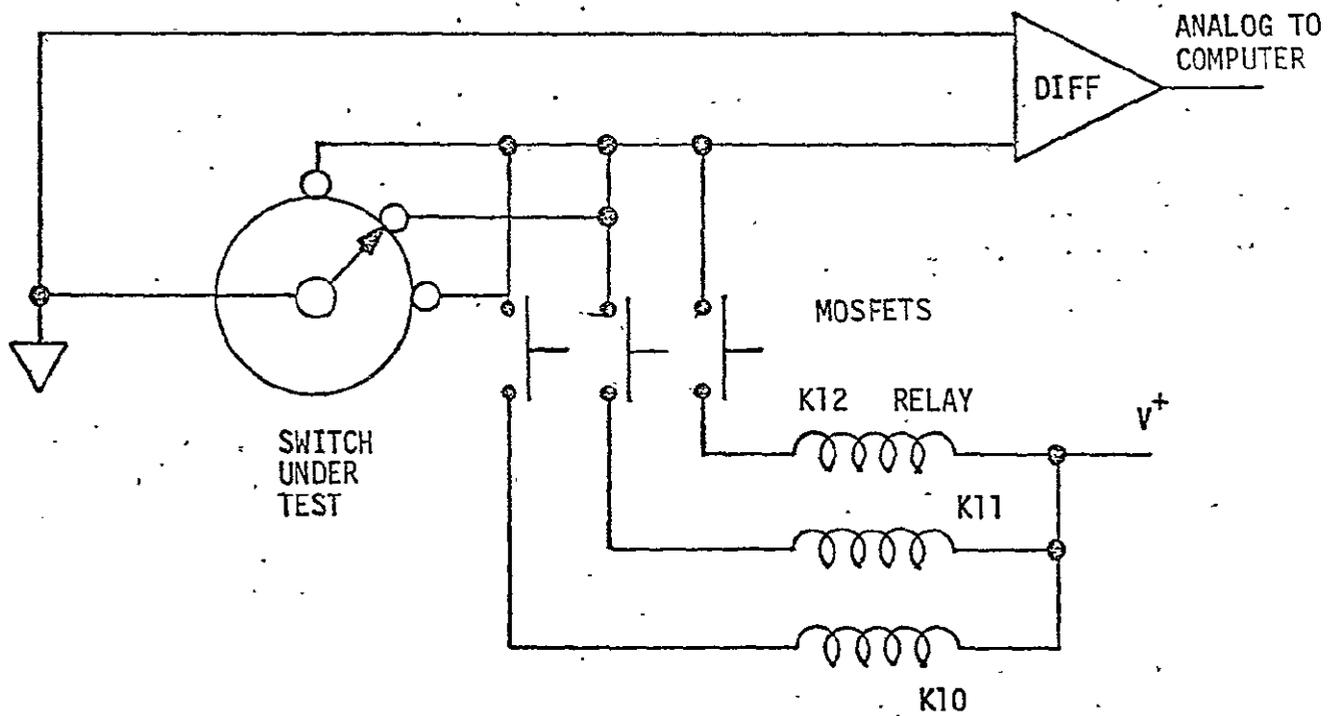
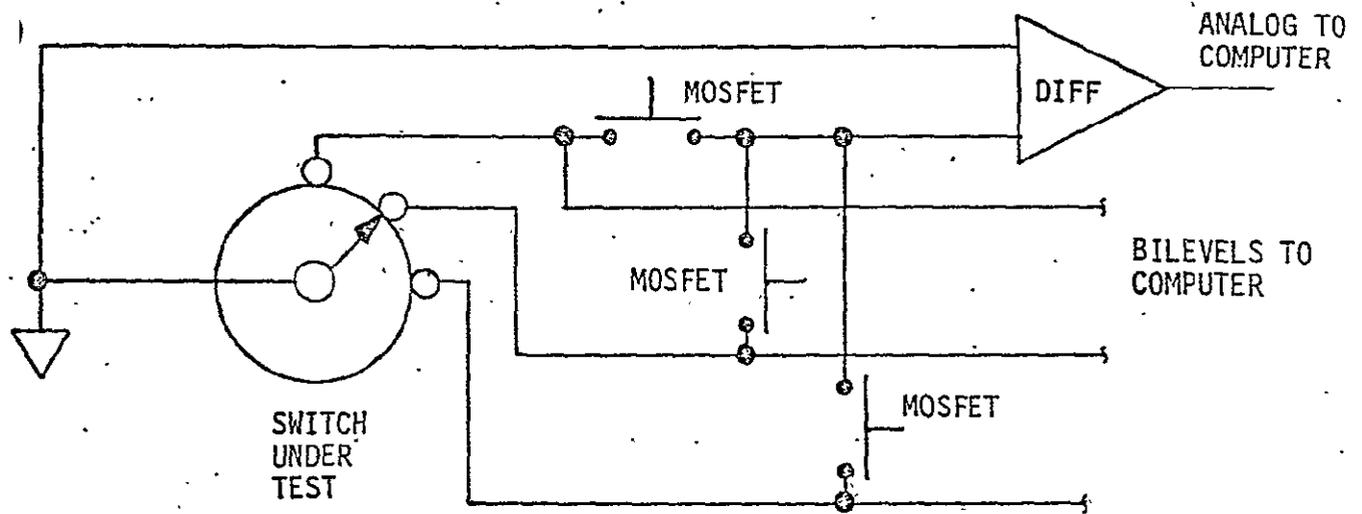


FIGURE 8.3-47 DIRECT MEASUREMENT OF VOLTAGE DROP ACROSS SWITCH CONTACTS

CANDIDATE TECHNIQUES	RELATIVE COST	HARDWARE MODIFICATIONS		SOFTWARE REQUIREMENTS	REMARKS
		MODS TO SWITCH	ADDITIONAL MODS TO EQUIPMENT		
MOTOR OPERATED SWITCH	MODERATELY LOW	ADD SMALL MOTOR TO REAR OF SWITCH	ADD ON/OFF, DIRECTION, SEQUENCING CONTROLS	PROCESSING OF DISCRETE FEED-BACK DATA. MULTIPLEXING OF OUTPUT SIGNALS TO MOTOR OR TO ROBOT.	SELECTED TECHNIQUE
THRESHOLD VOLTAGE COMPARATOR	MODERATE	NONE	ADD COMPARATOR, $V_{REF}$ , ISOLATION DIODES	PROCESSING OF DISCRETE DATA	GO/NO GO CHECKING EXTERNAL TO COMPUTER. LIMIT CHANGED BY HARDWARE ADJUSTMENT
ANALOG VOLTAGE DROP MEASUREMENT	MODERATE	NONE	ADD DIFFERENTIAL AMPLIFIER	PROCESSING AND EVALUATION OF QUANTITATIVE PARAMETERS	GO/NO GO CHECKING. LIMIT CHANGED BY PROGRAMMING
OPERATIONAL/ INPUT SIGNALS	LOWEST COST	NONE	NONE	PROCESSING OF DISCRETE DATA	SELECTED TECHNIQUE. USE EXISTING BILEVEL LINES TO COMPUTER

TABLE 8.3-25 CHARACTERISTICS OF CANDIDATE TECHNIQUES

### 8.3.12 Throttles, Hand Controllers, Rudder Pedals

This section deals with control devices that are physically larger than finger activated controls such as toggle and pushbutton switches, and which normally require the range of force that can be supplied by an operator's hand or foot. Included in this type of controllers are throttles, hand controllers, rudder pedals, and speed brake control levers. Some of these devices generate analog signals proportionate to the force applied or amount of deflection. Others generate discrete outputs that only notify the controlled subsystem as to the direction of the desired change.

#### 8.3.12.1 Hardware Description

Basically, all of these devices transform mechanical inputs from the crewman into electrical signals for vehicle subsystem control. Motion of the handle or pedal is monitored either by continuous position sensors, such as a potentiometer, or synchro/resolver arrangement, or by discrete switches that indicate the direction of motion without information as to the amount of deflection of the controller. In the first case, an analog signal is generated to the HOST computer, through the DCE. This signal is then used to determine velocity of change or whatever parameter is related to the controller motion in the affected subsystem math model, and thereby to determine the course of the simulation sequence. In the second case, the discrete information, also routed to the HOST through the DCE, is used to determine direction of change in the subsystem affected.

#### 8.3.12.2 Failure Modes and Characteristic Parameters

Failure modes and characteristic parameters for sensors used in the various controllers discussed here have been discussed in previous sections of this report. Table 8.3-26 shows the type of sensor that may be encountered in each type of controller and correlates these sensors to the section that dealt with the self test analysis for that type of sensor.

TABLE 8.3-26 CONTROLLER MOTION SENSORS

CONTROLLER TYPE	SENSORS	SENSOR SELF TEST ANALYSIS
THROTTLES	Potentiometers Synchro/Resolvers	Section 8.3.11 Section 8.3.4
HAND CONTROLLERS	Discrete Switches Potentiometers Synchro/Resolvers	Section 8.3.9 and 8.3.10 Section 8.3.11 Section 8.3.4
RUDDER PEDALS	Potentiometers Synchro/Resolvers	Section 8.3.11 Section 8.3.4
CONTROL LEVERS	Discrete Switches Potentiometers	Section 8.3.9 and 8.3.10 Section 8.3.11

It should be noted that, although the synchro/resolver analysis in Section 8.3.4 deals with display devices, the concepts discussed also apply to controllers in that the components involved and their characteristics are very similar. Only the direction of information flow is reversed, but, in both cases, an electrical signal is generated which is indicative of desired mechanical motion.

An additional source of failure applicable to the controllers discussed here is the possible mechanical linkage that would prevent motion of the controller or that would cause an improper signal to be generated for a particular motion of the specific controller.

### 8.3.12.3 Self Test Techniques

Self test considerations for these controllers are very similar to other control devices discussed previously, in that the primary requirement is to develop an automatic, calibrated means for control activation. Development of servo systems to actuate the controls is a relatively simple control design problem. The packaging problems are, of course, unique to each simulator to which they are applied. Although there is no substantial area for development of self test techniques for these devices, it should be noted that the basic types of concepts shown in subsection 8.3.9 for switch actuation could be adapted

to several types of control actuation and, with proper servo drives, could also be used for calibration. Calibration is a requirement even for manual testing of these devices and may, in some cases, require special hardware for this purpose.

Implementation of simple servo systems for control actuation presents no unusual problems. The packaging peculiarities of specific simulators are not an opportunity for development of generic self test techniques either. Consequently, analyses of the control actuation problem have not been extended any further than the discussion above.

### 8.3.13 Crew Station/IOS Controls and Displays Self Test Summary

The results of the analyses of self test techniques for the controls and displays in both the crew station and the instructor/operator station are characterized by the following properties:

- o The display instruments or control devices are the lowest level to which fault detection and isolation is intended to be carried in an automatic or self test mode. Effectively, there is only a requirement to test for proper functioning of each device. There is no need for further fault isolation testing.
- o In most instances modifications are required to the display instruments in order to implement a technique for sensing display instruments performance. This requirement for instrument modification creates a need for implementation of desired self test facilities during simulator development, rather than after delivery and acceptance.
- o Actuation of control devices does not necessarily require any modifications of the devices themselves, but the necessary actuation equipment, must be designed for compact stowage and concealment from the trainees.
- o Data base requirements can be substantial because of the number and variety of devices include in the controls and displays area. See the following subsection for a summary of the data base requirements.

- o The results of the present analysis represent self test techniques or approaches that may or may not be necessary for a specific simulator. The value of implementing any of these techniques depends on specific simulator requirements and other factors that are outside the scope of this study.

The implementation of the various control and display tests described requires a test executive to control and integrate the various component tests. The configuration of such an executive is discussed briefly in the following section. The overall data base requirements for the crew station and IOS are summarized in subsection 8.3.13.2.

#### 8.3.13.1 Crew Station/IOS Test Software

The software flow for an integrated crew station/instructor operator station test software structure is presented in Figure 8.3-48. The software modules called out by the crew station test module, CSTST, omit addressing requirements for communicating test signals out as well as for reading test results into the computer. Addressing requirements are dependent on the particular hardware configuration being considered and computer word sized, byte addressing capability, etc. The data base impact of addressing requirements even for the large number of instruments and controls in the crew station subsystems are negligible compared to the basic data that must be stored with respect to instrument performance.

The summary of the data base requirements for the crew station display and control test software are summarized in subsection 8.3.13.2.

#### 8.3.13.2 Test Software Data Base Impact

An initial estimate of the data base requirements for the test software and data to support a self test capability for a Shuttle simulator is presented in Table 8.3-27. The estimates shown assume that instruments in the IOS are duplicates of the instruments in the crew station and no additional data storage is allocated to the IOS instruments. In the case of discrete equipment such as indicators or toggle switches, it is assumed that any data required will be sub-merged in the context of the test software and will be extremely minor in amount.

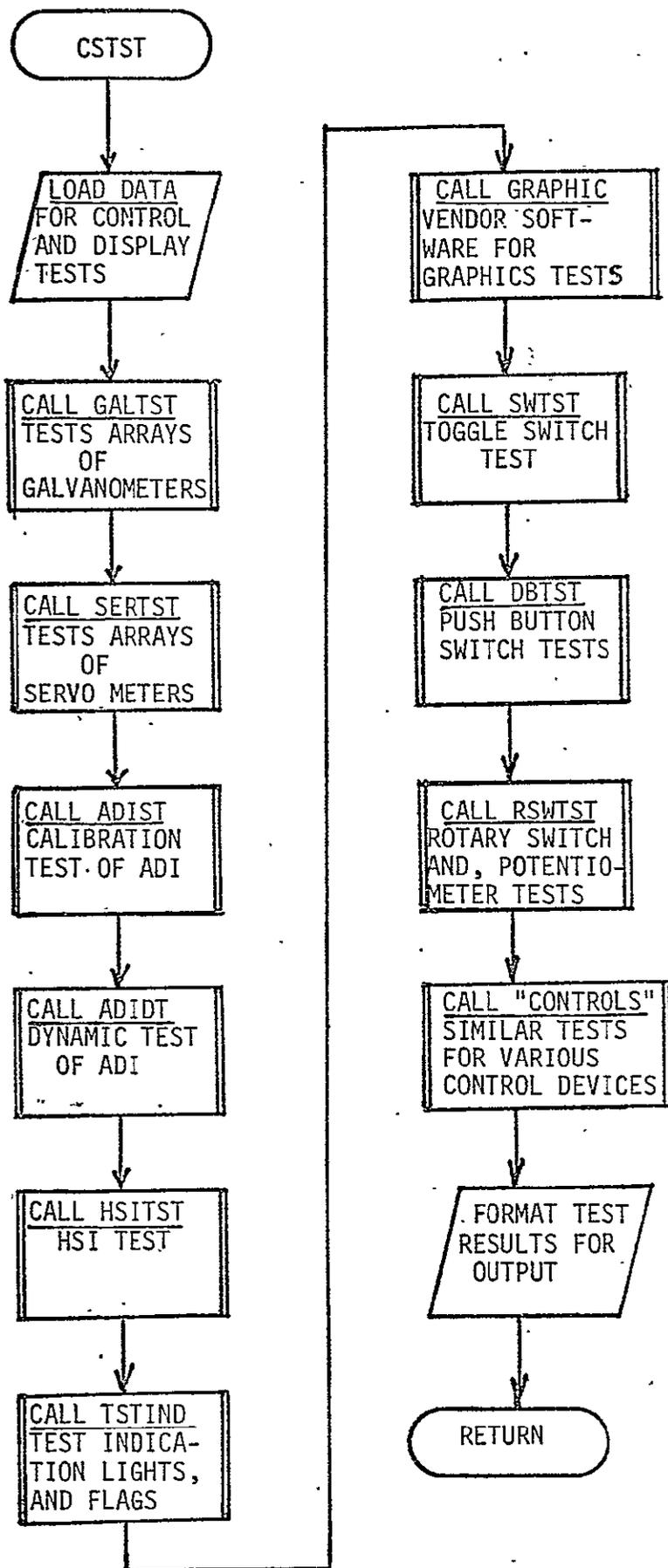


FIGURE 8.3-48 CREW STATION TEST SOFTWARE RESPONSIBILITY OF THE ORIGINAL PAGE IS POOR

The test software and data base for the crew station and IOS can be considered the section of the test software that will be most susceptible to impact by changes in the simulator configuration. Changes in the numbers of instruments, the type of instrument, or the arrangement of the instruments will impact the locations in the data base in which test values are stored as well as the magnitudes of the test parameters.

TABLE 8.3-27 CREW STATION TEST SOFTWARE DATA BASE IMPACT

COMPONENTS	NO. OF UNITS	SOFTWARE REQUIREMENTS	
		INSTRUCTIONS (NO. OF)	FLOATING POINT DATA WORDS (NO. OF)
GALVANOMETERS	70	300	630
DC SERVO DRIVEN METERS	36		324
ADI AND HSI	4	100	110
LIGHTED INDICATORS	700		
BILEVEL FLAGS	400		
TOGGLE SWITCHES	600	100	
PUSH BUTTON SWITCHES	75		
ROTARY SWITCHES	15		
POTENTIOMETERS	55		55
MISCELLANEOUS CONTROLLERS NEGLECTED			
<u>TOTALS</u>		<u>500</u>	<u>1119</u>

#### 8.4 VISUAL SIMULATION SUBSYSTEMS

The extent and complexity of visual simulation equipment required for future manned spacecraft, as typified by the Shuttle, can be readily inferred from schematic representations of the various crew members fields of view as shown in Figure 8.4-1, (a), (b), and (c). This information was obtained from Reference 8.4-1. The pilots and commanders views are the most demanding because of the variety of scenes to be displayed, the required fidelity, and the wide fields of view in the horizontal plane.

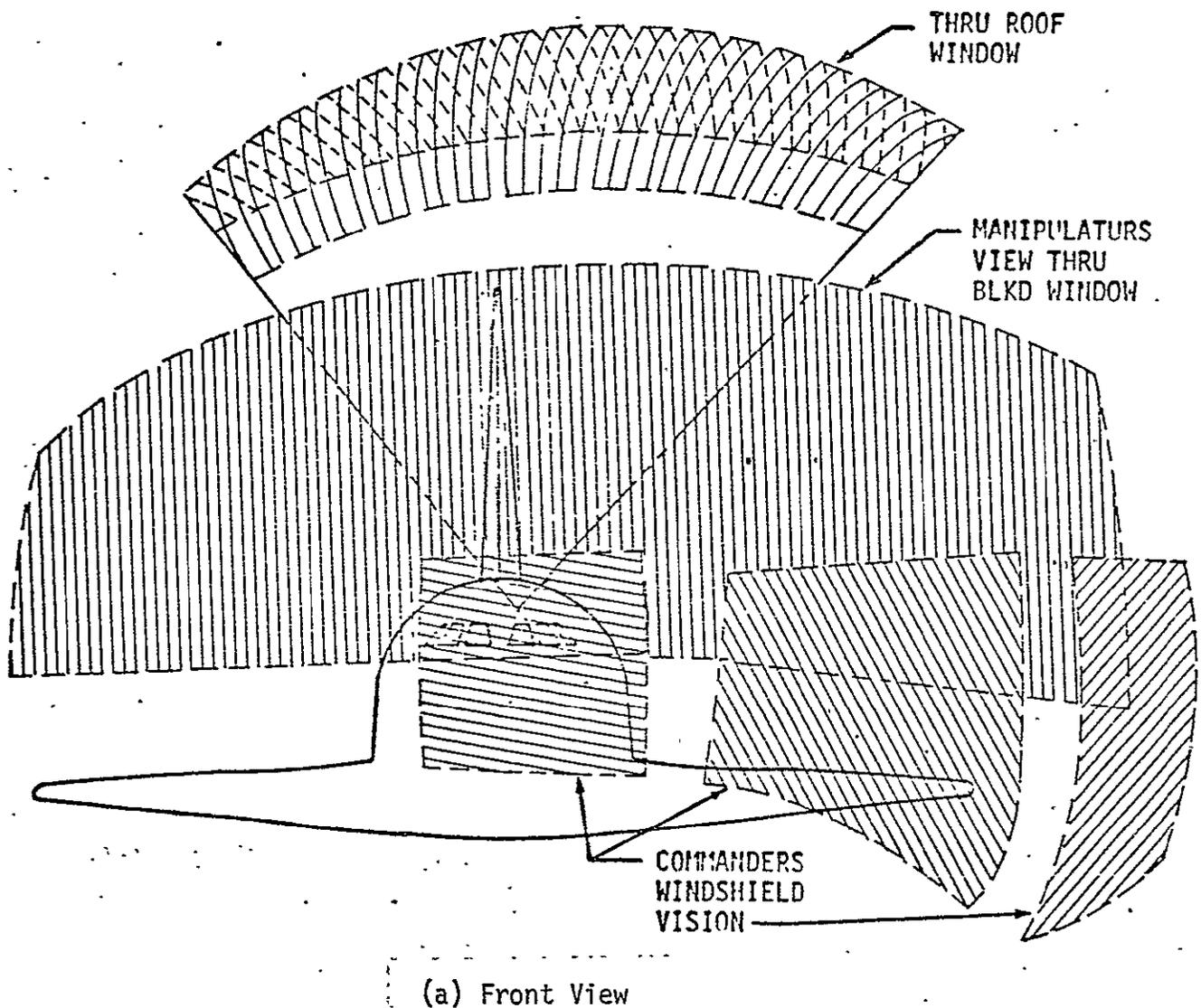
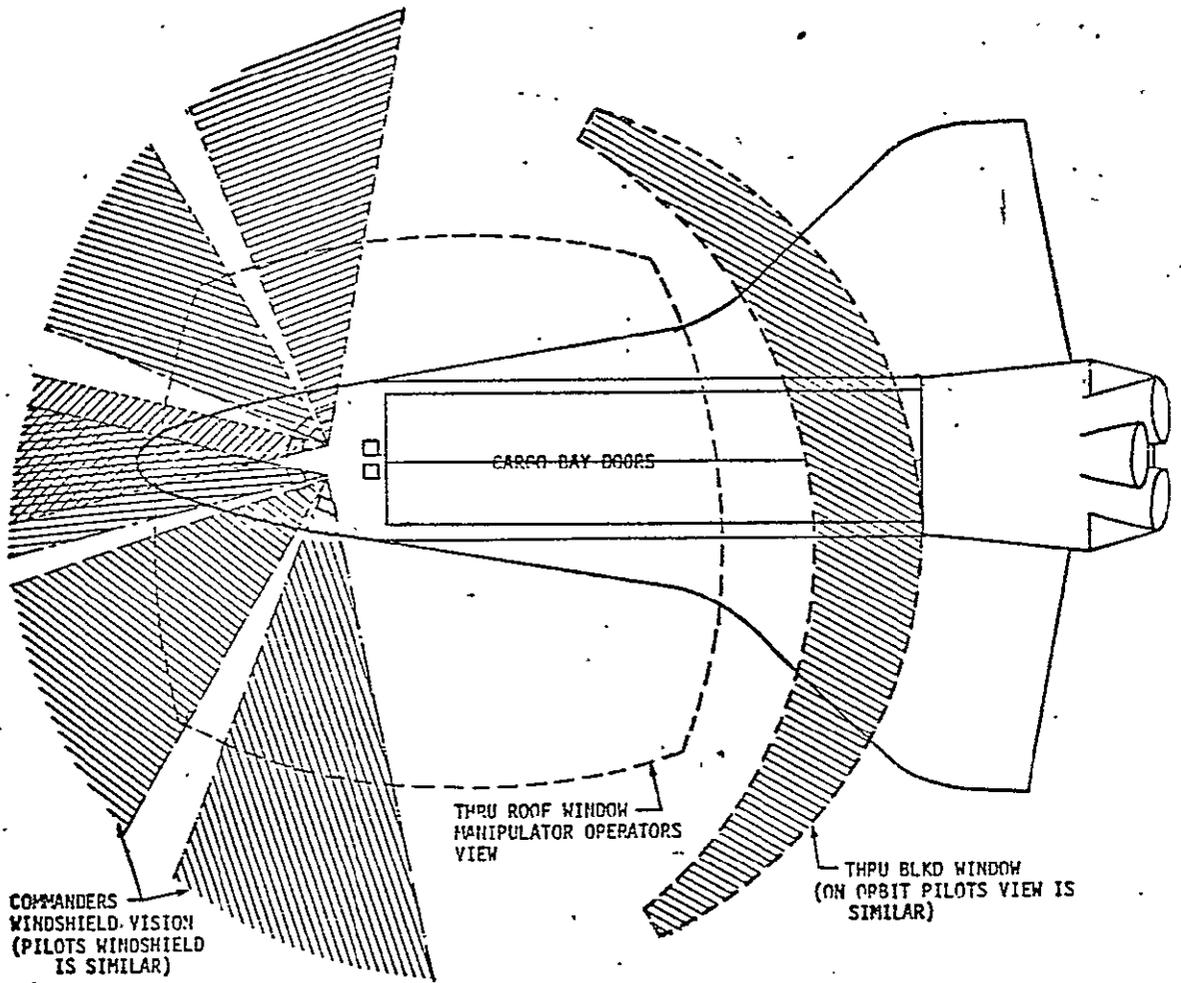
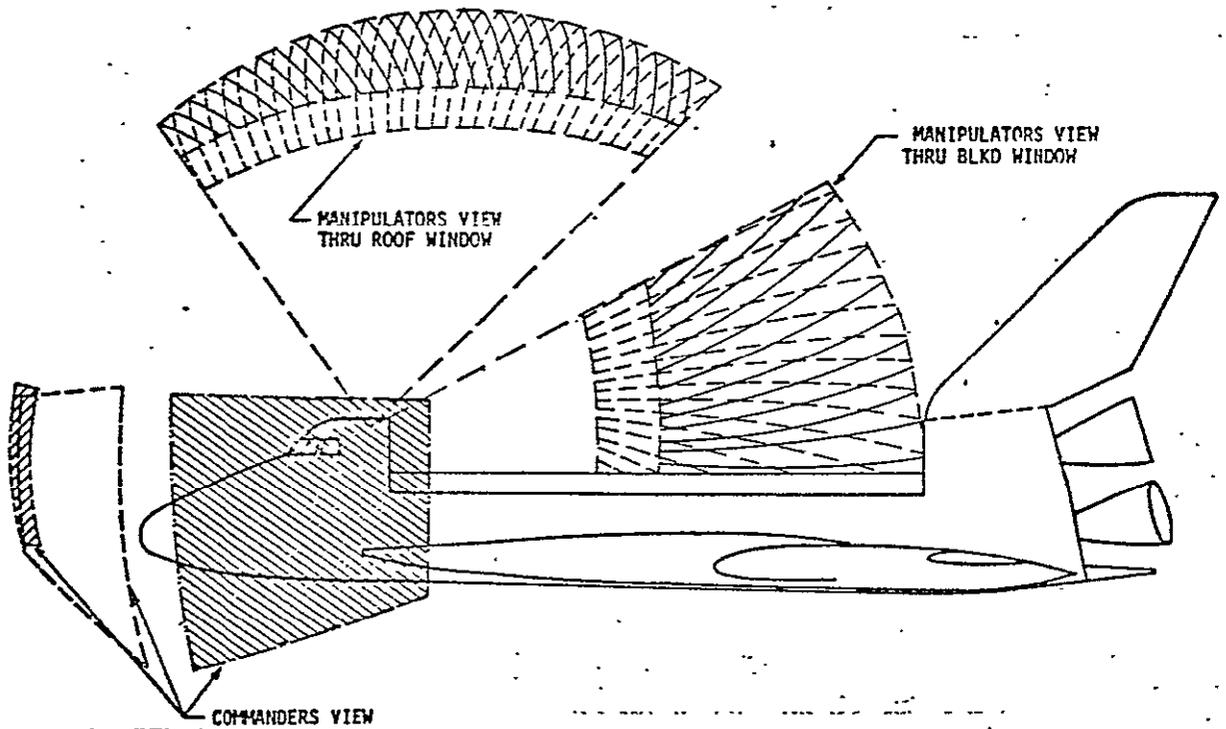


FIGURE 8.4-1 SHUTTLE ORBITER CREW STATION FIELDS OF VIEW



(b) Top View



(c) Side View

FIGURE 8.4-1 SHUTTLE ORBITER CREW STATION FIELDS OF VIEW (continued)

In the following analysis of self-test techniques for visual simulation equipment, we have emphasized those particular devices, concepts and systems most appropriate to Shuttle visual simulation requirements. The Reference 8.4-2 study concluded that these requirements necessitate a TV/model system for the forward crew station with a CRT display viewed through infinity optics. CGI techniques may be appropriate for certain scenes for the other stations.

The visual simulation equipment can be divided into several major subsystems which present their own peculiar set of checkout problems. Visual simulation subsystems considered for this analysis consist of the following:

- o Image generation equipment
- o Image transmission system
- o Display equipment

For purposes of this study, we will draw the line between these subsystems in such a manner as to best group together those hardware tests requiring similar techniques. Thus, we include the TV camera positioning servo drives and optical probe alignment tests with the image generation equipment analyses. All of the TV system electronics and the electrical performance of the camera tube are included in the image transmission system. In the display equipment, we consider only the CRT and the display optics.

In order to implement simulator self-test techniques, it is necessary to establish data paths to the host computer for test initiation, test signal generation and data acquisition. This is readily accomplished for the TV system electronics which can then be used, in conjunction with the camera, to monitor alignment and servo response parameters for the image generation equipment. Consequently, in this analysis, we will address first the self-test technique for the TV system electronics, Section 8.4.1, which comprise the image transmission system. In Section 8.4.2 we discuss the problems peculiar to the Display System, and in Section 8.4.3, we discuss the image generation equipment. The overall integrated visual system test approach is summarized in Section 8.4.4.

#### 8.4.1 Image Transmission System Self Test Techniques \*

In order to address the problems and techniques for self test of the three sections of the visual simulation hardware, it is necessary to develop a more detailed description than that provided by the schematic of Paragraph 5.2.6. In the instance of the image transmission system, this requires further assumptions with regard to the nature of the hardware expected and subsequent consideration of the impact of alternate simulation hardware approaches on the application of the self test techniques proposed. Consequently, in this discussion of self test techniques for a closed circuit TV (CCTV) system for image transmission in the simulator visuals, we will address the following topics:

- o CCTV system description
- o Test Techniques
- o Test Hardware
- o Test Software
- o Incipient fault detection and data base impact.

##### 8.4.1.1 CCTV System Description

A schematic of the video circuits and opto/electronic interfaces, as well as the interface to the simulator computer, is shown in Figure 8.4-2. Signal paths associated with the servo drives for camera positioning, probe controls, and model motion have been omitted and are considered in the following section on scene generation equipment. This configuration essentially represents the video components of the visual simulation equipment previously discussed.

The major unit functions may be summarized as follows:

- o Signals for controlling TV cameras are supplied by the camera Control Units (CCU). Signals such as vertical and horizontal sweeps, blanking, focus control, and beam controls are contained in these lines.

The formation of the analysis of television test requirements and techniques and the use of the Digital Processing Oscilloscope was suggested by Lawrence W. Lockwood of Lockwood Associates, Houston, Texas.

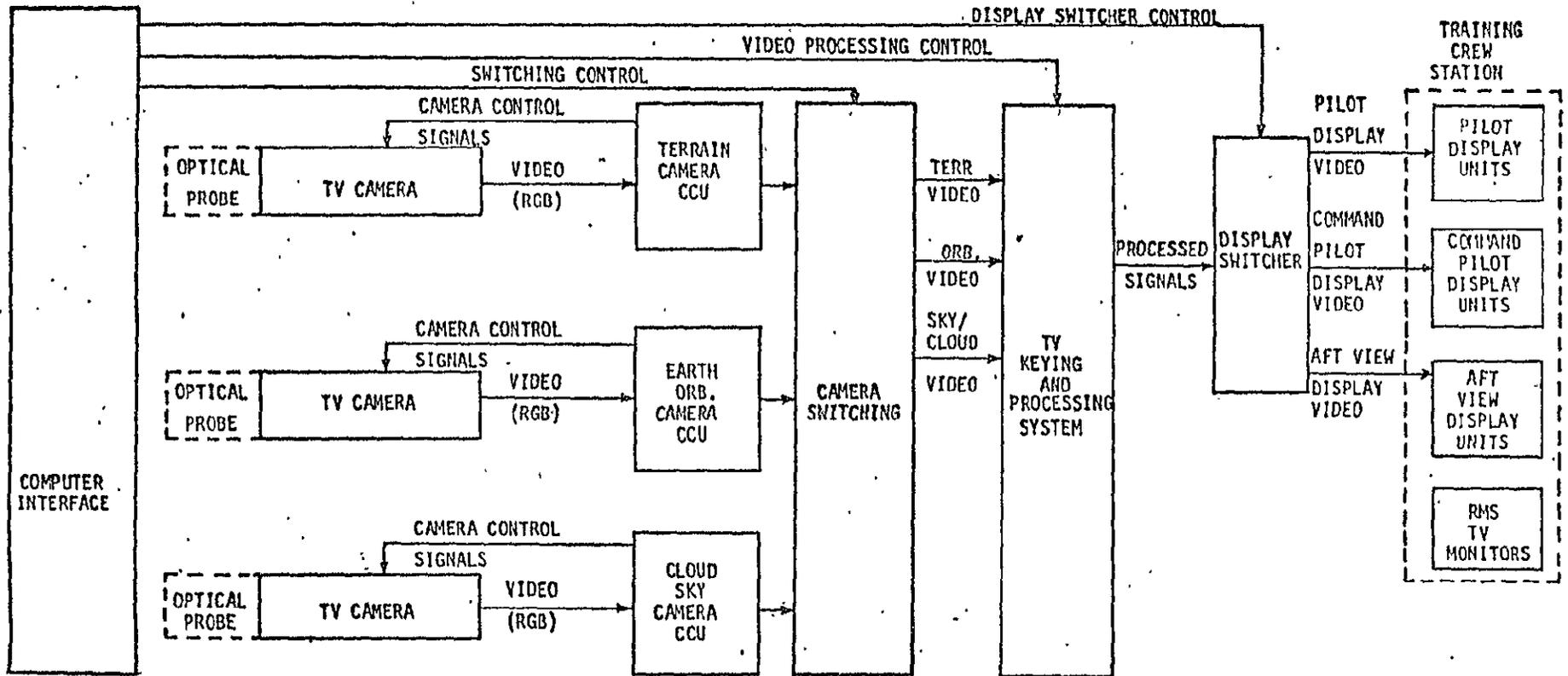


FIGURE 8.4-2 SIMULATOR TV SYSTEM SCHEMATIC

- o Under external computer control, the camera switching unit selects the appropriate earth model video signals required for the particular phase of the Shuttle mission. The basic scene element video signals of earth and cloud/sky are then supplied to the TV Keying and Video Processing System.
- o The TV Keying and Video Processing System accepts the basic video signals of earth video, cloud/sky video, terrain video, etc., and performs the appropriate processing to obtain composite scene video with the appropriate image inseting and special effects for realistic simulation of the out-the-window scenes. Certain special effects require computer inputs for gain control.
- o The Display Switcher under computer control routes the composite scene video to the appropriate display units of the crew training station.

In order to address the self test of the video system, it is necessary to establish more specifically the functional aspects of the configuration assumed. These functional factors are derived from previous studies, Reference 8.8-1, of Shuttle simulator visual simulation requirements and techniques. The most significant requirements are those for color in the visual simulation as well as for a wide field of view,  $120^{\circ}$  to  $140^{\circ}$ .

A widely used means of generating color TV images involves the use of dichroic filters to separate out the primary colors (red, green, and blue) from which the color image may be reconstituted. This is accomplished most directly in RGB systems (Red, Green, Blue) as shown in Figure 8.4-3. The three colors are separated out by filters. The camera tubes each effectively process a black and white image. The images are recombined to produce color in the display tube.

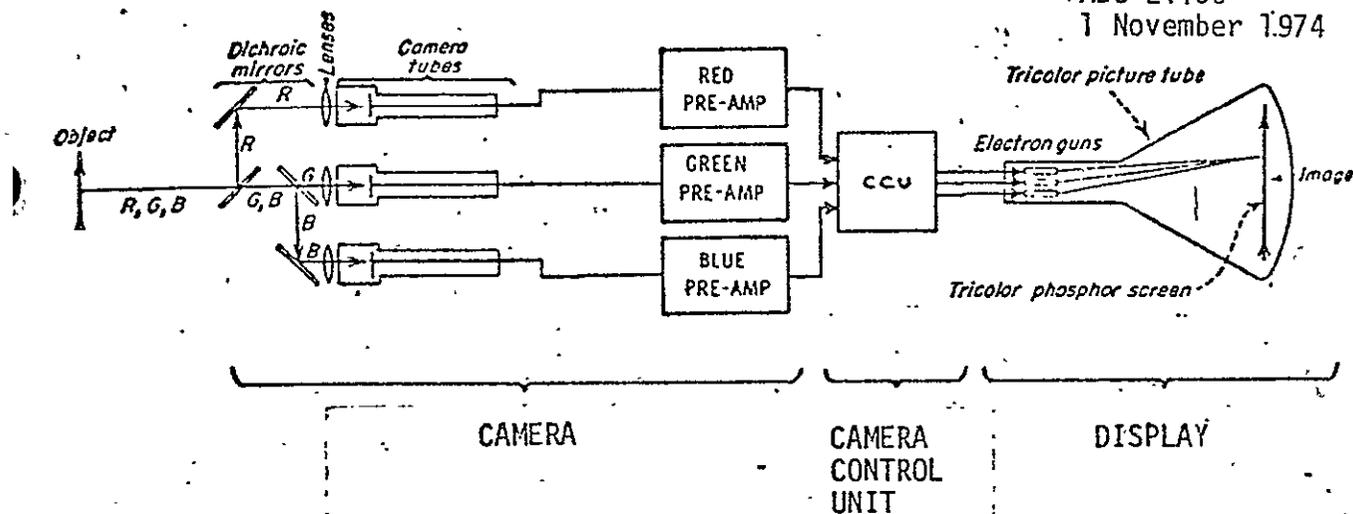


FIGURE 8.4-3 TYPICAL RGB SYSTEM SCHEMATIC

There is no encoding of the chrominance (color qualities) onto a subcarrier. Such an encoding is unnecessary since the signal containing the color does not have to be compressed into one channel for broadcast. Thus each "color" TV channel is effectively a separate black and white TV chain. In essence the verification techniques used for each of the TV electronics chain (R, G, or B) can then be the same as for a black and white (B & W) TV system since the video signals have no color qualities.

There are several factors which should be noted with respect to implementation of the above concepts. These are, first, the limitations on the probe/camera interface and secondly, the alternate color coding systems available.

The optical/electronic interface between the scene generation equipment and the image transmission system involves the interface between the optical probe, mounted on the TV camera, and the camera tube configuration. Figure 8.4-4 presents schematically the functional elements of an articulated probe as required for the Shuttle visual simulation. Reference 8.4-3 describes the only available articulated optical probe with an adequate field of view,  $140^{\circ}$ . An "articulated" probe has the rotational degrees of freedom required to generate views representing spacecraft motions in pitch, roll and yaw. Figure 8.4-5 presents a possible configuration for an interface with a single RGB camera.

Although the probe described in Reference 8.4-2 has a  $140^{\circ}$  field of view, it also has a limited effective aperture. This means there is a relatively limited amount of light transmitted through the probe. However, the  $140^{\circ}$  from the probe may be split into three narrow beams ( $40^{\circ}$  to  $45^{\circ}$  each, each of which is

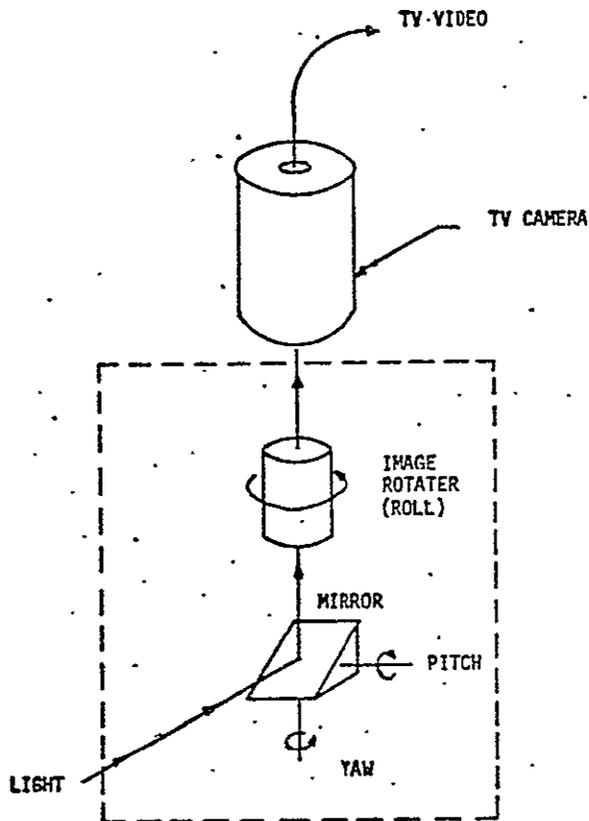


FIGURE 8.4-4 SCHEMATIC DIAGRAM OF ARTICULATED OPTICAL PROBE  
 (SIMULATES SPACECRAFTS THREE ANGULAR DEGREES OF FREEDOM)

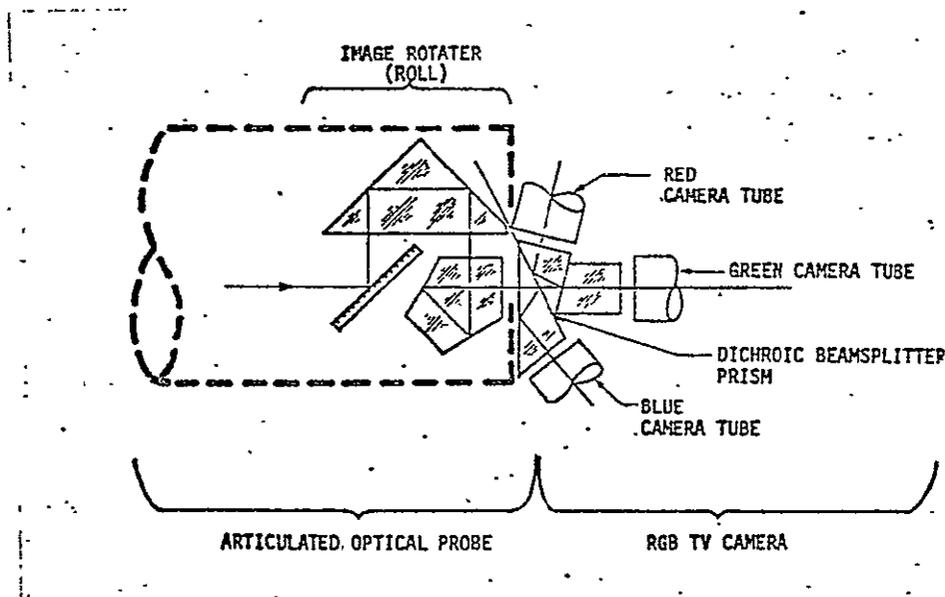


FIGURE 8.4-5 AN OPTICAL INTERFACE OF OPTICAL PROBE AND RGB CAMERA

directed to a separate TV camera in order to achieve greater resolution. If an RGB system is implemented, each of these would require additional splitting into three component colors. Consequently, in a three camera system, the availability of light at the camera tubes, for achieving the required image quality, becomes a problem. The alternatives are reduced requirements for color fidelity; reduced field of view and/or resolution requirements; or use of alternate color encoding schemes.

The color coding scheme proposed in Reference 8.4-4 enables coding of the color information with a single tube but with a reduction in color quality. Reference 8.4-5 graphically summarizes the various single and multiple tube color systems in common use. In either case, the major consideration with respect to the objectives of this study is that the signals of closed circuit TV systems using any of these concepts are essentially similar to closed circuit black and white TV signals. Consequently, although the test requirements and techniques may require some modification or extension, they are, nevertheless, characteristic of CCTV systems which may be incorporated in future visual simulations.

#### 8.4.1.2 Characteristic Parameters and Test Signal Requirements

Over many years of TV development, various TV test signals have been developed and applied to test specific TV parameters. For the present analysis, it is not practical to identify all of the parameters that are likely to be of interest since these will be dependent in part on the characteristics of the particular systems that are developed. However, it is possible to identify certain basic parameters which are very likely to be measured quantitatively if a self test concept is implemented and to then establish the test signal requirements for measuring these parameters.

The characteristic parameters to be considered for this discussion are summarized in Figure 8.4-6 along with the TV system units that they are applicable to. It should be noted that the switching devices have been omitted from the list of units since verification of proper switching imposes no unique requirements in this specific case of the closed circuit TV system. It should also be noted that the parameters characteristic of the camera optics and the display

	Characteristic Parameters Equipment Tested	(1) Voltages (P/P)		(2) Setup	(3) Reso- lution	(4) Gamma	(5) Signal Noise Ratio	(6) Differ- ential Gain	(7) Low Frequency Streaking	(8) High Frequency Ringing	(9) Linearity	(10) Convergence or Registration
		Composite Video	Video									
Readiness Tests	Optical End to Electrical End	X	X	X	X	X	X	X	X	X	X	X
	Electrical End to Electrical End	X	X	X	X	X	X	X	X	X		
Fault Isolation Tests (LRU'S)	Camera (Tube Output)		X		X	X	X	X	X	X	X	X
	Camera (Pre AMP Input)	X	X	X	X	X	X	X	X	X		
	CCU Camera Control Unit	X	X	X	X	X	X	X	X	X		
	Keying	X	X	X	X	X	X	X	X	X		
	Display Monitor (at grid)	X	X	X	X	X	X	X	X	X		
	Display Monitor (optical)				X	X			X	X	X	X

FIGURE 8.4- 6 CHARACTERISTIC PARAMETERS FOR TV READINESS AND FAULT ISOLATION TESTS

tube optics are also appropriate for implementation of an end to end check, beginning and ending with an optical display or image. Similarly the electrical parameters characteristic of the electronics may be used for an end to end electrical check.

The various characteristic parameters shown in this case represent either performance criteria or symptoms of malfunctions. The number of components in a TV signal chain is so numerous that it is not really feasible to consider the ramifications of failures of each of them. Consequently, the approach taken in this one instance is to select those characteristic parameters that have evolved as indicators of proper or faulty performance.

The characteristic parameters are discussed briefly below with reference to the typical test signal shown in Figure 8.4-7. This signal represents one line of the TV image. The parameters may be described as follows:

- (1) Voltage (Peak to Peak)
  - (a) The composite voltage is measured from peak white of the video (top) to the peak bottom of the sync pulse, Figure 8.4-7.
  - (b) Video voltage is measured from peak white of video (top) to the blanking level.
  
- (2) Setup is an arbitrary "setup" of the downward going black video peaks above the sync. level in each of the test signals shown. "Setup" values have ranged during black and white and color TV standards development from approximately 7.5% to 10% of the top of the sync pulse to 100% peak white video. The precise amount of setup will be determined during design phase. Setup prevents any accidental excursion of black picture information into sync. level interfering with the synchronization and also gives a little leeway in the brightness control setting on the display so that full black can be produced without the possible danger of seeing any artifacts of sync.

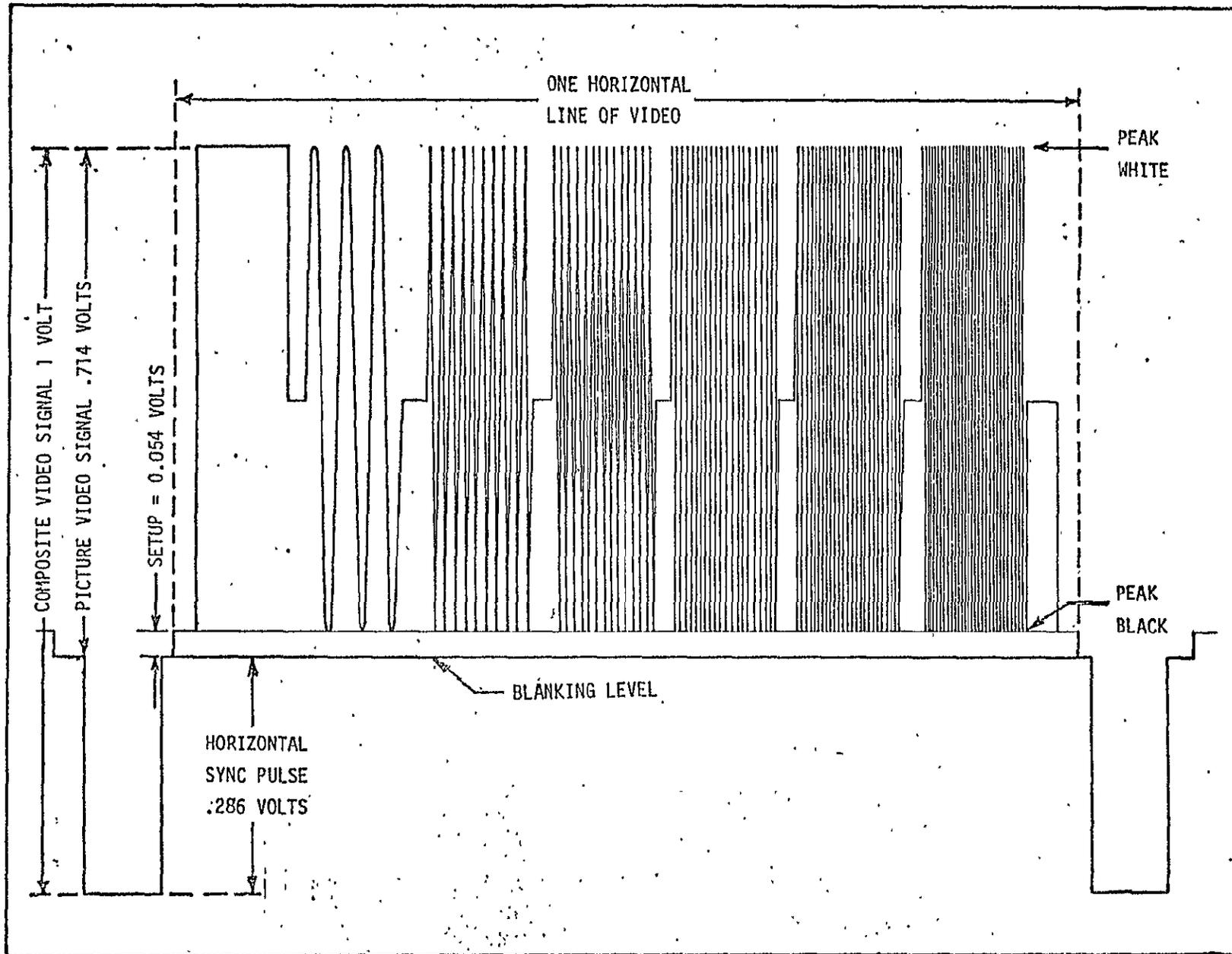


FIGURE 8.4-7 ONE HORIZONTAL LINE OF TV TEST SIGNAL (TYPICAL)

- (3) Resolution for test purposes refers to horizontal resolution since the vertical resolution is fixed by the number of scan lines for which the system is designed. Resolution is synonymous with frequency response since any degradation in high frequency response degrades the number of vertical lines which may be resolved.

- (4) Gamma is the exponent of the power law which is used to approximate the curve of output magnitude versus input magnitude over the region of interest, Gamma is usually applied to a camera tube, relating input luminous flux to output voltage but can also be applied to any TV video circuit (Reference 8.4-6).

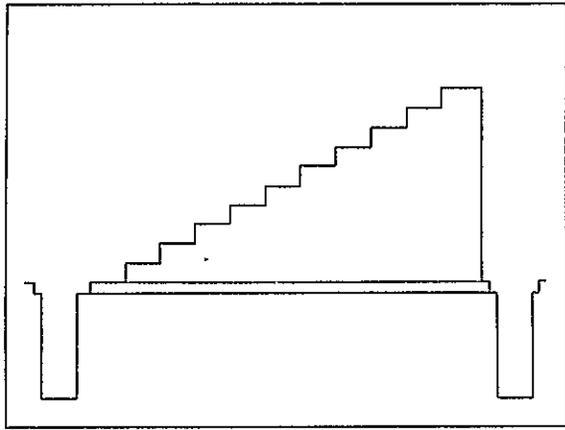
In engineering practice it is usual to plot input-output characteristics in linear units, e.g., foot-lamberts and volts. The general transfer function, in linear units, is

$$Q_o = KQ_i^g$$

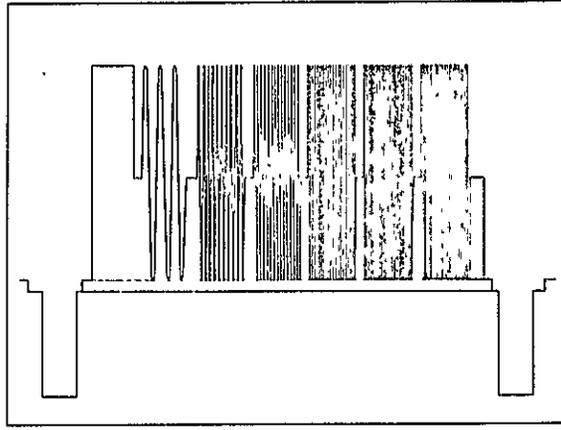
where  $Q_o$  is the output quantity resulting from the input quantity  $Q_i$ ,  $K$  is a factor relating the scales of the two quantities, and the exponent  $g$  is called the gradient or gamma

In simpler but much looser terms, gamma appears as though it were contrast. The steeper the curve, the more contrasty - the more black and white - the picture appears. The less steep the curve, the more "washed out" or grayish the picture.

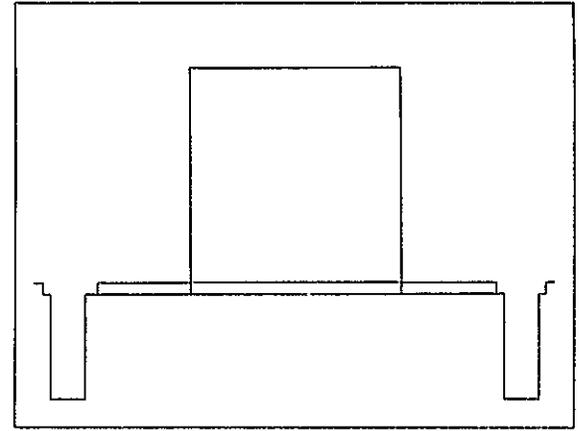
- (5) The signal to noise ratio is commonly defined as the ratio of the peak to peak of the video signal to the peak to peak noise. The ratio is commonly expressed in decibels.
- (6) Differential gain is the difference in gain at high signal levels as compared to the gain at low signal levels.
- (7) Low frequency streaking appears as the slow recovery to zero amplitude of the video signal following a very sharp drop or rise in signal level.



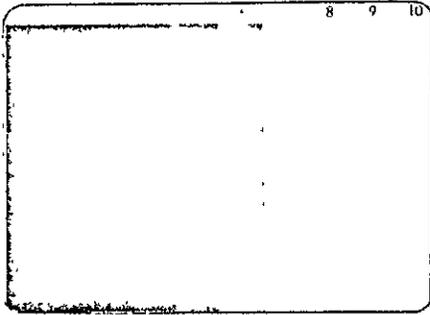
(a) Horizontal Line of Stairstep Signal



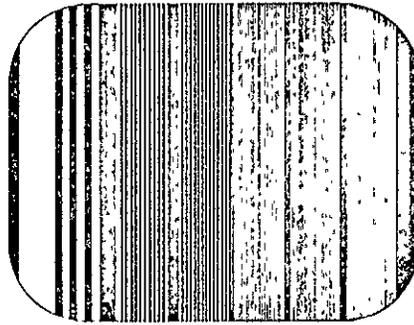
(c) Horizontal Line of Multiburst Signal



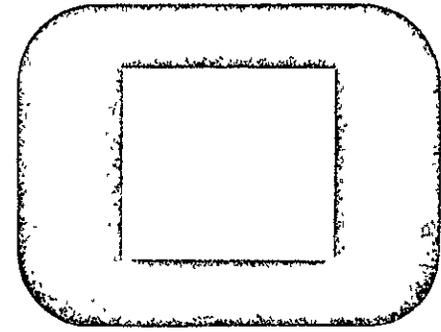
(e) Horizontal Line of Window Signal



(b) Display Image of Stairstep Signal



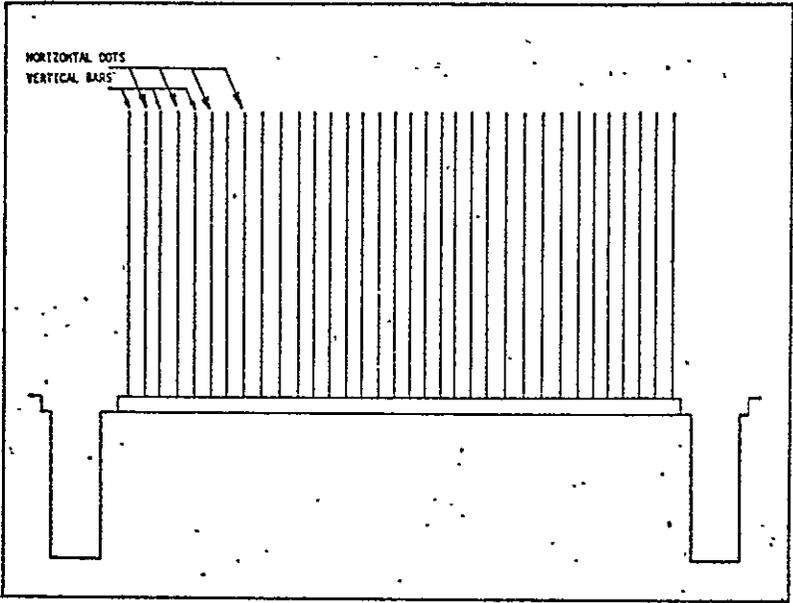
(d) Display Image of Multiburst Signal



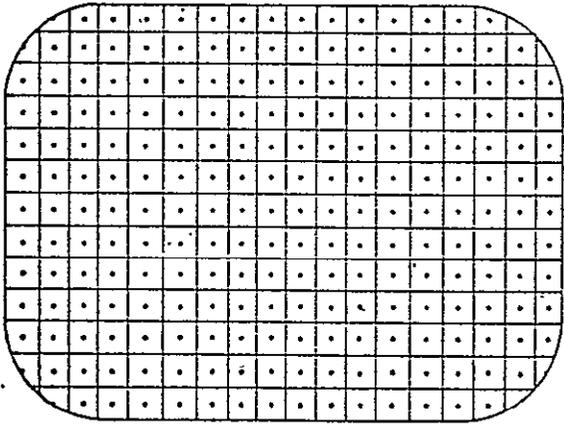
(f) Display Image of Window Signal

FIGURE 8.4-8 TV TEST SIGNALS AND THEIR IMAGES ON THE DISPLAY

REPRODUCTION OF THE ORIGINAL PAGE IS POOR



(g) Horizontal Line of Linearity Test Signal



(h) Display Image of Linearity Test Signal

FIGURE 8.4-8 TV TEST SIGNALS AND THEIR IMAGES ON THE DISPLAY (Continued)

- (8) High frequency ringing appears as decaying high frequency transient oscillations following a sharp signal level step up or down.
- (9) Linearity is a characteristic parameter for the camera and for the CRT's in the display. Linearity is measured by the capability of either the camera tube or the CRT to represent equally spaced lines or dots with an accurate representation of their spacial positions in any area of the image field.
- 
- (10) Registration is the capability required in a multi-image system, such as an RGB system, to superimpose the images with a high degree of coincidence. Lack of registration is quite often evident in color pictures printed in newspapers where the images printed with different colored inks do not coincide. Convergence is the related property of the multi-beam CRT tube. Again coincidence of the component images is the characteristic being assessed.

Over many years of TV development, various TV test signals have been developed to measure these parameters and others. The tests shown in Figure 8.4-8 are sufficient for measuring all of the above parameters in most instances by reference to the displays of a single line as produced on an oscilloscope. Before addressing self test techniques for the TV system it is therefore of interest to note briefly how the parameter measurements can be obtained from these test signals.

- (1) Peak to Peak voltages are readily measured from the oscilloscope trace of an infinitely repetitive signal such as the stair step or the multi-burst. (See Figure 8.4-7).
- (2) Set-up voltage is readily measured in a similar manner. (See Figure 8.4-7).
- 
- (3) Resolution can best be measured using the multiburst signal shown in Figure 8.4-8(c). The image it produces on a display is seen in Figure 8.4-8(d). As is seen this signal is a series of short bursts of signals increasing in frequency with each burst - hence "multiburst". In

essence this signal produces a sweep generator for the frequency span indicated and at a sweep rate of once per horizontal TV scan line. Quite normally the upper frequency bursts will decrease in amplitude after passing through the unit under test. Actually, the downward roll off of the envelope of the test output of the multiburst signal is the frequency response curve of the unit under test. Since the frequency response bears a direct relationship to the resolution, the latter is easily obtained.

- (4) Gamma is readily measured using the stair step signal. An illustration of gamma degradation is shown in Figure 8.4-9. It may prove more accurate to use a linear ramp instead of a stair step for the gamma measurement. The final determination can be made during test development.
- (5) Measurement of signal to noise ratio in a video signal is a relatively difficult problem. The presence of sync and blanking pulses and noise at frequencies outside the frequency range of the video system makes direct repetitive measurement difficult. The problem has been solved in a specialized metering system produced by Rhode and Schwartz, Reference 8.4-7. This unit contains the necessary processing, including filters, for obtaining accurate (.1-.2 dB), repeatable values for signal to noise ratio.
- (6) Differential gain is measured with the use of the staircase. It is the difference in gain between different steps - usually measured as a partial differential gain between the top most and the bottom steps. It might be parenthetically added here that if the gamma has been obtained the partial differential gain information already has been obtained (but not reduced to the parameter as explained above). Therefore, this quick and simple measurement provides the information for use in analysis.
- & 8) , Low frequency streaking and high frequency ringing can be measured by the use of the window signal (Figure 8.4-8(e)). The image it produces on a display is seen in Fig. 8.4-8(f). A low frequency streaking is measured as the offset from zero amplitude of the video following

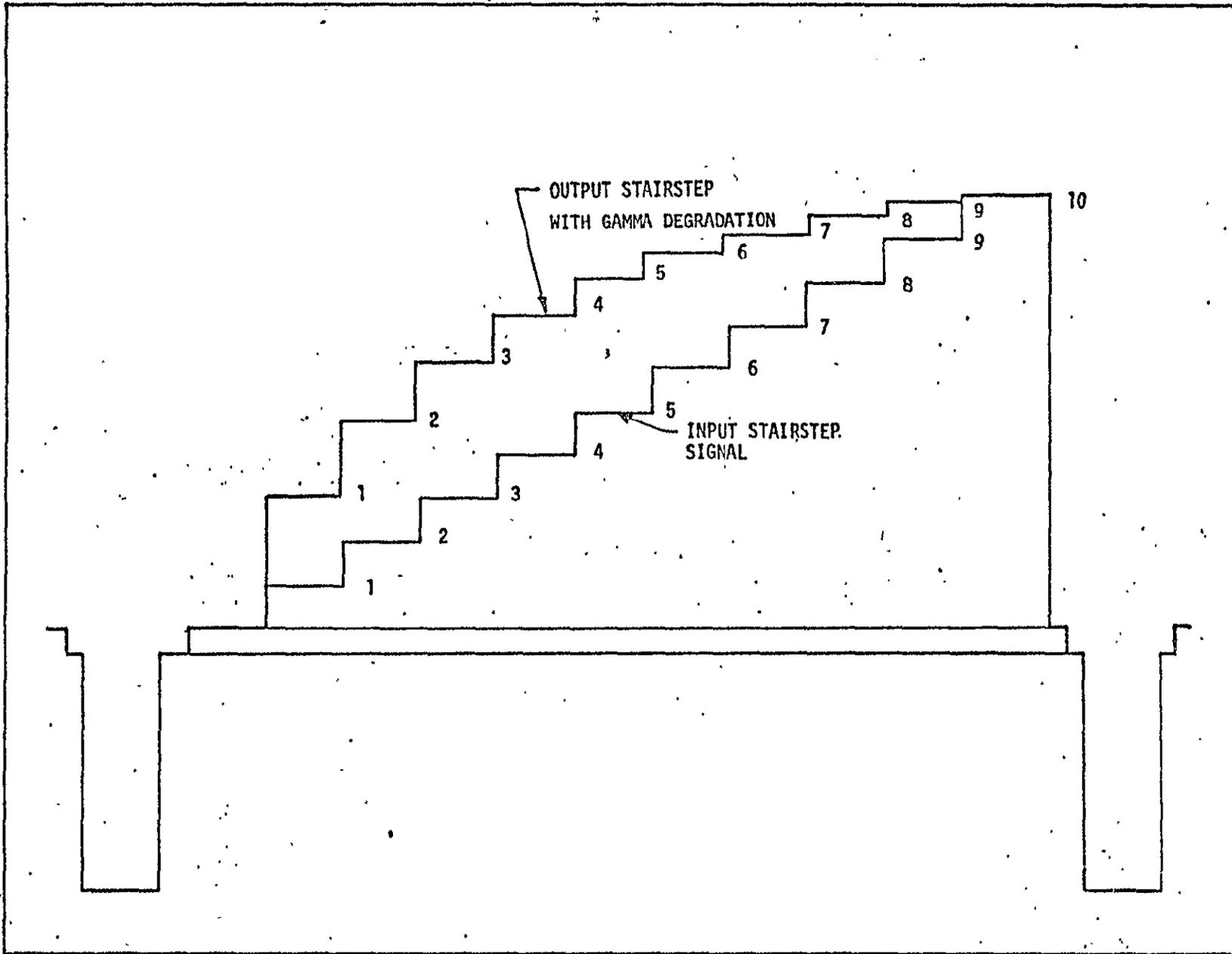


FIGURE 8.4-9 GAMMA DEGRADATION AS APPARENT FROM A STAIR STEP TEST SIGNAL

the window - or the slow recovery to zero from a falling signal with a sharp downtime. Other factors can be measured with the window signal such as tilt, i.e., how flat is the top of the window signal which in turn is a measurement of the DC response. It is apparent that high frequency ringing will also present itself for measurement on the sharp rise and fall times.

- (9) Camera linearity appears as the reproduction of equidistant points or lines in a test chart on which the camera is focused as equidistant voltage spikes in the camera output signal. Display linearity is perceived as the equidistant spacing of the lines or points in the display field as a result of equidistant voltage blips in the signal fed to the display. The test pattern in Figure 8.4-8(h) can be used to test camera linearity.
- (10) The registration of the three cameras in an RGB system can also be tested with the pattern of Figure 8.4-8(h). Superposition of the line trace from the cameras focused on this pattern can be used to measure registration errors. The signals generated by such a pattern can be applied to a color CRT to test for convergence of the three beams.

These ten parameters have been measured for TV systems for years by the use of a standard generator of the signals described and a trained operator interpreting their readouts on an oscilloscope. Such a standard TV test signal generator (or one modified to accommodate the upgraded TV requirements for simulation visuals, i.e., line scan rates, bandwidths, etc.) can be readily obtained. However, since these visuals tests are intended to be as fully automatic as feasible, the elimination of the operator/interpreter is desirable. An automatic test/validation of a complicated color TV Visuals system such as required in simulation/training would be much more rapid and accurate than with a human interpreter "in the loop". It would also facilitate storage of measurement data for day to day, and week to week comparisons for purpose of incipient fault detection.

#### 8.4.1.3 TV System Self Test Techniques and Hardware Requirements

In order to implement a self test concept for the TV system, it is necessary to sample the response signal produced by use of standard test signals, previously discussed, and digitize these samples for processing by digital computers. The A/D channels nominally available in the simulator DCE are wholly inadequate for the problem, the magnitude of which can be revealed by looking at a few critical parameters.

The nominal bandwidth of broadcast quality TV is approximately 5 MHz. Increased resolution requirements for visual simulations and the use of a 900 line system can easily increase the required bandwidth to 20 MHz. For the multiburst test the highest frequency burst represents nearly the bandwidth of the system. Sampling theorems require at least two samples per cycle to communicate the information present at a given frequency. This would require a 40 MHz D/A conversion rate which is well beyond the state-of-the art in A/D equipment. Consequently, an alternate means must be found for acquiring the data necessary for analysis.

Since the TV scan lines generated by test patterns or test signal generators are repetitive wave forms, sampling can be spread over several cycles or repetitions of the wave form, thus reducing sampling rate requirements drastically. Alternately, the data could be acquired from a single sweep or line by transient analysis techniques which might involve multiple parallel sample and hold amplifiers as well as parallel A/D converters. Obviously, the second approach would require hardware for the data acquisition function that would produce no added insight into the problem when dealing with repetitive signals. Consequently, the approach taken for the present study has been to explore the test capabilities available with current state-of-the-art oscilloscopes because of their basic applicability to repetitive signal analysis.

An additional capability available with oscilloscope systems that is particularly useful when working with TV test signals is the ability to select segments of a line sweep for further processing without having the data distorted by blanking signals or sync pulses. In particular, the processing of the multiburst is highly simplified since the amplitude response of each frequency burst may be measured directly by changing the gating for the test signal.

The gating capability of the oscilloscope may be used to eliminate the blanking and sync pulses when making signal to noise ratio measurements. However, the broad bandwidth of the oscilloscope may create problems by detecting noise signals outside the video transmission range. Since these signals do not affect TV system performance, they may contribute to unrealistic or irregular noise figure measurements. Should this be the case, a specially designed noise meter with the necessary filters, such as the Rhode and Schwartz type UPSF, may be required. The UPSF has an oscilloscope output which would permit it to be tied in to the data acquisition system for the other tests. Further information on the Rhode and Schwartz metering system may be found in Reference 8.4-7.

Currently, there is only one oscilloscope available commercially, that incorporates or provides the necessary A/D hardware and buffer memory required to interface signal measurements with digital computing equipment. Typically, however, when one system of this type is introduced, additional systems are likely to follow, and the initial manufacturer may be expected to come out with further variations on his own model in order to more efficiently tailor his system to specific applications. For purposes of this study, we will discuss the presently available system and how it can be introduced into the visual simulation system to implement self test capability.

Tektronix has recently introduced a new system based on a unit that is called the Digital Processing Oscilloscope or DPO, References 8.4-7 and 8.4-8. Figure 8.4-10 is a photograph of a stand alone system incorporating the DPO and using a dedicated DEC PDP 11 as the computer for the system. The Tektronix DPO has several key features which are required for implementing an automatic or self test capability for the TV system in a typical simulator. First, repetitive signals which can be displayed on the scope can be digitized and stored in a four thousand word buffer memory. The number of samples is 512 over whatever range of the signal is displayed on the scope. Since the sampling is spread over several cycles or repetitions of the signal as previously suggested, a ramp signal is generated with the same frequency as the sweep frequency. The ramp signal is sampled at the same times as the test signal

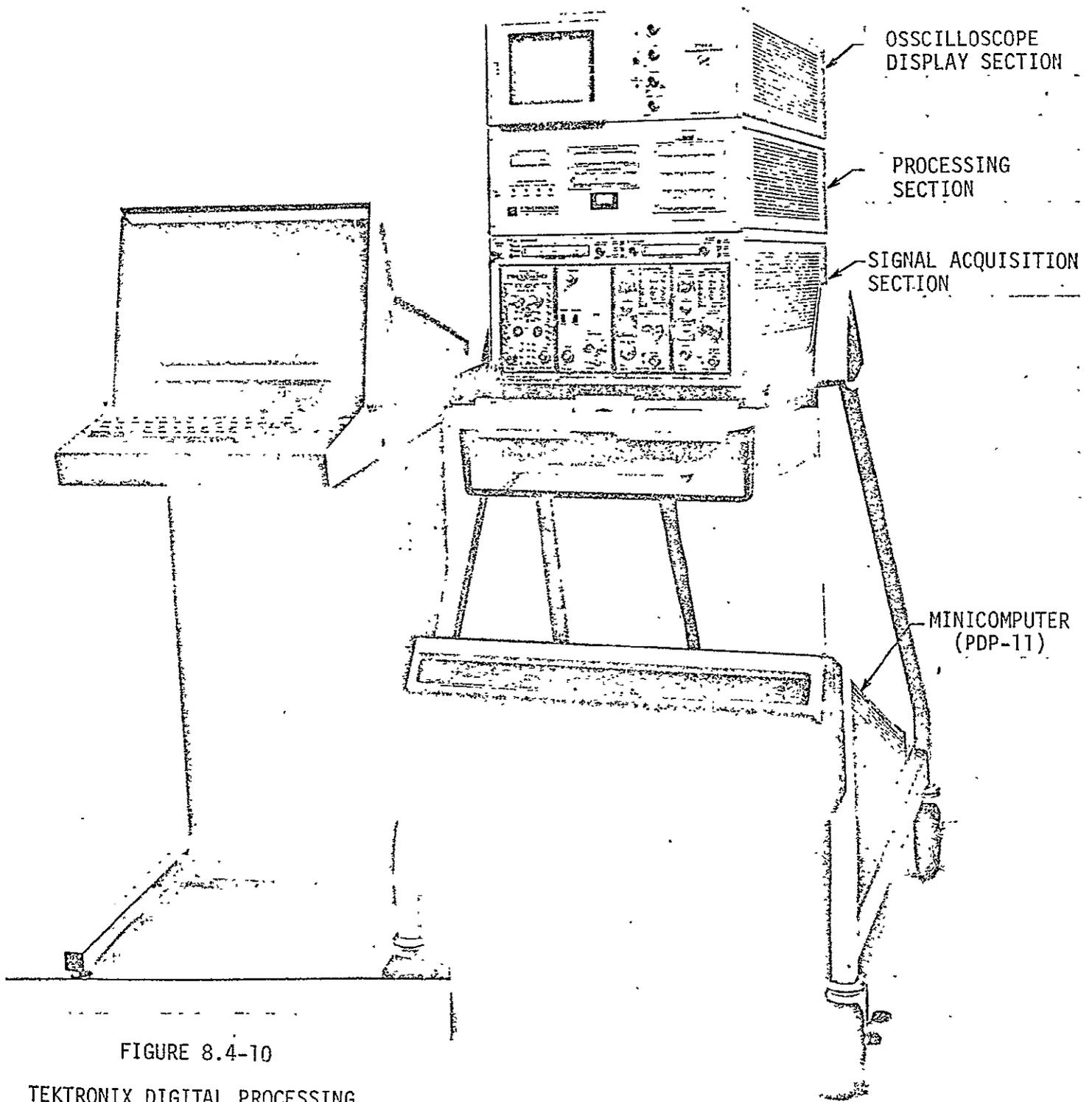


FIGURE 8.4-10

TEKTRONIX DIGITAL PROCESSING  
OSSCILLOSCOPE SYSTEM

REPRODUCIBILITY OF THE  
ORIGINAL PAGE IS POOR

And the value of the ramp signal is used for addressing the coincident data sample. Consequently, when a signal is sampled, it produces 512 data points and 512 addresses. The data storage precision is 10 binary bits. The available bandwidth is up to 14 GHz which is way beyond the requirements for the TV systems anticipated.

Secondly, Tektronix has available several time bases with TV sync separator triggering. This permits stable internal line or field rate triggering from displayed composite video or composite sync waveforms. Conventional waveform displays and measurements can be made from closed circuit TV systems with up to 1201 line, 60 Hz field rates. Individual lines can be displayed with the delayed sweep feature. Since anything displayed can be digitized for processing, this enables processing of waveforms such as those generated by test patterns such as the window and the grid, Figure 8.4-8(f) and (h).

The stand alone system shown in Figure 8.4-10 is exactly that. The PDP-11 computer shown under the oscilloscope, has a fully developed set of software to permit implementation of most processing operations that may be required, including Fast Fourier Transform, differentiation, integration, averaging, etc. The terminal permits programming to develop specific test sequences or special processing facilities. Any operation that can be implemented manually on the front of the scope can also be programmed for automatic operation or control from the PDP-11. Hard copy equipment can also be attached to produce permanent copies of displays from the scope or the terminal. Tektronix obviously has available the hardware for interfacing any of the DEC PDP 11 series to the DPO as well as support software compatible with this computer. Whether for particular applications such as a simulator visual simulation, the choice of this ancillary equipment would be the best approach is a question that must be decided for each particular case. The alternative is to integrate the DPO or its equivalent more fully with the simulator's available facilities and make maximum use of available hardware.

The DPO system as shown in the photo is not totally adequate for TV system testing. An additional unit is required to generate the electronic test signals that were described in Figure 8.4-8. It should be noted that, depending on

the unit under test, the signal source must be a test pattern image on which the camera may be focused, or it may be an electrical signal that is the output of a signal generator such as the Tektronix Test Signal Generators, Models 147A (\$3270) or 148 (\$3800). Similar equipment is available from other manufacturers.

The test circuitry recommended to instrument one chain of the TV electronics is shown schematically in Figure 8.4-11. Additional switching is required for the additional chains or video signal paths. Additional switching is also required to provide test signal selection capability at the TV test signal generator. This will be dependent on the particular unit selected and possibly on the numbers of tests used.

The use of solid state switches (MOS/FET) is recommended as for the DCE, Section 8.3. These switches can be installed such that normal simulation activities can be carried on as long as no switching command is applied to the switches.

A signal/noise ratio meter, such as the Rhode and Schwartz, is shown in the circuit. This meter would require switching to sample the signal at the same points as the scope. The meter may prove optional during development of the test equipment.

The computer interface shown in Figure 8.4-11 is not assumed to be the PDP-11 that Tektronix equipment is usually interfaced with. The decision as to what computer equipment is required must be made at the time the test system is developed for a specific simulator, since the nature and extent of simulator computer facilities must be considered.

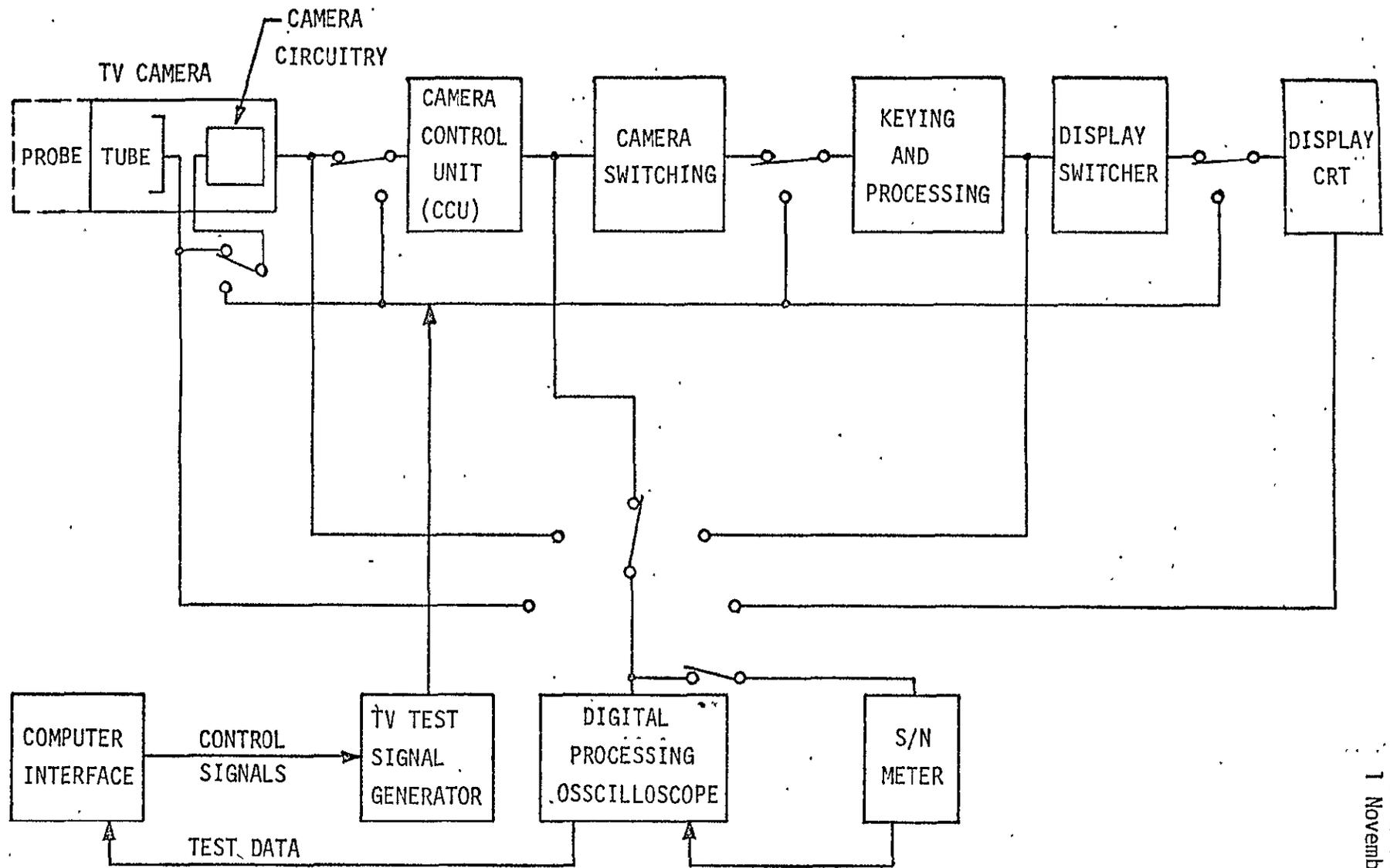


FIGURE 8.4-11 TV SELF TEST HARDWARE SCHEMATIC

## 4.2 Display Equipment

For the present study, the display equipment is assumed to consist of a shadowmask color kinescope viewed through a non pupil forming infinity display system. The self test problem with respect to this subsystem is basically that of providing an automatic means of perceiving or acquiring performance data generally available to only a human observer. In addressing this problem we will consider the following topics:

- o Description
- o Characteristic parameters
- o Self test techniques
- o Recommendations

### 8.4.2.1 Display Equipment Description

For purposes of this study it is assumed that the color display is presented on a shadow mask color kinescope such as shown in Figure 8.4-12. Recently a new high resolution shadow mask color tube has been introduced by RCA which offers a 900 TV line resolution picture at 50 foot lamberts peak white. The tube size is 25 inches diagonally. The RCA type number is C74957. As noted in Reference 8.4-4, it is believed that this is the only tube available which can achieve the resolution requirements of the Shuttle training simulators. Such a tube is shown incorporated into an infinity view optical display system, based on Reference 8.4-10, in Figure 8.4-13.

### 8.4.2.2 Characteristic Parameters

The characteristic parameters associated with the image as presented on the kinescope were identified in Figure 8.4-6 as consisting of the following:

- o Resolution
- o Gamma
- o Low frequency streaking
- o High frequency ringing
- o Linearity
- o Registration

Based on the test signal image displays shown in Figure 8.4-8, it can be concluded that the resolution, linearity and registration measurements require accurate positional readings of line and point locations on the CRT face either directly or through the optics. On the other hand, the gamma, streaking

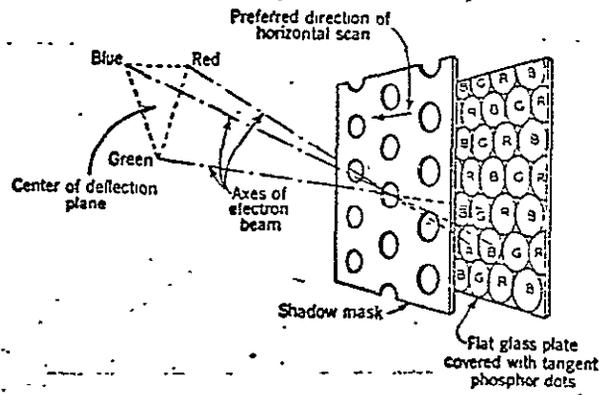
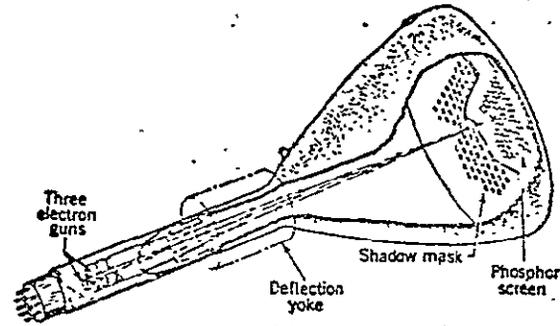


FIGURE 8.4-12 SHADOW MASK COLOR KINESCOPE DETAILS

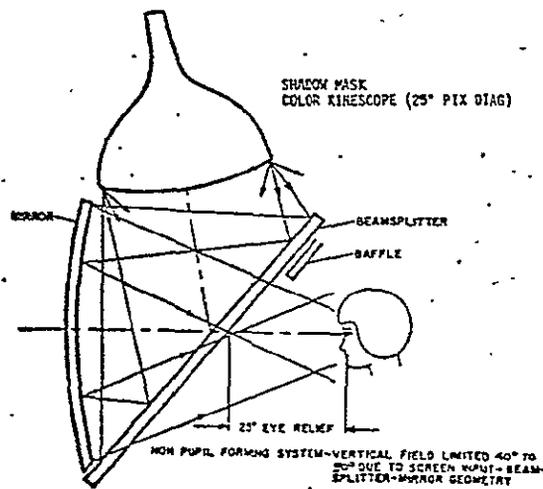


FIGURE 8.4-13 REFLECTIVE NON PUPIL FORMING INFINITY DISPLAY SYSTEM

and ringing measurements also require measurement of illumination level although with a lesser requirement for positional accuracy.

#### 8.4.2.3 Self Test Techniques

The characteristic parameters for the display CRT present no peculiar difficulties with respect to processing requirements. Many of these parameters are processed as part of the TV electronics testing. Consequently the only new difficulty in testing the display automatically is the data acquisition problem. Unfortunately, this is a major problem.

The use of TV cameras to view the CRT's has been ruled out categorically because of interference problems which can be expected between the camera scanning and the CRT sweep patterns. This interference would be apparent as Moire' patterns which would obviously obscure the desired signal information.

An alternate conceptual approach is the use of photo transistors as point light intensity sensors. Resolution capability of the detector which must look through the thick ( $\approx .5$  inches) layer of glass on the tube face to sense the intensity of illumination of the phosphors is a critical consideration. The sensor must be as close as possible to the tube face and may require a small lens to limit its field of view. To sense one line of a 900 TV line image on a 25 inch tube requires the capability of sensing the intensity of an image less than .03 inches in diameter. This is in the neighborhood of photo transistor dimensions commonly available. A final determination can only be made for a specific visual system design.

Accurate positioning of the sensor on the tube face requires servo positioning systems on two axes. The use of the hardware and software from a commercially available digital X-Y plotter suggests itself as readily available. The advantages of using a drive from a digital X-Y plotter are as follows:

- o No servo system development required
- o Extremely precise positioning of the sensor at any point on CRT
- o Available hardware interfaces to digital computers
- o Software available for driving plotter

The major development item and expected difficulty in applying this approach is

the repackaging required for the plotter drive motors and electronics which must be located so as not to interfere with or degrade normal simulation operations. The general shortage of space in display optics and the need for minimizing weight on the motion base are additional disadvantages. Finally, either the plotter drives and sensors must be provided for each CRT in the simulator or there must be provisions for them to be moved, possibly on a track, from one CRT to another, either by servos or operator intervention. In the latter case, the effort required by the operator to relocate the self test hardware may exceed the time required to verify CRT operation by direct operator observation. Provision of separate drives for each CRT is likely to be considered exorbitantly costly unless an unusually inexpensive source for the drives is located.

Photo transistors are relatively inexpensive and the alternate approach of using a grid of sensors that could be slipped into position on the face of the CRT has been considered as well as the option of using a single axis servo drive. The relative capabilities of these alternatives is summarized in Figure 8.4-14.

#### 8.4.2.4 Recommendations

The self test techniques identified above are highly dependent, with respect to their feasibility, on the particular features of the visual simulation equipment being addressed. Assuming basic feasibility, it is evident that substantial development activity is required for any of the alternatives. This development activity must consider packaging limitations of the display optics and the number of displays to be checked. The value of automating fully the checkout of the display hardware seems questionable considering that there may always remain additional checks or adjustments that must be performed manually.

The provision of self test features for the TV electronics, Section 8.4.1, would make available the capability of implementing effective and rapid interactive checkout of the display electronics by the operator. The automatic portion of this testing would consist of cycling desired test patterns, generated by the TV test signal generator hardware, to each display in turn.

CHARACTERISTIC PARAMETERS	DISPLAY SCANNING CONCEPTS		
	DISCRETE POINT SENSOR GRID	SINGLE AXIS SCAN (DIAGONAL)	X-Y SCAN
Resolution	Possibly	Yes	Yes
Gamma	Yes	Yes	Yes
Low Frequency Streaking	Possibly	No	Yes
High Frequency Ringing	No	No	Yes
Linearity	Yes	Yes	Yes
Registration	Unlikely	Yes	Yes

8-221

FIGURE 8.4-14 RELATIVE CAPABILITIES OF ALTERNATE DISPLAY SCANNING CONCEPTS

The operator could provide a ready signal to proceed to each succeeding test and this could be implemented using interactive devices available in the crew station on IOS. Such an interactive approach is recommended. The associated software flow is discussed in Section 8.4.4.

### 8.4.3 Image Generation Equipment Self Test Techniques

This Section discusses self test techniques for the Image Generation Equipment which, as defined above, includes the various spherical and flat models; the drive mechanisms for these models, and the drive servo systems for both camera positioning and optical probe orientation. This equipment is discussed after the Image Transmission and Image Display systems because the automatic self test process related here relies on correct performance of both of these systems; that is, in the Visual Simulation test sequence, the Image Generation Equipment would be tested last. A description of the scene generation equipment is presented, together with a listing of related characteristic parameters, followed by a discussion of failure modes and self test techniques recommended to detect each of these failures.

Only spherical models are discussed here as they present a variety of problems in self test techniques definition which includes those problems presented by flat models. A flat model, such as used in terminal area scene generation has no moving parts other than the camera and probe drives. Techniques used on camera and probe drive checkout for spherical models will also be applicable to the same components in the flat models.

#### 8.4.3.1 Equipment Description

The spherical model systems are implemented with a globe that is precisely controlled in both rotation and inclination. The globe is viewed by a closed circuit television camera through an optical probe. Angular accuracies of the globe mounting assembly are approximately  $\pm 1/2$  minute of arc in both axes during static and dynamic conditions.

The television camera and optical probe assemblies are mounted on top of a movable carriage. The optical system of the probe provides simulation for the roll, yaw, and pitch attitudes of the spacecraft. Movement of the carriage on which the camera and probe are mounted simulates a change in altitude. The carriage is positioned to an accuracy of  $\pm 0.10$  inch under dynamic conditions. Figure 8.4-15 shows, the basic components of the Earth Scene Generation subsystem.

8-224

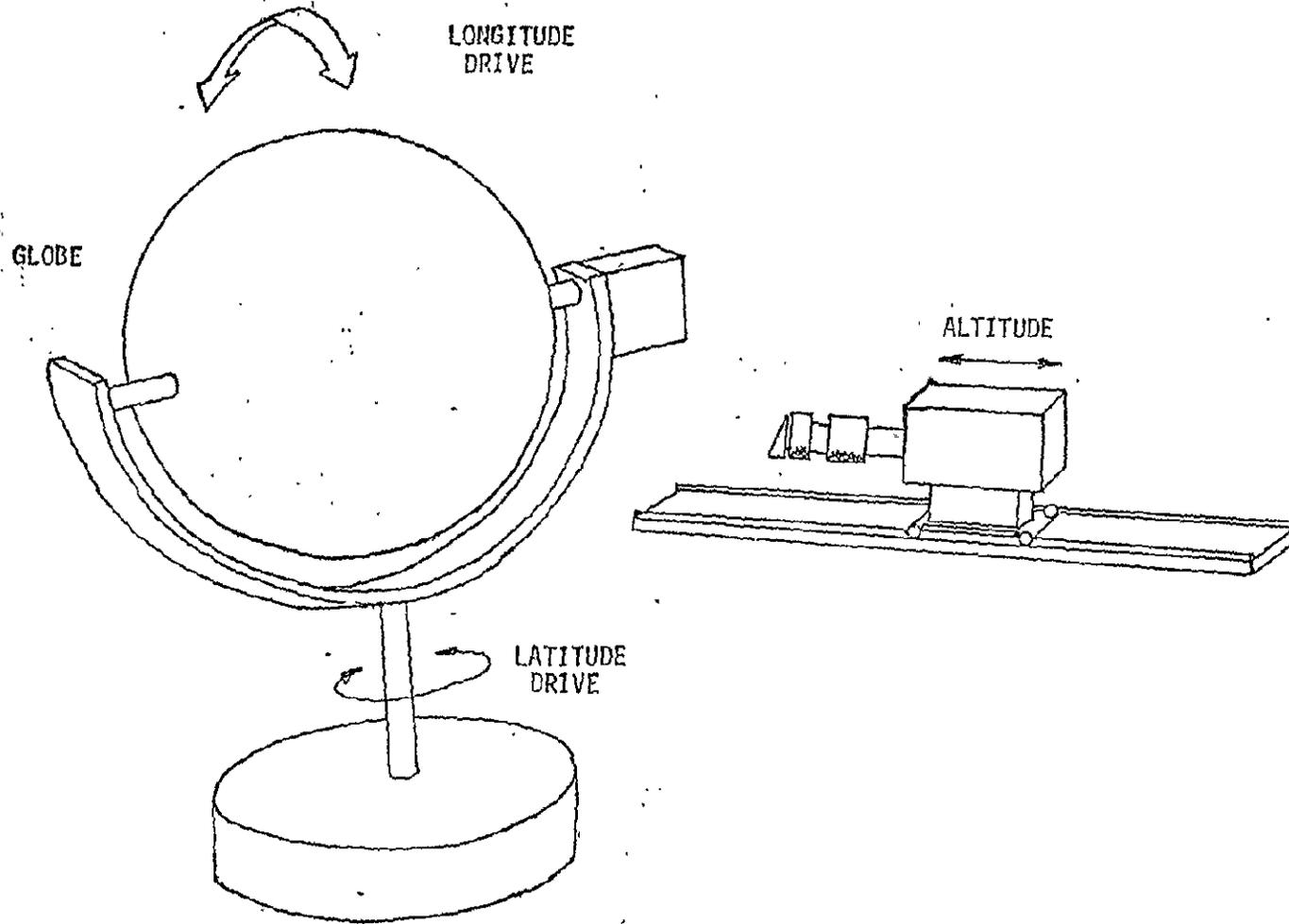


FIGURE 8.4-15 EARTH SCENE GENERATION - BASIC COMPONENTS

The globe is suspended by a gimbal yoke. Servo drive systems are used to rotate the globe about both the polar and equatorial axes. The servo system is a 15-bit digital system as shown in Figure 8.4-16. A shaft encoder outputs angular position in digital format. This output is compared to the digital signal from the computer. The error is converted to an analog signal for driving the DC motors. A tachometer provides velocity feedback to the servo loop for additional stability.

Pitch, roll and yaw motion of the spacecraft is simulated by servo driven prisms within the optical probes. The servo system is a 400 hz, synchro/resolver system as shown in Figure 8.4-17. Sine and cosine signals are sent from the digital computer; these signals are converted to 400 hz analog signals which have output magnitudes proportional to the digital sine/cosine values. These outputs drive a control transformer. The control transformer outputs an error signal whenever the angle input specified by the sine/cosine does not equal the shaft angle of the prisms. This error signal is amplified and demodulated. The resultant is a signal then drives a torque motor which turns the shaft until the command input angle is reached. A tachometer provides rate feedback for increased stability of the servo loop.

#### 8.4.3.2 Failure Modes

The failure modes, shown in Table 8.4-1, for the spherical model with the digital servo system, are described as follows:

No Shaft Movement - No shaft movement is indicated by the shaft encoder output remaining constant with changing input commands. This indicates that no current is reaching the torque motor. This failure is likely caused by a fault in the digital comparison circuit or in a power supply failure. Other possibilities include failures in the D/A converter, amplifier, or torque motor.

Shaft Position At Either Extreme - The shaft remains at the limit stops regardless of command input. This indicates that a constant drive is applied to the torque motor. This failure is likely caused by a shorted output in the amplifier driving the torque motor. Other possible causes include shorted first stage amplifier, shorted D/A converter output, or digital comparator failure.

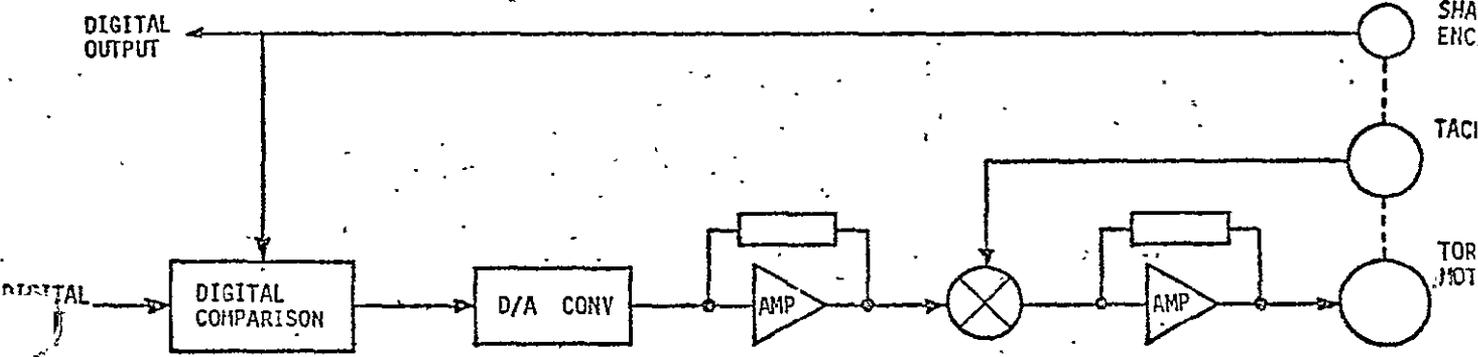


FIGURE 8.4-16 SERVO SYSTEM WITH DIGITAL COMPARISON

8-227

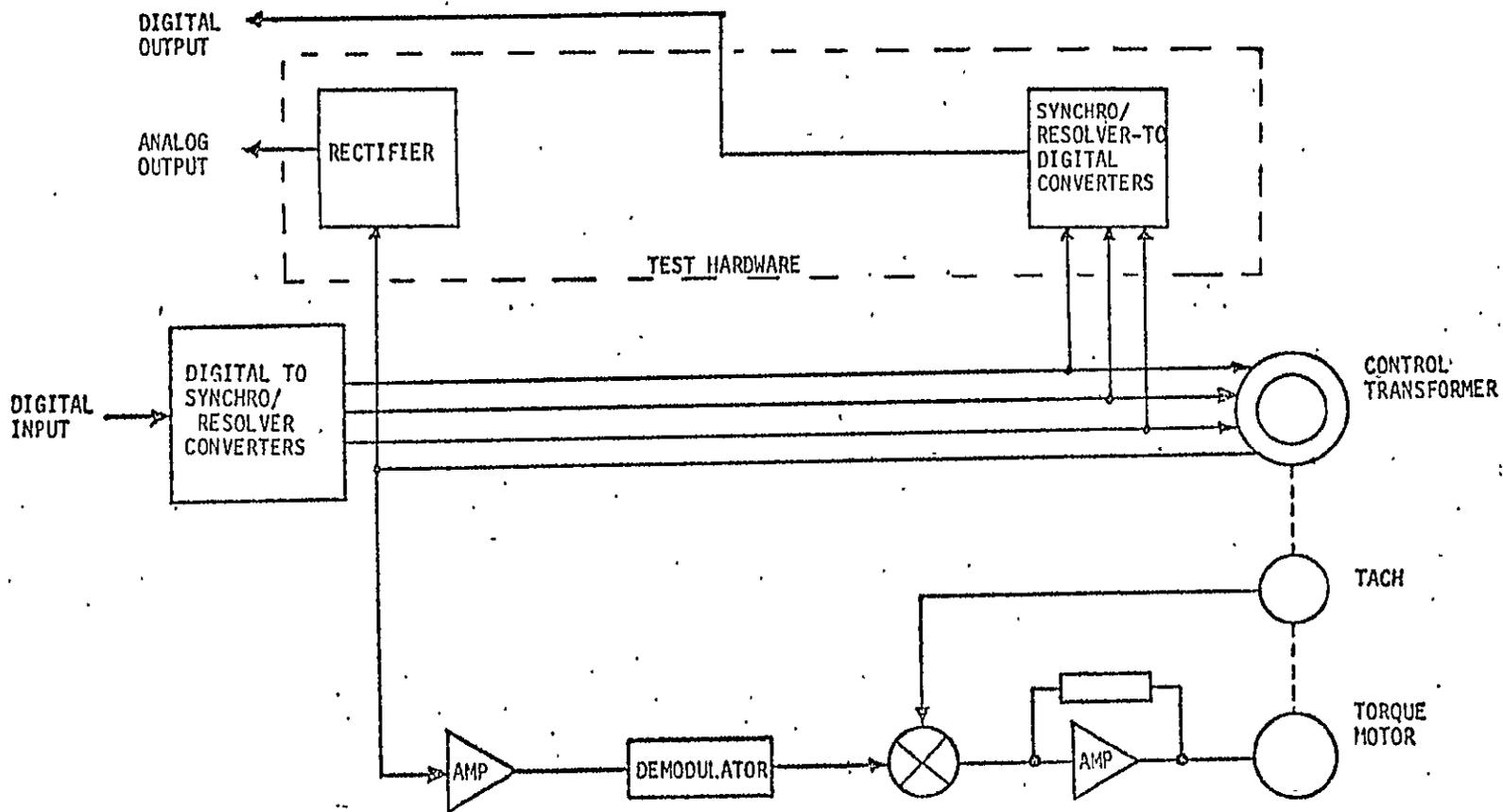


FIGURE 8.4-17 SYNCHRO/RESOLVER SERVO SYSTEM

EXTERNAL SYMPTOM	GLOBE	DIGITAL COMPARATOR	D/A CONVERTER	AMPLIFIER	SHAFT ENCODER	TACHOMETER	TORQUE MOTOR
No Shaft Movement		No Output	No Output	No Output			Possible short or open winding
Shaft Position at Either Extreme		No Comparison	Shorted Output	Shorted Output			
Shaft Position in Error	Mechanical Misalignment	Erroneous Comparison	Output not within Tolerance		Erroneous Output		
Instability in the System				Degraded Amplifier		Erroneous Output	

TABLE 8.4-1 FAILURE MODES FOR THE GLOBE SYSTEM

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

Shaft Position in Error - The shaft moves to a position with a commanded input, but the position is not within the specified tolerance range. This failure is likely caused by a mechanical misalignment of the globe or misalignment of the shaft encoder with respect to the globe shaft. Other possible causes include failures in the shaft encoder, digital comparator, or D/A converter.

Instability In The System - This failure is likely caused by a malfunction in the tachometer or by degradation or failure of components in the amplifiers.

The entire probe unit is assumed to be an LRU for this study. Association of faults with particular components of the probe is not necessary for implementing the desired tests.

#### 8.4.3.3 Characteristic Parameters

The measureable parameters for the spherical model driven by the digital servo system, shown in Figures 8.4-15 and 8.4-16 are as follows:

Input - The input is a 15-bit digital word specifying commanded shaft angle.

Shaft Encoder - The output of the shaft encoder is a 15-bit digital word specifying the shaft angle.

Digital Comparator Output - The output of the digital comparator is a 15-bit word that is the difference between the command input and the shaft encoder output.

Digital-to-Analog Converter Output - The output of the digital-to-analog converter is an analog representation of the digital comparison output.

Tachometer Output - The tachometer output is a DC signal proportional to rate.

The measureable parameters for the probe unit with the synchro/resolver servo system, shown in Figure 8.4-17 are as follows:

Control Transformer Output - The output of the control transformer is a 400 hz analog signal that indicates whether the commanded position has been reached. The output becomes zero when the shaft angle corresponds to the command angle.

Tachometer Output - The tachometer output is a DC signal proportional to rate.

Demodulator Output - The output of the demodulator is a DC representation of the control transformer output.

The characteristic parameters appropriate for evaluation of the performance of both the model and the probe unit servo systems are the following:

- Static calibration
- Rise time
- Peak overshoot
- Settling time
- Steady state error

These characteristic parameters can essentially be obtained by acquiring the above measureable parameters as shown in Figures 8.4-16 and 8.4-17.

#### 8.4.3.4 Image Generation Equipment Hardware Requirements

In considering hardware self test techniques for this equipment it is necessary to recognize that certain calibration procedures and dynamic tests for visual scenes involve an experimental determination of the gains or correlation matrices in the computer software that generates the commands for model or camera motion. These software adjustments may be used to bring into synchronism or proper juxtaposition the scenes generated by several different models or devices. For example, the images of a rendezvous target, the global earth scene, the star ball and the horizon mask are frequently superimposed. These procedures represent performance verification activities and, because of the judgment required for assessment of complex images, are best performed manually. There remain specific mechanical tests that can be automated, and that constitute useful checks in a daily readiness procedure.

The failure modes summarized in Table 8.4-1 represent faults that would appear as either static calibration type errors or dynamic response degradation effects. The characteristic parameters noted in the previous paragraph may be applied to check for both.

The static calibration checks required for model drive calibration are best implemented on the model drive system itself and are highly sensitive to the particular hardware configuration. Typically, a potentially useful approach would involve the imbedding of a light source in the moving element such as the globe. Alignment of the light source with a fixed reference point on the gimbal structure can then be detected by use of a light sensor such as a photodiode previously discussed. Alternately a minute reflecting surface might be provided on the moving component to avoid wiring runs to a moving device. Masking of both the light source and the sensor may be required to provide the necessary precision. The resulting test would simply be a direct check of a mechanical null such as might be implemented manually.

The dynamic response of the various servo systems can be checked to detect the faults that reveal themselves in dynamic performance degradation. The dynamic tests can be implemented using the standardized step response technique described in Section 7.1. The feedback signals from the closed loop servo systems are the most accessible sources of position response data.

A generalized software flow for the test software to assess scene generation equipment for proper static and dynamic operation are discussed in Section 8.4.4.

#### 8.4.4 Visual System Self Test Software Requirements and Data Base Impact

The test software required for implementing visual subsystem self test techniques has been divided into three major software modules. These three modules are VISTST, the top level visual subsystem test module, VIDTST, the video subsystems self test module, and GENTST, the image generation equipment self test software module. Each of these makes use of more fundamental software modules for implementing the required tests and will be discussed further.

VISTST, the overall visual subsystem self test module, shown in Figure 8.4-18, exercises the other two modules, VIDTST and GENTST as required. VISTST is primarily intended for the visual subsystem test executive. However, it seemed reasonable or efficient to make VIDTST a general purpose video or TV hardware test routine. As such, VIDTST is useful for testing either a single string of TV system

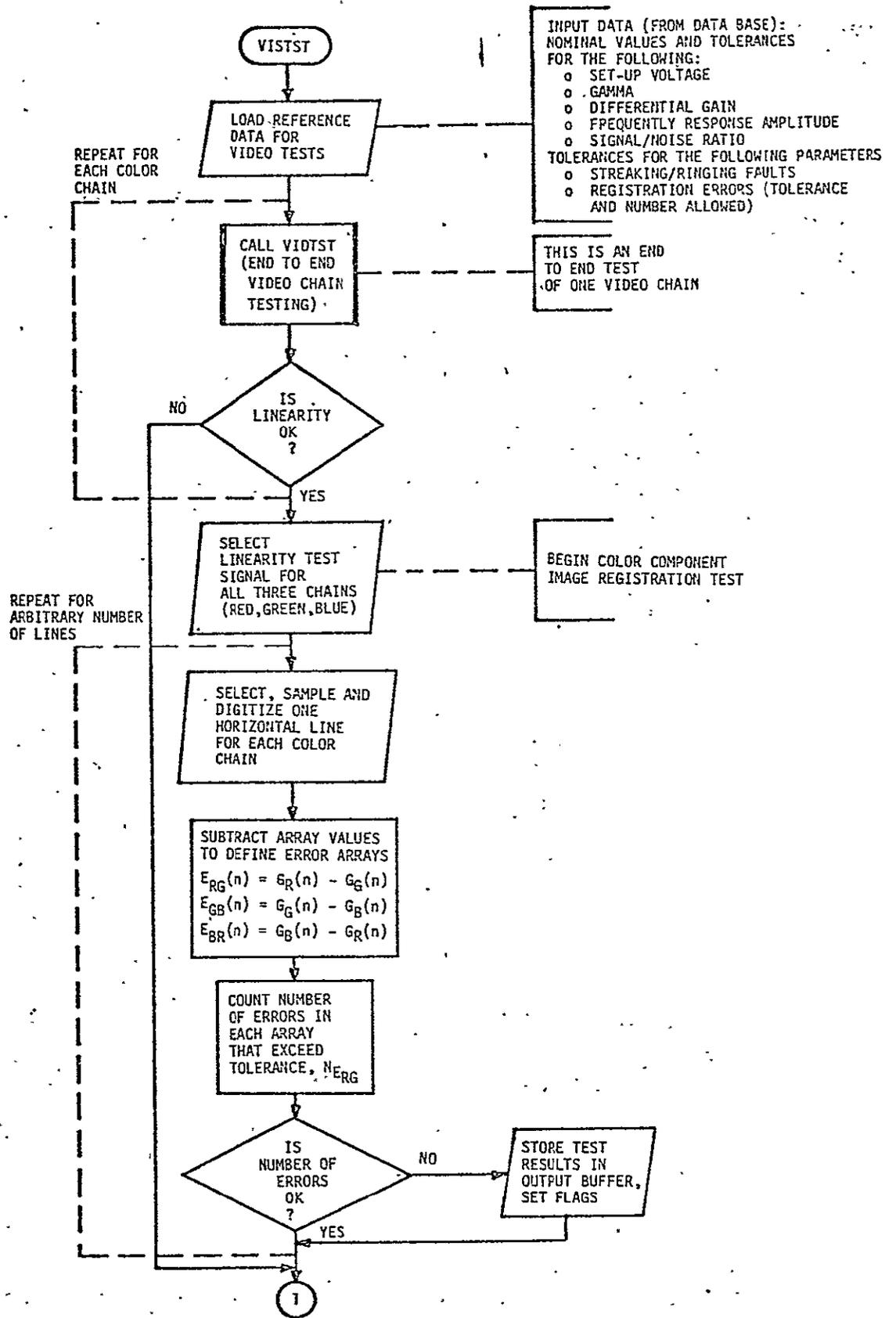
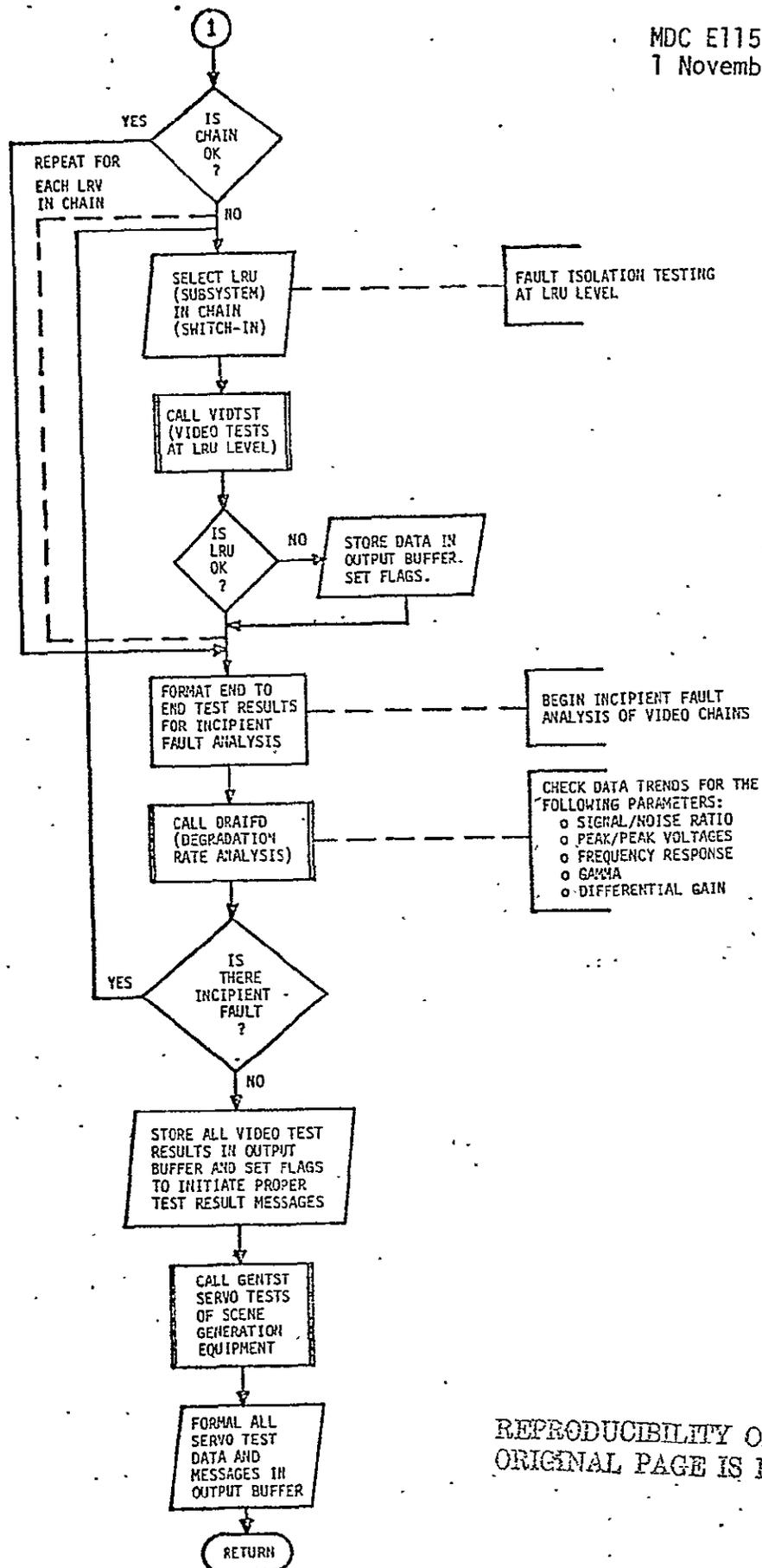


FIGURE 8.4-18 VISUAL SUBSYSTEMS TEST SOFTWARE, VISTST



REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

FIGURE 8.4-18 VISUAL SUBSYSTEMS TEST SOFTWARE, VISTST (CONTINUED)

components end to end, or for testing a single LRU within that string, or for testing any subset of components within the string. Consequently, when dealing with an RGB color TV system, the executive subprogram, VISTST, provides the logic for testing the red, the green, and the blue hardware strings individually and then implements the test for image registration. The registration test is only relevant if the optical interface of the TV camera is in the test circuit. The signals coming out of the camera are already divided into the component colors and the registration test is meaningless beyond that point. Following the registration tests, VISTST implements the incipient fault detection analyses using the fault isolation test procedures for either presently unacceptable end to end performance levels or for degraded performance levels which indicate incipient faults. Finally, VISTST calls GENTST which executes both static calibration and dynamic test for the servo systems in the scene generation equipment.

VIDTST, the basic video equipment test software, shown in Figure 8.4-19, executes the basic TV tests described in Section 8.4.1. VIDTST establishes values for the following parameters:

- o Set up voltage
- o Gamma
- o Differential Gain
- o Frequency Response (Amplitude)
- o Peak/Peak voltages
- o Signal/Noise ratio

In addition, VIDTST checks streaking and ringing faults against tolerable levels and also checks to determine if linearity deviations are within acceptable limits. This complete set of tests can be applied to a complete string of TV components including the camera tube optical interface if a test pattern is used for a test signal input. This set of tests may also be applied at the LRU or least replaceable component level for fault isolation purposes.

GENTST, the test software routine for testing static alignment or calibration as well as dynamic response of the servo systems in the model generation equipment is shown in Figure 8.4-20. GENTST uses previously defined routines to implement the transient response tests for the servo systems and then uses the marginal

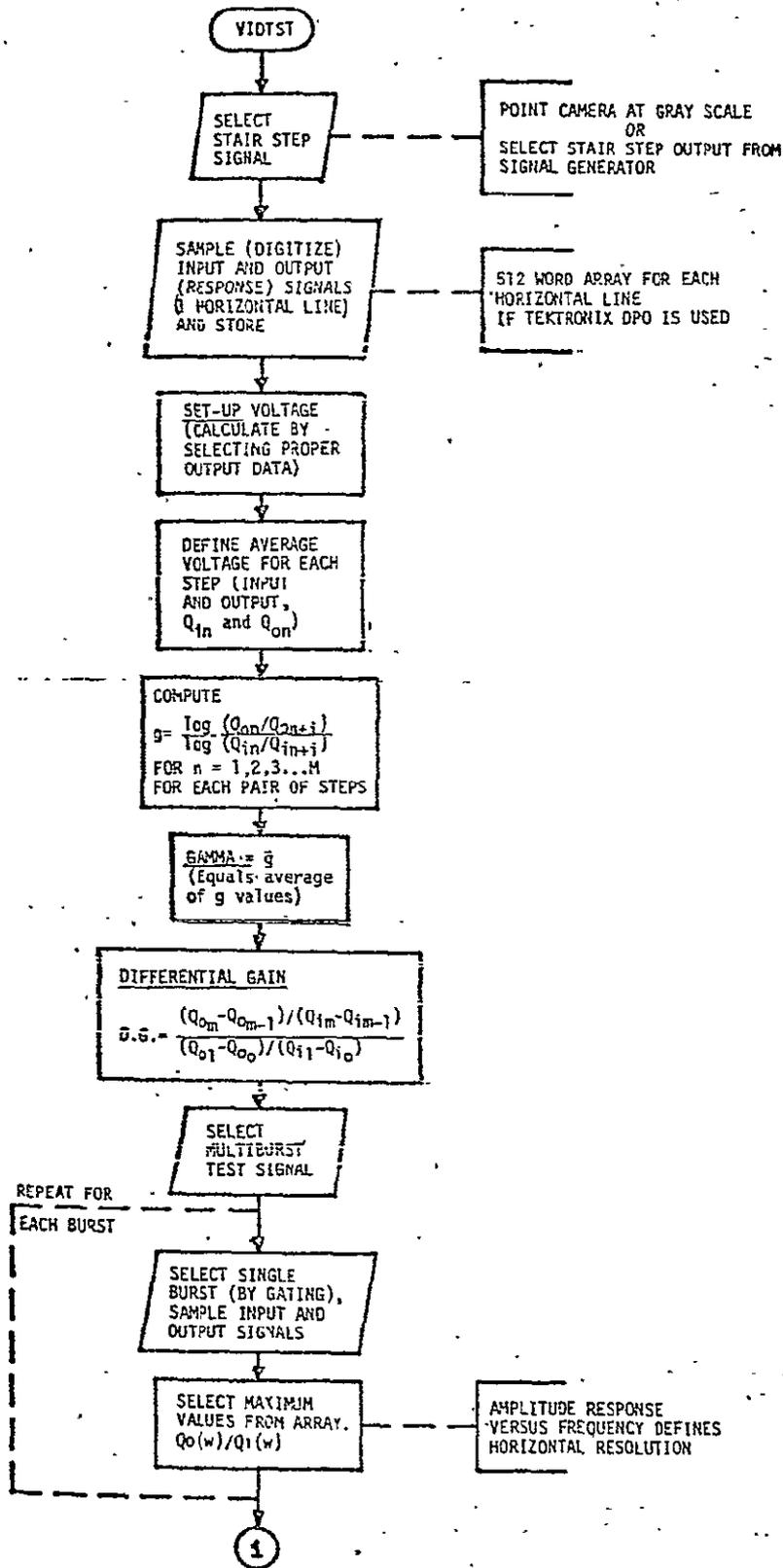


FIGURE 8.4-19 VIDEO EQUIPMENT TEST SOFTWARE, VIDTST

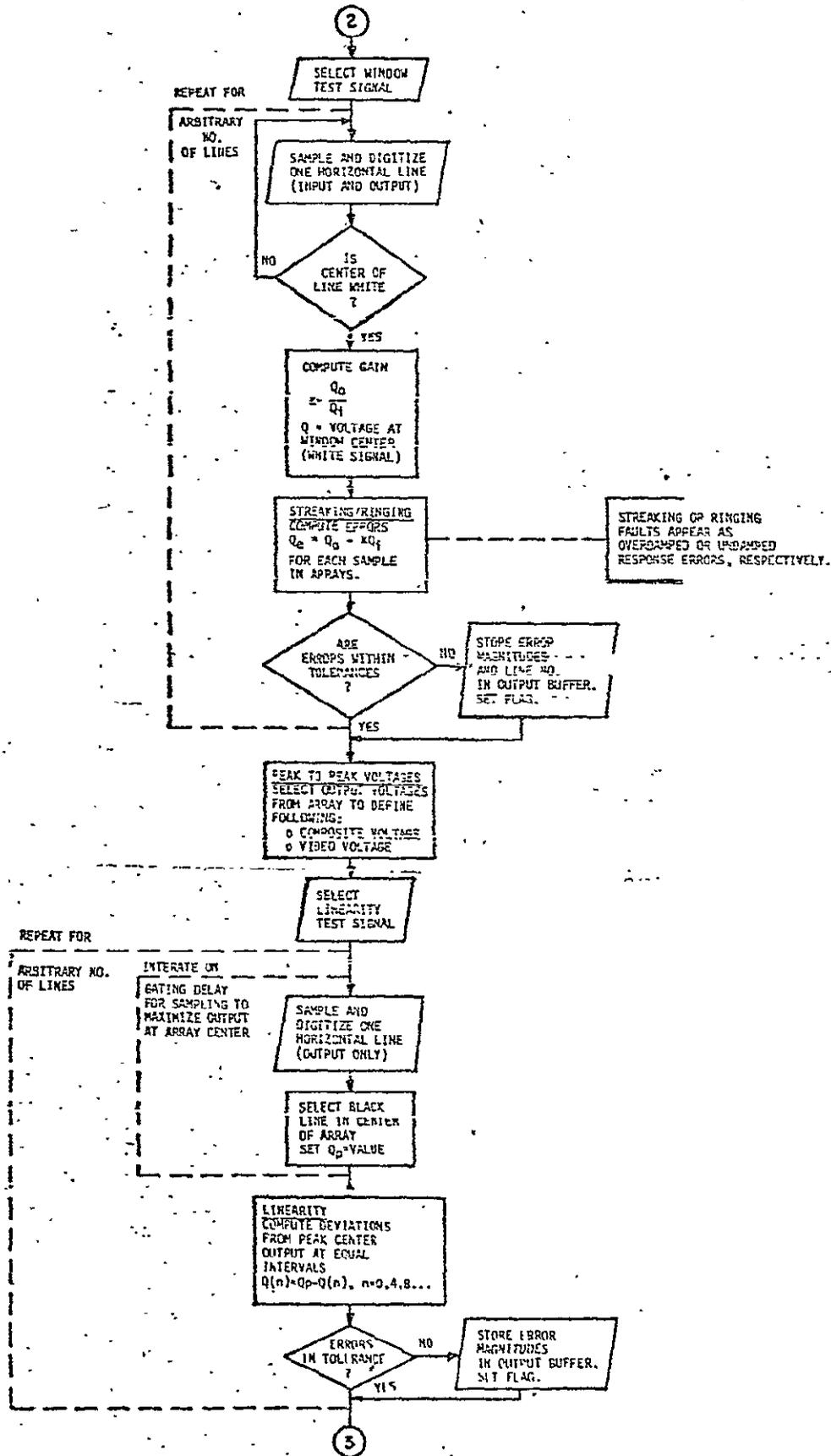


FIGURE 8.4-19 VIDEO EQUIPMENT TEST SOFTWARE, VIDTST (continued)

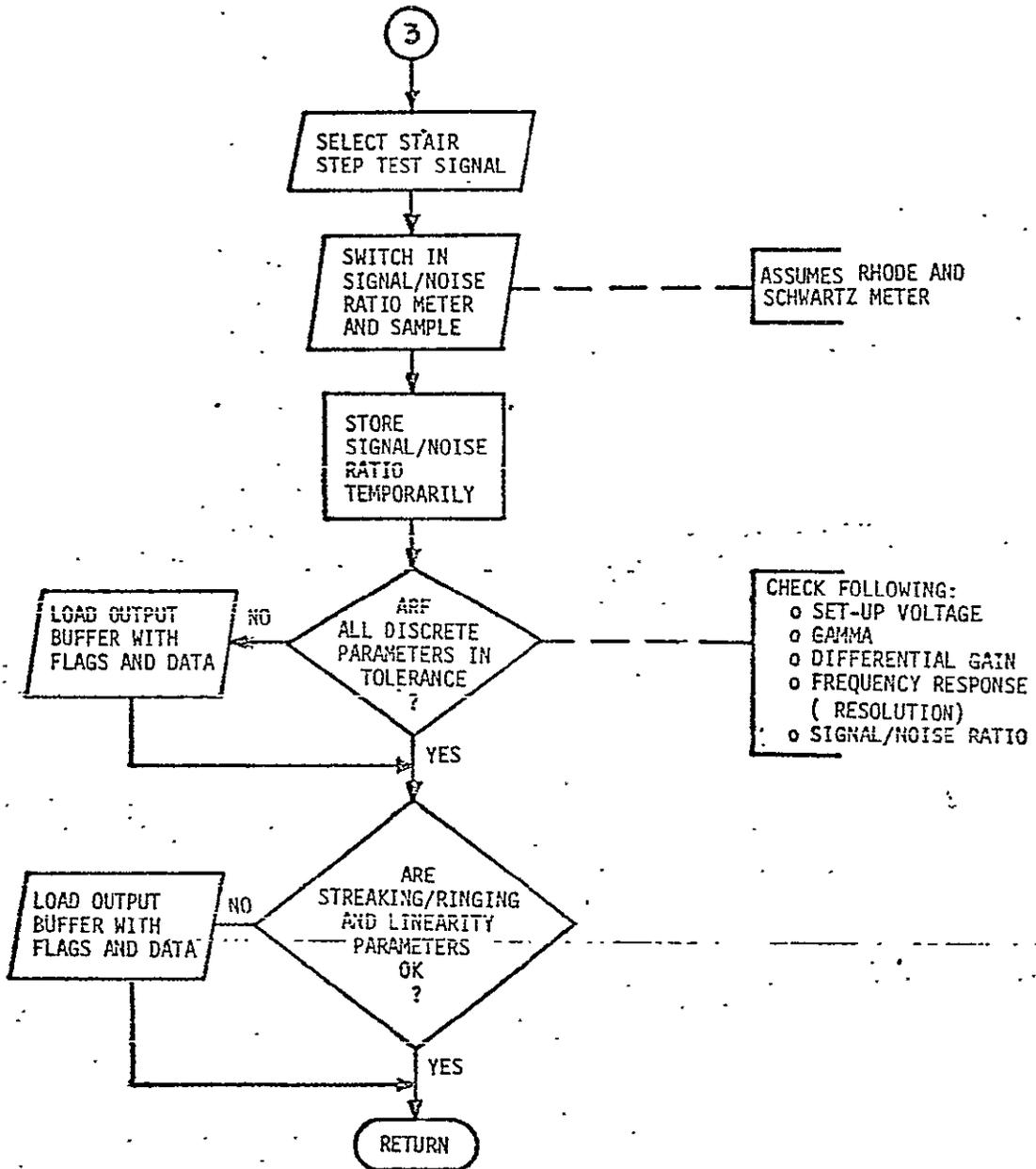
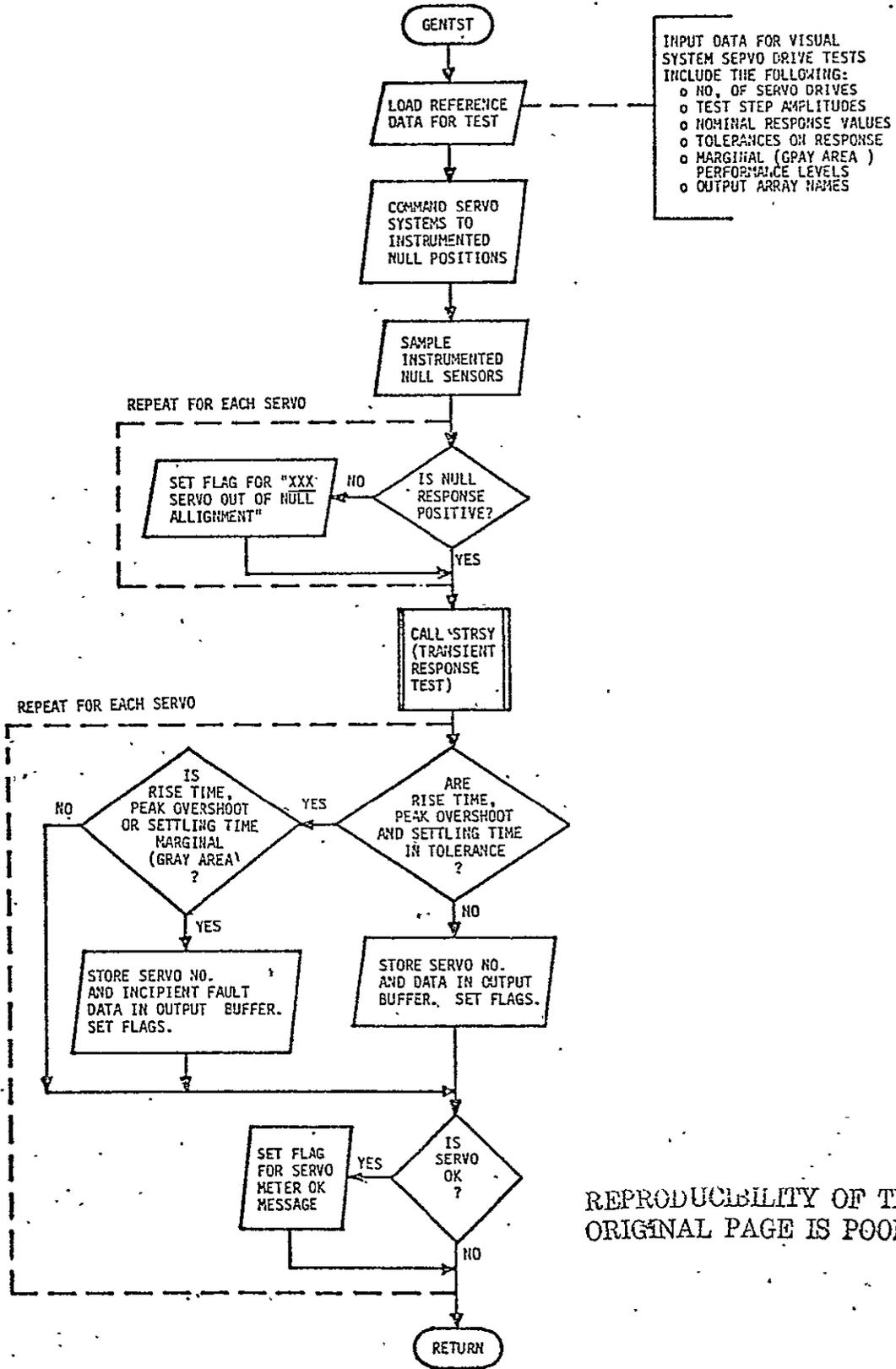


FIGURE 8.4-19 VIDEO EQUIPMENT TEST SOFTWARE, VIDTST (continued)



REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

FIGURE 8.4-20 SCENE GENERATION EQUIPMENT SELF TEST SOFTWARE FLOW, GENTST

performance, or gray area, approach to check for incipient faults. Sophisticated fault isolation techniques have not been shown because the choice of any particular technique requires development work of a higher magnitude than the tests generally outlined in the present software flows. The frequency response or fault dictionary type techniques are applicable to the servo systems of this type and may be applied if necessary.

The data base requirements of the tests described above are very nominal for several reasons. First, the signal generator used for the video subsystem tests is a hardware signal generator. Consequently, signal generation requires only switching commands. Test voltage levels for each test can be adjusted by built-in voltage dividers in each input line such that the signal generator always supplies an output voltage at a constant signal level no matter where the signal is inserted. In addition, the hardware components in each string of TV hardware are characterized by similar performance levels. Consequently, the tolerances for a given parameter are identical for each component color path, i.e., red or green or blue.

## 8.5 MOTION SYSTEM

This section defines hardware and software techniques for checking the operational status of a simulator motion base. It describes the system configuration and defines the parameters characterizing its output. It notes anticipated failure modes and associated symptoms. It discusses the processing requirements and transmission of sensed parameters to a computer for verifying proper operational performance as well as for isolating faults to a defective unit.

### 8.5.1 Description

Investigation of candidate motion bases for the Shuttle Mission Simulator (Reference 5.2-1) has indicated that a moving platform powered by hydraulically driven actuators has advantages over other possible kinematic schemes. These advantages in mass movement capability, range of displacements, velocities and accelerations have dictated that such a system will be used. State-of-the-art hardware also suggests using a six degree of freedom synergistic system.

The motion system with six degrees of freedom (DOF) is made up of two basic subsystems with peripheral hardware components. The first subsystem is a payload supporting a moving platform driven by a synergistic system of six hydraulically powered actuators. The second subsystem is the hydraulic system controlling the motion base movement through an electrohydraulic servo control loop. Interfacing hardware are a Tilt Frame mounted on top of the motion base platform, a redundant set of emergency cushioning legs, a hydraulically driven access walkway, a group of Control Panels and a number of Hardware Interlocks.

#### 8.5.1.1 Moving Platform

Figure 8.5-1 illustrates the motion base geometry. The moving base and the six hydraulic actuators form a space frame that is stable and determinate for any combination of lengths of the actuators. A set of six redundant legs may be installed as an emergency cushioning feature in the unlikely event of breakage of a servoed leg actuator. These additional legs are self contained hydraulic springs exerting a lifting force on the platform as a result of the pressure in independent accumulators mounted on each leg. A movable walkway is configured

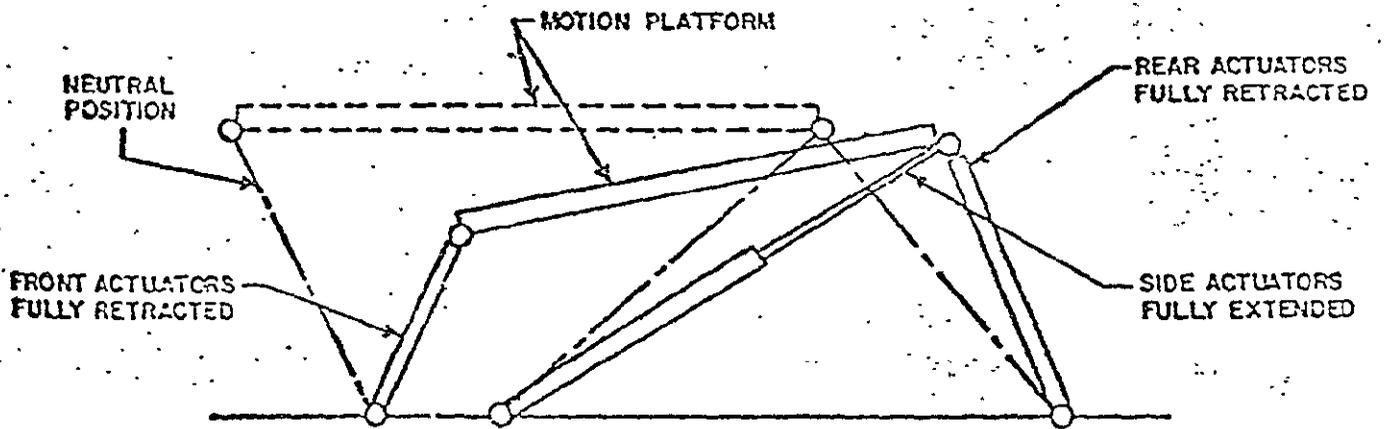
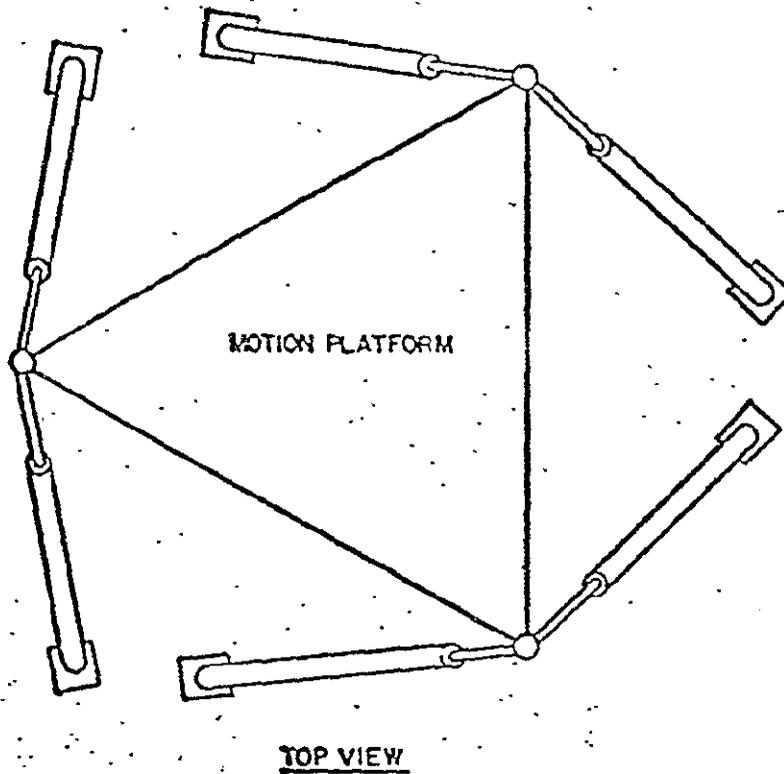


FIGURE 8.5-1 MOTION BASE GEOMETRY REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

for ingress and egress to the motion base platform. This access unit is positioned by independent hydraulic actuators.

A Tilt Frame will be mounted on the basic motion base platform to provide an increased pitch envelope capability and a higher pitch velocity (see 8.5.1.2). Lateral tilt frame motion is constrained by mechanical linkage.

#### 8.5.1.2 Hydraulic System

The hydraulic system is schematically illustrated in Figure 8.5-2, which also depicts the required system parameters (sensors) by bubble notation. The power for the motion base consists of electrically driven hydraulic pumps supplying high pressure oil to a system manifold where the flow capability to the hydraulic actuators through the electro-hydraulic servo valves is augmented by an accumulator system. System filtering and pressure relief hardware is provided. A hydraulic reservoir will supply fluid to the pumps at a satisfactory net positive suction head(NPSH). The oil temperature during periods of high activity of the motion base is controlled by a return line heat exchanger. Table 8.5-1 defines the system parameters and their characteristics.

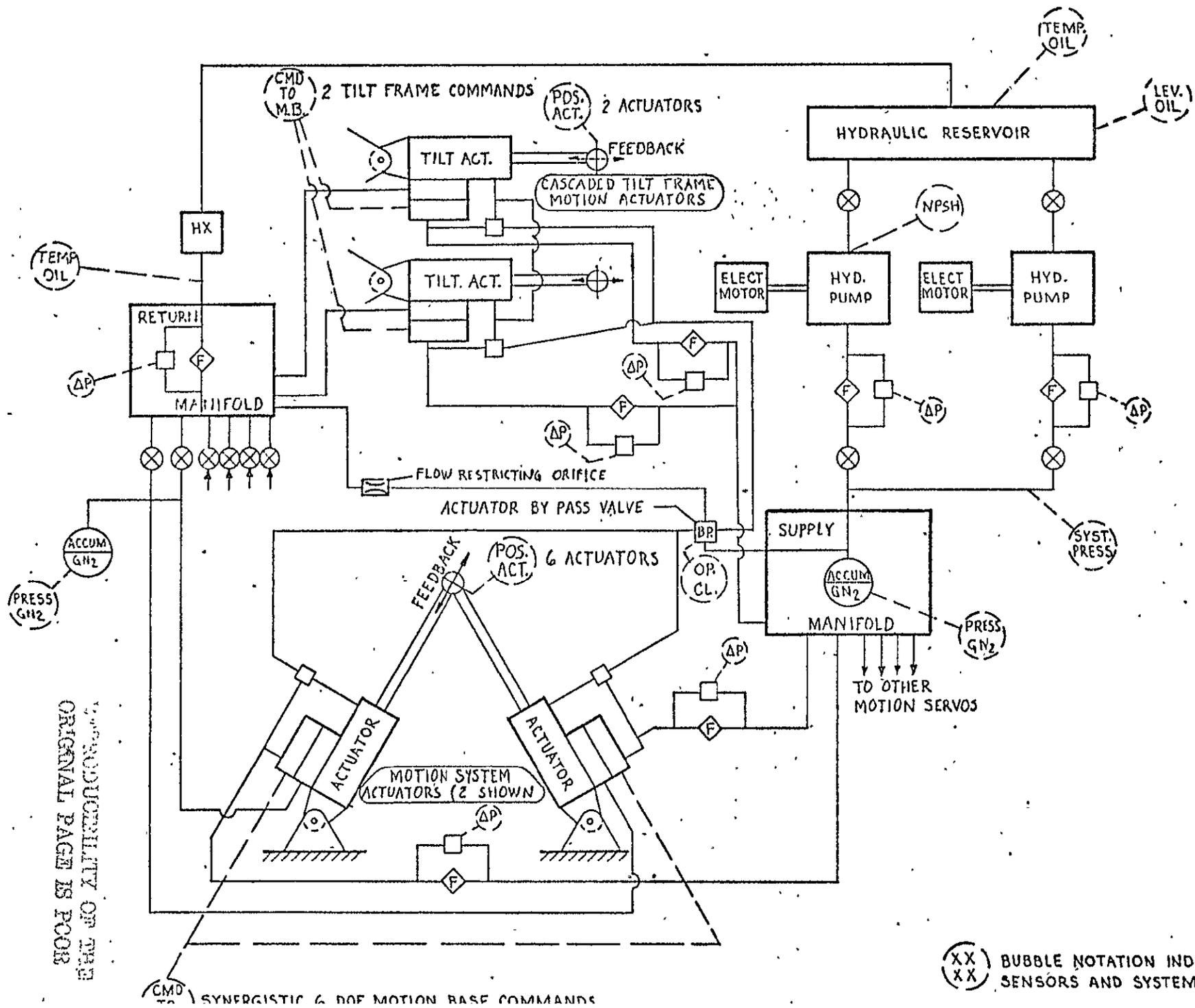
A Tilt Frame (see 8.5.1.1) will be cascaded on the Motion Platform to provide +75° positive pitch angle to the crew station and visual system independent of the pitch capability of the motion system. Two servoed hydraulic actuators, acting in parallel, are powered off the motion base hydraulic supply distribution manifold to provide a short duration maximum pitch velocity of 60 degrees per second when augmented with the motion system pitch velocity.

#### 8.5.1.3 Motion Base Hardware Interlocks

The system is protected by the following hardware interlocks:

- Proper sequencing for power up and power down with abnormal conditions inhibiting the application of power to the motion platform.
- Travel limits within the travel envelope are sensed and a smooth deceleration of less than 2.5g is assured when these limits are approached at maximum velocity.

REPRODUCIBILITY OF THIS ORIGINAL PAGE IS POOR



(XX XX) BUBBLE NOTATION INDICATES SYSTEM SENSORS AND SYSTEM LOCATION

TABLE 8.5-1 PARAMETER CHARACTERISTICS

PARAMETER	ESTIMATED SENSOR RANGE	NOMINAL SYSTEM OPERATING RANGE AND OPERATIONAL CONTROLLING FACTORS	RESULTS OF MALFUNCTION
HYDRAULIC SYSTEM PRESSURE (SYST PRESS)	0-3000 psig	2000 +200 psig - Controlled by specific pump outputs, pressure regulating valves and relief valves. On hydraulic system activation (pump On) the system pressure should rise and remain steady within approximately a $\pm 25$ psig long range variation caused by temperature effect on the pump hardware and oil viscosity.	If pressure indicates low a programmed maximum demand on the hydraulic system may not be met and a M.B. command may not be completed or there may be a command response lag. If pressure indicates high the response to all commands may be accelerated, destroying the preplanned verifying repeatability of the program and subjecting the M.B. and hydraulic system to excessive stress.
NET POSITIVE SUCTION HEAD PRESSURE (NPSH)	0-50 psig	5-20 psig - Controlled by the hydraulic pressure head between the hydraulic reservoir and the pump inlet.	If inlet pressure indicates low the hydraulic pump may cavitate resulting in possible pump damage or reduction of the hydraulic system pressure with resultant system response lag. Also a low indication could result in the introduction of gas into the system resulting in a spongy response lagging system.
HYDRAULIC RESERVOIR OIL LEVEL (LEV OIL)	0-100%	- Controlled by specified fill level. If level indicates low the system has not been properly serviced or there is a significant leak in the system.	If reservoir level indicates low the pump inlet line could have insufficient fluid head and cavitate when there is a high M.B. flow demand situation resulting in pump damage and or introduction of gas into the system.
SUPPLY LINE ACCUMULATOR PRECHARGE (PRESS GN <sub>2</sub> )	0-3000 psig	1500 $\pm$ 150 psig - Prior to hydraulic system activation - Controlled by accumulator servicing. 2000 $\pm$ 20 psig - After hydraulic system activated - Controlled by accumulator servicing and hydraulic system pressure.	If precharge indicates low there could be an insufficient pressurized volume of oil to complete a high demand command.
RETURN LINE ACCUMULATOR PRECHARGE (PRESS GN <sub>2</sub> )	0-300 psig	200 $\pm$ 20 psig - Prior to hydraulic system activation - Controlled by accumulator servicing.	If precharge indicates low the motion system might function in a somewhat erratic (jerky) manner, due to the non damped hydraulic flow.
TEMPERATURE OIL (OUT/IN) (TEMP OIL)	(-100) $\pm$ (+300) °F	40-160 °F - Controlled by diurnal temperature and frictional (fluid and mechanical) energy transferred as heat into the fluid. Excessive temperature indicates excessive activity by simulator, a malfunctioning return line heat exchanger, an internal friction source or a combination of the three.	160 °F is the high side of the operational specification requirement for MIL-H-5606 hydraulic oil in an open loop system. Also its minimum flash point is 200 °F.
FILTER DIFFERENTIAL PRESSURE ( $\Delta P$ )	0-100 psid	0-30 psid - Controlled by contamination of the filter.	If differential pressure is high oil flow in that specific loop could be reduced possibly causing a command response lag.
ACTUATOR POSITION FEEDBACK (POS)	0-100%	End to End (0-xx) inches = Mechanically controlled by electro hydraulic servo calibration. Dynamically controlled by the system response.	If response to command is not within a predetermined band the Motion Base is not verified as ready for operation.

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

- o Two sets of limit switches on each actuator are provided to:
  - 1) Shut down the motion system in the case of sensed actuator position beyond the servo electronic limit or against a mechanical limit.
  - 2) Attenuate motion system velocity in the event of a runaway servo.
- o Emergency deactivation switches are provided at the control station, at the hydraulic system and at the payload that will immediately remove all electric power from the system, returning the platform to the "settled position" at a smooth controlled rate.
- o Key switches will ensure exclusive local control of electrical and hydraulic power at the payload and at the hydraulic supply during power down maintenance. Additionally, "motion-on" will be interlocked with the seat harness and payload doors for personnel protection.

#### 8.5.1.4 Hydraulic Control Panels

There will be a Hydraulic Control Panel, a Motion System Maintenance Control Panel, a Cylinder Position Control Panel, and a Test Point Panel. The Hydraulic Control Panel includes provisions for semi-automatic trouble shooting of the Hydraulic Power Control Systems. This panel includes the following system displays. Those with an asterisk indicate sensors measuring analog levels.

Hydraulic Power Master Control System Status (e.g., Sim. Auto/Sim. Manual)

Pumps Start Status

Motion Base Actuator Bypass Valve Status

\*Hydraulic Fluid Level (Reservoir) and Temperatures

\*Hydraulic System Accumulator Precharges and Fluid Levels

\*Hydraulic Filter Differential Pressures

\*Hydraulic System Pressure and Temperatures

\*Motion Base Commands

\*Actuator Position Feedback (Servo Position) and Associated Servo Errors

\*Access Platform Accumulator Precharge and Fluid Levels.

The Maintenance and Cylinder Position Control Panels are equipped with controls and indicators for manual operation of the system, thereby isolating the computer inputs. The intention of this feature is to allow each servo to be checked operationally without removing the servos from the system or to allow operation of the system entirely through manual or analog means of control

(i.e., off-line). The Test Point Panel provides easily accessible test points of critical signals associated with each servo actuator.

### 8.5.2 Failure Mode Analysis

Table 8.5-2 lists the failure modes and related symptoms. Given a set of symptoms identified by a software program, the associated malfunction mode may isolate the problem to a specific replaceable unit. Unlike the black and white, go/no-go nature of electronic parameters, fluid system (hydraulic) parameters have a normal operating band trace (signature) indicating a range within which operational requirements for that system are met. If the parameter value is outside the signature trace but within the broader operational requirements band, an incipient hardware failure is suggested, and the anomaly should be resolved prior to continuing the checkout or simulation. Table 8.5-3 denotes checkout test techniques which will evaluate specific malfunction modes. The checkout test techniques are discussed in paragraph 8.5.3.

### 8.5.3 Checkout Techniques Evaluated

Table 8.5-4 lists characteristics of Candidate Checkout Testing Techniques. Table 8.5-3 notes Checkout Test Techniques which will evaluate specific malfunction modes.

#### 8.5.3.1 Static Checkout (Power ON and OFF)

A certain number of measurements and observations can be performed on a motion base while the system is in a static mode. Proper selection of these observations can yield a list of items that point to the operational status of the system without the need for actual movement of the base.

#### Static Power OFF Checkout

Prior to power ON activation of the Motion Base and Hydraulic Systems the following parameters status should be verified.

- Verify position of the motion base (actuator position feedbacks).
- Verify proper fluid level of hydraulic reservoir.
- Verify proper supply line accumulator precharge.
- Verify proper return line accumulator precharge.
- Verify proper hydraulic system operating pressure.
- Verify no hydraulic leaks.

MALFUNCTION MODE	SYMPTOMS	
	EXTERNAL	INTERNAL
ELECT. OPEN CIRCUIT	UNABLE TO ACTIVATE PART OR ALL OF THE COMPONENTS OF THE M.B.	NO CONTINUITY.
ELECT. SHORT CIRCUIT	AS ABOVE	LOW RESISTANCE.
COMPUTER AND OR DATA CONVERSION EQUIPMENT (DCE)	ERRATIC SYSTEM PERFORMANCE	INVALID COMMAND LOOPS AND/OR RESPONSE.
HYDRAULIC OIL RESERVOIR LEVEL LOW	LOW HYDRAULIC PRESSURE, PUMP NOISY	LOW RESERVOIR OIL LEVEL CAUSING HYDRAULIC PUMP TO CAVITATE.
HYDRAULIC PUMP PRESSURE OUTPUT LOW	LOW HYDRAULIC PRESSURE	PUMP INTERNAL LEAKAGE, PUMP DAMAGE, OR LOW RPM
ACTUATOR MOTION ERRATIC	MOTION BASE MOVEMENT JERKY OR LOCKS UP AND WILL NOT COMPLETE COMMANDED MOVEMENT.	ACTUATOR PISTON ROD BENT OR BEARING SEIZED.
ELECTRO HYDRAULIC SERVO VALVE	RESPONSE (OUTPUT) LAGS COMMAND (INPUT) IN AMPLITUDE IN VELOCITY IN FREQUENCY	MOVEMENT RESTRICTION (M.B. TABLE OR ACTUATORS). FLOW RESTRICTION OR INSUFFICIENT SUPPLY LINE ACCUMULATOR PRE-PRESS. MALFUNCTIONING SERVO VALVE
GENERAL HYDRAULIC SYSTEM	MUSHY SYSTEM	RESPONSE LAG
FILTER BLOCKED	HIGH FILTER DIFFERENTIAL PRESSURE	CONTAMINATED FILTER (DIRTY HYDRAULIC OIL)
FILTER OPEN	LOW FILTER DIFFERENTIAL PRESSURE	FILTER UNIT FLOW THROUGH (NOT FILTERING)
ACTUATOR POSITION FEEDBACK	DELAY IN RESPONSE TO COMMAND FOR ACTUATOR POSITION.	EXCESSIVE ERROR (LAG) ACTUATOR POSITION TO COMMAND.
SUPPLY LINE ACCUM LOW PRE-CHARGE	FAILURE OF ACTUATORS TO REACH HIGH DEMAND COMMANDED AMPLITUDE.	INSUFFICIENT SUPPLY LINE FLUID
RETURN LINE ACCUM LOW PRE-CHARGE	ERRATIC (NOT SMOOTH) M.B. MOVEMENT	RETURN LINE OIL SURGES NOT DAMPED

TABLE 8.5-3  
CHECKOUT TEST TECHNIQUE WHICH WILL EVALUATE SPECIFIC MALFUNCTION MODE

MODE MALFUNCTION	CHECKOUT TESTS					
	STATIC POWER OFF CHECKOUT	STATIC POWER ON CHECKOUT	MAXIMUM MISSION PROFILE (STEP INPUT)	TRAINING MISSION PROFILE +10% (STEP INPUT)	MOTION BASE LIMIT PROFILE (STEP INPUT)	FREQUENCY RESPONSE
ELECTRICAL OPEN CIRCUIT		✓				✓
ELECTRICAL SHORT CIRCUIT		✓				✓
COMPUTER AND/OR DATA CONVERSION EQUIPMENT (DCE)			✓	✓	✓	✓
HYDRAULIC OIL RESERVOIR LEVEL LOW	✓	✓				✓
HYDRAULIC PUMP PRESSURE OUTPUT LOW		✓				✓
ACTUATOR MOTION ERRATIC			✓	✓	✓	✓
ELECTRO HYDRAULIC SERVO VALVE			✓	✓	✓	✓
GAS IN HYDRAULIC SYSTEM			✓	✓	✓	✓
FILTER BLOCKED		✓				✓
FILTER OPEN		✓				✓
ACTUATOR POSITION FEEDBACK			✓	✓	✓	✓
SUPPLY LINE ACCUMULATOR LOW PRE-CHARGE	✓					✓
RETURN LINE ACCUMULATOR LOW PRE-CHARGE	✓					✓

TABLE 8.5-4 CHARACTERISTICS OF CANDIDATE CHECKOUT TESTING TECHNIQUES

CANDIDATE TECHNIQUE	RELATIVE COST	ADVANTAGES	DISADVANTAGES	HARDWARE REQUIREMENTS	SOFTWARE REQUIREMENTS
STATIC CHECKOUT	MINIMAL	1. PRE HYDRAULIC ACTIVATION CHECKOUT.	NO M.B. DYNAMICS,	ALL HYDRAULIC, ELECTRICAL AND MECHANICAL SYSTEMS PARAMETERS (SENSORS) NOTED ON FIGURE 2 HYDRAULIC SCHEMATIC.	ALL HARDWARE SENSORS PIPED INTO COMPUTER.
	POWER ON				
	POWER OFF	MINIMAL	1. POST HYDRAULIC ACTIVATION CHECKOUT.		
MAXIMUM MISSION PROFILE (STEP INPUT)	MODERATE	<ol style="list-style-type: none"> <li>SINGLE DAILY CHECKOUT.</li> <li>MODERATE STRESS ON M.B. AND PAYLOAD* PRIOR TO DAILY OPERATIONS.</li> <li>MODERATE CAPABILITY FOR DETECTION OF INCIPIENT FAILURES OF HARDWARE AND SYSTEMS.</li> <li>GOOD PROBABILITY OF HARDWARE OR SYSTEMS FAILURE PRIOR TO RATHER THAN DURING DAILY OPERATIONS.</li> </ol>	<ol style="list-style-type: none"> <li>MODERATE STRESS ON M.B. AND PAYLOAD*.</li> <li>LARGER CUMULATIVE STRESS (EXCESSIVE?) CAUSED BY CHECKOUT THAN BY TRAINING OPERATIONS IF RUN DAILY.</li> </ol>	AS ABOVE	AS ABOVE
TRAINING MISSION PROFILE +10% (STEP INPUT)	MODERATE	<ol style="list-style-type: none"> <li>MINIMUM STRESS ON M.B. AND PAYLOAD* PRIOR TO DAILY OPERATIONS.</li> <li>FAIR PROBABILITY OF HARDWARE OR SYSTEMS FAILURE PRIOR TO DAILY OPERATIONS.</li> </ol>	<ol style="list-style-type: none"> <li>ONLY NOMINAL CAPABILITY OF DETECTING INCIPIENT FAILURES OF HARDWARE AND SYSTEMS.</li> <li>SEVERAL DIFFERENT DAILY CHECKOUTS.</li> </ol>	AS ABOVE	AS ABOVE
MOTION BASE LIMIT PROFILE (STEP INPUT)	MODERATE	<ol style="list-style-type: none"> <li>SINGLE DAILY CHECKOUT.</li> <li>BEST CAPABILITY FOR DETECTION OF INCIPIENT FAILURES OF HARDWARE AND SYSTEMS.</li> <li>BEST PROBABILITY OF HARDWARE OR SYSTEMS FAILURE PRIOR TO RATHER THAN DURING DAILY OPERATIONS.</li> </ol>	<ol style="list-style-type: none"> <li>EXCESSIVE STRESS ON M.B. AND PAYLOAD*.</li> <li>PROBABLE CAUSE OF FAILURES DUE TO EXCESSIVE STRESSES IMPOSED DAILY.</li> </ol>	AS ABOVE	AS ABOVE
FREQUENCY RESPONSE	MODERATE	<ol style="list-style-type: none"> <li>BEST MISSION CHECKOUT SIMULATION.</li> <li>NOMINAL STRESS ON M.B. SYSTEMS PRIOR TO DAILY OPERATIONS.</li> <li>GOOD CAPABILITY FOR DETECTION OF INCIPIENT FAILURE OF HARDWARE AND SYSTEMS.</li> </ol>	<ol style="list-style-type: none"> <li>EXCESSIVE STRESS IMPOSED ON M.B. AND PAYLOAD* FOR MOST DAILY OPERATIONS.</li> <li>POST DAILY CHECKOUT EVALUATION TIME REQUIRED MAXIMUM.</li> </ol>	AS ABOVE	AS ABOVE

\*Payload consists of Cockpit, Personnel, Tilt Frames, and Visual Display.

### Static Power ON Checkout

In each of the following techniques the motion base checkout should be preceded by verifying the following parameters with the hydraulic system activated (hydraulic oil flowing) but with the motion base actuators bypassed or locked in place (motion base settled).

- Verify proper position of motion base (actuator position feedbacks)
- Verify proper fluid level of hydraulic reservoir
- Verify proper net positive suction head at pump inlet.
- Verify proper hydraulic system operating pressure.
- Verify proper differential pressure for all filters.

### 8.5.3.2 Step Input Checkout

Step input stimuli can be used to verify the operational and dynamic response capabilities of the motion base. This technique permits activation of the system in various and individual planes of motion, thereby permitting isolation of probable faults to specific components.

### Maximum Mission Profile (Step Input)

This is a technique that exercises the motion base by Step Input commands actuating all six synergistic actuators. The steps will be equivalent to those resulting in the maximum dynamic stress encountered in any of the training missions. The program must verify the actuator position feedback as well as discrete values for all hydraulic, mechanical and electrical parameters.

Comments: This technique would evaluate daily the operational status of the motion base to respond to the maximum programmed step input simulation activities without subjecting the motion base to excessive (out of maximum operational mode) stress. However, it would in most instances impose more stress than the daily training mission requires. In so doing it may force marginal components to fail during the actual training mission. It is only moderately likely that this technique would point to areas of incipient failure of hardware or systems as a more demanding frequency response analysis might do.

### Training Mission Profile Maximum Stress +10%

This is a technique that exercises the motion base by step input commands equivalent to those being imposed by the training mission being implemented that day plus a pre-established figure (e.g., 10%). The program must verify the actuator position feedback as well as discrete values for all hydraulic, mechanical and electrical parameters.

Comments: This technique would evaluate daily the operational status of the motion base to respond to that particular day's training mission's commands. It would exercise (stress) the motion base only 10% above the programmed daily requirements. If a hardware component failure was imminent it should fail during the checkout prior to and not during the training mission. Of the techniques submitted, this one imposes the least cumulative stress on the motion base system but also retains only a small capability to detect incipient hardware failures and degrading systems; also it entails the use of several different checkout profiles, of which the profile matching the most severe pertinent training activity would be utilized.

### Motion Base Limit Profile (Step Input)

This technique is basically the same as the Maximum Mission Profile except that the step input commands include the maximum design capability of the motion base under maximum dynamic load, maximum motion base velocity and maximum hydraulic fluid flow. The program must verify that discrete values for all hydraulic, mechanical and electrical parameters are attained.

Comments: This technique would evaluate daily the operational status of the motion base to respond to commands imposing the highest operational stress levels within the system's design capability of the motion base and hydraulics, (e.g., rotational acceleration of approximately  $50^\circ/\text{sec}^2$  and directional acceleration of approximately (0.6 to 0.8g). This level of checkout would undoubtedly be excellent in detecting areas of incipient hardware and systems failure, but the daily implementation of associated high stress might very well cause unnecessary failures.

### 8.5.3.3 Frequency Response

This technique has been implemented on motion bases in operation at NASA-AMES (Reference 2) and at Langley Research Center (Reference 5). The basic program, called SAFE (six axis frequency evaluation), operated an electro-mechanical motion base at AMES. Modifications were made at Langley to enable the program to operate a hydraulic motion base and perform the required inverse transformation.

A digital computer program SAFE drives each axis of the simulator near its acceleration, velocity and position limits, and records the simulator motions for comparison with nominal responses. It should also measure the frequency response and noise levels of each axis to provide a sensitive check of the entire motion system. If the frequency response and power capabilities of simulator drives are poor, or deteriorating, the motion simulation will be correspondingly poor.

To determine the current characteristics of the motion base the following measurements are made on a specified time-interval basis:

- Each axis of the simulator is driven near its acceleration, velocity, and position limits by acceleration commands, and the responses are compared with nominal values.
- Up to six axes of the simulator are simultaneously driven by a sum of sinusoids, and the position follow-up signals are used to calculate the frequency response and noise levels of each axis.

The results of these two measurements provide a sensitive check of the entire motion system and readily verify whether or not:

- there is continuity of signal from the digital computer to the motion drives.
- the position follow-up pots are working.
- there is continuity of signal from the position follow-up pots to the digital computer.
- the digital motion drive circuits, including gains, washouts, compensation networks and cross-feeds, are working properly.
- the simulator drives are responding normally.

- o there is undesired cross-talk between axes.
- o the system noise level is acceptably low.
- o there are unexpected non-linearities or resonances in motion response.

Reference 2 provides a complete description of the analysis technique, which involves:

- o Step or ramp acceleration commands scaled and timed to cause the simulator to perform near its acceleration, velocity, and position limits for each axis.
- o Recording of motion base feedback signals on stripcharts for comparison with standard or expected results.
- o The motion base frequency response is analyzed using a Fourier transform (in this case a fast Fourier transform to conserve computer resources) to determine the presence of noise, cross-coupling, or degradation in motion base response.

#### 8.5.3.4 Software Utilization

For convenience, a software package will be utilized to operate the motion base for the daily readiness tests. This software is intended to provide an orderly and logical test sequence that proceeds through the necessary verification steps to isolate discovered faults, with a minimum of operator intervention. Because an individual test may be desired at non-specific times, the test software will be modularized for maximum operational flexibility.

The different modules are illustrated in Figures 8.5-3, 8.5-4, and 8.5-5. The modules are sequenced by a master (executive) program that orders the modules into a logical verification procedure. See Figure 8.5-6.

The master program also controls the recording of verification test data for future comparison or analysis. Data of this nature may be valuable in establishing trends of events that lead to component failure, making it possible that through analysis of such trends corrective action may be possible before an actual failure. Data gathered through operation of the frequency analysis technique module may provide this type of clue.

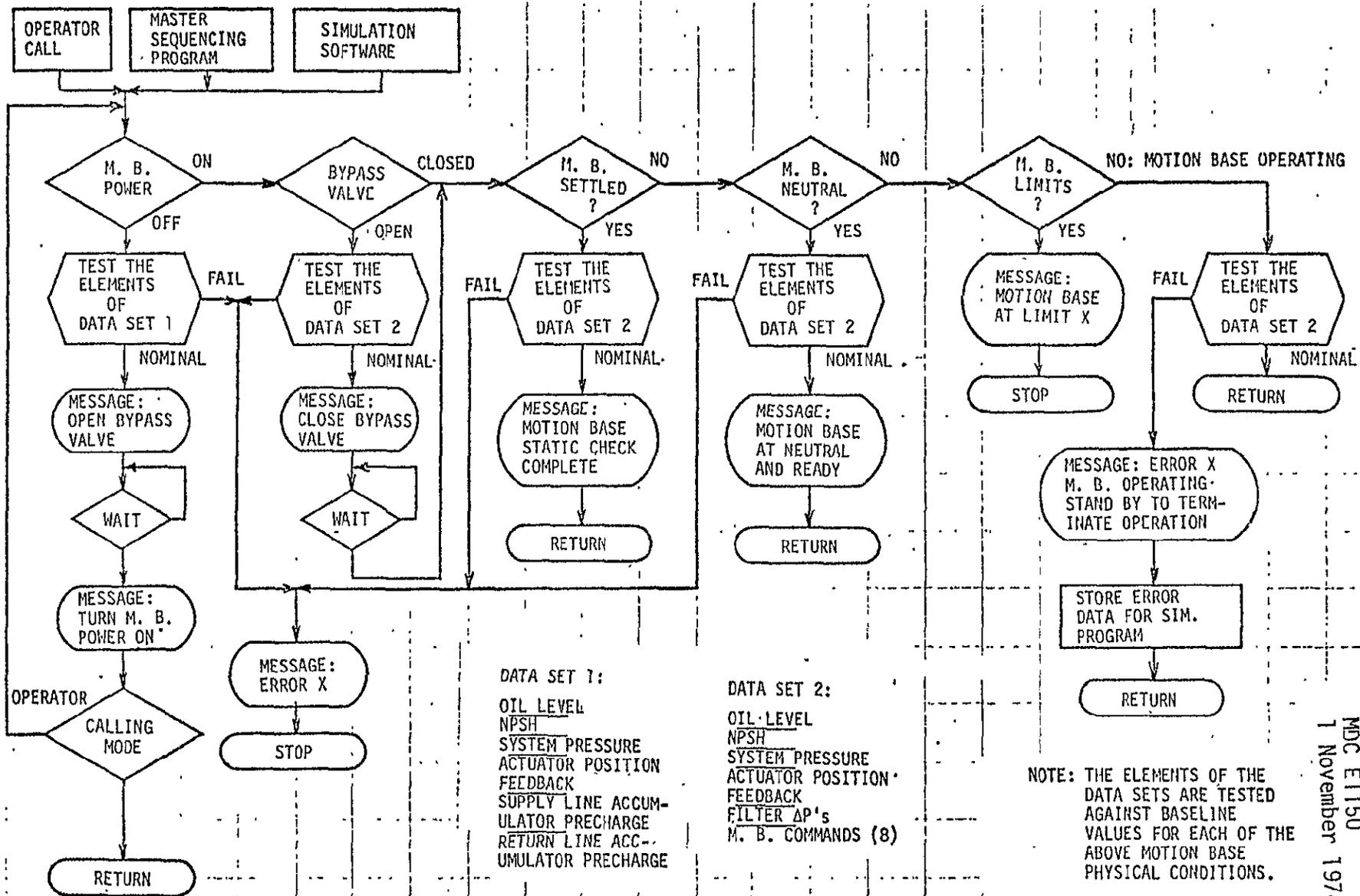
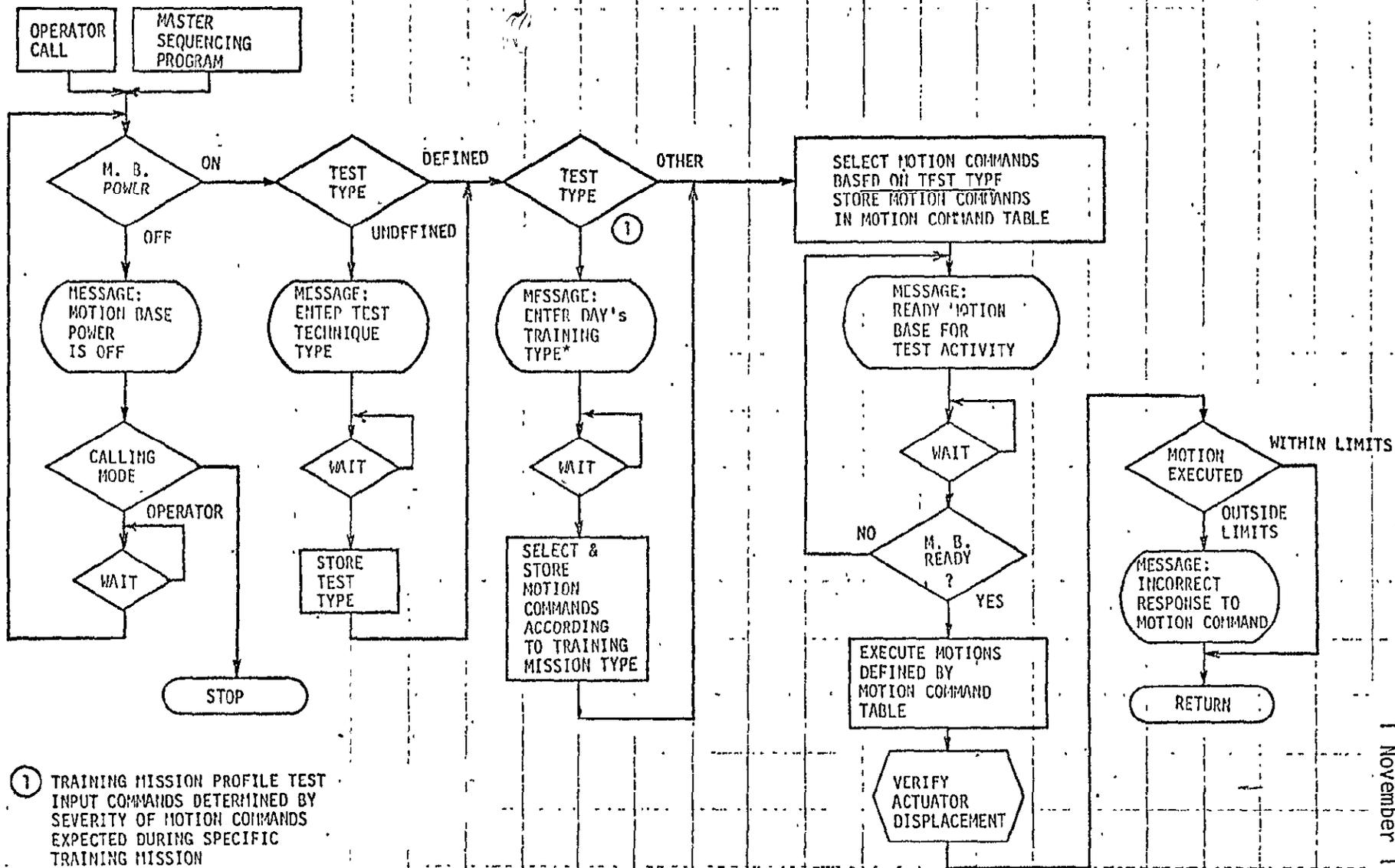


FIGURE 8.5-3 STATIC CHECKOUT MODULE



① TRAINING MISSION PROFILE TEST INPUT COMMANDS DETERMINED BY SEVERITY OF MOTION COMMANDS EXPECTED DURING SPECIFIC TRAINING MISSION

\* TRAINING TYPE REQUESTED IS THE CODE FOR THE TRAINING MISSION WHICH GENERATES THE MOST EXTREME MOTION COMMANDS

FIGURE 8.5-4 STEP INPUT CHECKOUT MODULE

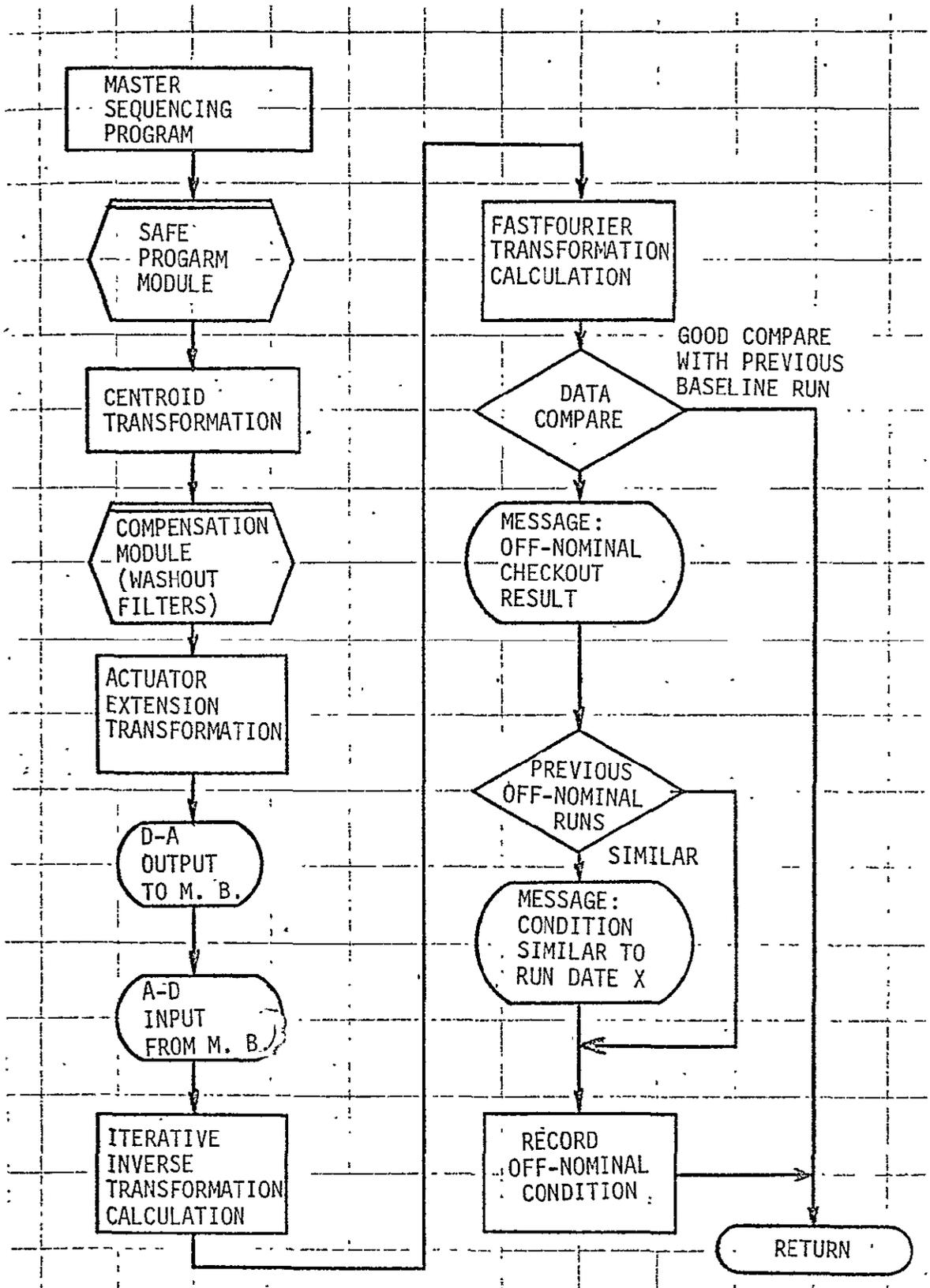
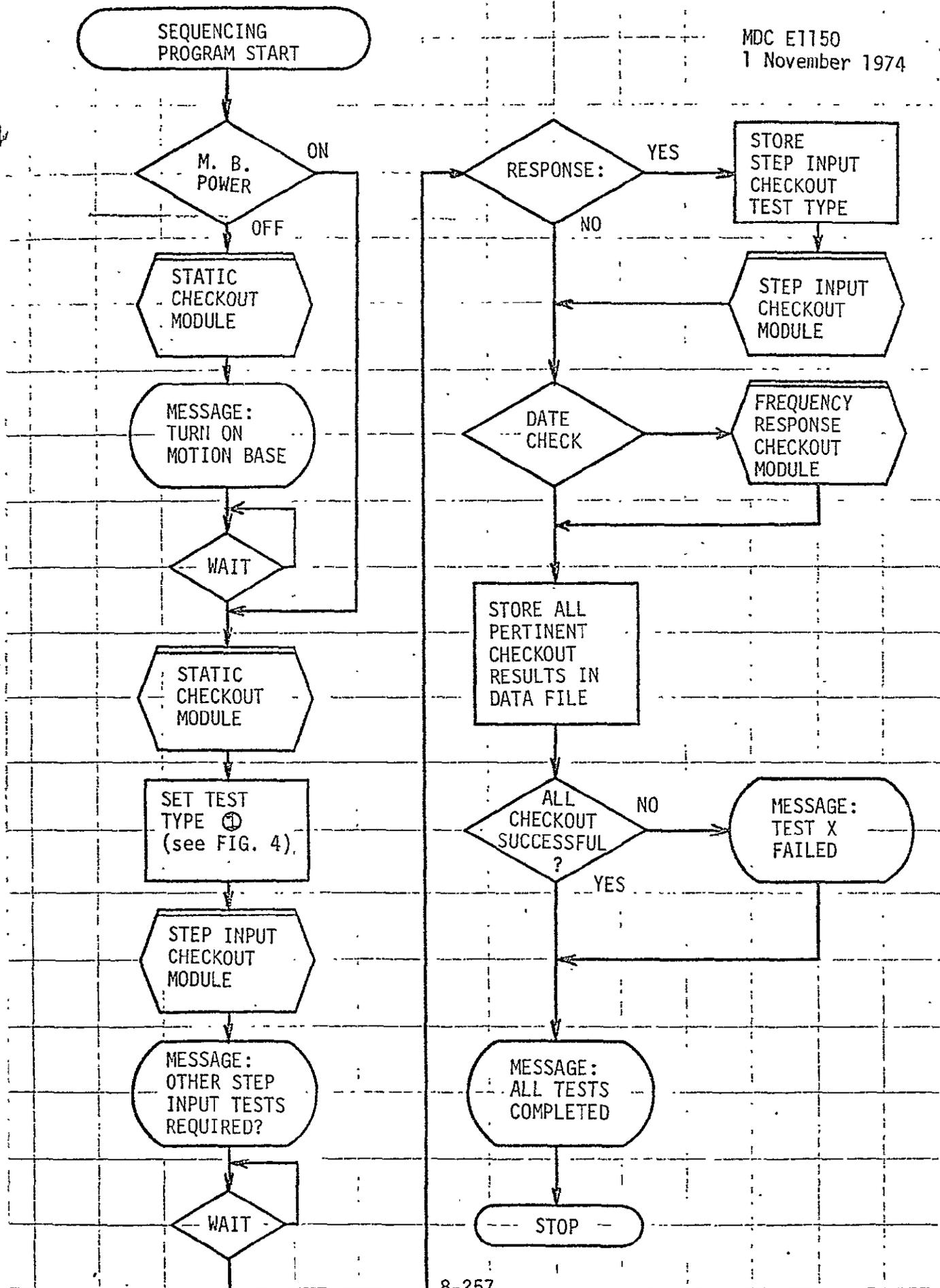


FIGURE 8.5-5 FREQUENCY RESPONSE CHECKOUT MODULE



8-257

FIGURE 8.5-6 MASTER SEQUENCING PROGRAM

#### 5.4 Recommended Checkout Approach

The major difficulties in the testing of a hydraulic motion base of the type expected in future simulators for the NASA is the potential for damaging the extensive visual displays and other equipment mounted on the motion base. The proposed techniques have examined the merits of various stress levels to which the motion base could be driven during transient response testing. Although, if a system exhibits truly linear performance, it is not necessary to test at any other than a nominal level to evaluate the transient response, there are other factors present. The frequency response of simulator motion bases of even the size considered here have the potential for extending beyond ten hertz. Consequently, it is a practice to filter all commands to the motion base to assure that it is never driven at those high frequencies with any substantial magnitude. This filtering, which may not be bypassed for testing, masks the dynamic response properties of the remaining hardware in the range where malfunctions are most likely to be revealed. Consequently, testing to stress levels comparable to daily training requirements is designed to at least assure system capability to achieve these levels.

The above difficulties have prompted us to suggest the following approach to motion base test implementation. First, it is recommended that maximum use be made of instrumentation which may be monitored on a day to day basis by an automatic self test system. Currently available sensors that would be applicable are summarized in Table 8.5-5. Sensors of this type can even be used to implement testing for incipient fault detection purposes as summarized in Table 8.5-6.

In addition, it is recommended that consideration be given to development of the on line monitoring techniques described in Section 7. These techniques have the potential of providing frequency response data from analyses of system performance recorded during normal operation. Use of these techniques to minimize the unnecessary stressing of motion base mounted simulator components seems like an ideal application for these concepts.

MCDONNELL DOUGLAS AERONAUTICS COMPANY - EAST  
8-259

SENSOR	HARDWARE	ACCURACY PERCENT OF FULL SCALE	OUTPUT	COST	REMARKS
3000 PSIG HYDRAULIC OIL PRESSURE	DYNISCO PT 320B-3M	<u>+0.5%</u>	0-5V	235.\$	DYNISCO DIVISION OF MICRODOT WESTWOOD, MASS. 617-326-8210
50 PSIG HYDRAULIC OIL PRESSURE	DYNISCO PT 320B-50	<u>+0.5%</u>	0-5V	255.\$	3000 PSIG OIL SENSOR
0-100% LEVEL HYDRAULIC OIL	R.F. BINDICATOR PRIMCO MODEL 770	<u>+0.5%</u>		945.\$	NOR-TEC INDUSTRIAL SALES HOUSTON, TEXAS 713-644-8026
	CAPACITANCE BINDICATOR CAP-LEVEL MODEL 660	<u>+1.0%</u>		463.\$	
3000 PSIG GAS PRESSURE	DYNISCO PT320B-3M	<u>+0.5%</u>	0-5V	235.\$	3000 PSIG OIL SENSOR
300 PSIG GAS PRESSURE	DYNISCO PT320B-3C	<u>+0.5%</u>	0-5V	255.\$	3000 PSIG OIL SENSOR
(-100) - (+300) <sup>0</sup> F TEMPERATURE HYDRAULIC OIL	RESISTANCE PLAT. ROSEMONT 152T10A	CALIBRATION ACCURACY @ 0 <sup>0</sup> C= <u>+0.04</u> <sup>0</sup> C		597.\$	ROSEMONT ENGINEERING CO. 213-866-8224
0-100 PSID HYDRAULIC OIL PRESSURE	DYNISCO DPT 361	<u>+1.0%</u>	<u>+3.MV/V MIN.</u>	395.\$	3000 PSIG OIL SENSOR
POSITION POTENTIOMETER	HOUSTON SCIENTIFIC POS. POT. 1800-150	<u>+0.1%</u>		475.\$	HOUSTON SCIENTIFIC HOUSTON, TEXAS ...713-681-6631
		<u>+0.05%</u>		505.\$	

TABLE 8.5-5 TYPICAL HYDRAULIC MOTION BASE INSTRUMENTATION

November 1974

TABLE 8.5-6 INCIPIENT FAULT DETECTION-HYDRAULIC MOTION BASE

FAULT	PARAMETER MEASURED	OPERATING RANGE (UNITS)	DATA STORED	IFD TECHNIQUE	IFD INDICATION LEVEL
Blocked Filter	Differential Pressure	3-30 (PSID)	1 day	Gray Area Performance	27 PSID with Max Flow Rate
Ruptured Filter	Differential Pressure	3-30 (PSID)	3 days	Rate of Degradation	Pressure Drop 1 PSID / Day with Max Flow Rate
Damaged Pump Seals	Hydraulic Pump Output Pressure	1800-2200 (PSIA)	20 days	Rate of Degradation	Pressure Drop 150 PSIA / Day
Sticky Actuator	Actuator Position Error	X (Inches)	60 days	Rate of Degradation	Average Error Increase 0.1 Inch / Day for Standard Manuever
Damaged Actuator Seals	Actuator Position Error	X (Inches)	10 days	Rate of Degradation	Average Error Increases 0.5 Inch / Day
Ruptured Sealing or Piping Component	Hydraulic Fluid Leak	None	?	Visual Inspection	Detectable Accumulation of Fluid External to Hydraulic System

8-260  
 MCDONNELL DOUGLAS ASTRONAUTICS COMPANY - EAST

## 8.6 MISCELLANEOUS EQUIPMENT

This section discusses self test implementation concepts for equipment which, although a necessary part of the simulator, are not particularly visible during actual operation. This equipment is usually procured as an integral, self contained, ready to operate subsystem. However, these subsystems do require extensive wiring installation in the simulator to provide the means for distribution of outputs.

The miscellaneous equipment covered here consist of the following:

- o Aural Cues System,
- o Power Supply System, and the
- o External Clocks and Timing Equipment.

### 8.6.1 Aural Cues System (ACS)

Operational experience with training simulators supports the importance level assigned to auditory cues by human engineering studies. These studies rate aural stimuli as second in importance only to visual stimuli. The complexity of modern aircraft and spacecraft has demanded aural cues components development to achieve a realistic simulation capability for all the various sounds normally heard in these vehicles. This section discusses one type of aural cues simulation equipment and a possible approach to the automatic self test of this equipment in a simulator.

#### 8.6.1.1 Aural Cues System Description

The system described here for reference purposes is the Poly-Voice Aircraft Sound Synthesizer proposed by Singer in the SMS Baseline Definition Report. This equipment has all the aural cue simulation capability needed to cover the various flight regions, flight phases, and configurations of the Space Shuttle. The Synthesizer, therefore, is a good representative design of the type of equipment that will eventually be selected for Shuttle simulators.

Figure 8.6-1 is a functional diagram of the Synthesizer. Sound is generated by the Voltage Controlled Oscillators (VCO's) or the Pseudo Random Noise Generators (PRNG's). Volume and duration (if applicable) of the resulting audio are controlled by the Voltage Controlled Amplifiers (VCA's). Controlling

8-262

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

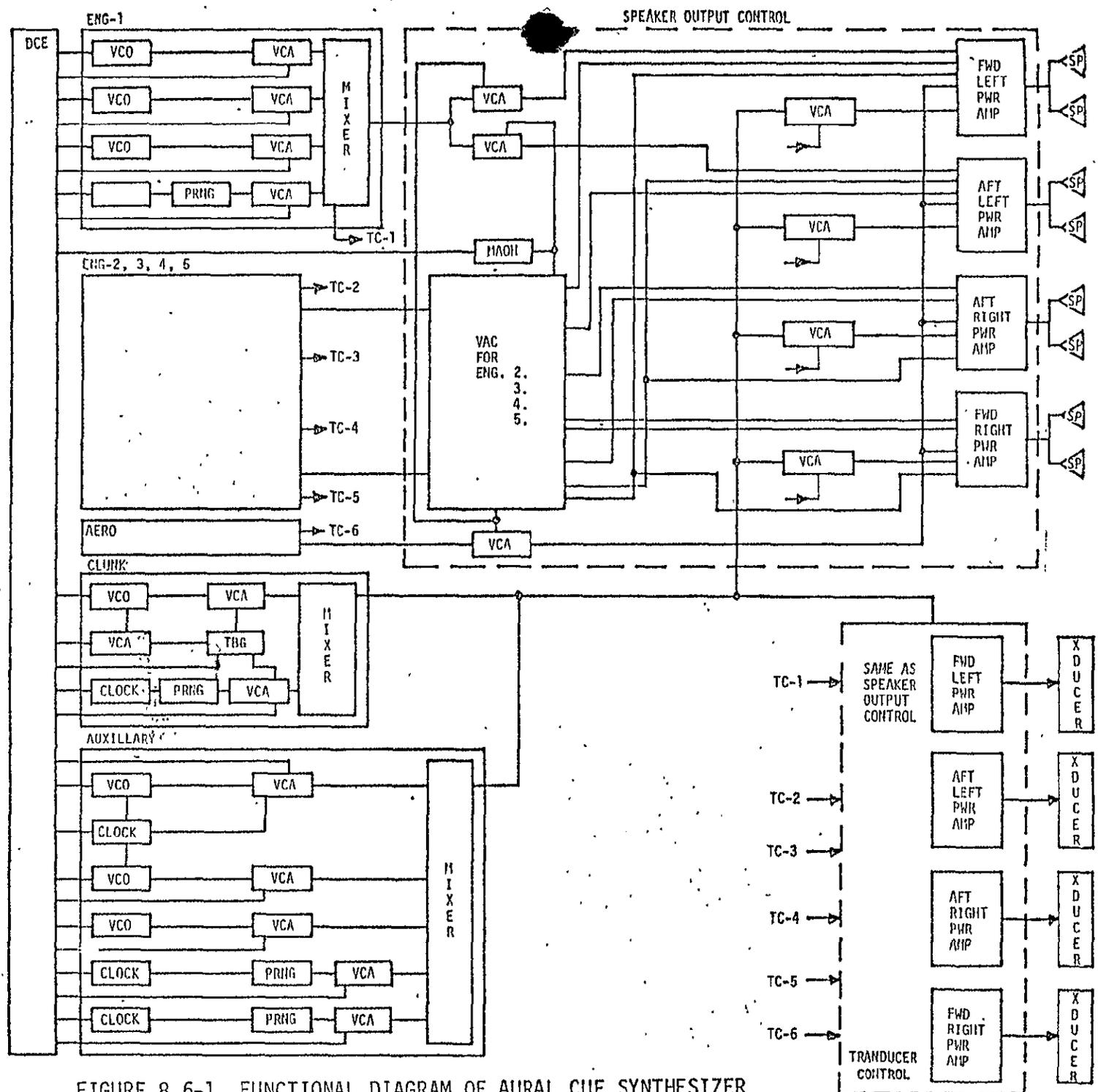


FIGURE 8 6-1 FUNCTIONAL DIAGRAM OF AURAL CUE SYNTHESIZER

commands for these modules are initiated under software control by the HOST Computer and routed to the ACS through the DCE.

Each of the various sections of the Synthesizer has been designed, wired, and hardware programmed to perform a unique function in sound generation. The basic functions included are:

1. Engine sound generation
2. Aerodynamic sound generation
3. "Clunk", "Thump", "Bang" generation
4. Miscellaneous sound generation

Engine sounds generated include thrust sound, vibration, start-up, shutdown, and thrust termination noises for all engines; that is, for SRM, MPS, RCS, and OMS engines. By virtue of the sequence of events in the mission time line, it is possible to use an engine sound simulation string to generate sounds for various engines that are not normally operated concurrently in the Shuttle.

The Aerodynamic Sound generation model includes simulation of air flow and structural noise as a function of vehicle attitude, altitude, and airspeed. Transonic and supersonic flight effects on sounds will be a part of the Aerodynamic model. Popping and cracking sounds associated with re-entry are also included.

The various "clunks", "thumps", and "bangs" related to events such as drag chute deployment, pyrotechnic separation, and docking will be simulated by the "clunk" section of the Synthesizer.

The Miscellaneous Sounds section generates sounds related to taxi pavement noise, landing gear activation, wheel touchdown, air conditioning duct blasts, fuel cell venting, and other similar, relatively smooth, continuous sounds.

Quadraphonic installation of the speakers and sound transducers in the crew station is shown in Figure 8.6-2. The Speaker Output Control section of the Synthesizer receives commands from the computer, and in some special cases, from the sound generation modules, to control the relative volume of the 4 channels and therefore the directional characteristics of the sound simulated.

LEGEND



= SPEAKER



= AUDIO TRANSDUCER

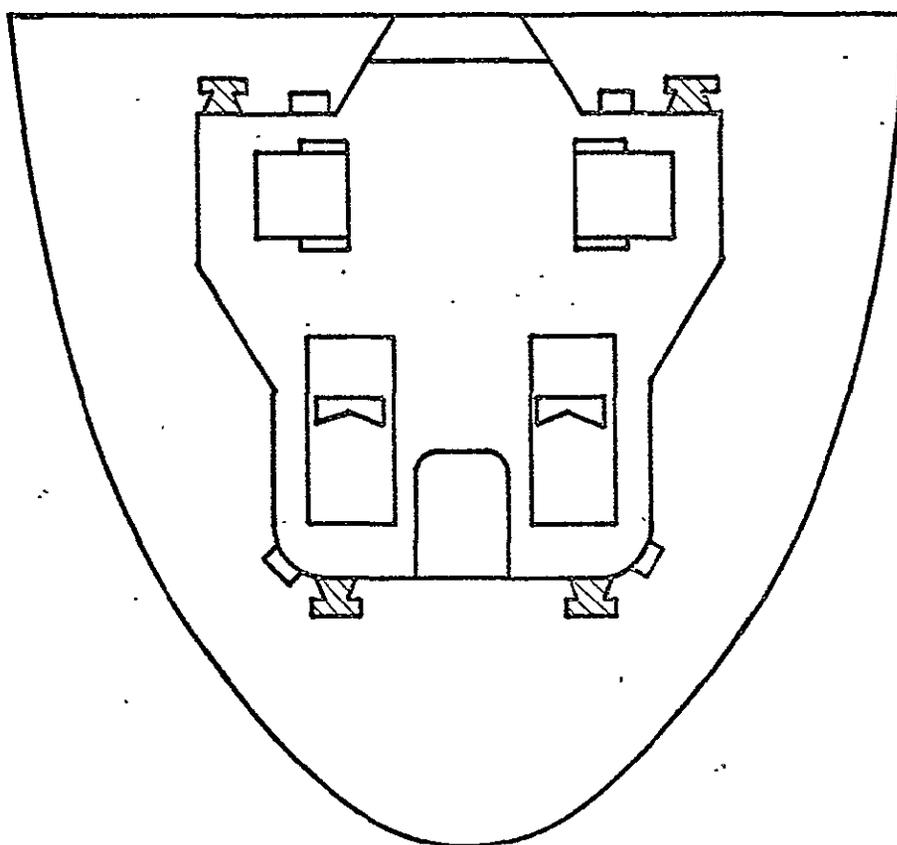


FIGURE 8.6-2 . AURAL CUE SPEAKER LOCATIONS

The aural cues will be presented within the cockpit enclosure as quasi elevated (nonplanar), bilateral quadriphonics with "sound in motion" where required to achieve realism in the training environment.

The parameters indicative of ACS operational status are shown in Table 8.6-1. These are the characteristic parameters measured during self test to determine system readiness, and in some cases, to determine the possible source of a failure.

TABLE 8.6-1. AURAL CUES SYSTEM CHARACTERISTIC PARAMETERS

PARAMETER	OPERATING RANGE
Frequency	10 to 18,000 Hz
Timbre	1 to 5 distinct harmonic frequencies
Volume	40 to 120 decibels
Duration	100 milsec to continuous

#### 8.6.1.2 ACS Self Test Approach

The approach described here for ACS self test is based on the sequential checkout of each section of the Synthesizer to verify its ability to simulate those sounds assigned to that section.

Under computer command, the section under test is directed to generate a particular sound. Microphones, installed in the cockpit for test purposes, pick up the audio signal. The microphone electrical output is digitized and routed to the I/O Interface Minicomputer (and possibly on to the HOST). Fast Fourier Transform processing is performed on the sampled data, and the resulting frequency distribution and amplitude information pattern or signature is compared with the stored reference signature for the commanded sound. If comparison is achieved within pre-defined tolerances, the test moves on to the next sound for the section under test, or to the next section, as required.

The only additional hardware required to implement the ACS self test described above is the four microphones, one for each of the four speaker and transducer installations, and the necessary circuitry to route analog signals to the ADC in the DCE. Additional software is described in Figure 8.6-3.

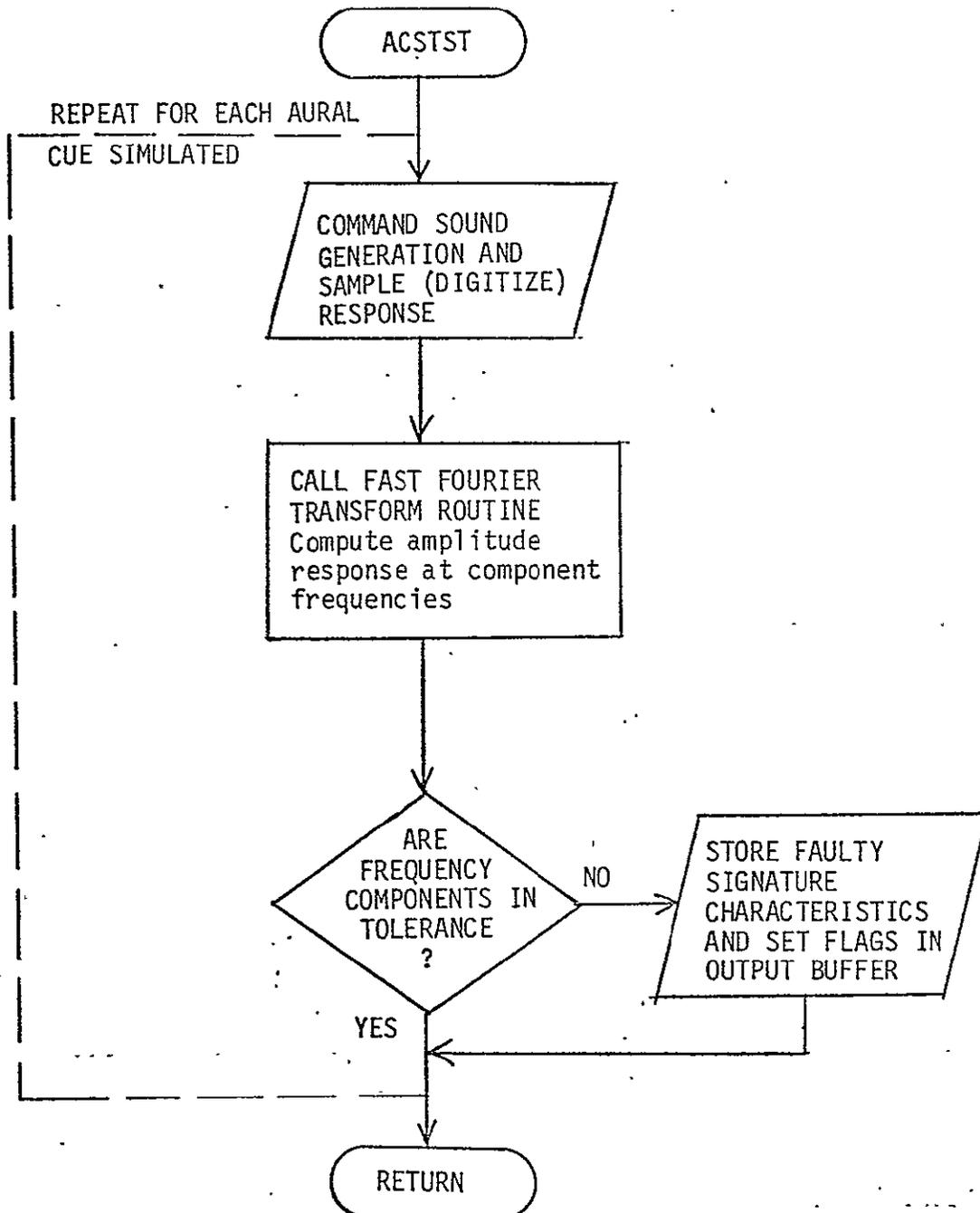


FIGURE 8.6-3 FFT PROCESSING OF AURAL SYSTEM, ACSTST

### 8.6.2 Power Supply System

There are two major areas in the Power Supply System. The AC portion of the system consists of the facility power and the power distribution and circuit protection components. The DC portion consists of the conversion equipment, the distribution cabinets with terminals and switching, and the circuit protection elements for DC supply lines.

Facility power consists of 277/480 V, 120/208V 60Hz, 3 phase power, and 115 V 400 Hz, single phase power. The 277/480 V power is used to drive high demand motors such as the hydraulic boost pump motor. The 120/208 V power energizes computer equipment, visual system, and the DC power supplies used by other subsystems. The 400 Hz source powers various elements of the simulator such as the flight hardware. Overload protection implementation consists of the use of circuit breakers on power output lines, and the use of a predefined power up sequence that minimizes motor and equipment start up current demands.

There are several power supplies that convert the 120/208 V power to DC voltages at the levels needed to drive electronic equipment in the simulator. Voltage levels include but are not limited to  $\pm 5$ ,  $\pm 10$ , and  $\pm 28$  volts. The supply lines for this power are also protected by means of circuit breakers, as well as visual and auditory alarm signals when voltages, currents, or temperatures exceed predefined limits.

The basic concern of the Self Test System in regards to simulator power is to verify that output voltage levels are within design specifications, and meet the requirements of the equipment they supply. It is also important that self test procedures verify that none of the protective devices has tripped to indicate a failure or potentially hazardous condition. Therefore, prior to initiating self test activities in the simulator system, it is necessary to make voltage measurements at the various supply busses, and to verify the status of overload and out of limit caution and warning indicators for the power system. These checks can be made automatically if the DCE and other elements in the measuring string can be used, or manually if automatic measurements are not possible.

### 8.6.3 External Clocks and Timing Equipment

Time synchronization and sequencing of mission events during training simulation is achieved by using a Mission Elapsed Time (EMT) generator and a Greenwich Mean Time (GMT) generator. In addition to these absolute time generators, there are several timing signals routed throughout the simulator in order to trigger cyclic operations, or to synchronize execution of certain events such as data transfers. For example, to enable the Shuttle Mission Simulator to operate in real time synchronism with the Mission Control Center, timing signals will be provided by MCC. These signals consist of the following:

- 1 pulse per minute at 66-2/3% duty cycle
- 1 pulse per second at 80% duty cycle
- 1 Megahertz at 50% duty cycle

The SMU will use these signals to assure synchronization with MCC related activities and also to generate other signals required to provide synchronization of events internal to the simulator. These signals are generated by the Central Timing Equipment and consist of the following clock signals: 128 KHz; 4KHz; 31.25, 25, 2.5, and 1.25 PPS for Flight Computer; 20 PPS; 10 PPS to Event Timer, 1 PPS, and 1 PPM. The generation of all those signals is simply a matter of tuning down in frequency and providing driving circuitry for the synchronizing signals received from MCC.

Self Test of the timing and synchronizing equipment, as in the case of the Power Supply System, consists simply of measuring the various timing distribution busses and verifying that the signal received is of the frequency and amplitude specified. Frequency and wave shape integrity can be verified by Fast Fourier Transform of sampled data for each of the clock signals except those that exceed 20 KHz as this is the upper frequency which can be accurately sampled and analyzed using operational capability of analog DCE.

For the higher frequencies, some specialized hardware would be required to either perform the total processing and notify the test control computer of go or no-go status in the clock signals, or hardware to digitize clock signals and then feed them to the computer in an expanded time scale for analysis. In either case, the higher frequency clock signals can be checked as part of the hardware verification activities on the simulator.

## SECTION 9

### INTEGRATED SYSTEM SELF TEST SUMMARY

In the previous sections, we have discussed the nature of the simulator hardware, the types of tests that are of interest for automatic implementation, the various generic techniques that are available for accomplishing the desired test results, and the hardware and software requirements for implementing each of these tests for each of the simulator subsystems. In this section, we will consider the overall sequencing of the tests for all of the simulator subsystems, as well as the executive software requirements and structure. We will also summarize the hardware and software requirements and will then use the results to consider the impact on the simulator system in terms of both a data base impact and a hardware cost impact.

In Section 9.1, we consider test sequencing requirements in order to implement in an integrated manner all of the subsystem tests previously described to accomplish the fault detection, fault isolation and incipient fault detection functions as required.

In Section 9.2, we address the implementation of this test sequence by description of an appropriate test software executive and discuss the assumptions that this design is based upon.

The hardware required is summarized in Section 9.3 and the data base impact of the test software, considered in total, is discussed in Section 9.4. The cost impact of the total hardware requirements is addressed in Section 9.5 and the impact of simulator system changes on both hardware and software is addressed in Section 9.6.

## 9.1 SIMULATOR SELF TEST SEQUENCING

Testing of the simulator as an integrated system involves the sequential and sometimes concurrent verification of simulator subsystems and functions.

There are three hierarchical classifications of tests determined by the level of detail to which the test is performed and the level of confidence obtained after successful completion of the test. These classifications are Readiness Tests, Fault Isolation Tests, and Incipient Fault Detection Tests.

Readiness Tests are used to verify the readiness of simulator subsystems to perform according to design and operational requirements. Generally these tests check each of the simulator subsystems and, whenever possible, use already verified subsystems to check other subsystems of the simulator. Ideally readiness tests are end to end tests. They are primarily concerned with testing a complete string of hardware.

The sequence of subsystem testing developed in the course of this study is based on the "building block" approach of establishing the operational readiness of a system element before that element can be used to test other portions of the system under test, in this case the simulator. Figure 9.1-1 shows the sequential and parallel arrangement for test execution during the Readiness Test. Individual tests shown at the same horizontal level can be executed simultaneously by the various relatively independent processors in the simulator. Checkout of the HOST computer is required prior to commencing with simulator checkout.

Each of the vertical strings indicate dependence of tests, lower on the chart, on successful completion of previous tests. For example, the simulator interface minicomputer must pass successfully its readiness self test prior to executing the routines for DCE self test. Failure of the minicomputer to pass this test would return information to the HOST regarding the point or function at which the test failed. On the other hand, a failure in the DCE would prevent Crew Station or Visual Simulation system checkout; and it would be up to HOST computer software to determine what automatic fault isolation tests should be run on the DCE.

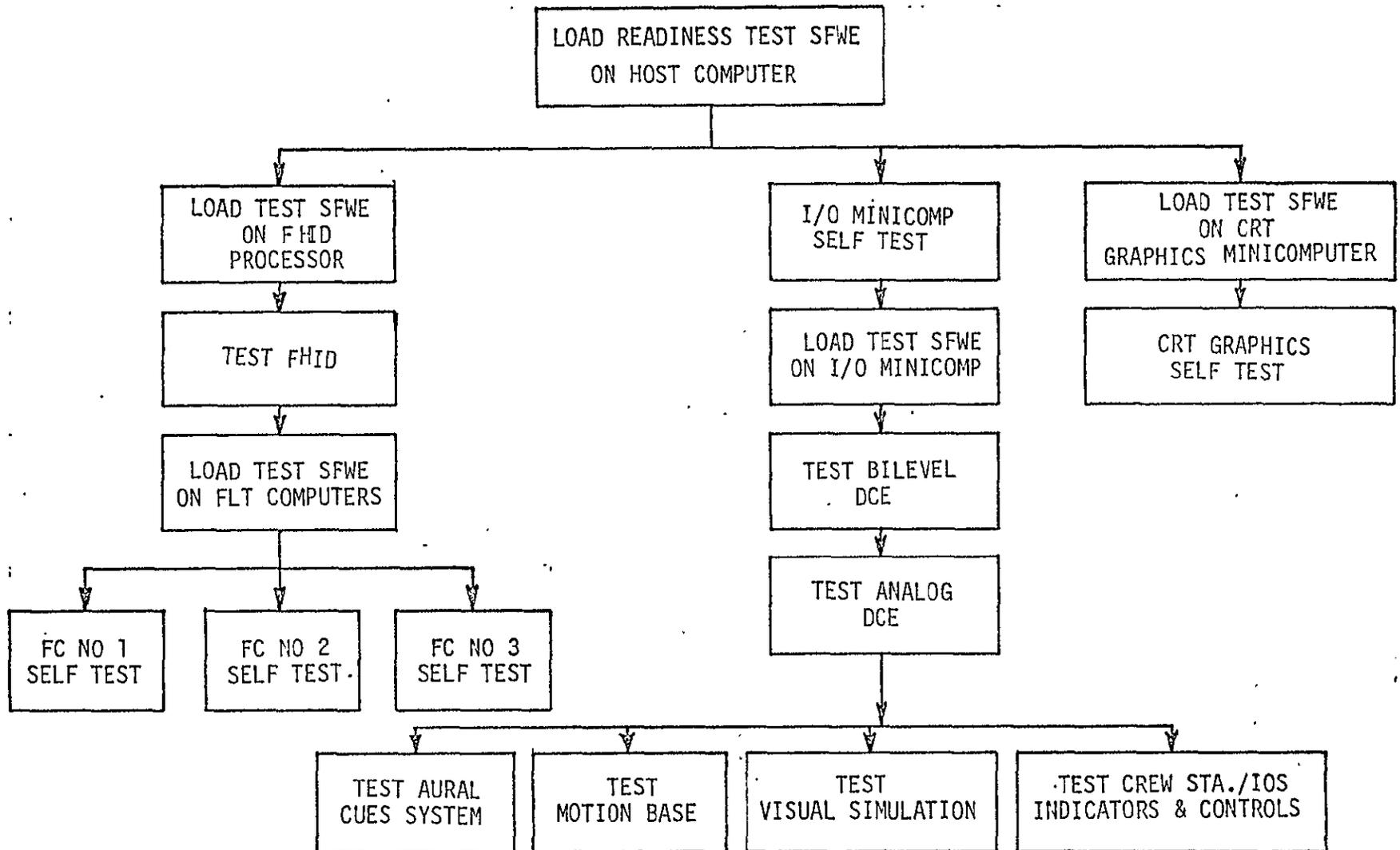


FIGURE 9.1-1 READINESS TEST ELEMENTS AND EXECUTION SEQUENCE

The self test software is initially in mass storage elements of the HOST computer complex, and is loaded into the system under control of and through the interface facilities of the HOST computer. At the conclusion of the load, each processor returns to the HOST a message indicating that the load has been completed, and some other information such as a checksum which the HOST can then use to verify that a successful, accurate load has been accomplished.

After the loading process is completed, and the HOST has verified that all went well, each processor is commanded by the HOST to execute the subsystem Self Test programs.

The simulator processing elements that receive this self test software load are the Flight Hardware Interface Device processors; the simulator interface for the DCE, crew station, visuals, and motion base; and the IOS Graphics minicomputer. Testing of the Flight Computers is not performed until after the FHID is thoroughly checked out; therefore, loading of the Flight Computer self test software does not take place until after completion of FHID Self Test. This particular sequence is necessary because FC Self Test requires correct performance of FHID hardware not only during the load, but also in checking FC capability to communicate with avionic components external to the FC and simulated by the FHID.

Completion of each level of Readiness Tests results in control being returned to the HOST, which then logs any subsystem operational data gathered during the test, issues appropriate messages to the Test Operator, and determines the next logical step in the test sequence.

The readiness tests are intended to be overall functional tests for the various simulator subsystems and as such should require only a limited amount of time to execute. It is assumed that the readiness tests would nominally be executed every day or at the beginning of each training shift. In the event that a failure or fault is detected by the Readiness Test, it may be necessary to execute a Fault Isolation Test in order to identify more precisely the LRU

which is the source of the problem. This is only necessary in the case of the DCE, the motion base, the visual system and the aural simulations.

If the fault isolation test is required, then a question arises as to whether it should be executed automatically or whether the operator should be required to initiate the additional test by further positive action. In design of the test software automatic execution is assumed since this establishes the necessary basic sequence elements. The introduction of pauses or overriding interrupt logic is a relatively minor and optional problem when the software is implemented.

Figure 9.1-2 illustrates the operational sequencing of Readiness, Fault, Isolation, and Incipient Fault Detection Testing. The basic data for the Incipient Fault Detection Test can be obtained in most cases during the readiness test execution as shown in this Figure. However, if a fault is detected during the readiness test, then the data obtained should not be entered into the Incipient Fault Detection data base. Tests for this contingency are shown in the software flows.

In summary, then, if the Fault Isolation Tests are initiated automatically, the operator will be provided with an output which identifies the following:

- o Subsystems tested that have successfully completed the readiness check
- o Subsystems tested that have failed the readiness check
- o Results of Fault Isolations Tests for faulty subsystems
- o Results of analyses of Incipient Fault Detection Test data, including the following:
  - o LRU's that have moved into a gray area and are approaching an unsatisfactory level of performance
  - o Projected failure dates/times for near term critical LRU's

The operator must then decide whether to constrain the type of activities allowed in simulator usage, or to initiate maintenance or replacement of the components identified as failed. The output from self tests then is an indication of the readiness status of the system, including information helpful in trouble shooting or restoring any faulty functions or components of the simulator.

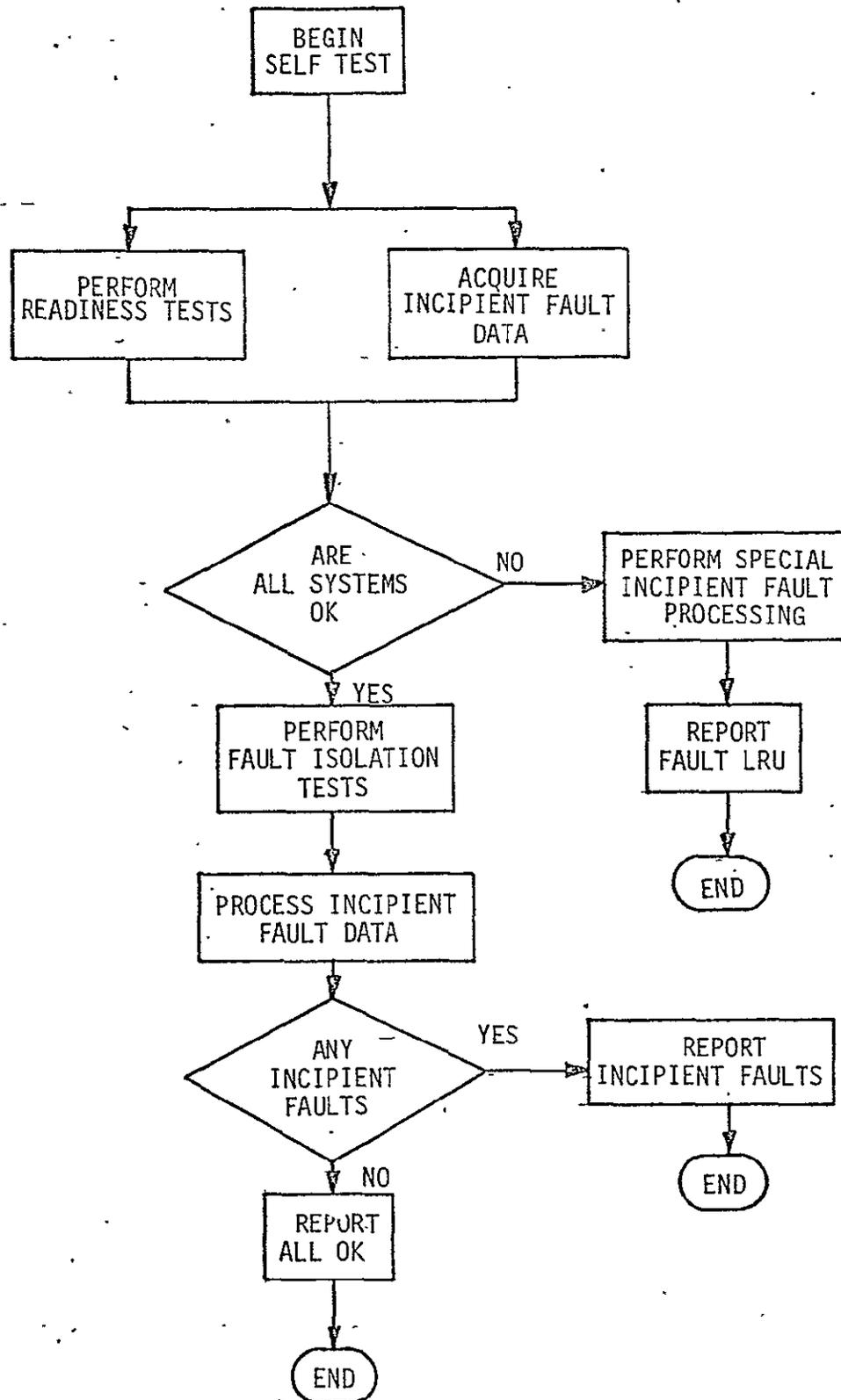


FIGURE 9.1-2 DAILY SELF TEST SEQUENCE

## 9.2 SIMULATOR SELF TEST SOFTWARE SUMMARY

The simulator self test software has been designed and structured so as to implement the test sequences discussed in Section 9.1 with respect to both the order of subsystem testing and the order in which the various types of tests are executed. Since the simulators of interest don't exist currently but are expected to be multi-computer systems with satellite processors available to handle input/output processing, data reformatting, and other miscellaneous operations, it is impossible to predict with much certainty the amount of computing power available in these peripheral processors. Therefore, it is also impossible to rationalize, at this point, the distribution of self test functions between the host computer and the interface computers. As a result, the approach taken in designing the software defers that problem to a later time and presents the software structure so as to define hierarchies of software modules, and the necessary sequencing of operations and channels of data communication.

A simulator self test software tree is shown in Figure 9.2-1. The software modules are identified by both an acronym, suitable for programming use, and a descriptive title. The modules shown in this tree do not have the relationship usually implied by a software tree in the sense that the modules at the two lower levels are not subprograms to the modules above them. Instead the implication of the structure of the tree is that modules at the same level may be exercised concurrently and independently, resources permitting; all modules in the same string but at a higher level must be exercised successfully before that module may be exercised; or, in other words, the modules in a string must be exercised in a top down sequence. All of these modules and programs will be controlled in their execution by the test executive and consequently in an ordinary tree would all be shown directly connected to the executive. Each of these modules has been described in detail in previous sections. In this section, we limit ourselves to describing the test executive software.

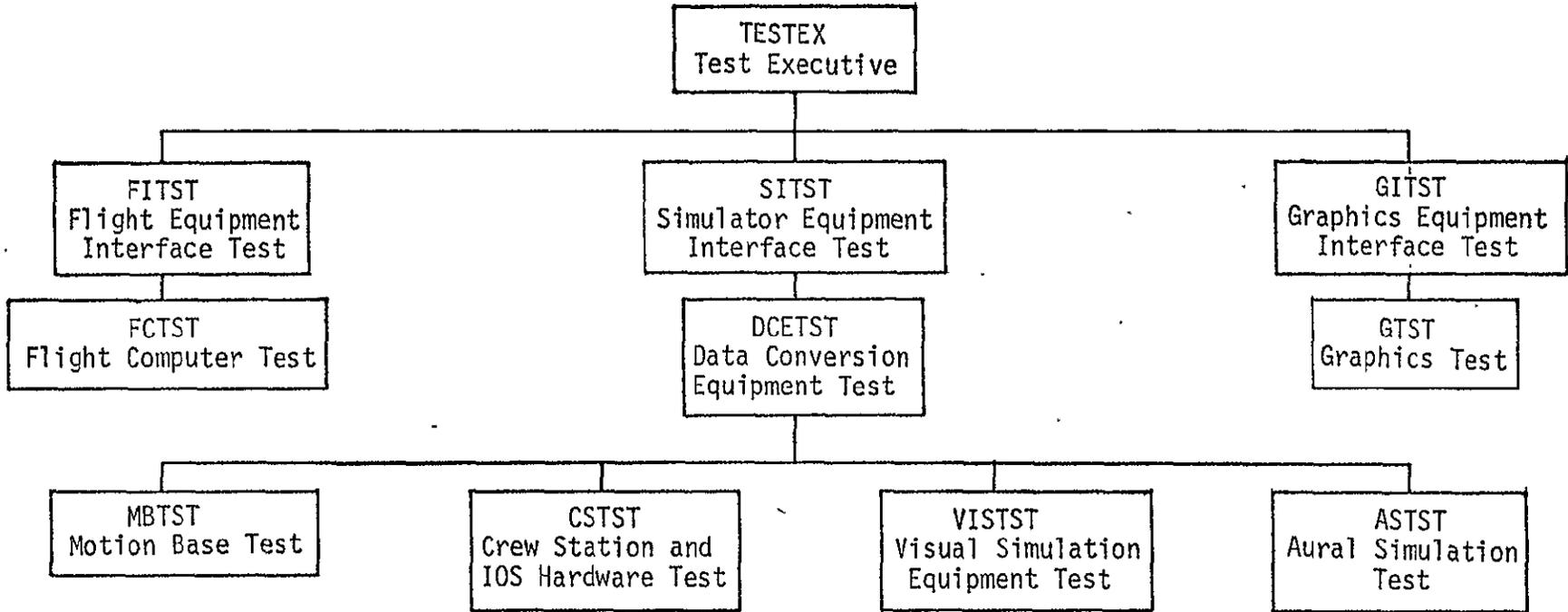


FIGURE 9.2-1 SIMULATOR SELF TEST SOFTWARE TREE

### 9.2.1 Self Test Executive Software

A software flow for the test executive is shown in Figure 9.2-2. The test executive is primarily concerned with controlling the order and extent of the testing executed and communicating the proper output messages and data for operator use as well as for storage in the data base for incipient fault detection. The test executive accomplishes its function by initiating and controlling the loading of all test software from mass storage facilities, commanding execution of the tests, reviewing the processed results from the tests and then selecting the proper course of action which will include in most cases, the presentation of data to the operator, the selection of additional tests, the storage of historical data in data base files, and the printing of test data reports.

It should be noted that the test executive does not become involved in the dynamic test timing function. The test data sampling rates are established by the test software for each subsystem being tested. Time tags for the data points are obtained from an external clock and are loaded into the controlling computer at the same time the data words are loaded. This minimizes the dependence of the subsystem tests on proper operation of all computer and interface timing functions and eliminates unpredictable interrupts and interrupt processing as an error source in test data timing.

The executive does contain the logic to decide on the course of action after each level of testing is completed. This logic requires access to the test results temporarily stored in a memory buffer.

When testing is completed or has proceeded to a standstill, the executive formats the data, if necessary, and communicates the appropriate information as follows:

- o Operator display - Operator action requirements, including "system ready"
- o Hardcopy printer - Daily test results summary and reference data
- o Data base file - Incipient fault data base update

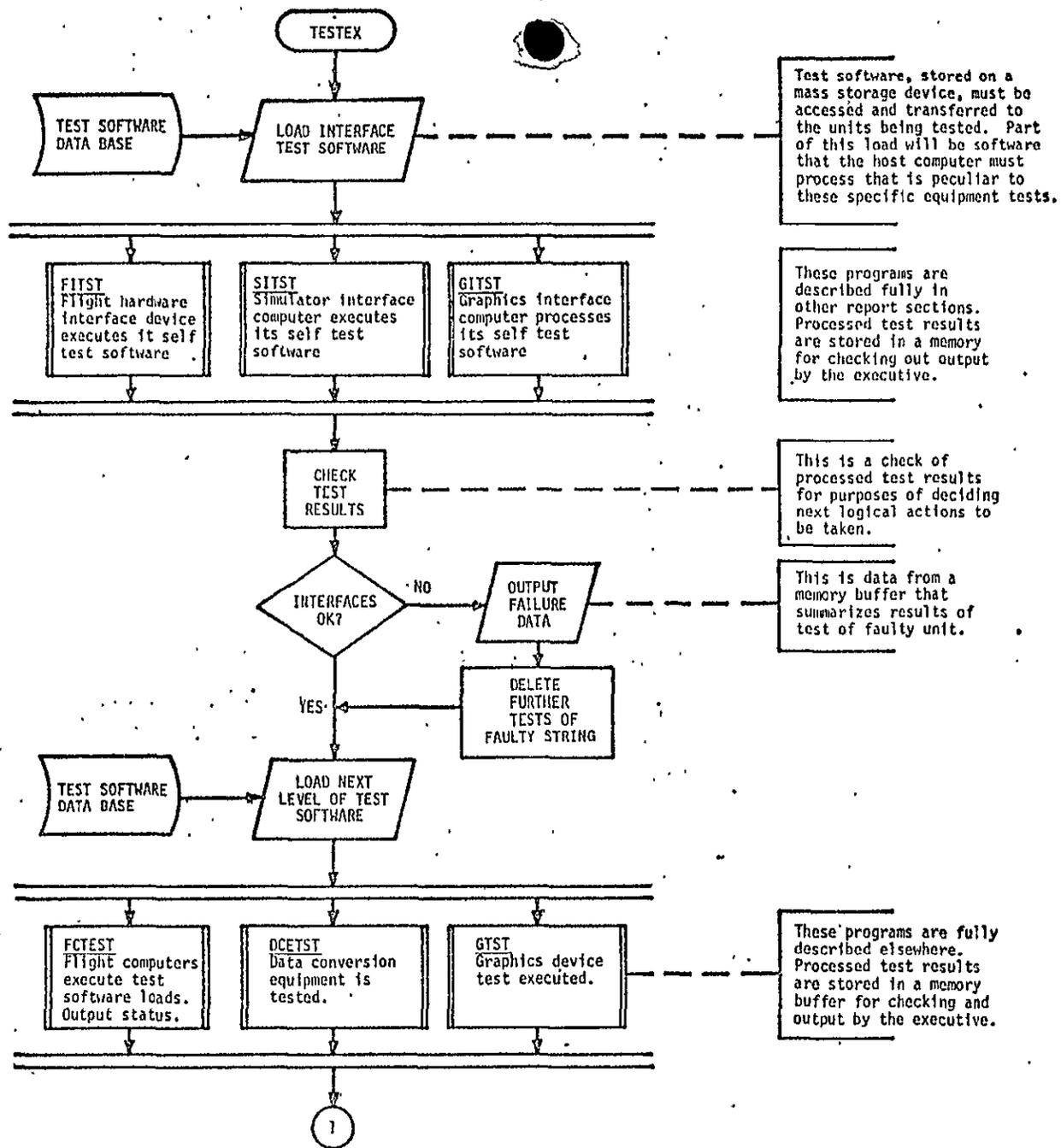
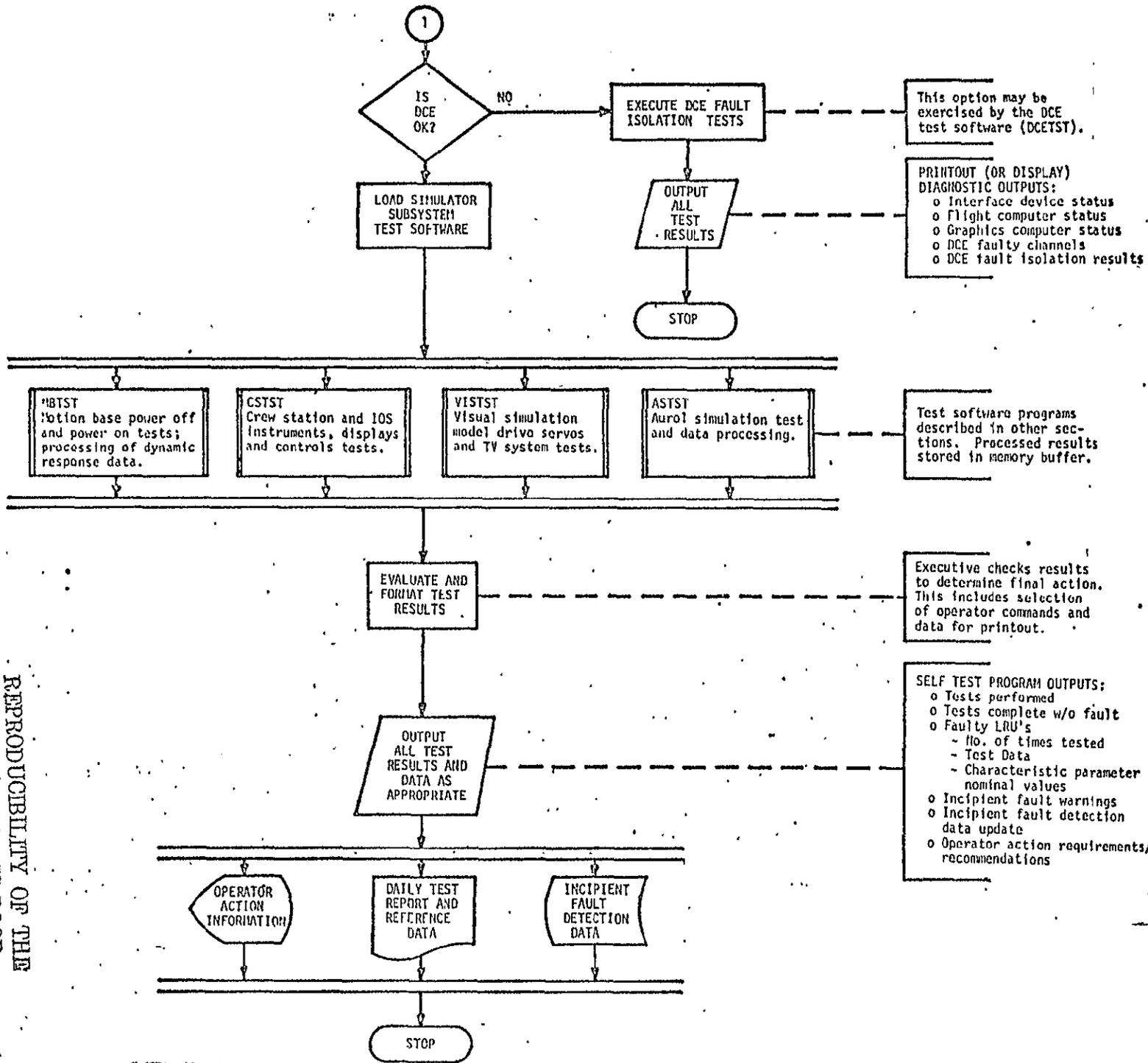


FIGURE 9.2-2 SELF TEST EXECUTIVE SOFTWARE FLOW

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR



### 9.2.2 Subsystem Test Software

The subsystem test software, such as FITST or MBTST, is loaded by the executive, TESTEX, and includes the programming which implements the tests, the characteristic parameter data associated with the hardware being tested, and the data processing software required for reducing the test data. The output of the subsystem test software is stored in a memory buffer for final disposition by the executive, TESTEX.

There are three different types of testing/processing at the subsystem level. These consist of readiness testing, fault isolation testing, and incipient fault detection processing, and are usually executed in that order.

The readiness tests are ideally end-to-end tests or total performance tests for a subsystem. If a readiness test indicates a fault and fault isolation tests are applicable, the decision to proceed with fault isolation can be taken automatically at the subsystem test software level, and the logic is shown accordingly as part of the subsystem tests. It is recognized that this decision could be exercised at the executive level where the results of other concurrent subsystem tests could also be considered. The decision can also be reserved for manual selection by the operator. Final choices can be made during software implementation without any significant impact on the software development.

Incipient fault detection data should nominally be gathered during the readiness tests. If a fault is detected at that time, the incipient fault detection data may be adversely affected. Consequently, the decision should be made at the subsystem level to delete this data and not add it to the data base. Otherwise, the incipient fault detection data is communicated to the executive through the memory buffer.

### 9.3 SELF TEST HARDWARE SUMMARY

This section summarizes the total hardware requirements to implement the Simulator Self Test System developed in the course of the study. The hardware listed includes only those components added for test purposes. A primary groundrule of the study was to maximize use of operational hardware in carrying out the Self Test job. This groundrule has been adhered to throughout the study, and hardware used in this manner is not specifically identified in this section. Cost data on Self Test System hardware is presented and discussed in Section 7.

Table 9.3-1 is a list of the various components added to each of the subsystems for Self Test purposes. Each photosensor used for galvanometer testing include a photodiode as a built-in light source. These devices are compatible with signal levels of logic circuitry used in the Data Conversion Equipment and therefore, no additional signal conditioning components are needed.

The current sensor circuits used to verify operation of lighted indicators are counted as a single component per indicator. These sensor circuits contain resistors, diodes, and transistors. However, each sensor circuit can be packaged in an individual module and using state of the art circuit integration technology, the circuits could be packaged right in the indicator assembly.

The loop closure switches listed for the Analog DCE are those solid state switches used to connect output channels to input channels in order to verify DCE operational status. Also under DCE is included the necessary components, ADC, switches, and buffer amplifiers needed to provide input channels for measurements taken during the Self Test process, measurements using instrumentation specifically added as part of the Self Test System.

Table 9.3-2 shows the breakdown of additional DCE channel requirements for Self Test. These requirements cover the data acquisition channels used specifically to measure characteristic parameters of simulator hardware during Readiness, Fault Isolation, or Incipient Fault Detection testing. No additional command channels were identified for applying test stimuli to operational hardware.

TABLE 9.3-1 HARDWARE ADDED FOR SELF TEST

COMPONENT	QUANTITY
Galvanometers (70) 3 photosensors in each	210
Lighted Indicator Current Sensor Circuits	700
Analog DCE - Loop Closure Switches Closure to test input channels	200
Crew station isolation and connection of output channels to the 200 operational input channels	200
Analog DCE Test Channel Components	
ADC	1
Buffer amplifiers	120
Solid state analog switches	120
Bilevel DCE - Loop Closure Gating and Digital Input Multiplex Selection Gates	2500
Visual Simulation System Self Test Hardware	
Digital processing oscilloscope	1
Signal generator	1
Signal to noise meter	1
Video switches	16
Video buffer amplifiers	36
Motion Base	
Pressure sensors	5
Level sensor	4
Temp sensor	2

TABLE 9.3-2 ADDITIONAL DCE CHANNEL REQUIREMENTS FOR SELF TEST

COMPONENT TESTED	CHANNEL REQUIREMENTS	
	Bilevel	Analog
Galvanometers	210	
Servometers		16
Tapemeters		20
ADI (4 units) 4 wire sin/cos, 3 channels each 6 pointer position feedback signals from each ADI	48	24
Lighted Indicator	700	
Electromechanical Flags	400	
Switches - No extra lines for testing	---	
Visual Model Servos 3 cameras x 6 servos Terminal area transport servos Orbital earth position resolver Cloud/sky/terminator altitude servos	30	18 4 3
Motion Base Sensors 11 Position, actuators 6		17
	<hr/> 1388	<hr/> 102
With Spares	1500	125

However, in the case of the video components of the Visual Simulation System, the host controlling computer, with DCE, must issue the necessary commands to re-configure the test hardware and control the test execution. These commands activate the DPO, signal generators, and measuring equipment used to test video elements. The number of these commands is relatively small (less than 20) and represent negligible impact on the total DCE.

On the other hand, the additional input channel requirements of 1500 bilevel and 125 analog channels for Self Test purposes represent an effective doubling of DCE capability when compared with the Reference Configuration magnitudes of 2000 bilevel and 100 analog channels. Considerable increase in parts and manufacturing cost should be expected from the fact that the increase in DCE capability is more accentuated on the analog instead of the bilevel side of the DCE. Analog channel components are larger, more complex, and more costly than the gating required to add bilevel channels.

Additional test hardware for the Visual Simulation System consists of the Digital Processing Oscilloscope, a video test waveform generator, a signal to noise meter, and the necessary switching to isolate each LRU during the testing sequence. No additional hardware is expected for testing model scene generation equipment other than that listed under DCE.

The Motion Base test hardware consists of the sensors and signal conditioning circuits required to route characteristic parameter data to the computer. Normally these parameters are only routed to the maintenance panel either electrically, mechanically, or hydraulically. In order to implement the automatic Self Test capability developed in this study, it is also necessary to digitize characteristic parameter data and supply them to the computer conducting the test.

The amount of hardware discussed above is a significant measure of the impact on simulator design and maintenance by the addition of automatic Self Test. Cost data related to this impact is discussed in Section 7. However, at this point it is important to point out that parts cost is a small part of total impact when compared with the total cost that includes generation of

-additional wiring instructions, manufacturing additional housing space for these components, actual wiring cost, and increase in acceptance and integrated testing of the resulting system. All of these costs, however, may be offset by the increase achieved in simulator availability and utility during the operational life of the system. They also represent a very small percent increase to simulator total procurement cost.

9.4 DATA BASE IMPACT

The impact on a data base system of the self test software requirements described throughout the report for the various simulator subsystems can be assessed in terms of the software loading (instructions plus data) imposed on the system. Since these size parameters are highly dependent on the specific configuration being considered, we will address the software loadings in terms of each of the individual subsystems.

The flight hardware interface device may be a custom built unit incorporating standard circuit modules or it may be derived from a high performance minicomputer. In either case, the components of the interface system are basic in nature and configuration to a simple computer structure. It is convenient to consider the test software for the interface device as if it were a minicomputer and to assume as an approximation that the flight computer test software is also similar to minicomputer practice. Based on data from a typical vendor, the following software loading is associated with the particular test software noted:

<u>Test</u>	<u>Storage Required (Decimal locations)</u>
ALU	1000
Memory	500
Instruction execution	1800
Instruction timing	1800
Real time clock	<u>600</u>
Total	5700

Rounding off the estimate and recognizing that essentially different software is required for the two units, it can be estimated that the software required for the flight computers and their interface will total about 6000 decimal locations each or about 12,000 locations total.

Software loadings imposed by the data conversion equipment test software should be nominal for a system that is designed with current technology as an integrated system. Requirements for older systems or systems derived from available equipment rather than newly designed systems can be expected to pose greater problems and require more software. For the proposed tests, the tolerances required can be expressed as percentages which reduces data requirements substantial although scale factors may still be required for the analog channels. These

parameters represent a minor part of the required software which should consist largely of problem logic. An estimate of 2000 locations for data and instructions should be ample.

The control and display software excluding the graphics display test software totaled 1600 locations including floating point data and instructions. The data requirements are extremely conservative in this estimate since it was assumed a unique set of hardware characteristics was associated with each analog hardware device although many of these devices may be physically similar.

Visual subsystems software requirements are extremely hard to estimate although they should not be expected to be overly large. This is because the proposed approach relies on hardware signal generators for test signal generation. The data requirements and software complexity are a function of the type of color TV systems implemented, how many separate strings of television hardware are involved, whether they are all alike or different for different scenes, and many other factors. Nevertheless, software requirements should be quite modest and 2000 locations should be adequate for the necessary logic and the minimal amount of data.

Motion base software requirements are extremely minimal. The software required to accomplish static power on and power off checks should not amount to more than 500 locations including data. If a more sophisticated analysis is implemented, using Fast Fourier Transform, this software does not consume much memory for itself; 300-400 locations is a reasonable estimate.

Totaling the above major items of test software produces a grand total of about 18.5 thousand locations.

## 5 SELF TEST SYSTEM RELATIVE COSTS

Relative cost data for addition of automatic self test capability to the Reference Configuration Simulator is shown in Table 9.5-1. No absolute dollar figures are shown because the selection of test component configuration, packaging, manufacturing, and installation techniques are dependent on the actual simulator design used, as well as the date in time when self test system implementation is to take place.

The first column shows cost of self test system elements relative to the basic cost of the subsystem to be tested. This rating ranges from Negligible (0-5%) to Moderate (20-30%). No "High" ratings were given to self test cost for any subsystem as this rating would have implied greater than a 50% increase in cost of the basic system. Consistent utilization of existing operational capability in the system for self test purpose is the basic reason for maintaining cost between Moderate and None.

The five columns to the right of the Table show cost of different factors implementing subsystem self test as a percentage of total Self Test System cost. Some of these factors are estimated as less than 1% and therefore are shown only as \*. The total cost of the Self Test System adds only to 98%; the remaining 2% is accounted for in the \* items and others which exceed the whole number shown.

The largest percentage of Self Test System cost appears in Visual Simulation CCTV Self Test. This high percentage is due to the cost of the Digital Processing Oscilloscope, the special video pattern test signal generators, and the various signal and noise meters. The cost of visual simulation video component self test is further increased by test software development cost. This software is relatively more sophisticated than other elements of the test software in that it must provide processing capability for various complex waveforms from the video string. These waveforms have a variety of frequency components, require sampling rates of up to 50 Megahertz, and include test patterns such as a stair step and frequency multiburst signals.

TABLE 9.5-1 RELATIVE COST FACTORS OF SELF TEST SYSTEM

SUBSYSTEM	SELF TEST COST RELATIVE TO SIMULATOR SUBSYSTEM COST	SELF TEST COST RELATIVE TO TOTAL SELF TEST SYSTEM COST				TOTAL PER SUBSYSTEM
		DESIGN	FABRICATION & ASSEMBLY	HARDWARE COMPONENTS	SOFTWARE	
ANCILLARY COMPUTERS	LOW	2	*	1	5	8
FLIGHT HARDWARE	NEGLIGIBLE	1	1	1	2	5
CREW STATION AND IOS	MODERATE	3	3	5	10	21
ANALOG DCE	MODERATE	1	2	3	3	9
BILEVEL DCE	MODERATE	1	3	1	3	8
VISUAL SIMULATION-CCTV	NEGLIGIBLE	3	3	11	9	26
VISUAL SIMULATION-MODELS	NEGLIGIBLE	2	*	2	4	8
MOTION SYSTEM	LOW	1	1	3	5	10
AURAL CUES SYSTEM	MOD-LOW	1	*	*	2	3
					TOTAL **	98%
* Items that are less than 1% of total Self Test System cost						
** 2% remain to cover costs not indicated numerically						

MCDONNELL DOUGLAS ASTRONAUTICS COMPANY - EAST

9-21

MDC E1150  
1 November 1974

Following closely in percentage of cost is the Crew Station and IOS self test factors. The high cost of controls and displays checkout is owed to the large number and variety of components to be tested, which in turn requires a fairly large characteristic parameter data base. In addition to the data base software, a generalized routine must be developed for each of the types of components used. Each component must be tested individually and therefore requires its own unique test calling and control sequence. With component count in the hundreds (thousands, if switches included), it can be seen that these sequences make up a considerable amount of software to be written, integrated, tested, and documented. Also, the large number of components affect design, manufacturing, and test component costs to make Crew Station and IOS self test a high portion of total Self Test System cost.

## 9.6 SELF TEST SYSTEM IMPACT BY REQUIREMENTS OR DESIGN CHANGES

This section discusses various types of changes that can occur after the simulator has become operational and the sources of changes. Particular attention is given to the impact of these changes both on the Simulator System as well as in the Self Test System. This discussion is the basis for assessing the effect of changes on the operational simulator and Simulator Self Test systems, and for formulating some recommendations that could minimize impact.

### 9.6.1 Sources of Change

As indicated above, this section only deals with changes initiated after the simulator has been installed, accepted, and is in its operational phase. Changes during design or development of the simulator per se are not considered here, because these changes are allowed a greater impact than those during the operational phase, and also because when considering the implementation of development changes, the impact of these changes on other parts of the system is evaluated before permitting implementation.

The major sources of changes during the operational life of the Simulator are:

1. Mission peculiar requirements variation from flight to flight.
2. Vehicle design changes.
3. Advancements in state of the art and
4. Performance enhancements.

These sources of change are discussed below.

#### 9.6.1.1 Mission Peculiar Requirements Variations

The primary mission of Shuttle simulators is to provide a training facility for the various piloting and mission oriented crews that will man Shuttle controls during the life of the program. Although the aerodynamic and vehicle performance characteristics will not vary considerably from flight to flight, it is reasonable to expect significant changes in mission objectives, payload management requirements, and general procedures as dictated by the type of trajectory, orbital characteristics, and maneuvering requirements needed to accomplish these mission objectives.

Mission related changes manifest themselves in the Simulator in the need for special payload management controls, payload related math models, and development of procedures that permit efficient utilization of vehicle and payload systems in accomplishing mission objectives. A dramatic example of mission related Simulator changes is the addition of payload environmental control and life support management facilities in the crew station, as well as the addition of math model restrictions in payload manipulation when changing Simulator configuration from a Shuttle carrying only unmanned payloads, to one carrying a mix of manned and unmanned payloads.

#### 9.6.1.2 Vehicle Design Changes

This source of Simulator changes is more difficult to visualize at this point in the Shuttle program. The design of the vehicle is an ongoing effort, and additions and changes are taking place on a daily basis. It would be ideal if this process could achieve an optimal configuration prior to the first space flight; however, experience in previous programs has shown otherwise, and therefore, it is reasonable to expect design changes to take place as problems become more apparent during early flights of the Shuttle.

Design changes may affect vehicle performance or payload accommodation facilities and consequently alter the control and displays layout in the crew station, as well as the looks, feel or sound of related cues.

#### 9.6.1.3 State of the Art Advancements

Even more difficult to visualize than vehicle design changes are those modifications that result from advances in the state of the art of simulator hardware. Generally, these modifications result in the replacement of multiple components for an integrated system that performs the same job more effectively. Effectiveness in this case refers to fidelity of simulation as a primary intention, and cost of operation when compared with the price of implementing the change.

#### 9.6.1.4 Performance Enhancement

The fourth source of simulator modifications is the need to improve performance of the simulator beyond that which was specified, and supposedly met as shown by acceptance test results. Again, the primary consideration in this

case is fidelity of simulation. The situation may develop such that although the simulator performs as specified, it does not present realistic cues to the crew so that positive training can take place in an environment that truly represents the vehicle in looks, feels, sound, and performance; consequently, modifications are needed to enhance the simulation capability of the system.

Performance enhancement changes would generally manifest themselves in the installation of new equipment, or replacement of existing components with units that have more capability. These changes would usually result in an increase of complexity in the simulator system, and therefore increase the need for control as well as data acquisition channels in quantity and sometimes in capability.

### 9.6.2 Impact of Simulator Changes

Implementation of changes on the simulator system requires additions, deletions, and reconfiguration of hardware and software. This impact will affect simulation as well as self test elements. In many cases, also, the characteristic parameter data base will be impacted. Table 6.2-1 itemizes the impact of various changes on the affected elements of the system.

The impact analysis for changes in interface minicomputers, visual simulation, and motion base is unnecessary since changes in these subsystems are unlikely after simulator acceptance. These subsystems are large, complex, and generally the subject of individually managed procurement efforts. This special attention in procurement yields comprehensive requirement specifications for components that are reliable, maintainable, and capable of meeting future needs of the simulator.

The Crew Station and IOS control and display elements may be changed either by vehicle changes, mission requirement changes, or the need to improve fidelity of simulation. Flight hardware used as part of the simulator system may also be the subject of change by either of the reasons mentioned above. The changes shown on Table 9.6-1 for the DCE generally result from changes in Crew Station and IOS controls and displays or in simulator Flight Hardware. Minor changes may develop to take care of needs for additional control or instrumentation of simulator functions.

Notice that the changes discussed may originate from needs to modify the simulation system, or from needs to modify the simulator Self Test System. Changes in one system generally impact configuration of the other system. This relationship is shown in the Table. Also, it should be noted that the characteristic parameter data base, as expected, is only affected when new hardware is introduced into the system, or when some hardware is removed or replaced.

TABLE 9.6-1 IMPACT OF SIMULATOR CHANGES

TYPE OF SIMULATOR CHANGE	IMPACT ON SIMULATOR HARDWARE	IMPACT ON SELF TEST HARDWARE	IMPACT ON SELF TEST SOFTWARE	IMPACT ON CHARACTERISTIC PARAMETER DATA BASE
<p>CREW STATION/IOS</p> <ul style="list-style-type: none"> <li>o ADD CONTROL OR DISPLAY COMPONENT</li> <li>o CHANGE CONTROL OR DISPLAY COMPONENT</li> <li>o DELETE CONTROL OR DISPLAY COMPONENT</li> </ul>	<ul style="list-style-type: none"> <li>o Panel Modification</li> <li>o Add command and response channels</li> <li>o Minor. Maybe require changes in mounting hardware</li> <li>o Reconfigure Panels</li> </ul>	<ul style="list-style-type: none"> <li>o Add sensors</li> <li>o Add stimuli hardware</li> <li>o Minor, if any</li> <li>o Minor removals</li> </ul>	<ul style="list-style-type: none"> <li>o Add calling sequence for self test routines for new component</li> <li>o Minor, if any</li> <li>o Delete calling sequence for deleted component self test</li> </ul>	<ul style="list-style-type: none"> <li>o Add new component characteristic parameter data</li> <li>o Change parameter data</li> <li>o Remove parameter data</li> </ul>
<p>DATA CONVERSION EQUIP</p> <ul style="list-style-type: none"> <li>o ADD CHANNELS - ANALOG OR BILEVEL</li> <li>o REASSIGN CHANNELS</li> </ul>	<ul style="list-style-type: none"> <li>o Add wiring, cabling, drivers/receivers, buffer amps &amp; mux gating</li> <li>o Implement wiring or patching changes</li> </ul>	<ul style="list-style-type: none"> <li>o Add loop-back channels and switches to check new channels</li> <li>o None --</li> </ul>	<ul style="list-style-type: none"> <li>o Extend channel check range in calling sequence to include new channels</li> <li>o None --</li> </ul>	<ul style="list-style-type: none"> <li>o Add new channel numbers &amp; special characteristics, if any</li> <li>o None --</li> </ul>
<p>FLIGHT HARDWARE</p> <ul style="list-style-type: none"> <li>o USE ACTUAL HDWE INSTEAD OF SOFTWARE SIMULATED HDWE</li> <li>o CHANGE FLT HDWE IN SIMULATOR</li> <li>o DELETE ACTUAL FLT HDWE. MAYBE REPLACE WITH MATH MODELS</li> </ul>	<ul style="list-style-type: none"> <li>o Add I/O channels</li> <li>o Possibly add I/O minicomputer</li> <li>o HOST I/O configuration modifications</li> <li>o Only that required to implement flight hardware change</li> <li>o Decreased requirements by deletion of flight hardware</li> </ul>	<ul style="list-style-type: none"> <li>o Minor, if any</li> <li>o Possible additional sensors &amp; instrumentation</li> <li>o May require different sensors</li> <li>o Minor reduction</li> </ul>	<ul style="list-style-type: none"> <li>o If Hdwe added is different from other in simulator-need new Self Test routines</li> <li>o Otherwise only need test calling sequence</li> <li>o Minor</li> <li>o Problem changes from hdwe verification to performance verification</li> </ul>	<ul style="list-style-type: none"> <li>o Add parameter data for new hardware</li> <li>o Change data base to reflect new hardware</li> <li>o Characteristic parameter data becomes part of math model data base</li> </ul>

MCDONNELL DOUGLAS ASTRONAUTICS COMPANY - EAST

9-27

MDC E1150  
1 November 1974

SECTION 10  
CONCLUSIONS AND RECOMMENDATIONS

The following general conclusions and recommendations have been derived during the course of the Hardware Verification Task and based on the data that was reviewed or derived.

- Extensive test software is expected to be available for the flight computers, the flight computer interface device, and any ancillary minicomputers that may be incorporated into the simulator systems. This software should be specified as part of the procurements of this equipment and utilized by the computer user as best fits the context of his operations.
- Test software for the data conversion equipment is already in use at the center. The switching techniques considered for the present study may be useful for future simulators.
- Instrumentation of the control and display devices in the crew stations is feasible. The value of instrumentation of these devices must be decided on an individual basis but there is no question that such test facilities are most economically and effectively provided as part of the initial simulator procurement.
- The video equipment in the visual simulation can benefit from recent advances in oscilloscope technology for implementation of self test techniques or for provision of test facilities for maintenance personnel. Provision of test capability should be considered when visual systems are procured.
- Motion base equipment is subjected to frequent response testing at several facilities. If the response of the motion base system is restricted by a filter, the value of such testing is limited. Sampling of system parameters with proper instrumentation may be an adequate substitute.
- Testing of the simulation of aural cues can benefit from readily available Fast Fourier Transform techniques for analysis of the spectral response of the synthesized sound.
- The software impact on a data base system of implementation of the complete gamut of tests considered is nominal in terms of magnitudes of software generally associated with a data base as well as relative to future simulation software loads.

SECTION 11

REFERENCES

- 5.1-1- McDonnell Douglas Astronautics Company-East, Development of Simulation Computer Complex Specification - Final Report, Report No. MDC E0857, 29 June 1973.
- 5.2-1 Simulation Products Division of The Singer Company, Shuttle Mission Simulator Baseline Definition Report - Volumes I-II, 22 June 1973.
- 6.2-1 To Kilin, R. E. Tulloss, "Automatic Test Systems," IEEE Spectrum, Vol. 11, No. 9, New York, September 1974.
- 7.1-1 NASA, Application of Modern Network Theory to Analysis of Complex Systems, by John C. Fakan, Rpt. No. NASA TN D-5511, Lewis Research Center, Cleveland, Ohio, October 1969.
- 7.1-2 H. Sriyananda, D. R. Towill, and J. H. Williams, "The Selection of Test Frequencies for System Fault Diagnosis," AIAA Paper 73-664, (Paper presented at the American Institute of Aeronautics and Astronautics, Guidance and Control Conference, Key Biscayne, Florida, August 20-22, 1974).
- 7.1-3 R. F. Garzia, "Fault Isolation in Complex Systems Via Bode Diagram Technique," (Paper presented at the Automatic Support Systems Conference, Barberton, Ohio, 1972).
- 7.3-1 R. S. Shirley and A. D. Jones, "SAFE, Six Axis Frequency Evaluation of a Motion Simulator," AIAA Paper 73-932, (Paper presented at the AIAA Visual and Motion Simulation Conference, Palo Alto, California, September 1973).
- 7.3-2 J. D. Lamb, "Direct Frequency Response Determination Exploiting the Deterministic Characteristics of Pseudo-Random Binary Sequences," (Paper presented at the IERE Conference on Automatic Test Systems, 1970).

- 7.3-3 Honeywell Inc., "Fault Diagnosis by White Noise Techniques," by Victor S. Levadi, Lawrence D. Turner; and Larry W. Christenson, USAF AFAPL -TR-66-43, May 1966.
- 7.3-4 J. R. Fischer, "FORTRAN Program for Fast Fourier Transform," NRL Report No. 7041, Project RF-05-552-403-4069, 1970.
- 7.3-5 E. Oran Brigham, The Fast Fourier Transform, New Jersey, Prentice-Hall, 1974.
- 7.3-6 Computer Sciences Corporation, Fault Isolation Computer Methods, by Ricardo F. Garzia, NASA Contractor Report, CR-1758, February 1971.
- 8.4-1 NASA, Lyndon B. Johnson Space Center, Space Shuttle System Payload Accommodations, Level II Program Definition and Requirements, JSC 07700, Volume XIV, Revision C, 3 July 1974.
- 8.4-2 McDonnell Douglas Electronics Co., Space Shuttle Visual Simulation System Design Study, Design Recommendation Report, Report MSC06741, 20 July 1973.
- 8.4-3 Farrard Optical Co. Inc., Wide Angle Infinite-Depth-f-Field Optical Pickup for Visual Simulation, by Albert H. Nagler, Anthony R. Mazurkewitz, USAF AFHRL-TR-71-41, November 1971.
- 8.4-4 McDonnell Douglas Electronics Co., Space Shuttle Visual Simulation System Design Study, Final Analytical Report, Report MSC 06742, 30 March 1973.
- 8.4-5 Joe Roizen, "Color Camera Basics", Broadcast Engineering, July 1973.
- 8.4-6 Donald G. Fink, Rd., Television Engineering Handbook, McGraw-Hill, 1957.
- 8.4-7 Rudolf Feldt, "Measurement of Noise in Video Systems", Journal of the SMPTE, Volume 77, April 1968.

8.4-8 Tektronix, Inc., "The Oscilloscope with Computing Power", by Hiro Moriyasn, Luis Navarro, Jack Gilmore and Bruce Hamilton, TEKSCOPE, March/April 1973.

8.4-9 Tektronix, Inc., 1975 Products, Beaverton, Oregon, 1974.

APPENDIX A  
SIMULATOR REFERENCE CONFIGURATION  
COMPONENT LISTINGS

APPENDIX A

SIMULATOR REFERENCE CONFIGURATION COMPONENT LISTING

The listings in the following pages summarize the numbers of various components expected to be present in various simulator subsystems. These components represent the types and quantities of devices likely to be present in future NASA simulators, but not necessarily in any specific simulator.

QTY	DEVICE	DESCRIPTION	REMARKS
	- CREW STATION -		
7	Graphics display	CRT, includes graphics, character, and refresh hardware. Serial data input, maximum transfer rate of 40 microseconds/32 bits multiplexed to each CRT.	Flight hardware. Data path and interface to the host computer separate from the crew station.
10	Digital display	7-segment numerical displays. Requires 4 bits per digit input. 3, 4 and 7 digits per display. Decimal digits.	Flight hardware
400	Lighted Indicators	Includes indicators in push button switches.	Flight hardware
20	Lighted Indicators	On-off discrete	Flight hardware
2	Flight Director Attitude Indicator	Pitch, yaw, roll 4-wire sin/cos inputs, 400 hz, 360° movement. 6 pointers DC input.	Flight hardware. Digital-to-resolver circuitry required for drive.
2	Horizontal Situation	Flight instrument modified to accept DC inputs.	Simulated flight hardware

A-4

QTY	DEVICE	DESCRIPTION	REMARKS
2	Air Speed/Mach Indicator	Flight instrument modified with DC servo systems to accept DC inputs.	Simulated flight hardware
2	Vertical Speed Indicator	Flight instrument modified with DC servo systems to accept DC inputs.	Simulated flight hardware.
2	Radio Altimeter	DC inputs	Simulated flight hardware
2	Barometric Altimeter	Flight instrument modified with servo systems to accept DC inputs.	Simulated flight hardware
2	Angle of Attack Indicator	DC inputs	Simulated flight hardware
2	Normal/Longitudinal Acceleration Indicator.	Flight instrument modified with DC servo systems to accept DC inputs.	Simulated flight hardware
2	Aero Surface Position Indicators	DC inputs, DC Meters	Flight hardware
2	Sideslip Indicator	DC inputs	Flight hardware
40	Vertical Meters	DC inputs, $\pm 2\%$ accuracy	Flight hardware
5	Horizontal Meters	DC input, $\pm 2\%$ accuracy	Flight hardware
20	Tape Meters <sup>2</sup>	DC input, $\pm 2\%$ accuracy	Simulated flight hardware
10	DC Meters	DC input, $\pm 2\%$ accuracy	Flight hardware
200	Flags	2 position, discrete input	Flight hardware
10	Flags	2 position, discrete input	Flight hardware

QTY	DEVICE	DESCRIPTION	REMARKS
4	Attitude Hand Controller	Pitch, yaw, roll, two levels of discrettes, proportional analog outputs, 400 hz.	Flight hardware
4	Rudder Pedals with Force feel Simulation	Analog output proportional to position, analog input proportional to surface pressure. Electro-hydraulic control loading servo, contains strain gage load cell, potentiometer amplifier, power supplies. Hydraulic power from motion base system.	Flight hardware with force feel simulation
2	Translation Hand Controller	Lateral, vertical, longitudinal; two levels of discrettes, proportional analog outputs, 400 hz.	Flight hardware
3	Throttles	Proportional analog outputs, DC. Electric motor driven servos, position sensing unit and slip clutch.	
50	Potentiometers	Proportional analog outputs, DC	Flight hardware
75	Pushbutton Switches	On-off; one discrete output	Flight hardware
10	Pushbutton Switches	On-off, one discrete output	Flight hardware
10	Toggle Switches	2 position, one discrete output, microswitch 2TL1, 2 pole.	Flight hardware
250	Toggle Switches, 2 position	On-off, one discrete output, Microswitch 2TL1, 2 pole	Flight hardware
350	3 position	On-off-on, two discrete outputs, Microswitch 2TL1, 2 pole	Flight hardware
400	Circuit Breakers	On-off, resettable, one discrete out, one discrete in.	Simulated flight hardware

A-6

QTY	DEVICE	DESCRIPTION	REMARKS
15	Rotary Switches	9, 7, 6, 5, 4 and 3 positions discrete outputs	Flight hardware
1	Audio Output Devices Communication System	Intercom built like flight hardware  S-band, VHF-1, VHF-2, intercom, caution/warning tones, Tacan, and ILS. Includes white noise generators, oscillators, switching, and voltage-controlled attenuators. Computer controlled attenuation, switching, and noise level.	
	- INSTRUCTOR/OPERATOR - STATION		
6	Graphics display	Inter-active CRT and keyboard, includes graphics and alphanumeric capability. A minicomputer with 16K 1.0 microsecond max. memory interfaces the terminals with the host computer.	Commercial hardware
10	Digital display	7-segment numerical displays; 4 bits per digit input. 3, 4 and 7 digits per display. Decimal digits.	Parallel to crew station displays
300	Lighted Indicators		Parallel to crew station indicators
100	Lighted Indicators	Includes indicators in pushbutton switches	

A-7

QTY	DEVICE	DESCRIPTION	REMARKS
2	Flight Director Attitude Indicator (or equivalent)	Pitch, yaw, roll 4-wire sin/cos inputs, 400 hz, 360° movement. 6 pointers DC input.	Parallel to crew station instruments
2	Horizontal Situation Indicator	Flight instrument modified to accept DC inputs.	Parallel to crew station instruments
2	Air Speed/Mach Indicator	Flight instrument modified with servo systems to accept DC inputs.	Parallel to crew station instruments
2	Vertical Speed Indicator	Flight instrument modified with servo systems to accept DC inputs.	Parallel to crew station instruments
2	Radio Altimeter	DC inputs	Parallel to crew station instruments
2	Barometric Altimeter	Flight instrument modified with servo systems to accept DC inputs.	Parallel to crew station instruments
2	Angle of Attack Indicator	DC inputs	Parallel to crew station instruments
2	Normal/Longitudinal Acceleration Indicator	Flight instrument modified with DC servo systems to accept DC inputs.	Parallel to crew station instruments
2	Aero Surface Position Indicator	DC inputs, flight hardware	Parallel to crew station, instruments
2	Sideslip Indicator	DC inputs, flight hardware	Parallel to crew station instruments

QTY	DEVICE	DESCRIPTION	REMARKS
20	Vertical Meters	DC inputs, $\pm 2\%$ accuracy	Parallel to crew station
5	Horizontal Meters	DC inputs, $\pm 2\%$ accuracy	Parallel to crew station
10	DC Meters	DC inputs $\pm 2\%$ accuracy	Parallel to crew station
200	Flags	2 position	Parallel to crew station
5	Potentiometers	DC analog output	
60	Pushbutton Switches	On-off, one discrete output.	
5	DC Meters	DC inputs, $\pm 2\%$ accuracy	Indicates direction of rotational hand controller.
- VISUAL SYSTEM -			
1	Star Scene	Computer generated image, display format same as CCTV. Navigation stars $\pm 3$ arc min., other stars $\pm 15$ arc min. position.	
1	Orbital Earth scene	Pitch, yaw, roll of spacecraft by servo driven prisms, D/R, 400 Hz. Globe 15 bit digital servo with shaft encoders, CCTV, $\pm 1/2$ min. arc.	More flexible and easier to maintain than film strips. Film strips difficult to synch for various windows.
1	Rendezvous and Docking	Computer generated image, display format same as CCTV.	
1	Landing Scene	TV/model, terrain map, 40'x 40', 3000:1 scale, 0.08" closest approach, D/R 400 Hz servos.	

QTY	DEVICE	DESCRIPTION	REMARKS
1	Payload Handling Scene	Computer Generated Image, display format same as CCTV.	
1	Cloud Cover and Terminator	TV/Transparency, large rear projection, hemispherical screen, film strip projector	Image high altitude, orbital and ascent sky/cloud scenes
5	TV Cameras	1248 lines, 3 channel optical probes, color	
11	TV Monitors	25" shadow mask CRT, 1248 lines, 900 line resolution center, 50 MHZ bandwidth, color	
	Video signal Processing	Mixes earth, cloud/sky, target/payload/RMS, and star field. Contains keying for selective blanking. Camera selection, display selection.	
3	Film strip Projector	Image high altitude, orbital and ascent sky/cloud scenes on a rear projection screen.	
	Aural Simulation	Poly-Voice aircraft sound synthesizer, consists of voltage controlled function generators with amplifiers and speakers, estimated 40 analog inputs.	Limited to environmental sound simulation.

QTY	DEVICE	DESCRIPTION	REMARKS
6	- MOTION BASE SYSTEM - Hydraulic Elements	6 DOF 60" stroke	

A-10

MCDONNELL DOUGLAS ASTRONAUTICS COMPANY - EAST

MDC E1150  
 1 November 1974

QTY	DEVICE	DESCRIPTION	REMARKS
61	Servo Control Components.	DC linear system, each axis position and velocity, and Vertical and lateral acceleration fed back to computer	
1	- DATA CONVERSION - EQUIPMENT Analog to Digital Converters	12-15 bits, 0.25%-0.025% accuracy, $\pm 10v$ , 50 Khz conversion	Families of converters to meet application
200*	Digital to Analog Converters	8-14 bits, 0.25%-0.05% accuracy, $\pm 10v$ , 10ma.	Families of converters to meet application
20	Digital to Resolver/Synchro Converters	10-16 bits, 11.8v, 26v, 90v, 400hz, 1 min. - 8 min. accuracy.	Families of converters to meet applications

MCDONNELL DOUGLAS ASTRONAUTICS COMPANY - EAST

A-11

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

MDC E1150  
1 November 1974

A-12

QTY	DEVICE	DESCRIPTION	REMARKS
100	Digital Multiplexer	2000 Digital input discretes 2500 Digital output discretes	
	Analog Multiplexer	FET, 0-4096 per minicomputer channel; to approximately 1 microsecond switching, 10-200 nanoseconds aperture with S/A amplifiers; relays, 1-10 milliseconds switching.	
	Minicomputer	16 bit word, less than 1.0 microsecond cycle time, minimum of four operating registers,	
	Minicomputer to Analog/Digital Interface	16 bit word parallel transfer, 11 bit address (2).	
	Minicomputer to Host Computer Interface	16 bit word parallel transfer, 32/16 bit word conversion, transfers two real-time clock interrupts.	
	Real Time Clock	2 outputs pulses programmable, 0.01% accuracy, 0.1 millisecond resolution, 0.001 to 1.0 second range.	

APPENDIX B

A GLOSSARY OF TERMS OF INTEREST FOR  
SIMULATOR SELF TEST APPLICATIONS

## APPENDIX B

### A GLOSSARY OF TERMS OF INTEREST FOR SIMULATOR SELF TEST APPLICATIONS

This Appendix contains definitions of a number of non-obvious specialized technical terms of interest to personnel involved in the Simulation Verification Techniques Study. The definitions of words, phrases and acronyms given here have been prepared with the aid of general dictionaries, specialized vocabularies (listed below) and other references.

In many cases, a word is found with different meanings in different sources. In such cases, we have indicated the variety of usage, but designated the usage preferred for this study.

Some notes on format:

- o Each word or phrase being defined appears once in upper case; cross-references to other defined words and phrases are indicated by initial caps.
- o Alternate terminologies are often shown together, as CHECKPOINT, RESET POINT.
- o Phrases are given in natural order, as CATASTROPHIC FAILURE, rather than FAILURE, CATASTROPHIC.
- o A number of abbreviations are used for compactness: usu., usually; syn., synonym; cf., compare; e. g., for example.

#### B.1 SPECIALIZED VOCABULARIES USED IN COMPILING THIS GLOSSARY

C. J. Sippl and C. P. Sippl, "Computer Dictionary and Handbook", Howard W. Sams & Co., 1972.

"Military Standard: Definition of Effectiveness Terms for Reliability, Maintainability, Human Factors, and Safety" (MIL-STD-721B), Dept. of Defense, 10 March 1970.

"Dictionary of Technical Terms for Aerospace Use" (NASA SP-7), National Aeronautics and Space Administration, 1965.

"IEEE Standard Dictionary of Electrical and Electronics Terms" (IEEE Std 100-1972, ANSI C42.100-1972), Wiley-Interscience, 1972.

"Microprogramming Handbook", Microdata Corp., 1971.

"International Dictionary of Applied Mathematics," Van Nostrand, 1960.

## B.2 DEFINITIONS OF WORDS AND PHRASES

**ACCEPTANCE:** a formal customer action which affirms that a procured article conforms to the agreement under which it was procured (specification, purchase order, etc.). (Vendor/contractor obligation often continues beyond acceptance; however, vendor normally becomes entitled to complete or substantial portion of contracted payment upon acceptance.)

**ACCEPTANCE TEST:** a test or series of tests intended to lead to acceptance.

**ALIASING:** error introduced by sampling a continuous signal at too low a rate. E.g., if the Nyquist Frequency is 10 Hz, power present at 11, 29, 31, 49, ... Hz will all contribute to the apparent power at 9 Hz. cf. Power Spectrum, Fourier Transform. (Low-pass filtering before sampling will reduce or eliminate aliasing.)

**APPLICATION SOFTWARE:** software generated by or for a computer user, to accomplish the purpose for which the computer was obtained; e. g., simulation.

**ASSEMBLER:** a Language Processor which translates programs written in assembly language into an executable form (typically machine language for a particular model computer).

**ASSEMBLY LANGUAGE:** a language characterized by the use of mnemonics for operations, operands, and addresses, a one-operation-per-statement format, and access to most machine hardware features.

**AUTOCORRELATION:** see Correlation.

**AVAILABILITY:** the percentage of its scheduled operational time (e. g., first shift) that a system is actually operable. Computed by the formula

$$A = 100 \times \text{MTBF} / (\text{MTBF} + \text{MTTR}).$$

**AVERAGING:** a technique for filtering uncorrelated noise from repetitive signals which are synchronously triggerable. Cancellation of the random noise occurring in different signal periods yields significant improvements in Signal-to-Noise Ratio.

**BACKUP:** intended for use in the event of a failure in primary equipment; said of equipment, procedures, etc. Often entails some degradation in performance or convenience. cf. Standby.

**BENCHMARK:** a program used to evaluate the performance of a computer system.

**BODE DIAGRAM, BODE PLOT:** a plot of the frequency response of a linear system. Gain in Decibels and phase in degrees are plotted against the log of the driving frequency.

**BUILT-IN-TEST:** hardware built in to a system to support Self-Test.

**CALIBRATE:** to adjust an apparatus to its design characteristics.

**CATASTROPHIC FAILURE:** a sudden failure which results in complete loss of function, equipment damage, or personnel injury.

**CERTIFICATION:** formal affirmation that a system is acceptable for its intended use; applies to initial certification of a new system, and to recertification of a system following modification.

**CHECK CASE:** a set of input data and corresponding output data, which provides a standard for evaluation of hardware/software operation.

**CHECKED OUT:** having passed a series of tests, and considered to be fault-free.

**CHECKOUT:** test or monitoring of a system to determine its condition or status; sometimes used as syn. for Verification.

**CHECKPOINT, RESET POINT, RESTART POINT, SAFE STORE POINT:** a point in a computer program at which data is stored to permit restarting its operation.

**CHECKSUM, CHECK SUM:** a number formed by addition operations upon a set of data (often without regard for carry); used for comparison with a reference value, to detect errors in storage, transmission, etc.

**CODE, n:** (1) a system of representation of data and instructions for computer manipulation, transmission, etc: e. g., BCD, EBCDIC, ASCII. (2) the text of a computer program.

**CODE, v:** to write out a computer program in an appropriate language; cf. Program.

**COHERENT RADIATION:** radiation with fixed or slowly-varying phase relationships; e. g., the output of a laser.

**COMBINATIONAL TESTING:** a Test Plan whereby a number of tests are performed before any results are analyzed.

**COMPILER:** a Language Processor which translates high-level compiler-language programs into an executable form, or into Assembly language. Compiler languages are characterized by symbolic representation of all entities and operations, a many-operations-per-statement format, and the Transparency of many machine hardware features.

**COMPONENT:** a hardware unit which, at a given tier of maintenance, is usually replaced rather than repaired. (An irreducible component is often called a "piece part".)

**CONTROL MEMORY, CONTROL STORE:** in a Microprogrammable computer, the memory in which microprograms are stored, for use by the Control Unit in executing programs from main memory.

**CONTROL UNIT:** the part of a computer which controls the sequencing and execution of instructions.

**CONVOLUTION:** for two time functions  $f$  and  $g$ , the convolution is the integral

$$f(t) * g(t) = \int_{-\infty}^{\infty} f(u) g(t-u) du$$

provided the integral exists; extremely important in the analysis of linear systems. (Convolution in the time domain corresponds to multiplication in the frequency domain.)

**CORRECTIVE MAINTENANCE:** maintenance performed in response to the detection or suspicion of a fault.

**CORRELATION:** for two random time series  $f_i$  and  $f_j$ , the Cross-Correlation is the integral

$$\varphi_{ij}(t) = \frac{1}{2T} \int_{-T}^T f_i(u) f_j(t-u) du$$

When  $f_i$  and  $f_j$  are the same function, the integral  $\varphi_{ii}(t)$  is known as the Autocorrelation.

**CRITERION (pl. CRITERIA):** (1) a rule or standard for making evaluations or decisions. (2) In formal Optimization, a function whose value is to be maximized (payoff function) or minimized (cost function); cf. Performance Index.

**CRITICAL DAMPING:** see Damping.

**CROSS-ASSEMBLER:** an Assembler which executes on one model computer, producing code executable on another model computer.

**CROSS-COMPILER:** a compiler which executes on one model computer, producing code executable on some other model computer.

**CROSS-CORRELATION:** see Correlation.

**DAMPING:** dissipation of energy by resistance internal to a system. In a second-order linear system, Critical Damping is the value of the damping ratio at which motion just ceases to be oscillatory. Systems with damping lower and higher than critical are called Underdamped and Overdamped, respectively. Cf. Overshoot.

**DATA:** numerical or alphabetical material serving as a basis of discussion or as the object of computer manipulation. In some cases (e. g., Checksums, Language Processors) Programs are treated as data.

DATA CONVERSION EQUIPMENT (DCE): equipment which changes the format of a signal passing between other pieces of equipment, e. g., A/D, D/A.

DEBUGGING: detection, isolation and correction of faults in a new system, usu. software. Often restricted to superficial errors, such as incorrect signs.

DECIBEL: the usual unit for measurement of gain (amplitude ratio) of a linear system (cf. Bode Diagram), and for Signal-to-Noise Ratio. Gain in dB =  $20 \log_{10}$  (amplitude ratio).

DIAGNOSTIC: (1) a procedure (often including a computer program) used to detect and isolate faults. (2) an error message generated by System Software.

DIRECT MEMORY ACCESS (DMA): Transfer of data between a computer peripheral device and Main Memory without CPU intervention.

DISCRETE EVENT SIMULATION: a form of Simulation in which simulated time progresses in accordance with the occurrence of events or the performance of tasks, rather than in fixed increments.

EMULATION: simulation of the instruction set of one model computer on another model computer. Often restricted to simulation approaches involving special hardware or firmware; cf. Interpreter.

ENERGY SPECTRAL DENSITY: the squared magnitude of the Fourier Transform of a "burst" signal (a non-periodic signal of finite duration); cf. Power Spectral Density.

EXECUTIVE: a program which controls the execution of other programs; sometimes syn. for Operating System.

FAIL-SAFE: having the property of counteracting or limiting the effects of certain anticipated failure modes.

FAILURE: unsatisfactory performance or non-performance of a normal function or functions (may be the manifestation of a fault, or a response to abnormal conditions or loads). Sometimes restricted to complete non-performance of a function, with "malfunction" used for out-of-spec performance.

FAILURE PREDICTION: analysis of current or historical data from a system or component to provide advance warning of hardware failures. Distinguished from Periodic Maintenance, which is based upon failure rates for a population of similar hardware units, rather than actual data from a specific unit.

FAST FOURIER TRANSFORM: a specific algorithm for computation of the discrete Fourier Transform, which reduces computational requirements dramatically; e. g., for 1024 data points, the computation time is reduced by about a factor of 200.

**FAULT:** a defect or deficiency in a hardware/software system, which will lead to failure in some or all circumstances.

**FAULT DETECTION:** a positive determination, from testing or monitoring a system, that a fault exists somewhere in the system.

**FAULT INSERTION:** deliberate faulting of a hardware/software system or a simulation of that system, for the purpose of diagnostic development, training, etc.

**FAULT ISOLATION, FAULT LOCALIZATION:** determining the location of a fault with increasing exactness. This process is normally carried as far as is required for remedial action; e. g., Reconfiguration, LRU replacement.

**FIDELITY:** actual or perceived accuracy of representation of the real world by a simulation.

**FILE MAINTENANCE:** modifying the contents of a file, while preserving its structure; includes inserting, deleting, modifying and rearranging data.

**FIRMWARE:** microprograms in the Control Store of a microprogrammable computer.

**FORMAT:** a specific arrangement or presentation of data, esp. on a computer input or output medium.

**FOURIER TRANSFORM:** an integral transformation which maps a time series into the frequency domain, given by

$$F(i\omega) = \int_{-\infty}^{\infty} e^{-i\omega t} f(t) dt$$

The inverse transformation is

$$f(t) = \int_{-\infty}^{\infty} e^{+i\omega t} F(i\omega) d\omega$$

The discrete form of the Fourier transform, used with sampled data, is usually computed using the Fast Fourier Transform. Cf. Power Spectral Density, Energy Spectral Density.

**FUNCTIONAL SIMULATION:** simulation of the observable effects of some hardware or software, without requiring representation of internal details.

**GATE:** (1) a low-order digital logic element; e. g., an AND gate or OR gate. (2) a hardware or software element which produces a discrete output based upon the magnitude of an input signal or the occurrence of a particular change in the input signal.

**GRAPH:** a mathematical construct consisting of nodes and arcs, usu. representing the interconnections between elements of a system; see Fig. B-1.

**IDENTIFICATION, SYSTEM IDENTIFICATION:** processing system inputs and outputs to determine values for the parameters of the system.

INCIPIENT FAULT DETECTION: see Failure Prediction.

INFORMATION: (1) that which has the capability to reduce uncertainty;  
(2) the reduction in uncertainty achieved by a particular message,  
measurement, etc.; usu. measured in bits.

INFORMATION THEORY: the study of messages, codes, transmission lines, etc.

INTEGRATED SYSTEM(S) TEST: a test conducted with all feasible subsystems  
of a system interconnected and energized.

INTERPRETER: a Language Processor which processes and executes computer-  
language text statement-by-statement; normally produces no corresponding  
text in another language. Often used for Emulation.

LANGUAGE PROCESSOR: a program which operates upon computer-language  
text (programs or individual statements).

LEAKAGE: "smearing" of spectral lines occurring when the discrete Fourier  
transform is evaluated over a time interval which is not an integral multiple  
of the base frequency of the input signal; cf. Windowing.

LIBRARY ROUTINES: multi-user routines which are made a part of an Appli-  
cation Software load; e. g., mathematical and input/output routines.

LOGGING: recording of data, machine indicators, etc., during the opera-  
tion of a system. May be continuous, periodic, on-command, or triggered  
by certain events.

MAIN MEMORY, MAIN STORAGE: the computer memory from which programs are  
executed.

MAINTENANCE: activities performed to keep a system operating satisfac-  
torily. See Corrective Maintenance, File Maintenance, On-Condition  
Maintenance, Periodic Maintenance, Preventive Maintenance, Program  
Maintenance.

MALFUNCTION: see Failure.

METALANGUAGE: a language used to define and discuss languages. A metalan-  
guage processor is a program used to generate language processors; e. g.,  
a meta-assembler generates Assemblers, a meta-compiler (compiler-compiler)  
generates Compilers.

MICROINSTRUCTIONS: instructions at a lower level of detail, and executed  
at higher speeds, than conventional machine instructions; e. g., a  
single "add" instruction consists of a sequence of microinstructions  
such as comparison, shifts, single-bit adds, and carries.

MICROPROGRAMMING: programming in terms of microinstructions, on a computer  
which makes microinstructions accessible.

MINICOMPUTER: loosely, any small general-purpose digital computer with word length of 16 bits or less.

MISSION PHASE: a portion of a space mission which is defined to begin and end with certain specified events; e. g., the Ascent phase runs from ignition to insertion, the On-Orbit phase runs from insertion to deorbit.

MODEL: a mathematical or physical representation of a real-world system; e. g., a math model, a scale model.

MODULAR, MODULARIZED: made up of interconnected Modules. (A modular approach to hardware/software design often entails sacrifices in terms of speed, compactness, and amount of planning required, but returns benefits in ease of documentation, modification, verification and repair).

MODULE: a set of hardware or software elements which are physically contiguous, invoked as a unit, and perform a defined function.

MODULO: number representation in which the magnitude of every number is less than or equal to the magnitude of a fixed number, called the "modulus." Any number greater than the modulus is converted to its remainder upon division by the modulus; e. g., in modulo-8 arithmetic, 10, 34, and 586 are all converted to 2.

MONITOR: to observe and evaluate system inputs and outputs while the system is operating; cf. Test. Also sometimes used to refer to the collection of operating data for later processing, which should properly be called Logging.

MONTE CARLO METHOD: a computational or simulation procedure involving repeated execution of a program, using data sampled randomly from a predefined population.

NYQUIST FREQUENCY, FOLDING FREQUENCY, TURNOVER FREQUENCY: the highest frequency which can be evaluated by a discrete Fourier analysis of sampled data. The Nyquist frequency is  $1/2T$ , where  $T$  is the Sampling interval. Cf. Aliasing, Sampling Theorem.

OBSERVABLE: an accessible variable which conveys information about the dynamics of a system.

OFF-LINE: independent of a specified system; not under control of that system, nor competing with it for resources; separated by time and/or space from the system's operation.

ON-CONDITION MAINTENANCE: see Failure Prediction.

ON-LINE: under control of, contributing to, or competing for resources with a specified system.

OPERATING SYSTEM: System Software which controls the execution of jobs and the allocation of computer resources to competing tasks.

OPTIMIZATION: (1) selection of the "best" system configuration, parameters or procedures. (2) selection of configuration, parameters, or procedures to minimize or maximize some objective Criterion function.

OVERDAMPED: see Damping.

OVERSHOOT: the percentage by which the first peak value exceeds the steady-state step response of an underdamped system; see Fig. B-2.

PARAMETER: (1) a quantity which is either constant, or is given a constant value for a specific period or process. (2) any quantity which is not an independent variable in the dynamics of a system. (3) a quantity in the calling sequence of a subroutine or entered on a control card.

PARAMETER ESTIMATION: see Identification.

PARITY: (1) a Modulo-2 checksum; it has the value 0 if the number of ONE bits in the subject data is even, and the value 1 if the number of ONE bits is odd. (2) an error-detecting coding scheme in which every legal code combination yields the same modulo-2 checksum. Both even-parity and odd-parity codes are used.

PERFORMANCE INDEX: a function of a system's inputs and outputs over a given time interval, used as a measure of the quality of system behavior; e. g., the integral-square-error (ISE) index is given by:

$$I = \int_0^T [r(t) - c(t)]^2 dt$$

where  $r(t)$  is the reference input and  $c(t)$  is the controlled output. Cf. Criterion.

PERFORMANCE VERIFICATION: see Validation.

PERIODIC MAINTENANCE, HARD-TIME MAINTENANCE: Maintenance performed at defined time intervals, based upon assumed failure rates and economic factors; cf. Failure Prediction.

PERIPHERAL PROCESSOR: a small computer which performs input/output and other service functions for the large computer to which it is attached.

POINTER: an index or address providing access to instructions or data.

POWER SPECTRAL DENSITY: the power per unit bandwidth of an input time function, as a function of frequency. Computed as the squared magnitude of the Fourier Transform; alternatively, the Fourier Transform of the Autocorrelation function. Cf. Energy Spectral Density.

PREVENTIVE MAINTENANCE: Maintenance intended to prevent in-service failures.

PROGRAM, n: a set of instructions for a computer to perform some specified task.

PROGRAM, v: to produce a computer program; includes both design and coding.

PROGRAM MAINTENANCE: modifying an existing program or system of programs, while preserving its basic structure; includes error correction, capability enhancement, and updating.

PSEUDO-RANDOM: generated by a deterministic process, but giving the appearance of Randomness and satisfying the standard tests for randomness. (The output of a pseudorandom sequence generator can be tailored to the statistical and/or spectral properties desired for a particular application.)

RANDOM: occurring by chance rather than by design; unpredictable as to the occurrence of a particular event or value, although having certain definable statistical properties when considered as an ensemble.

REAL-TIME COMPUTATION: computation which contributes no time lag to the dynamics of a real-world process.

REAL-TIME SIMULATION: simulation which proceeds at the same rate as the dynamics of the real-world process being simulated. (Man-in-loop simulation is almost always real-time.)

RECONFIGURE: to change the configuration of a system by activating or deactivating On-line or Backup subsystems; often entails some performance degradation. Cf. Repair.

REDUNDANCY: (1) in Codes, the use of characters which do not contribute to information content, provided for error detection and possibly correction; cf. Parity. (2) provision of more than one means to perform a function; e. g., extra hardware units. Cf. Backup.

RELIABILITY: (1) the quality of being reliable. (2) the probability of acceptable operation, for a specified length of time or for the accomplishment of a specified task.

REPAIR: to restore to service by replacement of parts or by restoration of parts to operational condition.

RESPONSE: a quantitative description of the output of a device or system as a function of the input (Stimulus), under conditions that must be explicitly stated.

RISE TIME: the time required, from the initiation of a Stimulus, for the output of a system to rise to a specified fraction (typically 90%) of its steady-state value; see Fig. B-2.

SAMPLING: the process of obtaining a sequence of discrete values of a continuous signal, generally at equal time increments; cf. Nyquist Frequency.

1 November 1974

**SAMPLING THEOREM:** a theorem which proves that a continuous signal which has been sampled can be perfectly reconstructed, if the Sampling frequency is greater than or equal to twice the highest frequency present in the original signal. Cf. Nyquist Frequency.

**SELF-TEST:** capability of a system (by means of hardware and/or software) to perform limited fault detection and isolation without external assistance; cf. Built-in-Test.

**SEQUENTIAL TESTING:** a Test Plan whereby the results of each test are analyzed to determine what should be done next.

**SETTLING TIME:** the time required, from the initiation of a Stimulus, for the output of a system to enter and remain within a specified narrow band centered upon its steady-state value; see Fig. B-2.

**SIGNAL-TO-NOISE RATIO:** ratio of the value of a signal to the value of the noise; usu. measured in Decibels. (Note that this ratio is commonly in terms of peak values for impulsive signals, and in terms of RMS values or power for continuous signals.)

**SIMULATION:** (1) the representation of real-world phenomena by some combination of hardware and software. (2) an assemblage of hardware and/or software which can perform simulation; includes pure software simulation.

**SIMULATOR:** a system providing a physical representation of real-world phenomena. (Our definition encompasses pure-hardware and hardware/software systems, but excludes pure software simulation (which can only generate non-physical representations, such as printouts and plots). However, this usage is not universal.)

**SOFTWARE:** (1) computer programs. (2) any information contributing to computer operation, including programs, data, documentation and procedures.

**SOFTWARE PROCESSORS:** routines which prepare applications software for execution; e. g., Language Processors, loaders, link editors. Cf. System Software.

**SPECIFICATION:** a document describing an item to be procured or developed. (A specification is often the basis of a contract procurement, and therefore written in a manner to facilitate legal enforcement.)

**STANDBY:** ready for rapid activation if required. Cf. Backup: Backup is a functional assignment, while standby is a mode or status.

**STIMULUS:** any change in input that effects the output of a system; e. g., a disturbance or change in reference input.

**SUBSYSTEM:** a System which is part of a larger system.

**SUPPORT SOFTWARE:** software which performs secondary tasks assisting a user in accomplishing his primary task; e. g., a simulator organization might have support software for simulation-software checkout, generation of initial conditions, etc.

**SYSTEM:** (1) an assemblage of hardware and/or software which interacts to perform some function or functions. (2) when unmodified, the word "system" will normally refer to a simulator.

**SYSTEM CONFIDENCE TEST:** a test which, in a reasonably short time, is intended to ensure the user that the complete System is operating properly.

**SYSTEM SOFTWARE, COMPUTER SYSTEM SOFTWARE:** software required for effective utilization of a computer for any likely application. Normally supplied by the computer manufacturer, but may be modified by the user for special applications. See Software Processors, Operating System, Library Routines, and Utilities.

**TEST, n:** a procedure to evaluate a system under controlled conditions.

**TEST, v:** to supply inputs to a system, and then observe and evaluate its outputs. Cf. Monitor.

**TEST CASE:** see Check Case.

**TEST PLAN:** an advance description of the number and order of tests to be conducted to accomplish a certain objective (system acceptance, diagnosis, etc.). The two main categories are Combinational and Sequential Testing. (Combinational strategies tend to be inefficient, particularly when system failure rates are low, because many of the tests performed are unnecessary. Mixed combinational/sequential strategies are often convenient.) Cf. Tree.

**TEST SIGNAL:** a Stimulus which is tailored to elicit a known Response from a system, providing a concrete indication of the system condition.

**TIMELINE:** a plan (usu. for a space mission) describing a planned sequence of events, tasks, etc.

**TIME-SHARING:** sharing of computer resources (CPU, storage, I/O channels, etc.) among various tasks, by giving each individual task control of a resource for a short time interval -- from a few milliseconds to a few seconds.

**TRACE:** an interpretive diagnostic technique which provides output from each computer operation (or each operation of a particular type, such as transfers). Often used for software Debugging.

REPRODUCIBILITY OF THE  
ORIGINAL PAGE IS POOR

**TRANSLATOR:** a Language Processor which converts text from one computer language to another; includes Compilers and Assemblers. Sometimes used to implement Emulation by converting machine language from one model computer to another.

**TRANSPARENT:** performing functions in such a manner that the user is unaware of the underlying implementation; e. g., most computer hardware features (registers, channels, etc.) are transparent to a high-level language programmer.

**TRAP:** a means of interrupting execution when a certain condition occurs (e. g., overflow, illegal transfer).

**TREE:** a form of directed Graph in which arcs "fan out" from each node, so that two paths which originate at different nodes are disjoint; see Fig. B-3. (Applications of tree structures include decoders, data management systems and fault-isolation test sequences.)

**TROUBLE-SHOOTING:** manual fault isolation, often without a preconceived plan.

**UNDERDAMPED:** see Damping.

**UNIT:** a set of hardware or software elements which performs a defined function; cf. Component, Module.

**UTILITY ROUTINES:** multi-user stand-alone routines; e. g., file maintenance, sort/merge. Cf. Library Routines, Support Software.

**VALIDATION:** the process of demonstrating that a simulation provides a satisfactory representation of the real world.

**VERIFICATION:** the process of demonstrating that a system is capable of the satisfactory performance of its intended functions; cf. Checkout. Verification is sometimes restricted (esp. in software usage) to refer to testing a system against its specifications and other documentation, while Validation refers to testing agreement with the real world.

**WINDOWING:** time-domain preprocessing of input data before taking its Fourier Transform, for the purpose of reducing Leakage. A variety of window shapes - triangular, cosine, Hanning, etc. - are in general use.

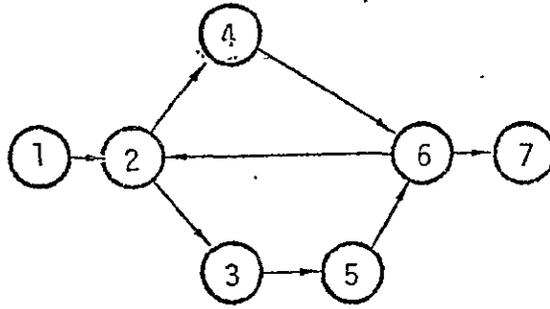


FIGURE B-1 A SIMPLE DIRECTED GRAPH

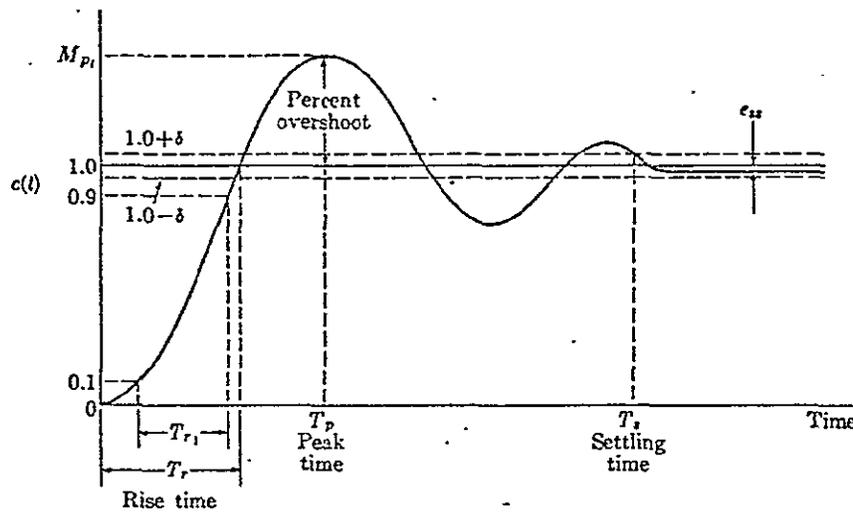


FIGURE B-2 STEP RESPONSE OF AN UNDERDAMPED SYSTEM

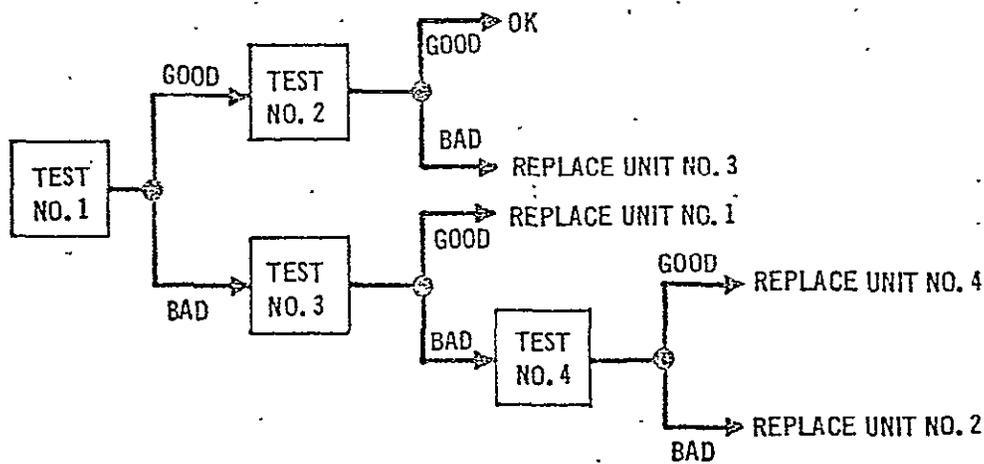


FIGURE B-3 A SIMPLE FAULT-ISOLATION TREE

### B.3 DEFINITIONS OF ABBREVIATIONS AND ACRONYMS

A/D	Analog to Digital (conversion)
ACE	Automatic Checkout Equipment, Acceptance Checkout Equipment
ACIDS	Aircraft Integrated Data System (Army)
AGARD	Advisory Group for Aviation Research and Development (NATO)
AGE	Aerospace Ground Equipment; cf. GSE
AIDS	Airborne Integrated Data System (commercial airlines)
ALGOL	Algorithmic Language: a FORTRAN competitor, especially in Europe
ARINC	Aeronautical Radio, Inc. (commercial aviation)
ASCII	American Standard Code for Information Interchange
ATE	Automatic Test Equipment
ATLAS	Automatic Test Language for Avionic Systems, Abbreviated Test Language...
ATS	Automatic Test System
BASIC	Beginner's All-purpose Symbolic Instruction Code: a programming language widely used in time-sharing systems, somewhat simpler than FORTRAN. Used for some ATE.
BCD	Binary - Coded Decimal
BETA	Basic English for Test Applications; an ATE programming language
BIT	Built-In Test
BITA	Built-In Test Access: provision of data-paths for effective interface with external test equipment
BITE	Built-In Test Equipment
C&D	Control & Display
CIG	Computer Image Generation (also CGI)
COBOL	Common Business-Oriented Language

CPU	Central Processing Unit
CRT	Cathode-Ray Tube
D/A	Digital to Analog (conversion)
dB	Decibel
DCE	Data Conversion Equipment
DMA	Direct Memory Access
D/R	Digital to Resolver (conversion)
DUT	Device Under Test; cf. UUT
EASY	Engine Analyzer System (Air Force)
FORTTRAN	Formula Translation: a popular scientific programming language
EBCDIC	Extended Binary-Coded Decimal Interchange Code (IBM)
FDAI	Flight Director/Attitude Indicator
FFT	Fast Fourier Transform
GPATS	General-Purpose ATS
GSE	Ground Support Equipment
HSI	Horizontal Situation Indicator
I/O	Input/Output
IOS	Instructor/Operator Station
ISE	Integral of Squared Error -
IST	Integrated System Test
LRU	Line-Replaceable Unit (e. g., on the "flight line" or the highest tier of maintenance) or Least Replaceable Unit
MADAR	Malfunction Detection, Analysis and Recording (Air Force)
MATE	Multi-band ATE (Navy)
MCC	Mission Control Center
MOPAS	Matrix-Oriented Production Assembly System

MTBF Mean Time Between Failures or Mean Time Before Failure; used in reliability studies

MTTR Mean Time To Restore or Mean Time To Repair; used in maintainability studies. The usage "Restore" is preferable.

PET Program Evaluator and Tester (formerly PTT)

PL/I Programming Language/I: an IBM FORTRAN/COBOL replacement (also PL/1)

PPU Peripheral Processing Unit

PRBS Pseudo-Random Binary Sequence

PTT Program Testing Translator (MDAC)

RMS Root-Mean-Square

ROM Read-Only Memory

SCT System Confidence Test

TAEM Terminal-Area Energy Management (Shuttle Mission Phase)

TBO Time Between Overhauls

UUT Unit Under Test; cf. DUT

VAST Versatile Avionics Shop Test (Navy)

APPENDIX C  
SIMULATOR VERIFICATION TECHNIQUES  
BIBLIOGRAPHY

## APPENDIX C

### SIMULATOR VERIFICATION TECHNIQUES BIBLIOGRAPHY

This Appendix provides a list of and subject index to the several hundred documents reviewed during the Hardware Verification Techniques Task Survey. The subject index is Appendix C.1; the document list is Appendix C.2 .

#### C.1 SUBJECT INDEX TO BIBLIOGRAPHICAL MATERIAL

This subject index is designed to support study technical activities by enabling ready access to documents needed by study personnel. Cross-indexing information is provided for more effective searching. The indexing approach used here was largely derived from experience with DoD and NASA information retrieval systems.

Headings - All terms in this index are given in natural order rather than inverted order; e. g., Integrated Circuits rather than Circuits, Integrated. In a few cases, parenthetical qualifiers are added to clarify the meaning of a term; e. g., Debugging (Software).

Cross-References - Cross-reference information is now provided within the index itself. The objectives of this cross-referencing are as follows:

- Maintain a degree of reference vocabulary standardization, rather than allowing uncontrolled proliferation of terms with similar meanings (e. g., Test, Checkout, Verification).
- Ease the task of finding relevant documents by balancing the need for "clustering" of related documents against the need for "resolution" of differences between related concepts.
- Illustrate relationships among various indexing terms, as a further aid in finding relevant documents.

Five types of cross-indexing "pointers" are employed in this index: BT, NT, RT, Avoid, and Use. "BT" (broader term) and "NT" (narrower term) indicate hierarchical scope relationships among index terms. "RT" (related term) is not meant

to imply any indication of the type or degree of relationship among terms. "Avoid" is added after headings whose use is undesirable due to poor resolution, ambiguity of definition, or some other reason. Alternative headings will be suggested. A document may still be indexed under this heading in some circumstances; e. g., when the term appears in the title of the document. Finally, headings marked with an asterick are not used as indexing terms. Alternative terms will be suggested, following the word "use". Examples of the various cross-indexing references are shown below.

<u>Computers</u>	BT	Digital Systems
	NT	IBM Computers, Minicomputers
<u>Acceptance Tests</u>	RT	Verification, Validation
<u>Checkout</u>	Avoid	
	RT	Diagnostics, Verification
<u>Backup*</u>	Use	Redundancy

Index Entries - The entries under each subject heading identify documents which are known to convey information about that subject. Each entry consists of a "retrieval tag" followed by a colon, then a short explanatory comment. The retrieval tag normally consists of the name of the author and date of publication of the document (see App. C.2).

Acceptance Tests

RT Verification, Validation

B. E. Cowart, 71: SYMBOL computer assemblies  
J. H. Boeckel, 69: Explorer spacecraft

Adaptive Control

V. S. Levadi, 66: learning algorithm for diagnosis  
V. S. Levadi, 65: pattern recognition applied to fault detection

AIDS\*

use Onboard Equipment, Data Acquisition

Aircraft

C. R. Hanke, 70: B747 modeling  
RAeS, 71: flight simulators  
CDU-MO, 71: FSI DC-10 maintenance  
CDU-MO, 71: JAL 747 maintenance  
CDU-MO, 70: 737 simulator maintenance  
C. R. Hanke, 70: B-747 Reference Data  
J. E. Love, 74: YF-12 onboard monitor  
M. L. Yaffee, 73: airline ATE  
R. T. Marshall, 73: F-104 dynamics, validation

Airlines

M. L. Yaffe, 73: airline ATE  
Interavia, 73/6: airline simulator market  
G. J. deWit, 68: airline simulator economics  
B. M. Elson, 72: UAL DC-10 maintenance  
CDU-MO: airline simulator manuals

Alignment

RT Calibration

J. P. Vincent, 72: solar simulation

ATLAS\*

use Test Languages

Automation

F. J. Hackl, 65: computer maintenance  
J. C. Richards, 68: hybrid computer checkout  
T. Kuroda, 71: MATE  
J. P. Verma, 72: PC board tester  
NYU, 1966+: Automation in Test Equipment  
V. S. Levadi, 66: automated learning  
J. E. Fay, 69: automated interface design  
E. F. Meyers, 69: automated checkout for IBM 4-pi  
M. Eleccion, 74: automatic testing overview

J. P. Vincent, 72: array alignment  
ASSC, 72: symposium  
ASSC, 73: symposium  
E. R. Aiken, 73: calibration laboratory  
L. G. Stucki, 72: "PTT" program experience  
L. G. Stucki, 73: Automatic software generation

Availability\* use Economics  
Avionics\* use Electronic Equipment, Onboard Equipment  
Backup\* use Redundancy  
BASIC\* use Test Languages

#### Bibliographies

A. B. Salisbury, 67: diagnostic programming  
IEEE, 71: vol C-20#11, IEEE transactions  
D. M. Goodman, 67 vol. 4: B.I.T. and Continuous Monitoring  
E. K. Gannett, 71: dictionary  
ACR, 73: quarterly bibliography  
H. S. Dordick, 66: advanced sensing techniques  
P. B. Schoonmaker, 7/73: Verification IRAD (TR-2a)

#### Built-in Test

H. L. Kay, 66: aircraft power systems  
U. R. Furst, 66: avionics  
R. Lund, 72: cars  
R. F. Wokasch, 69: on-board  
A. W. Haydon Co., 72: BITE indicators  
L. A. Whalen, 72: Simulator B.I.T.  
M. A. Mehta, 72: LSI chips  
C. V. Ramamoorthy, 71: blocking gates

#### Calibration

E. R. Aiken, 73: automation

#### Cathode Ray Tubes

RT Interactive Terminals, Displays & Controls,  
Television

D. M. Goodman, 67 vol. 3: color  
G. A. McKenzie, 70: TV waveforms  
W. C. Bauer, 69: oscilloscope  
RAeS, 71: in flight simulators  
J. W. Kenney, 72: test methods

Check Cases

RT Diagnostics, Test Signals

M. E. Fowler, 68: using SACCESS  
E. G. Dupnick, 73: minimize number by integer programming  
C. R. Hanke, 70: B-747 Reference Data

Check Sums

BT Test Methods

P. Kreager, 72: CRC  
D. C. Bossen, 72: parity  
J. F. McKeivitt, 72: parity  
G. Audin, 73: parity and CRC

Checkout

Avoid  
RT Diagnostics, Verification

K. C. Smith, 68: dual-processor system  
J. C. Richards, 68: hybrid computer system  
H. S. Dordick, 66: advanced sensing techniques  
E. F. Smith, 66: flexible automatic checkout system  
E. F. Meyers, 69: IBM 4-pi  
M&S, 71: SLS options

Circuit Boards\*

use Electronic Equipment

Codes

S. A. Szygenda, 71: memory error recovery  
G. Audin, 73: parity and CRC

Combinational Testing\*

use Test Plans

Computer Peripheral Equipment

Computest, 72: disk testing  
J. F. McKeivitt, 72: parity for memory testing  
D. Lignos, 72: disks  
G. W. Nelson, 68: on-line peripheral system test  
P. C. Jackson, 73: integrating ATE peripherals

Computer System Software

RT Support Software, Language Processors

IBM, S360-20: O/S debugging  
IBM, S360-25: FORTRAN debug  
UCC, 73: UCC-15 restart management  
H. H. Ipolyi, 73: System software requirements  
W. T. Musial, 73-2: System software survey

Computers

BT Digital Systems  
NT IBM Computers, Minicomputers, Memories

N. Bartow, 70: IBM-360 microdiagnostics  
B. Soucek, 72: minicomputers  
A. T. Chen, 73: event trigger  
D. L. Droulette, 71: IBM-360/370 recovery software  
ACR, 73: quarterly bibliography  
P. Kreager, 72: checksum  
J. C. Richards, 68: hybrid  
K. C. Smith, 68: dual-processor checkout system  
B. E. Cowart, 71: SYMBOL computer design  
M. A. Calhoun, 72: SYMBOL computer debugging  
M. J. Flynn, 72: Perspective on Computer Reliability  
A. Avizienis, 72: fault-tolerant, modular computer  
R. S. Chadwick, 73: recovery, redundancy, reconfiguration  
S. A. Szygenda: self-test diagnostics/self-repair  
W.T. Musial, 73-1: computer complex hardware  
W.T. Musial, 73-2: background survey  
Sci.Amer., 73-4: timing uncertainty, crashes  
L. E. Hart, 71: "system stethoscope"

Conferences

AA, 73: visual and motion simulation  
SME, 71: engineering software verification  
IEEE, 60+: switching circuit theory and logical design  
IEEE, 66: aerospace systems conference  
ASSC: automatic support systems (IEEE)  
IEEE, 71: fault-tolerant computing  
IERE, 70: joint conference on automatic test systems  
RAeS, 71: flight training simulators  
WESCON: electronics, software, etc.

Configuration Management

BT Management

R. L. Benbow, 73: Simulator management  
NASA, 70: hardware-software end item specifications, interfaces, change procedures

Costs\*

use Economics

Criteria

RT Optimization

C. V. Ramamoorthy, 72  
R. A. Johnson, 59: examples  
J. Wolkovitch, 62: linear dynamical systems

CRT\*

use Cathode Ray Tubes

Data Acquisition

RT Monitoring, Interfaces

B. Soucek, 72: minis for data acquisition  
H. R. Hegner, 66: sensors  
E. B. Nutt, 65: maintenance recording  
N. A. Landis, 71: software systems for multiple minicomputers  
D. M. Goodman, 67 vol. 4: B.I.T. and Continuous Monitoring  
St. Louis University, 67: "ACIDS"  
H. S. Dordick, 66: advanced sensing techniques  
P. C. Jackson, 73: integrating ATE peripherals  
Hewlett Packard, 73: 3050A  
E. Forster, 73: wave form testing  
D. J. Blecki, 73: digital transient recorder

Data Bases\*

use Data Management

Data Conversion Equipment\*

use Interfaces

Data Links

Data Industries, Inc.: transmission test set  
Philco Ford, 73: line test sets

Data Management

R. L. Benbow, 73: simulator management  
E. K. Gannett, 71: dictionary  
S. L. Prendergast, 72: commercial DBMS's  
D. Lefkovitz, 69: on-line file structures  
UCC, 73: UCC-15 restart management  
MCAIR; 10/16/72: F-15 Simulator SDRL  
Applied Data Research, 73: "Librarian"

Data Structures

NT Graphs, Trees

M. C. Colwell, 71: TREES software package  
J. A. O'Brien, 70: for automatic test systems  
C. V. Ramamoorthy, 67: graphs for diagnosis  
Independence, 73: decision table processor  
Trilog, 73: decision table compiler  
A. T. Berztiss, 71: graphs, decision tables, etc.  
K. W. Lord, 73: decision tables  
D. Lefkovitz, 69: on-line file structures

Debugging (software)

RT Programming  
BT Software Verification

IBM, S360-20: operating system guide  
IBM, S360-25: FORTRAN  
M. A. Calhoun, 72: SYMBOL hardware debugging  
IBM, 72: interactive - FORTRAN, COBOL  
S.A. Szygenda, 70H: software diagnosis  
W. Kocher, 69: Survey  
S&B, 72: SVCINTER  
B. W. Boehm, 70: "SDLab"  
ASME, 71: verification, etc.  
Solutions, Inc., 72: CHKSTOP  
R. Grishman, 70: AIDS on 6600  
Signetics, 72: debug microprograms  
Singer (Link), 73: monitoring; manual procedures; simulators  
E. G. Dupnick, 73: minimize check cases  
CIS, 73: breakpoint module  
Standard Data Corp., 73: COBOL SYMBUG  
Z. Jelinski, 73: spotting known errors  
Standard Data, 74: SYMBUG-F for FORTRAN  
Sider and Associates, 73: TXTM system  
Information Associates, Inc., 73: HEXDSPLY  
E. Fong, 73: compiler diagnostic capabilities  
A.T. Chen, 73: event trigger

Decision Tables\*

use Data Structures

Delays\*

use Timing

Design

Avoid

use the thing designed; e.g., Computers,  
Software, Test Equipment; and/or  
use the design attributes; e.g., Modular-  
ization, -ilities

Z. Kohavi, 67: diagnosable sequential machines  
C. V. Ramamoorthy, 71: blocking gates  
M. A. Mehta, 72: BIT on LSI chips  
U. R. Furst, 66: Avionics BIT  
J. E. Fay, 69: interface  
R. A. Marlett, 66: self-diagnosable  
J. L. Ogdin, 72: reliable software  
M. A. Breuer, 68: computer aided design  
D. Mactaggart, 70: test equipment (MELVIN)

Diagnostics

RT Check Cases, Checkout, Fault Detection,  
Fault Isolation, Self-Test, Test Methods

H. Jacobowitz, 67: logic-oriented program  
S. Seshu, 62: asynchronous switching systems  
S. A. Szygenda, 72D: test generation  
C. V. Ramamoorthy, 72: validating diagnostics  
C. V. Ramamoorthy, 67: use of graph theory  
M. A. Mehta, 72: Resolution for LSI  
Z. Kohavi, 67: diagnosable sequential machines  
R. E. Forbes, 65: self-diagnosable computer  
C. V. Ramamoorthy, 71: fault-tolerant computing  
W. E. Kehret, 72: LSI memory test patterns  
H. J. Shechtel, 67: missile testing  
R. Lund, 72: cars  
P. A. Hogan, 69: generating test programs  
D. C. Bossen, 72: parity  
D. Lignos, 72: discs  
S. Seshu, 65: self-diagnosis  
R. A. Marlett, 66: self diagnosable computers  
S.A. Szygenda, 69A: multidimensional paths  
R. A. Johnson, 59: sequential  
N. Bartow, 70:  $\mu$ -diagnosis on 360-85  
A. B. Salisbury, 67: Bibliography  
P. M. Casey, 70: detection/isolation programs  
F. J. Hackl, 65: IBM-360 self-test  
S. A. Szygenda, 69E: diagnostic generation  
D. E. Alderman, 71: CMPS "PSALT"  
S. A. Szygenda, 71FT: test generation problems  
S. A. Szygenda, 72SW: diagnostic tests

Digital Systems

NT Computers

M. A. Breuer, 71: sequential circuits  
R. F. Davis, 72: sequential and combinational circuits  
E. B. Eichelberger, 64: sequential and combinational circuits  
F. C. Hennie, 64: sequential  
IEEE, 60+: switching-circuit theory symposia  
Z. Kohavi, 67: sequential  
G. R. Putzulo, 71: asynchronous  
J. P. Roth, 67: algorithms to develop diagnostics  
S. Seshu, 62: sequential and asynchronous circuits  
T. F. Miotke, 72: boolean test patterns  
Hewlett Packard, 73: Logic Analyzer  
S. A. Szygenda, 71W: digital logic simulation  
C. W. Hemming, 72A: modular requirements for digital logic simulator  
S. A. Szygenda, 70S: realtime operation, on-line equipment  
H. D. Caplener, 73: modeling computer hardware

isk Memories\*

use Computer Peripheral Equipment

Displays & Controls

RT Electro-Optical Systems

ASC Berens, 70: TV, lasers  
Hewlett Packard, 73: Logic Analyzer  
J. Martin, 73: man-computer dialogues  
C. T. Morgan, 63: human engineering guide  
F. A. Hock, 73: visuals

Documentation

NT Flowcharts, Manuals

J. Toellner & Associates, 72: FORDOC for FORTRAN  
D. D. McCracken, 72: readable FORTRAN  
DataPro, 72: flowcharts  
F. McMains, 67: EDIT  
W. Kocher, 69: TIDY, EDIT, Autoflow, etc.  
NASA, 71: guidelines  
G. M. Weinberg, 71: requirements, use, structure

Dynamics

RT Simulation, Transient Response,  
Frequency Response

M. E. Fowler, 68: difference equations  
R. T. Marshall, 73: F-104 simulation  
R. F. Garzia, 70: fault detection and filtering  
C. R. Hanke, 70: B-747 Reference Data  
R. S. Shirley, 73: "SAFE" (FFT) program  
K. J. Szalai, 71: airborne simulator

Economics

G. J. deWit, 68: simulator utilization  
G. P. Chubb, 69: cost-optimized trees  
R. F. Wokasch, 69: evaluation  
RAeS, 71: simulator training  
R. Braby, 69: cost vs. utilization  
J. H. Boeckel, 69: Explorer spacecraft tests  
Interavia, 6/73: airline simulators  
S. L. Prendergast, 72: commercial DBMS's

Electrical Power

RAeS, 71: simulator power supplies  
W. F. Huseonica, 65: current measurements

Electro-Optical Systems

RT Displays & Controls

J. P. Vincent, 72: alignment of solar simulation  
ASC Berens, 70: mainly TV, some lasers  
E. A. Palmer, 73: night visual for landing  
F. A. Hock, 73: visuals  
AIAA, 73: visual and motion simulation  
L. R. Young, 73: visual sensations

Electronic Equipment

NT Digital Systems

B. M. Elson, 72: avionics  
H. J. Sheckel, 67: avionics  
J. P. Verma, 72: circuit boards  
ElecProd, 72: IC testing  
Teradyne, 73: NI51 backplane tester

Failure Detection\*

use Fault Detection

Failure Modes

RT Reliability

S. A. Szygenda, 71: memories

Fast Fourier Transforms

W. T. Cochran et al, 67: basics  
J. W. Cooley et al, 69: algorithm def.  
E. O. Brigham, 74: applications  
R. W. Ramirez, 74: error source  
G. D. Bergland, 69: guided tour of ...

Fault Detection

RT Diagnostics

M. A. Breuer, 68: problems for modern hardware  
F. C. Hennie, 64: sequential circuits  
E. B. Eichelberger, 64: "hazard" detection  
M. A. Breuer, 71: test-generation methods  
J. P. Roth, 67: test-generating algorithms  
V. S. Levadi, 66: automated learning  
H. R. Hegner, 66: passive/indirect sensing  
J. F. McKeivitt, 72: parity  
P. M. Casey, 70: diagnostic program  
V. S. Levadi, 65: Pattern recognition  
R. F. Garzia, 70: dynamic systems  
H. A. Cole, 73: failure detection and damping measurement  
D. D. Allen, 73: test digital IC boards

Fault Insertion

E. W. Thompson, 73: digital simulation  
S. A. Szygenda, 72F: digital logic simulation  
Z. Jelinski, 73: spotting known errors

Fault Isolation

RT Diagnostics, Trees

R. F. Garzia, 71: survey of methods  
J. P. Roth, 67: logic circuit  
C. V. Ramamoorthy, 67: graphs, trees  
M. A. Mehta, 72: LSI resolution  
V. S. Levadi, 66: automated learning  
Y. V. Lyubotov, 65: time-optimized sequence  
R. A. Johnson, 59: sequential  
P. M. Casey, 70: diagnostic programming  
G. P. Chubb, 69: trouble-shooting  
M. C. Colwell, 71: TREES, software package  
TJB Hannon, 68: cost-optimized trees  
T. K. Elliott, 67: manual trouble-shooting on MTS-2  
J. P. Rogers, 65: trouble-shooting manual  
R. F. Garzia, 72: Bode plots  
E. G. Cromer, 73: vendor survey  
W. F. Huseonica, 65: current measurement  
W. W. Anderson, 70: "EMMA"

Fault-Tolerant Systems

RT Self-Repair, Reliability

A. Avizienis, 72: computer reliability  
C. V. Ramamoorthy, 71: summary and overview  
IEEE, 71: symposium  
F. P. Mathur, 70: hybrid-redundant  
R.B. Conn, 72: fault-tolerant, modular computer

Filtering

RT Identification, Frequency Response

R. F. Garzia, 70: dynamic fault detection  
H. A. Cole, 73: fault detection  
H. Sriyananda, 73: selecting test frequencies  
H. J. Rome, 73: inertial platforms  
R. L. Hampton, 65: pseudo-random noise

Flight Hardware\*

use Onboard Equipment

Flight Software\*

use Onboard Equipment

Flowcharts

BT Documentation, Graphs

W. Kocher, 69: software aids  
Data Pro, 72: proprietary packages  
ANSI, 70: flowchart standards

Frequency Response

RT Test Signals, Dynamics, Fast Fourier Transform

A. M. Fuchs, 66: automated frequency response  
J. D. Lamb, 70: PRBS input  
J. E. Valstar, 66: sine-wave stimuli  
G. A. McKenzie, 70: TV waveforms  
H. A. Cole, 73: failure detection  
R. F. Garzia, 70: sampled response waveform  
R. F. Garzia, 72: Bode plots  
R. S. Shirley, 73: "SAFE" (FFT) program  
H. Sriyananda, 73: selecting test frequencies  
D. Luttrupp, 73: gain-phase meter  
R. C. Darf, 74: modern control theory

Graphs

BT Data Structures  
NT Flowcharts, Trees

C. V. Ramamoorthy, 67: use for diagnosis  
J. P. Roth, 67: use in logic diagnostics  
J. A. O'Brien, 70: for Auto-test systems  
J. C. Fakan, 69: review basics  
Z. Manna, 70: termination  
T. W. Pratt, 71: language extension for graphs  
K. Paton, 71: blocks and cutnodes  
E. G. Dupnick, 73: use for software check cases

Hardware

Avoid  
NT Computers, Test Equipment, Interfaces, etc.

L. E. Hart, 71: "system stethoscope"  
W. T. Musial, 73-1: hardware requirements  
A. T. Chen, 73: event trigger  
B.E. Cowart, 71: SYMBOL hard-wired language processor  
M. A. Calhoun, 72: SYMBOL hardware debug

Hazards

RT Security

E. B. Eichelberger, 64: in switching circuits  
B. Allen, 68: computer safeguards

Human Factors

RT Manual Procedures, Displays and  
Controls, Training

G. M. Weinberg, 72: improved programming performance  
G. M. Weinberg, 71: computer programming psychology  
J. Martin, 73: man-computer dialogues  
R. C. Wingrove, 69: pilot describing functions  
E. A. Palmer, 73: night visual for landing  
NTDC, 69: symposium  
A. Merten, 72: language impact  
Z. Jelinski, 73: spotting known errors  
C. T. Morgan, 63: human engineering guide

Hybrid Computers\*

use Computers

Hydraulic Equipment

RT Motion Systems

IBM Computers

BT Computers

D. L. Droulette, 71: IBM 360 and 370 recovery, software  
N. Bartow, 70: micro-diagnostics for IBM 360  
F. J. Hockl, 65: IBM 360 self-test  
E. F. Meyers, 69: 4-pi automated checkout

Identification

RT Filtering

J. E. Valstar, 66: indirect observation  
R. C. Wingrove, 69: pilot describing functions  
R. C. Wingrove, 71: pilot describing functions  
D. E. Stepner, 73: aerodynamic parameters

Incipient Failures

RT Monitoring

R. F. Barry, 70: failure prediction  
H. A. Cole, 73: damping measurement

Information RT Data Management, Codes

R. A. Johnson, 59: entropy optimization

Inputs\* use Test Signals

Integrated Circuits BT Electronic Equipment

ElecProd, 72: IC testing

M. A. Mehta, 72: LSI diagnostic resolution

M. A. Breuer, 68: fault detection problems

CompDes, 6/73: automated fault detection, isolation

Computest, 73: semiconductor testers

E. G. Cromer, 73: vendor survey

Fairchild, 73: FAIRTEST

D. P. Allen, 73: test digital IC boards

Integrated Systems

M. Eleccion, 74: systems

R. F. Miles, 73: systems concepts

W. F. Huseonica, 65: current measurements

Interactive Terminals RT Cathode Ray Tubes, Manual Procedures

IBM, 1972: FORTRAN/COBOL debug

W. C. Bauer, 69: oscilloscope

Standard Data, 74: FORTRAN SYMBUG-F

Interfaces

O. R. Batchelder: test equipment interfacing

H. A. Zimmerman, 72: test fixturing

J. E. Fay, 69: automated design

D. E. Alderman, 71: CMPS "PSALT"

A. T. Chen, 73: event trigger

J. Martin, 73: man-computer dialogues

Language Processors

RT Test Languages, Simulation Languages,  
Support Software

CHI Corp., 73: Language Implementation System  
M. N. Matelan, 70: BETA-to-BASIC translator  
E. A. Ehlers, 66: PLACE compiler for GPATS  
Independence, 73: decision table processors  
Trilog, 73: decision table compiler  
M. N. Matelan, 73: test language compiler-compiler  
Standard Data Corp., 73: COBOL SYMBUG  
M. H. Walshaw, 72: ATLAS→IDA  
W. Teitleman, 72: BBN-LISP aids  
Datamation, 2/73: FORTRAN compiler verification  
E. Fong, 73: compiler diagnostic capabilities  
B. B. Eiland, 70: Tree Meta study  
Datacraft, 73: Datacraft FORGO

Launch Vehicles

RT Spacecraft, Missiles

W. F. Huseonica, 65: current measurements

Learning\*

use Adaptive Control, Training

Logging\*

use Data Acquisition, Monitoring

Logic\*

use Digital Systems

Maintainability

RT Design, Economics

ASSC: various years  
F. J. Hackl, 65: IBM 360 self-test  
IEEE, 66+: automatic support systems symposia  
R. A. Johnson, 59: definition of maintainability; suggestions  
H. P. Nicely, 66: reliability and maintainability recording  
ASSC, 73: symposium  
C. T. Morgan, 63: human engineering guide

Maintenance

R. F. Barry, 70: failure prediction for preventive maintenance  
T. K. Elliott, 67: manual troubleshooting on MTS-2  
F. J. Hackl, 65: IBM-360 diagnostics  
E. B. Nutt, 65: maintenance recording  
J. T. Quatse, 65: time-shared oscilloscope (STROBES)  
R. E. Gunderman, 73: program maintenance  
Sider & Associates, 73: TXTM system  
Applied Data Research, 73: "Librarian"  
M. L. Yaffe, 73: airline ATE  
CDU-MO, 71: FSI DC-10 maintenance  
CDU-MO, 71: JAL 747 maintenance  
CDU-MO, 70: 737 simulator maintenance  
W. W. Anderson, 70: "EMMA"

Malfunctions\*

use Failure Modes

Management

NT Configuration Management

B. H. Liebowitz, 67: software specifications  
ESCON, 70: development of large software systems  
J. E. Butterworth, 73: Quality Assurance for SDM  
D. Smith, Jr., 72: project management  
R. L. Benbow, 73: simulator management  
S. D. Hansen, 73: software quality control

Manual Procedures

RT Human Factors, Interactive Terminals

N. C. Bauer, 69: oscilloscope  
T. K. Elliott, 67: proceduralized troubleshooting, MTS-2  
M. C. Colwell, 71: TREES software

Manuals

BT Documentation

CDU-MO, 71: FSI DC-10 maintenance  
CDU-MO, 71: JAL 747 maintenance  
CDU-MO, 70: 737 simulator maintenance  
T. K. Elliott, 67: proceduralized troubleshooting manual

Memories

BT Computers

S. A. Szygenda, 71R: memory self-test  
J. F. McKeivitt, 72: parity  
W. E. Kehret, 72: LSI memory test patterns  
ACL Chiang, 72: semiconductor memory tester

Microprogramming

BT Programming  
RT Computers

N. Bartow: 360/85  $\mu$  diagnostics  
Signetics, 72: debugging

Minicomputers

BT Computers

P. Kreager, 72: checksum  
P. C. Jackson, 73: integrating ATE peripherals

Missiles

RT Launch Vehicles

H. J. Shechtel, 67: test equipment for TALOS, etc.

Modelling\*

use Simulation

Modularization

RT Design

A. Avizienis, 72: fault-tolerant, modular computer  
C. W. Hemming, 72A: requirements for digital logic simulation  
W. W. Hinton, 73: simulation software modules  
N. Wirth, 71: stepwise refinement  
M. Iritz, 72: coding and testing

Monitoring

RT Data Acquisition

P. G. Bookman, 72: computer performance measurement  
H. P. Nicely, 66: in-service failures  
E. B. Nutt, 65: maintenance recording  
U. R. Furst, 66: Avionics B.I.T.  
R. F. Barry, 70: failure prediction  
J. R. Esser, 70: jet engine

J. M. Goodman, 67, vol. 4: B.I.T. and Continuous Monitoring  
C. Dudley Warner, 72: computer performance  
B. Miller, 73: failure detection  
W. W. Anderson, 70: "EMMA"  
J. E. Love, 74: YF-12 onboard monitor  
E. Fong, 73: compiler diagnostic capabilities  
AFIPS, 72: computer performance measurement  
Comress, 73: Dynaprobe-8000  
L. E. Hart, 71: "system stethoscope"

#### Monte Carlo Methods

M. A. Breuer, 71: random and algorithm for diagnostics  
M. A. Breuer, 72: random and algorithm for diagnostics  
C. Hammer, 67: validating math subroutines  
G. S. Fishman, 67: simulation validation and verification

#### Motion Systems

BT Simulators  
RT Hydraulic Equipment

R. S. Shirley, 73: "SAFE" (FFT) program  
AIAA, 73: visual and motion simulation  
L. R. Young, 73: visual sensation

#### On-Board Equipment

R. F. Wokasch, 69: evaluating onboard tests  
J. R. Esser, 70: jet engine monitoring  
B. M. Elson, 72: avionics  
H. J. Shechtel, 67: missile avionics  
B. Miller, 73: failure detection  
Electronics, 1/73: Honeywell 2600 for C-5A  
E. G. Solov, 73: inertial platforms  
H. G. Rome, 73: inertial platforms  
J. E. Love, 74: YF-12 onboard monitor  
W. W. Anderson, 70: "EMMA"

On-Line Equipment RT Time-sharing, Real-Time Operation

P. A. Hogan, 72: multi-station ATE

Operating Systems\* use Computer System Software

Optimization RT Criteria

R. A. Johnson, 59: various criteria  
Yu. V. Lyubotov, 65: minimize average time to localize  
J. H. Boeckel, 69: spacecraft test plans  
E. G. Dupnick, 73: integer programming; software checkout  
D. E. Stepler, 73: aerodynamic parameters

Oscilloscopes

W. C. Bauer, 69: Oscilloscope  
H. Moriyasu et al, 73: digital oscilloscope  
Tektronix, 74: catalog

Parity\* use Check Sums

Pattern Recognition

D. M. Goodman, 67 vol. 3: human color CRT  
V. S. Levadi, 65: applied to fault detection  
CompDes, 6/73: automatic IC fault detection  
H. Sriyananda, 73: selecting test frequencies

Performance Indexes\* use Criteria

Performance Verification\* use Validation

Peripherals\* use Computer Peripheral Equipment

Power Supplies\* use Electrical Power

Production Testing BT Testing  
RT Quality Control

ElecProd, 72: Automatic backplane testing  
T. Kuroda, 71: automated test system and its language  
J. P. Verna, 72: PC board tester  
E. F. Smith, 66: flexible automatic checkout system  
Computest, 73: semiconductor testers  
Teradyne, 73: L100 series  
R. Kitchen, 70: design impact  
Texas Instruments, 73: ATS-960  
Teradyne, 73: N151 backplane tester  
Fairchild, 73: FAIRTEST  
ADAR Associates, Inc., 73: memory and LSI testers  
F. Van Veen, 71: IC testing

Programming

BT Software Development  
RT Software Verification

D. D. McCracken, 72: readable FORTRAN  
CAR Hoare, 72: self-proving programs  
F. McMains, 67: FORTRAN EDIT  
P. A. Hogan, 69: automated diagnostic programming  
A. Merten, 72: language impact  
W. Teitleman, 72: BBN-LISP aids  
N. M. Cole, 73: plain English

Proving

RT Verification

Elsapas, 72: program-proving survey

Quality Control

RT Production Testing

J. E. Butterworth, 71: seminar on software  
N. F. Schneidewind, 72: software reliability  
S. D. Hansen, 73: computer programs

Random\*

use Monte Carlo Methods, Test Signals

Real-Time Operation

RT On-Line Equipment, Time-Sharing

M&S, 71: SLS logging

Reconfiguration

J. Goldberg, 66: error control  
A. Avizienis, 72: fault-tolerant, modular computer  
R. S. Chadwick, 73 (IBM): redundancy, reconfiguration

Recording\* use Data Acquisition, Monitoring

Recovery

D. L. Droulette, 71: IBM 360 recovery programming  
G. Oppenheimer, 68: hardware failures, protection  
Solutions, Inc., 72: CHKSTOP  
G. H. Maestri, 72: Retryable Processor  
R. S. Chadwick, 73: redundancy, reconfiguration  
UCC, 73: UCC-15 restart management

Redundancy RT Reliability

G. C. Hendrie, 66: "backup"  
P. Kreager, 72: checksum (CRC)  
F. D. Mathur, 70: self-repair  
A. Avizienis, 72: fault-tolerant, modular computer  
R. S. Chadwick, 73 (IBM): recovery, reconfiguration  
E. G. Solov, 73: inertial platforms  
H. J. Rome, 73: inertial platforms

Reliability RT Economics

G. C. Hendrie, 66: backup  
A. Avizienis, 72: fault-tolerant computing  
G. J. deWit, 68: flight simulator utilisation  
J. Goldberg, 66: logical design techniques  
RAeS, 71: flight simulator reliability  
J. L. Ogdin, 73: software reliability  
J. L. Ogdin, 72: software reliability  
F. P. Mathur, 70: redundancy and self-repair  
H. P. Nicely, 66: in-service MTBF, MTTR, etc.  
N. F. Schneidewind, 72: software reliability  
Z. Jelinski, 71: software reliability research  
J. E. Butterworth, 71: software quality assurance, not "reliability"

Requirements RT Specifications

MDEC, 73-1: visual simulation requirements  
W. T. Musial, 73-1: computer complex requirements  
J. F. Burke, 73-1: simulator hardware requirements

REPRODUCIBILITY OF THIS  
ORIGINAL PAGE IS POOR

Resetting

P. J. Waples, 67: initializing relay tree  
G. H. Maestri, 72: retryable processor

Security

B. Allen, 68: computer safeguards  
G. Oppenheimer, 68: protection of files

Self-Repair

RT Fault-tolerant Systems

I. Terris, 65: self-repairing computer  
B. E. Briley, 68: computer control section  
J. Goldberg, 66: logic reconfigurability  
F. P. Mathur, 70: hybrid-redundant

Self-Test

B. Allen, 68: HBR- self-test fails with failure  
R. E. Forbes, 65: "DX-1"  
E. F. Meyers, 69: IBM 4-pi  
ACL Chiang, 72: tester self-test  
S. Seshu, 65: improved diagnosis program  
R. A. Marlett, 66: self-diagnosing computers  
F. J. Hackl, 65: IBM-360 diagnostics  
L. G. Stucki, 73: "PET" program

Sensitivity

J. A. Alfieri, 66: logic-circuit design analysis

Sensors\*

use Data Acquisition

Sequential Testing\*

use Test Plans, Trees

imulation

M. E. Fowler, 68: difference equations  
R. T. Marshall, 73: F-104 dynamics  
S. A. Szygenda, 72: functional and gate-level simulation  
R.L.T. Hampton, 65: test signals  
E. W. Thompson, 73 (SMU): fault insertion  
S. A. Szygenda, ni: fault insertion  
C. W. Hemming, 72A: modular requirements for digital logic simulation  
S. A. Szygenda, 70S: realtime operation, on-line equipment  
H. D. Caplener, 73: modelling computer hardware  
C. R. Hanke, 70: B-747 modelling  
S. A. Szygenda, 71: simulated time flow

Simulation Languages

P. B. Dewan, 72: OSSL for computer simulation  
S. A. Szygenda, 72: functional simulation software setup

Simulation Studies

I. Terris, 65: self-repairing computer simulation  
A. Szygenda, 72sc: functional simulation, gate-level simulation  
P. Verma, 72: simulating PC boards  
W. T. Musial, 73-2: computer evaluation by simulation  
E. A. Palmer, 73: night visual for landing

Simulators

RT Training

G. J. deWit, 68: economics  
RAeS, 71: flight simulators  
T. K. Elliott, 67: MTS-2  
M&S, 71: SLS designs and procedures  
J. P. Vincent, 72: solar simulator alignment  
P. B. Schoonmaker, 73: verification problem areas  
R. T. Marshall, 73: F-104G validation  
Singer, 73: SLS procedures  
J. F. Burke, 73-1: simulator requirements  
J. F. Burke, 73-2: simulator baseline

MCAIR, 71: Simulation facility survey  
Interavia, 6/73: airline simulator market  
R. Braby, 69: cost vs. utilization  
W. F. Rhodes, 69: Simulator ATE  
NTDC, 69: symposium  
L. A. Whalen, 72: simulator B.I.T.  
K. J. Szalai, 71: airborne simulator  
F. A. Hock, 72: visuals  
CDU-MO, 71: FSI DC-10 maintenance  
CDU-MO, 71: JAL 747 maintenance  
CDU-MO, 70: 737 simulator maintenance  
D. E. Alderman, 71: CMPS "PSALT"  
P. B. Schoonmaker, 7/73: Verification IRAD (TR-2a)

Software Development

NT Programming

J. R. Metzner, 67: programming aids  
B. H. Liebowitz, 67: software specifications  
WESCON, 70: management  
TRW, 72: various authors; FSDVF  
N. F. Scheidewind, 72: software reliability  
S. D. Hansen, 73: quality control  
.. ASME, 71: verification, etc.  
J. E. Butterworth, 71: quality assurance seminar  
... J. L. Ogdin, 73: software reliability  
J. L. Ogdin, 72: software reliability  
G. M. Weinberg, 72: psychology  
G. M. Weinberg, 72: improved performance  
D. Smith, Jr., 72: project management  
B. B. Eiland, 70: Tree Meta study  
TRW, 72: space program history  
N. Wirth, 71: stepwise refinement  
N. M. Cole, 73: plain English  
M. Iritz, 72: coding and testing  
E. Fong, 73: compiler diagnostic capabilities  
Applied Data Research, 73: "Librarian"

Software Verification

NT Debugging (software)  
BT Verification

J. E. Butterworth, 71: quality assurance  
ASME, 71: symposium on engineering software  
Data-Pro, 72: proprietary software testing tools  
C. Hammer, 67: MC validation of math routines  
R. L. Nolan, 72: simulation models  
Synergetics, 72: fas/test for COBOL  
G. M. Weinberg, 71: software tools  
TRW, 7/72: software development  
Z. Jelinski, 71: "eutrophication" of errors  
L. G. Stucki, 72: "PTT" program experience  
B. Elspas, 72: program-proving survey  
Datamation, 2/73: FORTRAN compiler verification  
Sider and Associates, 73: TXTM system  
W. C. Hetzel, n/d: program test methods  
G. S. Fishman, 67: simulation validation and verification

Specifications

RT Requirements, Stan

B. H. Liebowitz, 67: basis of management  
C. E. Jones, 73: computer and DCE specs.

Standards

RT Specifications

J. R. Metzner, 67: software  
J. Toellner & Associates, 72: FORDOC for FORTRAN  
DoD, 70: NIL-STD-721B  
NASA, 71: computer program documentation  
ANSI, 70: flowchart standards

Statistical Techniques\*

use Monte Carlo Methods, Filtering,  
Optimization

Statistics

R. F. Wokasch, 69: a, B decision errors

Support Software

Independence, 73: decision table processor  
Trilog, 73: decision table compiler  
S. L. Prendergast, 72: commercial DBMS's  
L. G. Stucki, 72: "PTT" program experience  
L. G. Stucki, 73: "PET" program  
Sider & Associates, 73: TXTM system  
Information Associates, Inc., 73: HEXDSPLY  
Applied Data Research, 73: "Librarian"  
DataPro, 72: verification tools, flowcharters

Switches

S. Givens, 74: FET applications  
L. J. Sevin, Jr., 65: FET's

Switching Circuits\* use Digital Systems  
Systems\* use Integrated Systems  
Television RT Cathode Ray Tube

ASC Berens, 70: mainly TV  
R. Feldt, 68: noise measurement  
D. G. Fink, 57: handbook  
MDEC, 72: state of art  
MDEC, 73-1: simulator requirements  
MDEC, 73-2: proposed approaches  
S. Muth, 74: digitizing video signals  
A. H. Nagler, 71: wide FOV probe  
A. Nagler, 72: Wide FOV probe  
J. Roizen, 73: color cameras  
L. W. Lockwood, M. L. Noble, 70: high resolution TV  
G. A. McKenzie, 74: waveform measurement

Test Equipment

IERE, 70: symposium  
B. M. Elson, 72: airline maintenance (UAL DC-10)  
H. J. Shechtel, 67: Navy missile test  
R. Lund, 72: cars  
W. A. Scanga, 69: Navy ATS  
W. C. Bauer, 69: oscilloscope  
J. E. Fay, 69: interface  
A. C. L. Chiang, 72: semiconductor memory test  
Elec Prod, 72: backplane testing  
T. Kuroda, 71: MATE system and its language  
R. J. Meyer, 70: software survey  
K. C. Smith, 68: dual-processor checkout system  
O. R. Batchelder, 66: interfacing  
E. F. Smith, 66: flexible automatic checkout system  
J. P. Verma, 72: PC board tester  
B. E. Cowart, 71: SYMBOL computer assemblies  
D. Mactaggart, 70: MELVIN  
Elec Prod, 3/73: good survey  
P. A. Hogan, 72: multistation ATE  
CompDes, 6/73: <sup>1</sup>IC pattern recognition fault detection  
<sup>2</sup>automatic and semi-automatic PCB fault isolation  
W. F. Rhodes, 69: Simulator ATE  
J. Lyman, 73: portable and microprogrammable  
Electronics, 73: Honeywell 2600 for C-5A  
K. J. Stein, 74: shipboard VAST  
E. G. Cromer, 73: vendor survey  
D. P. Allen, 73: test digital IC boards  
P. C. Jackson, 73: integrating ATE peripherals

### Test Equipment (Vendors)

Macrodata, 71: MD-100 memory tester  
Tektronix, 72: S-3260  
Hewlett-Packard, 72: 9500 series  
Tektronix, 74: catalog  
Data Numerics, 73: continuity tester  
Teradyne, 73: backplane tester  
Hewlett-Packard, 73: IC troubleshooters, probes  
Signetics, 73: debug microprograms  
Computest, 73: semiconductor testers  
Teradyne, 73: L100 series  
Datatron, 73: 4000 series  
Digital General, 73: Model DC-VIII  
Addison, 73: Auto Trak, self-programming  
Texas Instruments, 73: ATS-960  
Teradyne, 73: N151 backplane tester  
Fairchild, 73: FAIRTEST  
ADAR Associates, 73: memory and LSI testers  
Data Industries, Inc.: transmission test set  
Philco Ford, 73: line test sets

### Test Languages

### RT Language Processors

J. W. Anstead, 70: ATLAS  
-ASSC, 70: VITAL, BETA; surveys  
- E. A. Ehlers, 66: PLACE, for GPATS  
- M. T. Ellis, 70: VITAL and "dialects"  
R. Grishman, 70: AIDS, for debugging  
Hewlett-Packard, 72: ATS BASIC  
ARINC, 72: ATLAS Spec. (416-6)  
W. Kocher, 69: TESTRAN, etc. (software debugging)  
M. N. Matelan, 70: BETA-to-BASIC compiler  
R. J. Meyer, 70: survey of  
M. N. Matelan, 73: test language compiler-compiler  
H. L. Fischer, 72: augmented FORTRAN  
D. MacTaggart, 70: MELVIN - assembly language  
T. Kuroda, 71: high-level language for MATE

Test Methods

RT Diagnostics

M. A. Breuer, 71: algorithms  
W. C. Hetzel, n/d: program test methods  
C. V. Ramamoorthy, 72: test validation  
J. P. Roth, 67: algorithms

Test Patterns\*

use Test Signals

Test Plans

M. A. Breuer, 71: sequential  
J. F. Poage, 64: sequential  
J. H. Boeckel, 69: acceptance-test optimization  
E. G. Dupnick, 73: minimize software check cases

Test Points

J. A. Alfieri, 66: test logic analysis  
C. V. Ramamoorthy, 71: "blocking gates" in LSI  
M. A. Mehta, 72: "ambiguity resolver" in PC board  
R. Lund, 72: cars

Test Signals

T. F. Miotke, 72: Boolean functions, test patterns  
J. D. Lamb, 70: pseudo-random binary sequence  
RLT Hampton, 65: A/D pseudo random noise generator  
R. C. Wingrove, 69: pilot describing functions  
R. C. Darf, 74: testing of control systems  
R. F. Garzia, 72: Bode plots  
A. J. Martin, 69: PRBS's  
D. E. Stepner, 73: aerodynamic parameters  
H. Sriyananda, 73: selecting test frequencies  
W. E. Kehret, 72: LSI memory test patterns  
R. L. Hampton, 65: pseudo-random noise  
S. W. Golomb, 67: shift-register sequences

Test Systems\*

use Test Equipment

Testability

- D. C. Bossen, 72: parity network test patterns
- M. A. Mehta, 72: LSI diagnostic resolution
- J. P. Roth, 67: distinguishing between logic failures
- B. E. Cowart, 71: SYMBOL testing aspects
- R. Kitchen, 70: design impact
- A. Kohavi, 67: diagnosable sequential machines

Testing

Avoid

- NT Production Testing, Acceptance Tests
- RT Checkout, Diagnostics, Verification, Test Methods

- B. E. Cowart, 71: testing SYMBOL computer

Time-Sharing

RT Real-Time Operation, On-Line Equipment

- J. T. Quatse, 65: STROBES: oscilloscope
- W. Nelson, 68: on-line peripheral testing

Tolerances\*

use Sensitivity, Failure Modes

Training

RT Simulators, Human Factors

- T. K. Elliott, 67: MTS-2 for troubleshooting
- RAeS, 71: training simulators
- R. Braby, 69: cost vs. utilization
- NTDC, 69: symposium

Transfer Functions\*

use Dynamics, Frequency Response

Transient Response

RT Dynamics, Frequency Response

- E. Forster, 73: waveform testing
- D. J. Blecki, 73: digital transient recorder

Trees

BT Data Structures, Graphs  
RT Fault Isolation, Flowcharts

R. F. Davis, 72: combinational/sequential tree circuits  
P. J. Waples, 67: relay tree  
G. P. Chubb, 69: cost-optimized troubleshooting  
TJB Hannon, 68: optimum troubleshooting  
M. C. Colwell, 71: software package  
B. B. Eiland, 70: Tree Meta study  
N. Wirth, 71: stepwise refinement

Trouble-Shooting\*

use Fault Isolation, Manual Procedures

Validation

RT Verification, Acceptance Tests

ASME, 71: engineering software symposium  
M. E. Fowler, 68: analytic check cases  
R. T. Marshall, 73: F-104G dynamics  
R. L. Nolan, 72: simulation models  
R. D. Wright, 72: predictive tests, dynamic models  
K. J. Szalai, 71: airborne simulator

Verification

RT Acceptance Tests, Checkout, Diagnostics  
NT Software Verification

J. A. Szygenda, 72H: digital system design verification  
P. B. Schoonmaker, 7/73: Verification IRAD (TR-2a)

Visual Systems

RT Displays & Controls, Electro-Optical  
Systems, Cathode Ray Tubes, Television

M. Aronson, 72: wide angle visuals  
MDEC, 72: survey of technology  
MDEC, 73-1: simulator requirements  
MDEC, 73-2: baseline configuration  
A. H. Nagler, 71: wide angle optical pick-up  
A. Nagler, 72: wide angle optical pick-up  
L. W. Lockwood, R. D. McCafferty, 69: LMS  
L. W. Lockwood, M. L. Noble, 70: high resolution  
R. D. McCafferty, L. W. Lockwood, 70: CMS.visuals

## 5.2 BIBLIOGRAPHIC DOCUMENT LISTING

The following pages contain a bibliographic listing of all documents reviewed during the Hardware Verification Task. These documents are ordered and tagged with a retrieval tag which enabled the concise subject cross index presented in the previous section. The documents are listed in nominal alphabetical order by authors' name.

KEY

BIBLIOGRAPHIC LISTING

-A-

- ACR, 73 (report) Applied Computer Research, Quarterly Bibliography of Computers and Data Processing: A Subject/Author Index to Computer Literature, Phoenix, Arizona, 1973
- ADAR Associates, Inc., 73 Adar Associates, Inc., Digital Text Systems, Cambridge, Mass., 1973
- Addison, 73 Addison Electric Company, Addison Auto Trak Self-Programming Fault Finder System for Testing - Backplanes, Harnesses, and Cables, Bulletin No. AT-3-73-5M, Mountainside, New Jersey, 1973
- ADR, 73 Applied Data Research, The Librarian: A Source Program Maintenance And Retrieval System for IBM System 360/370 Installations - OS and DOS, Princeton, New Jersey, 1973
- AFIPS, 72 Cureton, H.O., "A Philosophy to System Measurement," AFIPS Conference Proceedings, Vol. 41, pages 965-969, 1972
- AFIPS, 72 Rudd, Richard J., "The CPM-X-A Systems Approach to Performance Measurement," AFIPS Conference Proceedings, Vol. 41, pages 949-957, 1972
- AIAA, 73 "AIAA Visual and Motion Simulation Conference," September 10-12, 1973
- E.R. Aiken, 73 Aikin, Robert, and Minck, J.L., "Automating the Calibration Laboratory," Hewlett-Packard Journal, Vol. 24, No. 8, Palo Alto, California, April, 1973
- D.E. Alderman, 71 Lockheed Electronics Co., "Procedures Simulators Analog Linkage Test (PSALT) System," by Alderman, D.E. and McGiffin, M.R., Vol. 1 and 2, Houston, Texas, January, 1971
- J. A. Alfieri, 66 Northrop Nortronics Co., "Analytical Approach for Determination of Effective Test Logic Design," by Alfieri, J.A., Anaheim, California, 1966
- B. Allen, 68 Allen, Brandt, "Danger Ahead! Safeguard Your Computer," Reprinted from HBR, Nov.-Dec., 1968
- D.P. Allen, 73 Allen, Donald P., "Logic Tester Uses Single Trial Procedure to Troubleshoot Digital IC Boards," Electronics, Nov. 1973

KEY

BIBLIOGRAPHIC LISTING

- W. W. Anderson, 70  
NASA, "A Computerized Onboard Expeditious Monitor and Maintenance Analyst (EMMA) for Spacecraft Systems", by W. W. Anderson; R. W. Will, TN D-5725, May 1970
- ANSI, 70  
(std X3.5-1970)  
American National Standards Institute, American National Standard Flowchart Symbols and Their Usage for Information Processing, New York, New York, 1970
- J. W. Anstead, 70  
(IERE ppr)  
Anstead, J.W. and Challenger, M.2., "The Use of ATLAS by an Equipment Manufacturer," IERE Conference on Automatic Test Systems, pages 365-371, Cheltenham, Gloucestershire, 1970
- ARINC, 72  
(spec 416-6)  
Aeronautical Radio, Inc., Abbreviated Test Language for Avionics Systems (ATLAS) ARINC Specification 416-6, October 31, 1972
- M. Aronson, 72  
M. Aronson, "A New Assessment of Wide-Angle Visual Simulation Techniques", Proceedings of the Fifth Naval Training Device Center and Industry Conference, NAVTRADEVGEN IH-206, February 15-17, 1972
- ASME, 71  
American Society of Mechanical Engineers, Pressure Vessels and Piping Division, Symposium, On Engineering Computer Software: Verification, Qualification, Certification, pages 6,7,8,9,61,62, and 63, 1971
- ASSC, 65  
(proceedings)  
Proceedings of the Automatic Support Systems Symposium for Advanced Maintainability, St. Louis, Missouri, June 7-8-9, 1965
- ASSC, 68  
(symposium)  
Proceedings of the 1968 Automatic Support Systems Symposium, St. Louis, Missouri, Nov. 12-13-14, 1968
- ASSC, 69  
(symposium)  
Proceedings of the 1969 Automatic Support Systems Symposium, St. Louis, Missouri, Nov. 3-4-5, 1969
- ASSC, 70  
(symposium record)  
Symposium Record - Automated Support Systems for Advanced Maintainability, Nov. 1972
- ASSC, 73  
Automatic Support Systems for Advanced Maintainability, (Symposium), Arlington, Texas, Nov. 5-6-7, 1973
- G. Audin, 73  
Audin, Gary, "How Are Errors Detected?" Modern Data, August, 1973, page 39
- A. Avizienis, 72  
Avizienis, Algirdas, "Computer Reliability and Fault-Tolerant Computing" (First International Symposium on Fault-Tolerant Computing, Pasadena, Calif., March, 1972)

KEY

BIBLIOGRAPHIC LISTING

-B-

- R.F. Barry, 70 R. F. Barry, "Failure Prediction for Preventive Maintenance" (paper presented at the IERE Conference on Automatic Test System, 1970)
- N. Bartow, 70 N. Bartow, "IBM System/360 Model 85 Microdiagnostics," (paper presented at the Spring Joint Computer Conference, Kingston, New York, 1970)
- O. R. Batchelder, 66 O. R. Batchelder, "Interface Problems in a Multipurpose Tester," (paper presented at the IEEE Automatic Support Systems Conference, Ft. Worth, Texas, 1966)
- W. C. Bauer, 69 W. C. Bauer, "Integrating an Oscilloscope into a General Purpose Automatic Test System," (paper presented at the IEEE Automatic Support System Conference, St. Louis, Mo., 1969)
- R. L. Benbow, 73 McDonnell Douglas Astronautics Co.-East, Development of Simulation Computer Complex Specification, Simulation Management Plan Report, by R. L. Benbow, P. B. Schoonmaker, NASA Contract NAS 9-12882, Report No. MDC E0824, 31 May 1974
- A. S. C. Berens, 70 A. S. C. Berens and R. A. Welter, "Automatic Testing of Electro-Optical Systems," (paper presented at the ASSC, Burlington, Mass., 1970)
- G. D. Bergland, 69 G. D. Bergland, "A Guided Tour of the Fast Fourier Transform," IEEE Spectrum, July 1969
- A.T. Berztiss, 71 A. T. Berztiss, Data Structures, Academic Press, 1971
- D.J. Blecki, 73 David J. Blecki, "Trapping Transients Digitally," Electronic Products, 21 May 1973, pages 211-219
- J.H. Boeckel, 69 Goddard Space Flight Center, Test Plan Optimization for an Explorer-Size Spacecraft, by J. H. Boeckel and A. R. Timmins, NASA TND-5283, Greenbelt, Md., July, 1969
- B.W. Boehn, 70 Rand Report: "Some Information Processing Implications of Air Force Space Missions: 1970-1980," AD 703279, 1970
- P.G. Bookman, 72 Phillip G. Bookman and Barry A. Brotman, "Use Measurement Engineering for Better System Performance," Computer Decisions, 1972, pages 28-32
- D.C. Bossen, 72 D. C. Bossen, "Optimum Test Patterns for Parity Networks," Computer Design, April, 1972, pages 93-98
- raby, 69 Richard Braby, "The Impact of Utilization Patterns on the Cost Effectiveness of Training Devices," (paper presented at the NTDC Conference, 1969)

C-36

KEY

BIBLIOGRAPHIC LISTING

- M. A. Breuer, 68 M. A. Breuer, "Hardware Fault Detection," (paper presented at the FJCC, 1968)
- M.A. Breuer, 71 M. A. Breuer, "A Random and an Algorithmic Technique for Fault Detection Test Generation for Sequential Circuits," IEEE Transactions on Computers, Nov., 1971, Pages 1364-1365
- M.A. Breuer, 72 M. A. Breuer and Stephen Y. H. Su, "Automatic Test Generation for Digital Networks - Some Experimental Results," (paper presented at the SCSC, 1972)
- E. O. Brigham, 74 E. Oran Brigham, The Fast Fourier Transform, New Jersey, Prentice-Hall, 1974
- B. E. Briley, 68 Bruce E. Briley, "A Self-Healing Control," Bell System Technical Journal, 30 April 1968, page 2367
- J.F. Burke, 73-1 Singer Company, Simulation Products Division, Shuttle Mission Simulator Requirements Report, Volumes I and II, by J.F. Burke, NASA Contract No. NAS 9-12836, Binghamton, New York, December 1973
- J.F. Burke, 73-2 Singer Company, Simulation Products Division, Shuttle Mission Simulator Baseline Definition Report, Volumes I and II, by J.F. Burke, NASA Contract No. NAS 9-12836, Binghamton, New York, December 1973
- I.E. Butterworth, 71 I. E. Butterworth, "Software Quality Assurance," (paper presented at ASOC Seminar, 18 September 1971)

BIBLIOGRAPHIC LISTING

-C-

- M.A. Calhoun, 72 M. A. Calhoun, "Symbol Hardware Debugging Facilities," (paper presented at the SJCC, Manhattan, Kansas, 1972)
- H. D. Caplener, 73 Harold D. Caplener and Jan A. Janku, "Improved Modeling of Computer Hardware Systems," Computer Design, August, 1973, pages 59-64
- P.M. Casey, 70 Patrick M. Casey and Keith W. Oliver, "Detection and Isolation In A Diagnostic Program," ASSC Record, 1970
- CDC, 71 Control Data Corporation, 3100, 3200, 3300, 3500-MSOS/RTS Diagnostic Handbook, #60236700B, St. Paul, Minn., 10 February, 1971
- CDU-MO, 70 Conductron-Missouri, Division of Conductron Corporation, Operation and Maintenance Instructions, CPA 737 Flight Simulator, TM-68A027-3, St. Charles, Mo., 15 May 1970
- CDU-MO 71 Conductron-Missouri, Division of Conductron Corporation, Operation and Maintenance Instructions, Japan Airlines 747 Flight Simulator, Vol I, TM-68A036-3, St. Charles, Missouri, 15 February 1971
- J-MO 71 Conductron-Missouri, Division of Conductron Corporation, Operation and Maintenance Instructions, Flight Safety Inc. DC-10 Flight Simulator, Vol. I, TM-68A037-3, St. Charles, Missouri, July, 1971
- R.S. Chadwick, 73 R. S. Chadwick, "Recovery of a Large Computer System," 1973 SWIEEECO Record of Technical Papers, 73 CHO 719-5-SWIECO, page 378. (paper presented at the 25th Annual SWIEEECO and Exhibition, Houston, Texas, 4-6 April 1973)
- CHI Corp. 73 CHI Corporation, Systems Development, Language Implementation System, Cleveland, Ohio, 1973
- A.T. Chen, 73 A. T. Chen and R. S. Siebert, "A Data-Triggerable Program Debugging Aid," Computer Design, Sept., 1973, pages 90-93
- A.C.L. Chiang, 72 Albert C. L. Chiang, "A Self-Testable Semiconductor Memory Tester," Computer Design, June, 1972, pages 75-78
- G.P. Chubb, 69 Gerald P. Chubb and Robert G. Mills, "Development and Preparation of Cost-Optimized Troubleshooting Decision Trees," (paper presented at the ASSC, Aerospace Medical Research Laboratory, Wright-Patterson Air Force Base, Ohio, 1969)
- CIS, 73 CIS, Computer Interface Systems, Program Debugging Aid, Piscataway, New Jersey, 1973

KEY

BIBLIOGRAPHIC LISTING

- W.T. Cochran et al, 67 William T. Cochran, James W. Cooley, David L. Favin, Howard D. Helms, Reg A. Kaenel, William W. Lang, George C. Maling, Jr., David E. Welson, Charles M. Roder, Peter D. Welch, "What is the Fast Fourier Transform?", Proceedings of the IEEE, Vol. 55, No. 10, October 1967
- H.A. Cole, 73 Henry A. Cole, Jr., "On-Line Failure Detection and Damping Measurement of Aerospace Structures by Random Decrement Signatures," NASA Contractor Report, NASA CR-2205, March 1973
- N.M. Cole, 73 Ned M. Cole, Jr., and Michael J. Suke1, "Solving A Software Design Problem Using Plain English," Datamation, October 1973, pages 101-106
- M.C. Colwell, 71 M. C. Colwell, D. M. Risk, L. E. Reed, "Tree's User's Guide: A Computer Software System for Handling Information in Branch Form," AFHRL-TR-71-27, Dayton, Ohio, 1971
- Comp. Des., 73/6 "Integrated Circuits Pattern, Recognition Test Equipment," Computer Design, June 1973
- Comress, 73 Comress, "Dynaprobe - 8000: System Concepts and Capabilities," CR4-0046, Rockville, Maryland, 1973
- Computest, 72 Computest, Disk Testing: Reasons, Methods, Equipment, and Costs, brochure, Cherry Hill, New Jersey, 1972, pages 3-20
- Computest, 73 Computest, Semiconductor Test Equipment and Systems, Bulletin #73-3-1, Cherry Hill, New Jersey, 1973
- R.J. Conn, 72 Ralph B. Conn and Nikitas A. Alexandridis and Algindas Avizienis, "Design of a Fault-Tolerant, Modular Computer with Dynamic Redundancy," (paper presented at the Fall Joint Computer Conference, 1972)
- J. W. Cooley et al, 69 J. W. Cooley, P. A. W. Lewis, P. D., "The Finite Fourier Transform" IEEE Transactions on Audio and Electroacoustics, Vol. AU-17, Vo. 2., June 1969
- B. E. Cowart, 71 B. E. Cowart, R. Rice, and S. F. Lundstrom, "Physical Attributes and Testing Aspects of the Symbol System," (paper presented at the Spring Joint Computer Conference, Palo Alto, California, 1971)
- E.G. Cromer, 73 Earl G. Cromer, "Fault Isolation and Repair Techniques: Current Strategies and Equipment," Electronic Packaging and Production, June 1973, pages 28-36

KEY

BIBLIOGRAPHIC LISTING

-D-

- R. C. Dorf, 74 Richard C. Dorf, Modern Control Systems, Massachusetts, Addison-Wesley, 1974
- Data Industries, Inc. 73 Digitech Data Industries, Data Transmission Test Set the 2056, Ridgefield, Connecticut, 1973
- Data Numerics, 73 Data Numerics, Automatic Continuity Test System, Farmingdale, New York 1973
- Datacraft, 73 Datacraft, Forgo Series 6000, Bulletin #PB61111-00, Fort Lauderdale, Florida
- Datamation, 73 Dennis C. Phipps, Bob Blease, Jack M. Wolfe, Neal Paris and Donald C. Harder, "Letters on Programming and Documentation," Datamation, February 1973, pages 23-24-157
- Datamation 73/2 "Soon They'll Also Validate FORTRAN," Datamation, February 1973, page 119
- Data Pro, 72 Data Pro Research Corporation, Proprietary Software Testing Packages, 70E-188-01a Software, January, 1972
- Data Pro, 72 Data Pro Research Corporation, Proprietary Flowchart Packages, 70E-052-03c Software, May 1971
- Datatron, 73 Datatron Incorporated, Automatic Testers, 1973
- R.F. Davis, 72 Richard F. Davis, Jr., "Fault Testing in Combinational and Sequential Tree Circuits," (paper from Applied Computer Science Department, Albuquerque, New Mexico, presented at SCSC 1972)
- P.B. Dewan, 72 Prem Bhushan Dewan, C.E. Donaghey, and Joe B. Wyatt, "OSSL-A Specialized Language for Simulating Computer Systems," (paper presented at SJCC, 1972)
- G.J. deWit, 68 G. J. deWit, "Reliability and Utilisation of Flight Simulators," Aeronautical Journal of the Royal Aeronautical Society, January, 1968
- Digital General, 73 Digital General Corporation, Diagnostic Computers and Test Equipment: Model DC-VII, Cleveland, Ohio, 1973
- DoD, 70 Department of Defense, Definitions of Effectiveness Terms for Reliability, Maintainability, Human Factors, and Safety, MIL-STD-721B, Washington, D.C., 10 March 1970
- " S. Dordick, 66 H. S. Dordick and J. W. Ranftl, "Advanced Sensing Techniques for Automatic Checkout: A Comparative Discussion," Rand RM 4937, NASA, Contract #NASr 21 (08), March 1966

KEY

BIBLIOGRAPHIC LISTING

- L. Droulette, 71 Donald L. Droulette, "Recovery Through Programming System/360 - System/370," (paper presented at the SJCC, Kingston, New York, 1971)
- E.G. Dupnick, 73 National Aeronautics and Space Administration, Johnson Space Center, A Zero-One Integer Programming Solution to Determine the Minimum Number of Test Cases Required for Software Checkout, Internal Note #73-FM-63, Houston, Texas, 27 April 1973
- E-
- E.A. Ehlers, 66 Erwin A Ehlers, "Development of a Compiler for GPATS," (paper presented at Automatic Support Systems Symposium, 1966)
- E.B. Eichelberger, 64 E. V. Eichelberger, "Hazard Detection in Combinational and Sequential Switching Circuits," (paper presented at Switching Circuit Theory Symposium, 1964)
- M. Elección, 74 Marce Eleccion, "Automatic Testing: Quality: Raiser, Dollar Saver," IEEE Spectrum, August 1974, pages 38-43.
- Elec. Prod., 72/5 Esinbyen and Harvey Goldberg, "The IC Testing Puzzle," Electronic Products Magazine, 15 May 1972, pages 65-86
- Elec. Prod., 72/12 "Two New Approaches to Automatic Backplane Testing," Electronic Products Magazine, December, 1972, pages 21-22
- Elec. Prod., 73/3 Jerome Lyman, "A Cost -Effective Way to Test Complex Electronics," Electronic Products Magazine, 19 March 1973, pages 117-135
- Electronics, 73/1 "C-SA Testers Get New Automatic Gear," Electronics, 18 January 1973, pages 112-113
- T.K. Elliot, 67 Aerospace Medical Research Laboratories, "The Effect of Electronic Aptitude on Performance of Proceduralized Troubleshooting Tasks," by T.K. Elliot, USAF AMRL TR 67-154,1
- M.T. Ellis, 70 Michael T. Ellis, "Vital - A General Purpose, Multiple Dialect Automatic Test Language," (paper presented at Automatic Support Systems Conference, June, 1970)
- B.M. Elson, 72 Benjamin M. Elson, "United Automates DC-10 Avionics Tests," Aviation Week and Space Technology, 7 August 1972, pages 36-38
3. Elspas, 72 Bernard Elspas, Karl Levitt, Richard Waldinger and Abraham Waksman, "An Assessment of Techniques for Proving Program Corrections," Computing Surveys, Volume 4, June 1972, page 1

KEY

BIBLIOGRAPHIC LISTING

J.R. Esser, 70

John R. Esser, "Flight Test Evaluation of an Advanced On-Board Jet Engine Monitoring System," (paper presented at Automatic Support Systems Conference, 1970)

KEY

BIBLIOGRAPHIC LISTING

-F-

- Fairchild, 73      Fairchild Systems Technology, Introducing FAIRTEST: A Computer Aided Test Generation System, brochure, Palo Alto, California, 1973
- J. C. Fakan, 69      NASA, Application of Modern Network Theory to Analysis of Complex Systems, by John C. Fakan, Rpt. No. NASA TND-5511, Cleveland, Ohio, October 1969
- J. E. Fay, 69      J. E. Fay and D. H. Lovegrove, "Automated Interface Design for ATE," (Paper presented at the ASSC, 1969)
- R. Feldt, 68      Rudolf Feldt, "Measurement of Noise in Video Systems", Journal of the SMPTE, Volume 77, April 1968
- D. G. Fink, 57      Donald G. Fink, ed., Television Engineering Handbook, McGraw-Hill, 1957
- H.L. Fischer, 72      H. L. Fischer and A. G. Vallance, "A Fortran Solution to a Higher Order Test Language Requirement," ASSC Record 72-173, 1972, pages 173-179
- J.R. Fischer, 70      J. R. Fischer, "Fortran Program for Fast Fourier Transform," NRL Report #7041, Project RF-05-552-403-4069, 1970
- G.S. Fishman, 67      Rand Report, Digital Computer Simulation: Statistical Considerations, RM 5387 PR, November 1967
- M.J. Flynn, 72      M. J. Flynn and S. A. Szygenda, "A Perspective on Computer Reliability," (from the Proceedings of the 1972 Annual Reliability and Maintainability Symposium, Dallas, Texas, 1972)
- R.E. Forbes, 65      R.E. Forbes, D. H. Rutherford, C. B. Stieglitz, and L. H. Tung, "A Self-Diagnosable Computer," (paper presented at the FJCC, Endicott, New York, 1965)
- E. Fong, 73      National Bureau of Standards, A Set of Debugging and Monitoring Facilities to Improve the Diagnostic Capabilities of a Compiler, Washington, D.C., March 1973
- E. Forster, 73      Ernst Forster, "Computerized Waveform Testing Improves Accuracy," Electronic Products, 16 July 1973
- M.E. Fowler, 68/11      McDonnell Douglas Astronautics Co., Analysis and Check Solutions for Evaluation Simulations, DC-10, Design Note No. 4, Houston, Texas, 4 November 1968
- A.M. Fuchs, 66      A. M. Fuchs and F. Post, "Automated Frequency Response Measurement Compatible with Computer Control and Readout," (paper presented at the ASSC, 1966)

KEY

BIBLIOGRAPHIC LISTING

1 R. Furst, 66

U. R. Furst, "Automatic Built-In Test of Advanced Avionics Systems," Supplement to IEEE Transactions on Aerospace and Electronic Systems, Vol. AES-2, No. 4, July 1966 (paper presented at the 1966 Aerospace System Conference, Culver City, California, July 1966)

KEY

BIBLIOGRAPHIC LISTING

-G-

- U.O. Gagliardi, 65 Dunlap and Associates, Inc., Man-Computer Relationships In Fault Isolation, by Ugo O. Gagliardi, NASA Contract NASW-954, Darien, Conn., June 1964
- E. K. Gannett, 71 E. K. Gannett, An American National Standard IEEE Standard Dictionary of Electrical and Electronics Terms, Wiley-Interscience, 13 October 1971
- R. F. Garzia, 70 Ricardo F. Garzia, "A Fast Technique for Dynamic Fault Detection," ASSC Record 70, (paper presented at the ASSC, 1970)
- R. F. Garzia, 71 Ricardo F. Garzia, Fault Isolation Computer Methods, NASA Contractor Report, CR-1758, MSFC, Huntsville, Ala., 1971
- R. F. Garzia, 72 Ricardo F. Garzia, "Fault Isolation in Complex Systems Via Bode Diagram Technique," (paper presented at the ASSC, Barberton, Ohio, 1972).
- S. Givens, 74 Shelby Givens, "Field-Effect Transistors as Analog Switches," Computer Design, June 1974, pp. 106-110
- J. Goldberg, 66 Jack Goldbert, "Logical Design Techniques for Error Control" (paper presented at U. C., Menlo Park, California, 1966)
- S. W. Golomb, 67 Solomon W. Golomb, Shift Register Sequences, Holden-Nay, Inc., 1967
- D.M. Goodman, 66 David M. Goodman, Automation in Electronic Test Equipment, New York University, 1966
- B. M. Gordon, 74 Bernard M. Gordon, "Noise-Effects on Analog to Digital Conversion Accuracy," Computer Design, March and April, 1974
- R. Grishman, 70 Ralph Grishman, "The Debugging System AIDS," (paper presented at the SJCC, New York, New York, 1970)
- R.F. Gunderman, 73 Richard E. Gunderman, "A Glimpse Into Program Maintenance," Datamation, June 1973, Pages 100-101

-H-

- F. J. Hackl, 65 F. J. Hackl and R. W. Shirk, "An Integrated Approach to Automated Computer Maintenance," (paper presented at the IEEE Conference, Poughkeepsie, New York 1965)
- C. Hammer, 67 Carl Hammer, "Statistical Validation of Mathematical Computer Routines," (paper presented at the SJCC, Washington, D.C. 1967)

KEY

BIBLIOGRAPHIC LISTING

- R.L.T. Hampton, 65 R. L. T. Hampton, "A Hybrid Analog-Digital Pseudo Random Noise Generator," (paper presented at the SJCC, Tucson, Arizona, March, 1965)
- R. L. Hampton, 65 R. L. Hampton, "Experiments Using Pseudo-Random Noise," Simulation, April 1965, pages 247-254.
- C.R. Hanke, 70 Boeing Company, The Simulation of a Large Jet Transport Aircraft, by C. Rodney Hanke and Donald R. Nordwall, Contract No. NAS2-5524, Wichita, Kansas, September 1970
- T.J.B. Hannom, 68 T. J. Barry Hannom, Anthony J. Azzari, Alan I. Brooks, John R. Hunter, Leon Steinberg and Gerald P. Chubb, "Development of a Computer Program for Generating Trouble-shooting Decision Trees," USAF AMRL-TR-67-83, September 1967
- S.D. Hansen, 73 S. D. Hansen and Q. McHarg, "Reliability and Quality Control of Production Engineering Computer Programs," (AIAA paper 73-356, paper presented at the AIAA/ASME/SAE, 14th Structures, Structural Dynamics, and Materials Conference, Williamsburg, Virginia, March 20-22, 1973
- L.E. Hart, 71 Larry E. Hart and George J. Lipovich, "Choosing a System Stethoscope," Computer Decisions, November 1971, pages 20-23
- A.W. Haydon Co., 72 A. W. Haydon Company, a North American Philips Company, Indicators (brochure), Waterbury, Conn.
- W. A. Heffner, 70 William A. Heffner; Automatic Support Systems For Advanced Maintainability, ASSC Record 70, (paper presented at the ASSC, Denver, Colorado, 1970)
- H. R. Hegner, 66 H. R. Hegner, R. S. Braman, and J. E. Bridges, "Indirect Sensing of Quiescent Parameters for Built-in Test Equipment," (paper presented at the ASSC, Chicago, 1966)
- C.W. Hemming, 72 C. W. Hamming and S. A. Szygenda, "Modular Requirements for Digital Logic Simulation at a Predefined Functional Level," (from the proceedings of the National ACM Conference, 1972)
- G.C. Hendrie, 66 G. C. Handrie, "Reliability Still Means Backup," Control Engineering, September 1966, pages 131-132
- F.C. Hennie, 64 F. C. Hennie, "Fault Detecting Experiments for Sequential Circuits, (paper presented at the IEEE Switching Circuit Theory and Logical Design Symposium, Cambridge, Mass. 1964)

KEY

BIBLIOGRAPHIC LISTING

- w.C. Hetzel, 72 William C. Hetzel, ed., Program Test Methods, North Carolina, Prentice-Hall, Inc., Englewood Cliffs, New Jersey (based on the proceedings of the Computer Program Test Methods Symposium held at the University of North Carolina, June 21-23, 1972)
- Hewlett-Packard, 72 Hewlett-Packard, HPATS BASIC, by John G. Kemeny and Thomas E. Kurtz, November 1972
- Hewlett-Packard, 72 Hewlett-Packard, Configuring Guide to 9500 Series Automatic Test Systems, Technical Data, January 1972
- Hewlett-Packard, 73 Hewlett-Packard, IC Troubleshooters, May 1973
- Hewlett-Packard Hewlett-Packard, The 3050A Automatic Data Acquisition System, Technical Data, 1 July 1973
- Hewlett-Packard, 73 Hewlett-Packard, Logic Analyzer, 1973
- W. W. Hinton, Jr., 73 McDonnell Douglas Astronautics Co.-East, Development of Simulation Computer Complex Specification, Simulation Software Sizing Report, NASA Contract NAS 9-12882, Report No. MDC E0739, 15 January 1973
- C.A.R. Hoare, 72 C.A.R. Hoare, "Proof of a Program: FIND," Communications of the ACM, Vol. 14, No. 1, January 1971
- E.A. Hock, 73 Eugene A. Hock, "A-Visual Display System Approach for An Advanced Spaceflight Simulator," AIAA paper 73-923 (paper presented at the AIAA Visual and Motion Simulation Conference, Palo Alto, California, September 10-12, 1973)
- P.A. Hogan, 69 Phillip A. Hogan, "Automatic Generation of Test Programs," (paper presented at the ASSC, 1969, Minneapolis, Minnesota)
- P.A. Hogan, 72 Honeywell, Inc., Aerospace Division, A Time Shared Computer Data System for Automated Test Stations, by Philip A. Hogan and Aerial E. Nelson, Minneapolis, Minn. 1972
- W.J. Hurd, 74 William J. Hurd, "Efficient Generation of Statistically Good Pseudonoise by Linearly Interconnected Shift Registers," IEEE Transactions on Computers, Vol. C-23, No. 2, February 1974
- W.F. Huseonica, 65 NASA, Technical Note, The Current Measurement in Launch Vehicle Checkout, by William F. Huseonica, TND-3585, Washington, D.C., October 1966

BIBLIOGRAPHIC LISTING

I

- IBM, 66 IBM, IBM System 360 Operating System: Fortran IV(G) Programmers Guide, File No. 5360, 25, Form No. C28.6639.1, 1966.
- IBM, 67 IBM, IBM System/360 Operating System Programmer's Guide to Debugging, File No. 5360-20; Form No. C28-6670-0, Houston, Texas, 1967.
- IBM, 72 IBM, Cobol Interactive Debug, C520-2557-0, White Plains, N. Y.
- IEEE, 64 Institute of Electrical and Electronics Engineers, Switching Circuit Theory and Logical Design, (proceedings of the fifth Annual Symposium, Princeton University, Princeton, N. J., Nov. 11-13, 1964).
- IEEE, 66 Institute of Electrical and Electronics Engineers, Aerospace and Electronic Systems, (proceedings of the 1966 Aerospace Systems Conference Olympic Hotel, Seattle, Washington, July 11-15, 1966).
- IEEE, 69 Institute of Electrical and Electronics Engineers, Automatic Support Systems Symposium for Advanced, Maintainability, (proceedings of the 1969 Automatic Support Systems Symposium, St. Louis, Missouri, Nov. 3-5, 1969).
- IEEE, 71 C. V. Ramamoorthy, "Fault-Tolerant Computing: An Introduction and an Overview," IEEE Transactions on Computers, Vol. C-20, No. 11, Nov. 1971.
- IERE, 70 IERE, Joint Conference on Automatic Test Systems, (proceedings of the Joint Conference on Automatic Test Systems, Univ. of Birmingham, 14-17 April 1970).
- Independence, 73 Datapro Research Corporation, Decible III Independence Computing and Software Corp., 70E-496-019, Debran, N. J., Jan 1974.
- Information Associates, Inc. 73 Information Associates, Inc., Hexadecimal Display Utility (HEXDSPY), No. 108, Rochester, N. Y., 1973.
- H.H. Ipolyi, 73 McDonnell Douglas Astronautics Co.-East, Development of Simulation Computer Complex Specification, System Software Requirements Final Report, by H.H. Ipolyi, NASA Contract NAS 9-12882, Report No. MDC E0799, 30 April 1973.
- Iritz, 72 Maxine Iritz, "Modular Programming Makes Systems Satisfy User Needs", Computer Decisions, Ridgewood, N. J. 1972.

BIBLIOGRAPHIC LISTING

J

- P. C. Jackson, 73 Phillip C. Jackson, "Integrating Nonstandard Peripheral Devices into Automated Test Systems", Computer Design, pages 83-88; Sept. 1973.
- H. Jacobowitz, 67 Herman Jacobowitz, "A Logic Oriented Diagnostic Program," (paper presented at the Spring Joint Computer Conference, Camden, N. J., 1967).
- Z. Jelinski, 71 Z. Jelinski and Dr. P. B. Moranda, "Software Reliability Research," (paper presented at Conference on Statistical Methods for the Evaluation of Computer System Performance, Brown University, 1971).
- Z. Jelinski, 73 Z. Jelinski and P. B. Moranda, "Application of a Probability-Based Model to a Code Reading Experiment" (paper presented to IEEE Symposium on Computer Software Reliability, N. Y. 30 Apr-2May 73).
- R. A. Johnson, 59 Syracuse University Research Institute, Diagnosis of Equipment Failures, by Richard A. Johnson, Earl Kletsky, and John Brule, Tech. Rpt. No. 1, RADC -TR-59-47, AD-213876, Arlington, Virginia, April 1959.
- C. E. Jones, 73 McDonnell Douglas Astronautics Co.-East, Development of Simulation Computer Complex Specification, Training Simulator Computer Complex Specification, by C. E. Jones and H. H. Ipolyi, NASA Contract NAS 9-12882, Report No. MDC E0833, 29 June 1973.

KEY

BIBLIOGRAPHIC LISTING

-K-

- L. Kay, 66 Harold L. Kay, BITE in Aircraft Electric Power Systems, Supplement to IEEE Transactions on Aerospace and Electronic Systems, Vol. AES-2, No. 4, Lima, Ohio, July, 1966.
- L. E. Karels, 73 L. E. Karels, "New Techniques in Automatic Test Equipment," (paper presented at the Automatic Support Systems Conference, 1973.)
- W. E. Kehret, 72 Tektronix, Test Patterns and Their use in LSI Memory Diagnostics, January, 1972.
- J. W. Kenney, 72 J. W. Kenney, "Test Techniques for Advanced Avionics Displays," (paper presented at the Automatic Support Systems Conference, Fort Worth, Texas, 1972).
- R. Kitchen, 70 R. Kitchen, "The Impact of Automatic Testing on Product Design and on the Production Testing Organization," (paper presented at the IERE Joint Conference on Automatic Test Systems, 1970)
- W. Kocher, 69 NASA Contractor Report, A Survey of Current Debugging Concepts, by Wallace Kocher, NASA CR-1397, 1969.
- Z. Kohavi, 67 Zir Kohavi and Pierre Lavallee, "Design of Diagnosable Sequential Machines," (paper presented at the Spring Joint Computer Conference, Cambridge, Mass. 1967.)
- Paul Kreager Paul Kreager, "Small Computer Cyclic Redundancy Checksum Capability," Computer Design, May 1972.
- T. Kurodo, 71 Terushsia Kruroda and Thomas C. Bush, "Multiband Automatic Test Equipment - A Computer Controlled Check-Out System," (paper presented at the Spring Joint Computer Conference, New York, N. Y. 1971).

KEY

BIBLIOGRAPHIC LISTING

L

- J. D. Lamb, 70 J. D. Lamb, "Direct Frequency Response Determination Exploiting the Deterministic Characteristics of Pseudo-Random Binary Sequences," (paper presented at the IERE Conference on Automatic Test Systems, 1970).
- N. A. Landis, Jr., 71 NASA, Generalized Software System For Multiple Small Data-Acquisition Systems, by Norman A. Landis, Jr., Rpt. No. TMX-2331, July 1971.
- D. Lefkowitz, 69 David, Lefkowitz, "File Structures for On-line Systems," New York, Spartan, 1969.
- V. S. Levadi, 65 V. S. Levadi, "Pattern Recognition Applied to Fault Detection," (paper presented at Conference on Military Electronics, Washington, D. C., Sept. 1965).
- V. S. Levadi, 66 V. S. Levadi, "Automated Learning Applied to Fault Diagnosis," (proceedings of the IEEE-Automatic Support System Conference, Minneapolis, Minn. 1966)
- V. S. Levadi, 66 Honeywell, Inc., "Fault Diagnosis by White Noise Techniques," by Victor S. Levadi, Laurence D. Turner, and Larry W. Christenson, AFAPL-TR-66-43, May '66.
- H. Liebowitz, 67 Burt H. Liebowitz, "The Technical Specification-Key to Management Control of Computer Programming" (paper presented at the Spring Joint Computer Conference, Washington, D. C., 1967).
- D. Lignos, 72 Demetrios Lignos, "A Hardware Diagnostic Capability for Disc Drives" Computer Design, April, 1972.
- L.W. Lockwood, 69 Lawrence W. Lockwood and Riley D. McCafferty, "Visual Simulation in the Lunar Module Mission Simulator" (paper presented at the 1969 ISA Annual Conference and Exhibit, Houston, Texas, October 27-30, 1969).
- L.W. Lockwood, 70 Lawrence W. Lockwood and Milton L. Noble, "Very High Resolution Television for Visual Simulation", Journal of the SMPTE, Vol. 79, No. 4, April, 1970.
- K. W. Lord, 73 Kenniston W. Lord, Jr., A Communicator's Guide Decision Tables, Ware Associates, 1973.
- J. E. Love, 74 NASA, Flight Study of a Vehicle Operational Status and Monitoring System, by James E. Love, William J. Fox, and Edward J. Wicklund, TND-7546, Washington, D. C., January 1974.

KEY

BIBLIOGRAPHIC LISTING

- LTV, 68 LTV Electrosystems, Inc., Greenville Division, System Simulation Using the Fast Fourier Transform, Rpt. No. G 3685.29.03B, Nov 1968.
- R. Lund, 72 Robert Lund, "How Plug-in Troubleshooting Will Save You Money," Popular Mechanics, Oct. 72, pages 132-135.
- D. Luttröpp, 73 Dave Luttröpp, "Test/Measurement Applications," Electronic Products, 1973.
- J. Lyman, 73 Jerome Lyman, "News of Products (portable and micro programmable ATE; graphics)," Electronic Products, 19 Mar 1973, pages 83-88.
- Y. V. Lyubaton, 65 Yu V. Lyubaton, "Optimal Procedures for Localization of Breakdown in a Modulated Radioelectronic System," Technical Cybernetics, No. 4, 10 Nov 1964 (see N65-10757)

KEY

BIBLIOGRAPHIC LISTING

M

- M & S Computing, Inc., 71 M & S Computing, Inc., SLS Structural Designs & Real Time Procedures - Final Report, Rpt. No. 71-0006, Rev.1, Contract No. S-62483, Binghamton, N. Y., 23 August 1971.
- Macrodata, 71 Macrodata Co., Test System Division, Macrodata MD-100 Test Systems, Chatsworth, Calif., 1971.
- D. Mactaggart, 70 Donald Mactaggart, "MELVIN, or Why Buy Racks of Conventional Test Equipment When You Have a Computer in the Loop?", (paper presented at the Automatic Support Systems Conference, 1970.)
- G. H. Maestri, 72 George H. Maestri, "The Retryable Processor," (paper presented at the Fall Joint Computer Conference, Phoenix, Arizona, 1972.
- Z. Manna, 70/5 Zohar Manna, "Termination of Programs Represented as Interpreted Graphs," (paper presented at the Spring Joint Computer Conference, Pittsburgh, Penn.:1970.)
- R. A. Marlett, 66 Ralph Allan Marlett, "On the Design and Testing of Self-Diagnosable Computers," University of Illinois, Coordinated Sciences Report R-293, July 1966.
- R. T. Marshall, 73 NASA, Modeling of Airplane Performance from Flight-Test Results and Validation with an F-104G Airplane, by Robert T. Marshall and William G. Schweikhard, TN D-7137, February 1973.
- A. J. Martin, 69 A. J. Martin, "PRBS can fool the system," Electronics, 28 Apr 1969.
- A. J. Martin, 73 James Martin, Design of Man-Computed Dialogues, Prentice Hall, New York, N. Y., 1973.
- M. N. Matelan, 70 Matthew N. Matelan, "A Method for Conversational to Interpreter Test Language Translation," (paper presented at the Automatic Support Systems Conference, Fort Worth, Texas, 1970).
- M. N. Matelan, 73 Matthew N. Matelan, "A Compiler-Compiler System for Avionic Test Languages," (paper presented at the Automatic Support Systems Conferences, 1973.)
- F. P. Mathur, 70 Francis P. Mathur, and Algiddas Avizienis, "Reliability Analysis and Architecture of a Hybrid Redundant Digital System: Generalized triple Modular Redundancy with Self-repair" (paper presented at the Spring Joint Computer Conference, Los Angeles, Calif., 1970.)

KEY

BIBLIOGRAPHIC LISTING

- R.D. McCafferty, 70 Riley D. McCafferty and Lawrence W. Lockwood, "Apollo Mission Simulation with Visual Presentation", Journal of the SMPTE, Vol. 79, No. 2, February 1970.
- MCAIR, 71 McDonnell Aircraft Company, An Inventory of Aeronautical Ground Research Facilities, by C. J. Pirrello, R. D. Hardin, J. P. Capellupo, and W. D. Harrison, NAS-2-5458, St. Louis, Missouri, Vol. IV, 1 March 1971.
- MCAIR, 72 McDonnell Aircraft Company, Space Shuttle Simulation Definition Report, Vol. I, Part I, MDC E0863, Saint Louis, Missouri, 15 Nov. 1972.
- D. D. McCracken, 72 Daniel D. McCracken and Gerald M. Winberg, "How to Write a Readable Fortran Program," Datamation, Oct 1972.
- G. A. McKenzie, 70 G. A. McKenzie and R. H. Vivian, "Automatic Measurement of Television Waveforms", (paper presented at the I.E.R.E. Joint Conf. on Automatic Test Systems, University of Birmingham, Apr. 70).
- G. A. McKenzie, 74 G. A. McKenzie, "Automated Television Waveform Measurement by Use of a Digital Computer", Journal of the SMPTE, Vol. 83, September 1974.
- J.F. McKeivitt, III, 72 James F. McKeivitt, III, "Parity Fault Detection in Semiconductor Memories", Computer Design, July 1972.
- F. McMains, 67 Picatinny Arsenal, EDIT, A Fortran Program for Renaming Variables in a Source Program, by Forrest McMains, AD 659 340, Dover, N. J., July 1967.
- MDEC, 72 McDonnell Douglas Electronics Co., Space Shuttle Visual Simulation System Design Study, Contemporary Study Report, MSC 06743, October 27, 1972.
- MDEC, 73-1 McDonnell Douglas Electronics Co., Space Shuttle Visual Simulation System Design Study, Final Analytical Report, Report MSC 06742, 30 March 1973.
- MDEC, 73-2 McDonnell Douglas Electronics Co., Space Shuttle Visual Simulation System Design Study, Design Recommendation Report, Report MSC 06741, 20 July 1973.
- M. A. Mehta, 72 Madhukumar A. Mehta, Henry P. Messinger, and William B. Smith, Functions for Improving Diagnostic Resolution in an LSI Environment," (paper presented at the Spring Joint Computer Conference, Holmdel, N. J., 1972).

BIBLIOGRAPHIC LISTING

- { } Merten, 72 Alan Merten and Daniel Teichroew, "The Impact of Problem Statement Languages on Evaluating and Improving Software Performance," (paper presented at the Fall Joint Computer Conference, Ann Arbor, Mich., 1972).
- J. R. Metzner, 67 John R. Metzner and Bruce H. Barnes, "Specialized Software for Digital Simulation Program Production" from Prospects for Simulation of Dynamic Systems, by G. Shapiro and J. Rogers, Spartan Books, 1967.
- R. J. Meyer, 70 Robert J. Meyer, "Software Considerations for Computer Controlled Test Equipment," (paper presented at the Automatic Support Systems Conference, 1970).
- E. F. Meyers, 69 E. F. Meyers, "Automatic Checkout Techniques for Aerospace Digital Computers," (paper presented at the IEEE Automatic Support Systems Conference, Owego, N. Y., 1969)
- R. F. Miles, 73 R. F. Miles, ed., Systems Concepts: Lectures on Contemporary Approaches to Systems, Wiley-Interscience, 1973.
- Barry Miller, 73 Barry Miller, "Navy Aims to Cut A-7E Losses with Inflight Engine Monitoring," Aviation Week & Space Technology, 30 April 1973, pages 82-83.
- T. F. Miotke, 72 Thomas F. Miotke, "Systematic Testing of Boolean Functions" Computer Design, Mar 1972, page 99.
- C. T. Morgan, 63 C. T. Morgan (et. al), ed. Human Engineering Guide to Equipment Design, McGraw-Hill, 1963.
- H. Moriyasn et al, 73 Tektronix, Inc., "The Oscilloscope with Computing Power", by Hiro Moriyasn, Luis Navarro, Jack Gilmore and Bruce Hamilton, TEKSCOPE, March/April 1973.
- W.T. Musial, 73-1 McDonnell Douglas Astronautics Co.-East, Development of Simulation Computer Complex Specification, Computer Complex Hardware Requirements Final Report, by W.T. Musial, C.E. Jones, R.L. Benbow, NASA Contract NAS 9-12882, Report No. MDC E0798, 30 April 1973.
- W.T. Musial, 73-2 McDonnell Douglas Astronautics Co.-East, Development of Simulation Computer Complex Specification, Background Survey Final Report, by W.T. Musial, H.H. Ipolyi, and C.E. Jones, NASA Contract NAS 9-12882, Report No. MDC E0813, 29 June 1973.
- S. Muth, 74 Steve Muth, "Digitizing Video Signals," Electronic Engineering Times, 26 August 1974, pp. 22-23.
- J. Myers, 73 Glenford J. Myers, "Characteristics of Composite Design," Datamation, page 100, 1973.

KEY

BIBLIOGRAPHIC LISTING

-N-

- A.H. Nagler, 71 Farrand Optical Co. Inc., Wide Angle Infinite-Depth-Field Optical Pickup for Visual Simulation, by Albert H. Nagler, Anthony R. Mazurkewitz, USAF AFHRL-TR-71-41, November, 1971.
- A.H. Nagler, 72 A. Nagler, "140° Close Approach Optical Probe for Visual Simulation", Proceedings of the Fifth Naval Training Device Center and Industry Conference, NAVTRADEVCCEN IH-206, February 15-17, 1972.
- NASA, 70 NASA, Asollo Configuration Management Manual, NHB 8040.2, NASA (hq.), Washington, D.C., Jan. 1970.
- NASA, 71 NASA, Computer Program Documentation Guideline, NHB 2411.1, Jul 1971.
- G. W. Nelson, 68 G. W. Nelson, "OPTS-600-On-line Peripheral Test System," (paper presented at the Fall Joint Computer Conference, Phoenix, Arizona, 1968).
- H. P. Nicely, Jr., 66 H. P. Nicely, Jr., "A System for Maintainability Surveillance and Improvement," (paper presented at the Automatic Support Systems Conference, Daytona Beach, Florida, 1966).
- R. L. Nolan, 72 Richard L. Nolan, "Verification/Validation of Computer Simulation Models," (paper presented at the Summer Computer Simulation Conference, Boston, Mass., 1972)
- NTDC, 69 Naval Training Device Center, Proceedings of the Fourth Annual Naval Training Device Center and Industry Conference, NAVTRADEVCCEN IH-173, Orlando, Florida, 18-20 Nov. 1969.
- NTDC, 72 Naval Training Device Center, Proceedings of the Fifth Annual Naval Training Device Center and Industry Conference, NAVTRADEVCCEN IH-206, Orlando, Florida, 15-17 Feb 1972.
- E. B. Nutt, 65 Evan B. Nutt, "Maintenance Recording," (paper presented at the Eleventh National ISA Aerospace Instrumentation Symposium, Ontario, Calif., Oct 1965)
- NYU, 66 New York University, Automation in Electronic Test Equipment, Edited by David M. Goodman, Volume IV, Built-In-Test and Continuous Monitoring, 29 Aug-2 Sept 1966.
- A. O'Brien, 70 Joseph A. O'Brien, "Data Structures for Automatic Test Systems," (paper presented at the IERE Joint Conference on Automatic Test Systems, Univ. of Birmingham, 1970).
- J. L. Ogdin, 72 Jerry L. Ogdin, "Designing Reliable Software," Datamation, July, 1972. C-56

0

BIBLIOGRAPHIC LISTING

- J. L. Ogdin, 73      Jerry L. Ogdin, "Improving Software Reliability,"  
Datamation, Jan 1973.
- G. Oppendeimer, 68      G. Oppenheimer and K. P. Clancy, "Considerations for  
Software Protection and Recovery from Hardware Failures  
in a Multiaccess, Multiprogramming, Single Processor  
System," (paper presented at the Fall Joint Computer  
Conference, 1968).
- P
- E. A. Palmer, 73      E. A. Palmer and F. W. Cronn, "Touchdown Performance With  
a Computer Graphics Night Visual Attachment," (paper  
presented at the AIAA Visual and Motion Simulation  
Conference, Palo Alto, Calif., 10-12 Sept 1973).
- K. Paton, 71      Keith Paton, "An Algorithm for the Blocks and Cutnodes  
of a Graph," Communications of the ACM, Vol. 14, No. 7,  
Jul 1971.
- Philco, Ford, 73      Philco Ford Corporation, "Sierra Electronic Operation,  
Test and Measurement Instruments, SPL-5M-0573, Menlo  
Park, Calif. 1 June 1973.
- J. F. Poage, 64      J. F. Poage, and E. J. McCluskey, Jr., "Derivation of  
Optimum Test Sequences for Sequential Machines," (paper  
presented at the IEEE Switching Circuit Theory Symposium,  
Princeton, N. J., 1964).
- T. W. Pratt, 71      Terrence W. Pratt and Daniel P. Friedman, "A Language  
Extension for Graph Processing and Its Formal Semantics,"  
Communications of the ACM, Vol. 14, No. 7, Jul 1971.
- S. L. Prendergast, 72      S. Lawrence Prendergast, "Selecting a Data Management  
System," Computer Decisions, Aug 1972, pages 12-15.
- Q
- J. T. Quatse, 65      Jesse T. Quatse, "Strobes: Shared-Time Repair of Big  
Electronic Systems," (paper presented at the Fall Joint  
Computer Conference, Pittsburgh, Pennsylvania, 1965).
- R
- RAEs, 71      Royal Aeronautical Society, "Two-day Symposium on Flight-  
Training Simulators for the 70s", October 1970.

BIBLIOGRAPHIC LISTING

- V. Ramamoorthy, 67  
(SJCC ppr) C. V. Ramamoorthy, "A Structural Theory of Machine Diagnosis," (paper presented at the Spring Joint Computer Conference, 1967).
- C. V. Ramamoorthy, 71 V. C. Ramamoorthy, "Fault-Tolerant Computing: An Introduction and an Overview, IEEE Transactions on Computers, Vol. C-20, No. 11, November 1971.
- C. V. Ramamoorthy, 71  
(et al) C. V. Ramamoorthy and W. Mayeda, "Computer Diagnosis Using the Blocking Gage Approach," IEEE Transactions on Computers, Vol. C-20, No. 11, November 1971.
- C. V. Ramamoorthy, 72 C. V. Ramamoorthy, "Criteria for Diagnostic Test Validation," (paper presented at the Spring Joint Computer Conference, 1972).
- R.W. Ramirez, 74 Robert W. Ramirez, "The Fast Fourier Transform's Errors are Predictable, Therefore Manageable," Electronics, 13 June, 1974, pp. 96-110.
- J. C. Richards, 68 Jesse C. Richards, "Automatic Checkout of a Large Hybrid Computer System," (paper presented at the Fall Joint Computer Conference, 1968).
- T. Rhodes, 69 W. J. Rhodes, "The Application of Automatic Test Equipment to Training Devices," (paper presented to the Naval Training Device Center and Industry Conference, 1969).
- W. B. Riley, 73 Wallace B. Riley, "4,096-bit Memories Pose Test Woes," Electronics, December 20, 1973.
- J. P. Rogers, 65 J. P. Rogers & H. W. Thorne, "The Development and Evaluation of an Improved Electronics Troubleshooting Manual," N65-26068, AD 614 606, George Washington University, Mar 65.
- J. Roizen, 73 Joe Roizen, "Color Camera Basics", Broadcast Engineering, July 1973.
- H. J. Rome, 73 H. James Rome, "Automated Degradation Detection and Isolation for Sins" (AIAA Paper No. 73-849, paper presented at the AIAA Guidance and Control Conference, Key Biscayne, Florida, 1973).
- J. P. Roth, 67 J. Paul Roth, Willard G. Bouricius and Peter R. Schneider, "Programmed Algorithms to Compute Tests to Detect and Distinguish Between Failures in Logic Circuits", Vol. EC-16, No. 5, IEEE Transactions on Electronic Computers, October 1967.
- H. Rowe, 70 Ian H. Rowe and Ian M. Kerr, "A Broad-Spectrum Pseudo-random Gaussian Noise Generator," IEEE Transactions on Automatic Control, Vol. AC-14, No. 5, October 1970.

BIBLIOGRAPHIC LISTING

-S-

- S & B Software Products, 72 S & B. Software Products, SVCINTER (debugging aid), Northridge, California, 1972.
- St. Lous Union, 67 St. Louis University, The Aircraft Integrated Data System "ACIDS", DAAJ01-67-C-0733(G), East St. Louis, Illinois, May 1967, Rev. Feb. 1968.
- A. B. Salisbury, 67 United States Military Academy, Diagnostic Programming For Digital Computers: A Bibliography, by Alan B. Salisbury, Technical Report Elec.-67-2, West Point, New York, 1967.
- W. A. Scanga, 69 W. A. Scanga, "Naval Air Rework Facility 5500 Automatic Test Set", (paper presented at the IEEE Automatic Support System Conference, Baltimore, Maryland, 1969).
- G. J. Schick, 72 G. J. Schick, "Assessment of Software Reliability", (paper presented at the 11th Annual Meeting of German Operations, Research Society, Hamburg, Sept. 1972).
- N. F. Schneidewind, 72 Norman F. Schneidewind, "An Approach to Software Reliability Prediction and Quality Control", (paper presented at the Fall Joint Computer Conference, Monterey, Calif. 1972).
- P. B. Schoonmaker, 73/7 McDonnell Douglas Astronautics Company, Simulator Verification Study, by P. B. Schoonmaker, MDC E0861, St. Louis, Missouri, 30 July 1973.
- Sci. Amer., 73/4 "Decisions, Decisions", Scientific American, 1973.
- S. Seshu, 62 S. Seshu and D. N. Freeman, "The Diagnosis of Asynchronous Sequential Switching Systems", IRE Transactions on Electronic Computers, Vol. EC-11, August 1962.
- S. Seshu, 65 Sundaram Seshu, "On An Improved Diagnosis Program", IEEE Transactions on Electronic Computers, Vol. EC-14, Feb. 1965, pp. 76-79.
- L. J. Sevin, Jr., 65 Leonce J. Sevin, Jr., Field-Effect Transistors, New York, McGraw-Hill, 1965.
- H.J. Shechtel, 67 H. J. Shechtel, "An Outline of the History and Development of the Terrier, Tarter and Talos Missile Test Equipments", (paper presented at the Automation in Test Equipment Conference, Washington, D.C., Vol. 3, 1967).

BIBLIOGRAPHIC LISTING

- W. S. Shirley, 69      NASA, Application of a Modified Fast Fourier Transform To Calculate Human Operator Describing Functions,  
By Richard S. Shirley, NASA TM Y-1762, Washington, D.C.
- R. S. Shirley, 73      R. S. Shirley and A. D. Jones, "SAFE: Six Axis Frequency Evaluation of a Motion Simulator", (paper presented at the AIAA Visual and Motion Simulation Conference, Palo Alto, Calif., Sept. 10-12, 1973).
- Sider & Assoc., 73      Joseph Sider and Associates, TXTM, The Text Maintenance System That Can Slash Program Maintenance Costs by Fifty Percent, b Joseph Sider, Encino, California, 9 August 1973.
- Signetics, 72      Signetics Memory Systems, Debug Your Micro-Programs Fast, Mountain View, Calif., Feb. 1973.
- Singer, 73      Singer-General Precision, Inc., Link Division, Skylab Operating Procedures Manual, Rept. Vo. SLS-61, January 1973.
- D. Smith, Jr., 72      Don Smith, Jr., "An Organization For Successful Project Management," (paper presented at the Spring Joint Computer Conference, 1972).
- F. Smith, 66      Everard F. Smith, "A Concept For a Flexible, Automatic Checkout System," (paper presented at the IEEE Automatic Support System, Denver; Colorado, 1966).
- K. C. Smith, 68      Kenneth C. Smith, "A Dual Processor Checkout System", (paper presented at the Fall Joint Computer Conference, Denver, Colorado, 1968).
- E. G. Solov, 73      Edwin G. Solov, and Joseph R. Thibodeau III, "Failure Detection and Isolation Methods For Redundant Gimbaled Inertial Measurement Units", (paper presented at the AIAA Guidance and Control Conference, Key Biscayne, Florida, Aug. 20-22, 1973).
- Solutions, Inc., 73      Solutions, Inc., How to Use Chkstop (debugging aid), Moretown, Vermont, 1973.
- B. Soucek, 72      Branko Soucek, Minicomputers in Data Processing and Simulation, New York, Wiley-Interscience, 1972.
- H. Sriyananda, 73      H. Sriyananda, D. R. Towill, and J. H. Williams, "The Selection of Test Frequencies for System Fault Diagnosis", (paper presented at the American Institute of Aeronautics and Astronautics, Key Biscayne, Florida, Aug. 20-22, 1973).

KEY

BIBLIOGRAPHIC LISTING

- Standard Data Corp, 73      Standard Data Corporation, Symbug (Cobol Debugging Aid), New York, New York, 1973.
- Standard Data Corp, 74      Standard Data Corporation, Standard Data Introduces: Symbug - System For Fortran Symbolic Debugging, New York, New York, 1974.
- K. J. Stein, 74              Kenneth J. Stein, "Initial Versatile Avionic, Shop Test Systems Near Operation", Aviation Week and Space Technology, Feb. 11, 1974.
- D. E. Stepner, 73            NASA Contractor Report, "Maximum Likelihood Identification and Optimal Input Design For Identifying Aircraft Stability and Control Derivatives", by David E. Stepner and Raman K. Mehra, NASA CR-2200, March, 1973.
- L. G. Stucki, 72             Leon G. Stucki, "A Prototype Automatic Program Testing Tool", (paper presented at the Fall Joint Computer Conference, Anaheim, Calif., 5-7 Dec., 1972, MDAC WD 1985).
- L. G. Stucki, 73             Leon G. Stucki, "Automatic Generation of Self Metric Software," (paper presented at 1973 IEEE Symposium on Computer Software Reliability, May 1973).
- Synergetics, 72              Synergetics Corporation, Fas/Test System 360 Testing Aid, Burlington, Mass.
- K. J. Szalai, 71             NASA, "Validation of a General Purpose Airborne Simulator for Simulation of Large Transport Aircraft Handling Qualities," by Kenneth J. Szalai, NASA TN D-6431, Octo. 1971.
- S.A. Szygenda, 69A         S. A. Szygenda and G. C. Goldbogen, "Implementation and Extension of Multi-Dimensional Path Sensitizing In a Simulation and Diagnosis System," (proceedings of the Seventh Annual Allerton Conference on Circuit and Systems Theory, Rolla, Missouri, Oct., 1969).
- S.A. Szygenda, 69E         Stephen A. Szygenda, "A Software System for Diagnostic Test Generation and Simulation of Large Digital Systems," (proceedings of the 1969 National Electronics Conference, Rolla, Missouri, 1969).
- S.A. Szygenda, 70H         S. A. Szygenda and R. J. Smith, II, "A Preprocessor For Software Diagnosis and Simulation of Digital Systems," (Second Annual Houston Conference on Circuits, Systems, and Computers, Houston, Texas, April 1970).

KEY

BIBLIOGRAPHIC LISTING

- Szygenda, 70S Stephen A. Szygenda, David M. Rouse, and Edward W. Thompson, "A Model and Implementation of a Universal Time Delay Simulator for Large Digital Nets," (paper presented at the Spring Joint Computer Conference, Rolla, Missouri, 1970).
- S.A. Szygenda, 71 S. A. Szygenda and M. J. Flynn, "Coding Techniques For Failure Recovery in a Distributive Modular Memory Organization," (paper presented at the Spring Joint Computer Conference, Dallas, Texas, 1971).
- S.A. Szygenda, 71D S. A. Szygenda, "A Simulator For Digital Design Verification and Diagnosis," (from the Second Workshop on Fault Detection and Diagnosis in Digital Systems, Lehigh University, Dec. 1971).
- S.A. Szygenda, 71FT Stephen A. Szygenda, "Problems Associated With the Implementation and Utilization of Digital Simulators and Diagnostic Test Generation Systems," (from Digest, 1971 International Symposium on Fault-Tolerant Computing, Dallas, Texas, 1-3 March 1971).
- S.A. Szygenda, 71R Stephen A. Szygenda and Michael J. Flynn, "Failure Analysis of Memory Organization for Utilization in a Self-Repair Memory System", IEEE Transactions on Reliability, Vol. R-20, No. 2, May 1971.
- S.A. Szygenda, 71W Stephen A. Szygenda, Cliff W. Hemming, and John M. Hemphill, "Time Flow Mechanisms for Use in Digital Logic Simulation," (from the proceedings of the 1971 Winter Simulation Conference, Dec. 1971).
- S.A. Szygenda, 72D S. A. Szygenda, "Anatomy of a General Purpose Test Generation and Simulation System for Digital Logic", (from the proceedings of the ACM-IEEE Design Automation Workshop, Dallas, Texas, June 1972).
- S.A. Szygenda, 72F Stephen A. Szygenda and Edward W. Thompson, "Fault Insertion Techniques and Models for Digital Logic Simulation," (paper presented at the Fall Joint Computer Conference, Dallas, Texas, 1972).
- S.A. Szygenda, 72H S. A. Szygenda, J. Hemphill, C. Hemming, J. Fike, A. Lekkos, E. Thompson, "Implementation of Synthesized Techniques for a Comprehensive Digital Design, Verification and Diagnosis System," (from the proceedings of the fifth Hawaii International Conference on System Sciences, Hawaii, 1972).

KEY

BIBLIOGRAPHIC LISTING

- S.A. Szygenda, 72SC S. A. Szygenda and C. Hemming, "Functional Simulation - A Basis for a Systems Approach to Digital Simulation and Fault Diagnosis", (paper presented at the Summer Computer Simulation Conference, Dallas, Texas, 1972).
- S.A. Szygenda, 72SH S. A. Szygenda, "Diagnostic Test Generation for Digital Logic Containing only Packaged Flip-Flop Memory Elements," (from the Southwestern IEEE Conference, Dallas, Texas, April 1972).
- T-
- W. Teitelman, 72 Warren Teitelman, "Automated Programming - The Programmer's Assistant", (paper presented at the Fall Joint Computer Conference, Cambridge, Mass. 1972).
- Tektronix, 72 Tektronix, S-3260 Software System, by W. E. Kehret, Feb. 1972.
- Tektronix, 73 Tektronix, Transient Digitizer: New Ways to Look at Transients, 1973.
- Tektronix, 74 Tektronix, Inc., 1975 Products, Beaverton, Oregon, 1974.
- Teradyne, 73 Teradyne, Inc., N151 Computer-Operated Backplane Test System, Boston, Mass. 1973.
- Teradyne, 73 Teradyne, Inc., L100 Series Automatic Circuit-Board Test Systems, Boston, Mass., 1973.
- I. Terris, 65 I. Terris and M. A. Melkanoff, "Investigation and Simulation of a Self-Repairing Digital Computer", (paper presented at the IEEE Conference on Switching Circuit Theory and Logical Design, Los Angeles, California, 1965).
- Texas Instruments, 73 Texas Instruments, Inc., ATS-960 Automated Test System, 1973.
- E. W. Thompson, 73 E. W. Thompson and S. A. Szygenda, "The Specification and Generation of Different Fault Classes for Digital Simulation", (paper presented at the Southwestern IEEE Conference, Houston, Texas, 1973).
- J. Toellner & Assoc., 72 J. Toellner and Associates, Fordoc...Automatic Fortran Documentation, Los Angeles, Calif., 1 August 1972.
- Trilog, 73 Trilog Associates, Inc., SMP System Management Processor, Philadelphia, Pa., 1973.

KEY

BIBLIOGRAPHIC LISTING

- TRW, 71 TRW Systems Group, Flight Software Development and Verification System Requirements, by C. G. Gibson, Rpt. No. 99900-H016-RO-00, Houston, Texas, 1971.
- TRW, 72/7 TRW Systems, STS Software Concepts Development Study, Vol. 2: Investigation of Software Problems, Houston, Texas, 31 July 1972.
- TRW, 72 TRW Systems, Space Transportation System Software Concepts Development Study, Vol. 8 Verification, 21455-6009-RO-00, Los Angeles, Calif., 31 July 1972.

-U-

- UCC, 73 University Computing Company, UCC-15 Restart Management System, 3 Oct. 1973.

-V-

- J. E. Valstar, 66 J. E. Valstar, "Some Fundamental Limitations on Indirect Testing of Dynamic Systems", (paper presented at the Aerospace Systems Conference, July 1966).
- F. Van Veen, 71 Frederick Van Veen, "An Introduction to IC Testing", IEEE Spectrum, Dec. 1971.
- J. P. Verma, 72 Jai PraKesh Verma, "Automatic Tester For PC Boards Using Fault Injection Simulation", (paper presented at the Summer Computer Simulation Conference, Pasadena, California, 1972).
- J. P. Vincent, 72 James P. Vincent, Leo F. Polak, and P. S. Fisher, "An Automated System for the Alinement of Large Area Modular Solar Simulators", Space Simulation, NASA S10-298, p. 251, N72-22265, May 1972.

-W-

- M. H. Walshaw, 72 M. H. Walshaw, "Some UK Views on the Development and Procurement of AIDS, BIT and ATE for Avionics", (paper presented at the Automatic Support Systems Conference, 1972).
- P. J. Waples, 67 P. J. Waples, "Discussion on the Techniques of Satellite launcher checkout", Transactions of the Society of Instrument Technology, June, 1967, p. 94.
- C. D. Warner, 72 C. Dudley Warner, "System Performance and Evaluation -- Past, Present, and Future", (paper presented at the Fall Joint Computer Conference, Santa Clara, Calif. 1972).
- G. M. Weinberg, 71 G. M. Weinberg, The Psychology of Computer Programming, Van Nostrand Rheinhold, 1971.

C-64

KEY

BIBLIOGRAPHIC LISTING

- G. M. Weinberg, 72 Gerald M. Weinberg, "The Psychology of Improved Programming Performance," Datamation, Nov. 1972.
- Wescon, 66 "Systems, Space Electronics", (papers presented at the Western Electronics Show and Convention, Part 5, Aug. 23-26, 1966).
- Wescon, 70<sup>---</sup> Marvin L. Luther, "Managing the Development of Large Software Systems", (papers presented at the 1970 Western Electronic Show and Convention, 1970).
- L. A. Whalen, 72 L. A. Whalen and M. P. Gerrity, "Built-In Test (BIT) For Training Devices", (paper presented at the Fifth Naval Training Device Center and Industry Conference, February, 1972).
- R. C. Wingrove, 69 NASA, A Technique For Identifying Pilot Describing Functions From Routine Flight-Test Records, by Rodney C. Wingrove and Frederick G. Edwards, NASA TN D-5127, May 1969.
- R. C. Wingrove, 71 NASA, Comparison of Methods For Identifying Pilot Describing Functions From Closed-Loop Operating Records, by Rodney C. Wingrove, NASA TN D-6235, March, 1971.
- M. Wirth, 71 Miklaus Wirth, "Program Development by Stepwise Refinement", Communications of the ACM, Vol. 14, No. 4, April 1971.
- J. G. Wohl, 65 Dunlap and Associates, Analysis of Man-Computer Roles In Prelaunch Fault Anticipation, by Joseph G. Wohl, Contract No. VASW-954, Darien, Conn., June 1965.
- R. F. Wokasch, 69 Russell F. Wokasch, "A Practical Technique For On-Board Test Adequacy Evaluations", (paper presented at the IEEE Automatic Support System Conference, 1969).
- J. Wolkovitch, 62 Systems Technology, Inc., Performance Criteria For Linear Constant-Coefficient Systems With Deterministic Inputs, by Julian Wolkovitch, AD 275 515; Feb. 1962.
- R. D. Wright, 72 Richard D. Wright, "Validating Dynamic Models, An Evaluation of Tests of Predictive Power", (paper presented at the Summer Computer Simulation Conference, Boston, Mass. 1972).
- L. L. Yaffee, 73 Michael L. Yaffee, "New Maintenance Systems Aid Efficiency", Aviation Week and Space Technology, 22 October 1973.

-Y-

KEY

BIBLIOGRAPHIC LISTING

L. R. Young, 73

L. R. Young, C. M. Oman, R. E. Curry, and J. M. Dichgans, "A Descriptive Model of Multi-Sensor Human Spatial Orientation With Applications to Visually Induced Sensations of Motion", (paper presented at the AIAA Visual and Motion Simulation Conference, Palo Alto, Calif., Sept. 10-12, 1973).

-Z-

H. A. Zimmerman, 72

H. Allen Zimmerman, "Test Fixturing -- The Critical Path", Electronic Products, Beaverton, Oregon, 1972.