AD/A-006 411

DIGITAL FLIGHT CONTROL SYSTEM REDUNDANCY STUDY

John McGough, et al

Bendix Corporation

Prepared for:

Air Force Flight Dynamics Laboratory

July 1974

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

This report has been reviewed and cleared for open publication and/or public release by the appropriate Office of Information (OI), in accordance with AFR 190-17 and DODD 5230.9. There is no objection to unlimited distribution of this report to the public at large, or by DDC to the National Technical Information Service.

This technical report has been reviewed and is approved for publication.

Daniel K. Bird 18 Dec 74
DANIEL K. BIRD
Project Engineer/Technical Monitor

FOR THE COMMANDER

Paul E. Blatt
PAUL E. BLATT
Chief
Control Systems Development Branch
Flight Control Division

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

AIR FORCE/56780/11 February 1975 — 300

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER
AFFDL-TR-74-83

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER
AD/A-006411

4. TITLE (and Subtitle)
Digital Flight Control System
Redundancy Study

5. TYPE OF REPORT & PERIOD COVERED
Final
March 1973 - May 1974

6. PERFORMING ORG. REPORT NUMBER
Technical Prop. AP733

7. AUTHOR(s)
John McGough
Kurt Moses
Walter Platt
Gibson Reynolds
John Strole

8. CONTRACT OR GRANT NUMBER(s)
F33615-73-C-3035

9. PERFORMING ORGANIZATION NAME AND ADDRESS
The Bendix Corporation
Flight Systems Division
Teterboro, New Jersey 07608

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
62201F
1987-01-32

11. CONTROLLING OFFICE NAME AND ADDRESS
Air Force Flight Dynamics Laboratory
Air Force Systems Command
United States Air Force
Wright-Patterson AFB, Ohio 45433

12. REPORT DATE
July 1974

13. NUMBER OF PAGES
341

14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)

15. SECURITY CLASS. (of this report)
UNCLASSIFIED

15a. DECLASSIFICATION DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Flight Control Systems
Fly-by-Wire
Redundancy
Digital
Reliability
Failure Detection
Self-Test
Software
Test Validation

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Redundancy requirements and trade-off criteria are established for flight critical digital flight control systems with particular emphasis on the fly-by-wire application. The use of general purpose digital computers is considered, with self-test and cross-channel comparison monitoring techniques to obtain the necessary flight safety reliability. A reliability model is presented which includes the effects of detected and undetected failures and provides a basis for establishing in-flight and preflight test coverage

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

20. ABSTRACT

requirements consistent with a given reliability goal.

System characteristics that are pertinent to flight safety are discussed in detail. Among these are signal selection and cross-strapping, software, self-test, secondary actuator characteristics, digital computer architecture and I/O organization, equalization, multiplex communications, synchronization and test validation requirements.

FOREWORD

This document is the final report on a study entitled, "Digital Flight Control System Redundancy Study". The work was performed from March, 1973, to May, 1974, by the Flight Systems Division of The Bendix Corporation, Teterboro, New Jersey under Air Force Contract No. 333615-73-C-3035 AFFDL.

The work was administered under the direction of the Air Force Flight Dynamics Laboratory, Wright-Patterson Air Force Base, Ohio, 45433, by Mr. D. Bird, Program Manager.

The principal contributors to this study, which was made under the direction of John McGough, Senior Engineer, are: Kurt Moses, Assitant Chief Engineer, Walter Platt, Assistant Chief Engineer, Gibson Reynolds, Senior Engineer, and John Strole, Senior Engineer, all of the Flight Systems Group of the Bendix Flight Systems Division.

This manuscript was released by the authors in July, 1974.

TABLE OF CONTENTS

TABLE OF CONTENTS (CONCLUDED)

APPENDICES

APPENDICES (CONCLUDED)

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS

APPENDICES

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS

LIST OF TABLES

LIST OF SYMBOLS

LR	Loss rate (losses/flight hour)
AR	Abort note (aborts/flight hour)
F	Event that an LRU fails during a mission
A	Event that an LRU alarms during a mission
$\bar{F}$	Not F
$\bar{A}$	Not A
T	Mission time (hours)
P(E)	Probability of event E
α	$P(\bar{A}\|F)$ = Test defeciency
$1-\alpha$	Test coverage
β	$P(\bar{F}\|X)$ = Nuisance alarm sensitivity
Ƶ	$P(F) = 1-e^{-\lambda T}$
λ	Failure rate (failure/flight hour)
γ_α	Failure rate of the untested portion of an LRU
E_l	Event of loss of system
E_a	Event of mission abort
f_N	Event of a latent failure at the start of the Nth mission
P_N	P (fN)
F_N	Event of a failure of an LRU inflight during the (Nth) mission
A_N	Event of an alarm of the preflight test prior to the (N+1)the mission.
α_i	Inflight test deficiency
$1-\alpha_i$	Inflight test coverage

LIST OF SYMBOLS

α_p	Preflight test deficiency
$1-\alpha_p$	Preflight test coverage
L_N	Event of loss of airplane during the N the mission given that the airplane survived the previous N-1 mission
$P(L\infty)$	$\lim_{n \to \infty} P(LN)$
Q_N	Event that the control system is not operational at the start of or during the Nth mission.
q_N	Union of all failure combinations which are not consistent with event, L_N
MFR	Mean failure rate (average losses/flight hour)
S_N	Event that the airplane failed sometime during the first N missions
q_K	$P(L_K)$
MTFF	Mean time to first failure (hours)
SL	Service life of the airplane (hours)
η_K	Number of airplanes lost during the Kth mission
η	Number of airplanes in sample
N_p	Number of missions between periodic tests of 100% coverage
F_t	Event of failure of the test
Z_t	$P(F_t)$
f	Event of a latent failure

SECTION 1

INTRODUCTION

1. Introduction

The subject of this study is "Redundancy" in digital flight control systems. One of the objectives of the study is to identify those characteristics of the digital computer which tend to improve or lessen mission and flight safety reliability and to suggest requirements and design and validation procedures which will insure compliance with these objectives without compromising performance. In this context the following specific areas (among others) were considered:

a. Failure detection capability of the digital computer

b. The effects of undetected failures

c. Inflight and preflight test requirements

d. Flight safety evaluation criteria

e. Reduction in the number of redundant channels through improved failure detection

f. Techniques of signal selection as a means to improve flight safety reliability

g. Isolation, buffering and I/O requirements

h. Validation of test procedures

i. Multiplexed communications

Unfortunately, time did not permit the inclusion of the important topic of survivability and the effects of battle damage.

Throughout the study, emphasis was placed on identifying general problem areas and formulating design data rather than on proposing solutions to specific problems. The justification for this approach is that there is hardly any task in the flight control application which is not specific to a particular set of conditions; i.e., noise environment, configuration, mission and reliability objectives, etc. As a consequence, a solution in one situation may be invalid in another. There is another area in which a certain restraint is desirable and that is when imposing requirements to insure that a particular objective is achieved. Too frequently such requirements are based on inadequate data, and therefore could become an impediment to a good design or could tend to replace sound engineering judgment. It is therefore hoped that, in those few instances when general requirements have to be proposed, due consideration is given to alternatives which may be better for specific applications.

SECTION 2

SUMMARY

2. Summary

The major areas of investigation are summarized in the following paragraphs.

Section 3

- Mission and flight safety reliability goals are established based on field data of existing military and commercial aircraft.
- Failure detection capability of a test is defined in terms of test coverage and sensitivity to nuisance alarms. In terms of these parameters, the failure detection requirements of a redundant configuration can be specified.

Section 4

- Ground rules are established for the tradeoffs of redundant configurations, including those characteristics of secondary actuators and signal selection devices which are pertinent to the study.
- The effects of combinations of detected and undetected failures and nuisance alarms on several candidate redundant configurations are discussed.
- Abort strategies are defined and abort rate computed for each candidate configuration.

Section 5

This section contains the results of the flight safety reliability tradeoffs of the candidate configurations.

Section 6

The techniques and methods of the previous sections are applied to the longitudinal axis of the 680-J airplane using F-4 component reliability data.

Section 7

Pertinent differences between analog and digital implementation of a FBW FFCS are discussed.

Section 8

Based upon results of the study additional requirements for inclusion in MIL-STD-9490D are recommended.

Section 9

Conclusions and recommendations for future action are presented.

3. Appendices

The Appendices generally contain either detailed mathematical derivations, reference and supporting data, or subject matter which, although important from the point of view of redundancy, was not considered appropriate for the main text. Included in this latter category are discussions of:

a. Redundant Secondary Actuators

b. Signal Selection and Monitoring

c. Self Test Considerations

d. Multiplex Communications

e. Test Validation Considerations

SECTION 3

EVALUATION CRITERIA FOR REDUNDANCY STUDIES

1. Design Goals Established

In the following chapters, tradeoff studies of digital flight control configurations will be reported. It is assumed that the control system is flight critical and its loss would result in loss of the airplane. In particular, the intended application is either a fly-by-wire (FBW) primary flight control system (PFCS) or a control and stability augmentation system (CAS/SAS) for an aircraft that could be statically or dynamically unstable in certain portions of its flight regime.

In these studies, configurations are evaluated from the point of view of mission and flight safety reliability. Other factors, perhaps equally important, were considered to the extent that they imposed constraints on the candidate configurations. Cost, size, weight, power, maintainability, survivability and reliability are some of these factors.

a. Flight Safety Reliability Goals

The following estimates of flight safety reliability (and mission reliability in the next section) were obtained from surveys of military fighter and cargo aircraft in the time period 1960-1973 and commercial aircraft in the time period 1950-1960. The estimates are given in terms of loss rates (designated LR=losses/flight hour) involving either all flight controls or primary flight controls. The flight controls category includes:

- primary flight control
- secondary flight control
- automatic flight control
- hydraulic and electrical power supplies

The primary flight controls include:

- rudder, aileron, elevator (stabilator) actuators
- control linkages
- feel and trim system

In a survey (Ref. 1) of several types of naval fighter aircraft (e.g., F-4, F-8, A-5, A-6, A-7) in the time period 1960-1970, the following estimates are given:

Flight Controls

$LR = 11.6 \times 10^{-6}$ (averaged over all aircraft types)

$LR = 10.35 \times 10^{-6}$ (for the F-4)

Primary Flight Controls

$LR = 5.5 \times 10^{-6}$ (averaged over all aircraft types)

$LR = 6.6 \times 10^{-6}$ (for the F-4)

The loss rates involving thc PFCS weie attributed to either the power actuators or control linkages, the estimated average distribution being

Power Actuator:	$LR = 3.2 \times 10^{-6}$
Control Linkage:	$LR = 2.3 \times 10^{-6}$
Total	$LR = 5.5 \times 10^{-6}$

The cited estimates include the additional hazard of carrier operations. When losses are deleted which could be attributed to the carrier environment, the resultant estimate is

$LR = 4.63 \times 10^{-6}$ (for the F-4)

as compared with

$LR = 6.6 \times 10^{-6}$ for carrier operations.

In another survey (Ref. 2) of USAF aircraft (e.g., F-4, F-101, F-111) in the time period 1966-1970 the following estimates are given:

Flight Controls (Excluding Hydraulic and Electrical Power Supplies)

$LR = 30.0 \times 10^{-6}$ (averaged over all aircraft types)

$LR = 5.8 \times 10^{-6}$ (for the F-4)

Primary Flight Controls

$LR = 13.7 \times 10^{-6}$ (averaged over all aircraft types)

$LR = 3.8 \times 10^{-6}$ (for the F-4)

It is interesting to note the loss rate due to all causes. From data supplied by Tactical Air Command covering all types of military aircraft the estimates are:

$LR = 120.0 \times 10^{-6}$ (for fighter aircraft)

$LR = 20.0 \times 10^{-6}$ (for cargo aircraft)

for the time period 1966-1970. Supporting data is given in Ref. 14. There the loss rate due to all causes, for the year 1967, is

$LR = 140.0 \times 10^{-6}$

(averaged over 7 types of fighter aircraft)

$LR = 141.0 \times 10^{-6}$ (for the F-4).

An estimate of the loss rate of the present F-4 primary flight control system, longitudinal axis, is

$LR = 1.145 \times 10^{-6}$

assuming a stabilator actuator failure rate of 1.0×10^{-6} failures/hour. This estimate includes hydraulic and electrical power supplies and current equipment failure rates.

Footnote: In a survey of commercial aircraft(Ref. 16) in the time period 1950-1960 the loss rate of the PFCS is estimated to be

$LR = 0.23 \times 10^{-6}$.

Reference 19 (MIL-F-9490D User's Guide) cites the following estimates:

Flight Controls (Including Hydraulic and Electrical Power Supplies)

LR = 0.55×10^{-6} (averaged over B-52, C-135, C-141 aircraft, 1964-1973)

LR = 8.97×10^{-6} (F-4, 1960-1970)

LR = 2.88×10^{-6} (rotary wing aircraft averaged over H-1, H-3, H-43, H-53)

Summarizing these estimates and making allowances for improvement in equipment the following projection is considered a reasonable goal for flight safety reliability of a primary flight control system which includes hydraulic and electrical power supplies:

LR = 3.0×10^{-6} (for fighter aircraft)

b. Mission Reliability Goals

In Ref. 2 mission reliability estimates are given in terms of in-flight abort rate (AR = Aborts/Flight Hour) for the referenced aircraft. Since aborts are not normally reported as such when accidents occur on the homeward leg of the mission, the following estimates have been modified (by the factor 1.5) to reflect the rate throughout the entire mission:

Flight Controls (Excluding Hydraulic and Electrical Power Supplies)

AR = $2{,}295.0 \times 10^{-6}$ (averaged over all aircraft types)

AR = $1{,}710.0 \times 10^{-6}$ (for the F-4)

Primary Flight Controls

AR = 450.0×10^{-6} (averaged over all aircraft types)

AR = 420.0×10^{-6} (for the F-4)

In Ref. 1 the estimated abort rate is

AR = 165×10^{-6} (F-4, Navy)

Loss rate data is summarized in Table 1.

TABLE 1. SUMMARY OF LOSS RATE/FLIGHT HOURS

	COMMERCIAL 1950 - 1960	USAF 1966 - 1970		MACAIR PRESENT	NAVY 1960 - 1970		
	PFC	All Flight Controls	PFCS	PFCS* Long. Axis Only	All Flight Controls*	PFCS	
AVERAGE	0.23 x 10^{-6}	30.0 x 10^{-6} Abort Rate =2,295 x 10^{-6}	13.7 x 10^{-6} Abort Rate =450 x 10^{-6}		11.6 x 10^{-6}	5.5 x 10^{-6}	Power Actuator 3.2 x 10^{-6} Control Linkage 2.3 x10^{-6}
F-4		5.8 x 10^{-6} Abort Rate =1710. x 10^{-6}	3.8 x 10^{-6} Abort Rate =420. x 10^{-6}	1.145 x 10^{-6} Stabilator Actuator =1.0 x 10^{-6}	10.35 x 10^{-6} 6	6.6 x 10^{-6} Abort Rate =165. x 10^{-6}	

*Includes Hydraulic and Electrical Power Supplies

USAF AVG: F-4, F-101, F-111

NAVY AVG: F-4, F-8, A-5, A-6, A-7

	TAC 1966 - 1970 Due to All Causes	MIL-STD-8785 1967 Due to All Causes
AVERAGE	120. x 10^{-6} (Fighter) 20.0 x 10^{-6} (Cargo)	140 x 10^{-6}
F-4		141. x 10^{-6}

		MIL-F-9490D (DRAFT) USER'S GUIDE ALL FLIGHT CONTROLS *
AVERAGE,	Heavy Military Transport B-52, C-135, C-141 1964-'1973	.55 x 10^{-6}
F-4	1960 - 1970	8.97 x 10^{-6}
AVERAGE,	Rotary Wing H-1, H-3, H-43, H-33	2.88 x 10^{-6}

2. Inflight and Preflight Test Coverage

Of the many elements which influence flight safety reliability, the following are primary:

- Component Reliability
- Configuration (Redundancy, end-to-end, etc)
- Failure Detection Capability

At the present time the reliability of a non-redundant FBW PFCS is not sufficient to achieve the reliability goals established in the previous section. Excluding sensors and primary actuators, the combined failure rate of digital controller and a secondary actuator would probably exceed 300 x 10^{-6} failures/hour. Because of this deficiency of the basic components it is necessary to resort to redundancy techniques to improve system reliability.

Regardless of the levels of redundancy, every redundant configuration inherently depends upon some form of failure detection and subsequent removal or rerouting of failed components either before or during each mission. One of the objectives of this study is to define a measure of failure detection capability which can be used to specify failure detection requirements for a given redundant system and reliability goal and to show to what extent system reliability is compromised by non-perfect failure detection.

The basic unit of the system is the LRU (Line Replaceable Unit) which for purposes of this discussion, is the smallest field-replaceable system element. Associated with each LRU is a failure detection device whose function is to alarm if the LRU does not conform to some model characteristics. The LRU is assumed to consist of a large number of components, each with a small probability of failing during a mission of duration, T. The LRU is considered to have failed when at least one component fails. Finally, it is assumed that failures of all components including the LRU are Poisson* distributed in time.

* See Appendix III

Define:

F = Event that the LRU fails during the mission

A = Event that the LRU alarms during the mission

$\bar{F}$ = Not F

$\bar{A}$ = Not A

The probability model consists of the events F, $\bar{F}$, A, $\bar{A}$, F$\bar{A}$, FA, FA, $\overline{FA}$ together with their probabilities of occurrence during the mission. In this context,

F$\bar{A}$ is an undetected failure.

$\bar{F}$A is a nuisance alarm,

FA is a detected failure.

According to this model the occurrence of a failure and an alarm during the mission is a detected failure regardless of their order of occurrence or the time interval between failure and alarm. The alarm device is a failure detector and annunciator which may be either dedicated hardware, as with a comparator, or a self-test soft ware program or a combination of both.

Details of the following discussion can be found in Appendix III.

Define

$$\alpha = P(\bar{A}|F) = P(F\bar{A})/P(F) \qquad (1)$$

$$\beta = P(\bar{F}|A) = P(\bar{F}A)/P(A) \qquad (2)$$

i.e., α, β are the conditional probabilities of $\bar{A}$ given F and $\bar{F}$ given A, respectively.

It is shown in Appendix III that the following relationships apply:

$$P(F\bar{A}) = \alpha z \qquad (3)$$

$$P(FA) = (1-\alpha)z \qquad (4)$$

$$P(\bar{F}A) = \frac{\beta\ (1-\alpha)}{1-\beta}z \qquad (5)$$

$$P(\overline{FA}) = 1-\frac{(1-\alpha\ \ \beta\ )}{1-\beta}z \qquad (6)$$

$$P(A) = \frac{1-\alpha}{1-\beta z} \qquad (7)$$

where z = P(F).

From the above expressions it can be seen that the probabilities of the events $F\bar{A}$, $\bar{F}A$, FA, $\overline{FA}$, A can be obtained in terms of the three parameters α, β, P(F). These parameters may be selected independently* subject only to the constraint imposed by the inequality

$$0 < \left(\frac{1-\alpha}{1-\beta}\right)z + \alpha z < 1. \qquad (8)$$

The quantities, α and β, are measures of the failure detection capability of a test relative to the LRU being tested. The quantity β is a measure of the sensitivity to nuisance alarms and it is desirable that β be small for a given test. The quantity α, however, does not reflect the detection capabilities of the test depending, as it does, on the interaction between nuisance alarms and alarms which are the result of detected failures. Thus, a small value of α is not necessarily a good indicator of detection capability. This is not surprising since the probability model does not distinguish between causal

* α and β may be functionally related, depending upon the detection procedure.

and non-causal alarms.* However, in the complete absence of nuisance alarms; i.e., $\beta = 0$, α is equal to the ratio of undetected to total failures, assuming that all failures are equiprobable** In this case α is called the test deficiency and $1-\alpha$ is called the test coverage. Observe that $1-\alpha$ is equal to the ratio of detected to total failures. It is shown in Appendix III tnat, if the mission time is sufficiently small, then

$$\text{Test Deficiency} = P(\bar{A}|F), \text{ approximately.}$$

*Causal alarm = an alarm caused by a failure.

**If λ = failure rate of the LRU and

λ_α = failure rate of that portion of the LPU which is not tested then

$$\text{Test deficiency} = \frac{\lambda_\alpha}{\lambda}$$

if the mission time is sufficiently small.

a. Applying the Probability Model

In the context of estimating the flight safety reliability of a redundant control system the procedure for applying the probability model is:

(1) Determine α , β and P(F) for each LRU. This presumes that a test procedure exists for each LRU.

(2) Define the event, E, of the loss of the airplane or loss of system, as the case may be, in terms of the events F, $\bar{F}$, A, $\bar{A}$, $F\bar{A}$, $\bar{F}A$, FA, $\overline{FA}$ for each system LRU. The event, E, is application dependent and will differ for each configuration, servo characteristics, etc.

(3) Compute P(E).

Implicit in this procedure is the assumption that α and β can be determined for each LRU independently of any other LRU or even of the configuration itself. It is recognized that this assumption may not always be valid for certain kinds of tests, notably comparison monitoring, where an upstream failure could prevent detection of failures downstream or where a failure in one channel could seriously degrade coverage in the other channels. While such characteristics are undesirable in any test and should be avoided whenever possible, it is necessary to include such considerations in the evaluation of a given test.

b. Example

Consider a dual, standby configuration consisting of a single actuator commanded by one of two computers. In the event that the command computer fails the standby computer is switched onto the driving channel. Assume that

(1) the servo has a zero failure rate,

(2) each computer is in a non-failed state at the start of the mission,

(3) the standby computer is powered throughout the mission,

(4) loss of system occurs when

$$E_{\ell} = A_1F_2+F_1\bar{A}_1$$

(5) a mission abort occurs when

$$E = (A_1+A_2)\ \bar{E}_{\ell}$$

where the subscripts 1 and 2 designate the active and standby channels, respectively. The probability of loss of system is

$$P(E_\ell) = P(A_1)P(F_2) + P(F_1\bar{A}_1)$$

$$= \frac{1-\alpha}{1-\beta} z^2 + \alpha z$$

where

$$z = z_1 = z_2 = P(F_1) = P(F_2) = 1-e^{-\lambda T}$$

In the absence of nuisance alarms, i.e., $\beta = 0$,

$$P(E_\ell) = (1-\alpha)z^2 + \alpha z.$$

In order to estimate inflight test requirements assume

(a) $z = 300 \times 10^{-6}$, which is a typical single channel failure rate, and

(b) two thirds of the flight safety reliability goal of 3.0×10^{-6} for fighter aircraft is allocated to the servos. Then, in order to meet the flight safety reliability goal, it is necessary that

$$\alpha z = \alpha \times 300 \times 10^{-6} \leq 1.0 \times 10^{-6}$$

or $\alpha \leq .00333...$

or $1-\alpha \geq .99666...$

i.e. 99.66% of all failures must be detected.

It should be noted that in the active/standby arrangement inflight failures must be detected and acted upon almost immediately as they occur in order to prevent the failure transients from propagating to the surfaces. This imposes a severe additional requirement on inflight test. The effects of nuisance alarms on loss of system can be obtained by setting $\alpha = 0$. Thus

$$P(E_\ell) = \frac{z^2}{1-\beta} .$$

Obviously nuisance alarms have a neglible effect on loss of system in this example. However, nuisance alarms have a very significant effect on mission aborts. From the event of mission abort, the probability is seen to be

$$P(E_a) = 2\,\frac{1-\alpha}{1-\beta}\,z, \text{ approximately.}$$

If 1 out of every 2 alarms is a nuisance alarm then

$$P(E_a) \cong 4z = 4 \times 10^{-4} = 400 \times 10^{-6} \quad (\text{if } \alpha \text{ is small})$$

which is approximately the abort rate for the F-4.

3. Latent Failures

In estimating the probability of success of a given mission two types of failures must be considered:

a. Inflight Failures: Failures which occur during the mission,

b. Latent Failures: Failures which occurred previously and were not removed or detected by inflight monitors on successive applications of preflight tests.

Latent failures can be subdivided into active and passive. Active failures directly affect a computation in the signal chain and are presumed to have failed the entire LRU. Passive latent failures do not directly affect the signal chain unless accompanied by additional, and possibly remotely occurring, failures or even system states. Examples of such failures would include limiters, states or state transition paths of MSI devices such as random access memories, inflight monitors, ground test equipment etc. The effects of passive latent failures on flight safety reliability are difficult to establish since such failures can exist simultaneously in all channels of a redundant system without adversely affecting system operation. As a consequence, a reliability model which includes the effects of passive latent failures does not appear to be feasible. For purposes of this study all latent failures are presumed to be active. This approach, although somewhat unrealistic, is at least conservative.

While some redundant configurations are less sensitive to latent failures than others, latent failures tend to compromise flight safety reliability in all configurations. The extent of this compromise will be determined in subsequent sections. We proceed now to derive an expression for the probability of a latent failure of an LRU.

Here and throughout the remainder of the report it will be convenient to distinguish between inflight monitoring and preflight tests. Inflight monitoring is performed during the mission and for the purpose of removing failures in order to reduce failure transients or to improve the benefits of cross strapping. Preflight test is administered on the ground and before every mission for the purpose of detecting latent failures. It is desirable, at least from the operations point of view, that preflight test be built-in.

The major system components whose failure must be detected either inflight or in preflight test are

- sensors
- digital computers
- actuators
- displays and controls
- monitoring, testing and disengage devices
- communications paths
- redundant system - associated-components such as signal selection devices, intercomputer links, etc.

With the possible exception of the displays, an undetected failure of any of these components could seriously compromise the operational capability and safety of the aircraft.

According to the assumption which regards the LRU as the smallest field replaceable system element, a detected failure of any component within an LRU will cause the entire LRU to be replaced. As a consequence, a latent failure will be removed if the failure is detected or if some other failure of the LRU occurs and is detected. Regarding the existence and detection of latent failures, we make the following additional assumptions:

- The existence of a latent failure of an LRU does not impair detection of subsequent failures provided that it was not the test or alarm mechanism that failed.

- A failure, once undetected, will remain undetected no matter how frequently the test is administered.

This latter assumption tends to be more valid for computer self test than for comparison monitoring. In any case it is a conservative assumption.

Let f_N = Event of a latent failure at the start of the Nth mission.

F_N = Event of an inflight failure during the N-1 th mission

A_N = Event of an alarm of the preflight test prior to the N+1 mission.

The preflight test may incorporate inflight monitoring. Thus, A_N may include inflight monitoring during the Nth mission.

A latent failure at the start of the Nth mission can occur if and only if

$$f_N = \bar{F}_{N-1} f_{N-1} + (F_{N-1} \bar{A}_{N-1}) \bar{f}_{N-1} + \left(F_{N-1} \bar{A}_{N-1}\right) f_{N-1} \qquad (9)$$

In other words, a latent failure can occur at the start of the Nth mission if and only if

- A latent failure existed at the start of the (N-1) th mission and no inflight failure occurred, or

- No latent failure existed at the start of the (N-1) th mission and an inflight failure occurred and was not detected, or

- A latent failure existed at the start of the (N-1) th mission and an inflight failure occurred and was not detected.

Taking probabilities of both sides of (9) yields

$$\begin{aligned} P(f_N) &= P(\bar{F}_{N-1})P(f_{N-1}) \\ &\quad + P(F_{N-1}\bar{A}_{N-1})\, P(\bar{f}_{N-1}) \\ &\quad + P(F_{N-1}\bar{A}_{N-1})\, P(f_{N-1}) \\ &= (1-z)\, P(f_{N-1}) + \alpha_p z \end{aligned} \tag{10}$$

where $z = P(F_{N-1})$ and α_p is the preflight test deficiency. Solving difference equation (10) for the initial condition $P(f_1) = 0$ yields

$$P(f_N) = \alpha_p \left[1-(1-z)^{N-1}\right] \tag{11}$$

as the probability of a latent failure at the start of the Nth mission. Observe that, since $z = 1-e^{-\lambda T}$,

$$P(f_N) = \alpha_p \left(1-e^{-\lambda(N-1)T}\right) .$$

Thus, $P(f_N)$ approaches α_p exponentially with a time constant equal to $1/\lambda$ hours. If the LRU incorporates the whole channel then $z = 10^{-6}$, approximately, and $1/\lambda = 3,333$ hours. Because of the existence of latent failures the probability of loss of airplane will be a function of elapsed operational time. Define

L_N = Event of loss of airplane during the Nth mission given that the airplane survived the previous N-1 missions.

The probability, $P(L_N)$, is the primary measure of flight safety. However, before evaluating $P(L_N)$ it is necessary to obtain the connection between L_N and the event of loss of system. Let

Q_N = Event that the control system is not operational at the start of or during the Nth mission.

The event, Q_N is configuration dependent and will consist of all failures, detected now and undetected earlier, which render the configuration non-operational. Some of these failure combinations, however, are not consistent with the premise that the airplane survived the previous missions. These combinations will involve the number and location of latent failures. Let q_N denote the union of those failure combinations which are not consistent with this premise. Then we define

$$P(L_N) = P(Q_N | \bar{q}_N).$$

This equation relates loss of airplane to loss of system given that the airplane survived the previous missions.

Example

Consider a triplex configuration with no inflight monitoring and assume that the digital computers are the only system LRU'S with a non-zero failure rate and that loss of system occurs when two or more channels fail. Then

$$Q_N = (f_{1N}+F_1)(f_{2N}+F_2) + (f_{1N}+F_1)(F_{3N}+F_3) + (f_{2N}+F_2)(f_{3N}+F_3) \qquad (12)$$

where

f_{KN} = Event of a latent failure of the Kth channel,

F_K = Event of an inflight failure of the Kth channel.

Observe that, if $P(F_K) = z$, then

$$P(Q_N) = 3z^2 - 2z^3 \qquad (13)$$

in the absence of latent failure, as expected. It is apparent that the only permissible combinations of latent failure are

$$a_N = f_{1N}\bar{f}_{2N}\bar{f}_{3N}$$

$$b_N = \bar{f}_{1N}f_{2N}\bar{f}_{3N}$$

$$c_N = \bar{f}_{1N}\bar{f}_{2N}f_{3N}$$

$$d_N = \bar{f}_{1N}\bar{f}_{2N}\bar{f}_{3N}$$

Thus

$$\bar{q}_N = a_N + b_N + c_N + d_N$$

and

$$P(L_N) = P(Q_N|\bar{q}_N) = P(Q_N|a_N)\frac{P(a_N)}{P(\bar{q}_N)} + P(Q_N|b_N)\frac{P(b_N)}{P(\bar{q}_N)} \qquad (14)$$

$$+ P(Q_N|c_N)\frac{P(c_N)}{P(\bar{q}_N)} + P(Q_N|d_N)\frac{P(d_N)}{P(\bar{q}_N)}$$

since a_N, b_N, c_N, d_N are mutually exclusive.

From (12) it can be seen that

$$P(Q_N|a_N) = P(F_2 + F_3) - 2z - z^2 \qquad (15)$$

$$P(Q_N|b_N) = P(F_1 + F_3) = 2z - z^2$$

$$P(Q_N|c_N) = P(F_1 + F_2) = 2z - z^2$$

$$P(Q_N|d_N) = P(F_1F_2 + F_1F_3 + F_2F_3) = 3z^2 - 2z^3$$

From the expression (11) for the probability of a latent failure

$$P(a_N) = P(b_N) = P(c_N) = p_N(1 - p_N)^2 \qquad (16)$$

$$P(d_N) = (1 - p_N)^3$$

$$P(q_N) = 3p_N(1 - p_N)^2 + (1 - p_N)^3$$

where $p_N = P(f_{KN})$.

Substituting these quantities into (14) yields

$$P(L_N) = \frac{3(2z - z^2)\, p_N (1 - p_N)^2 + (3z^2 - 2z^3)(1 - p_N)^3}{3\, p_N (1 - p_N)^2 + (1 - p_N)^3} \qquad (17)$$

$$= \frac{3(2z-z^2) + (3z^2-2z^3)(1-p_N)}{1 + 2\, p_N}$$

Observe that

$$P(L_N) = 3z^2 - 2\,z^3 \text{ when } \alpha_p = 0$$

If the service life of the airplane is large compared with the time constant $1/\lambda$ where $z = 1-e\ \lambda T$ then we can replace P_N by α_p. If, in addition, $\alpha_p << 1$, then

$$P(L_N) \cong 3(2z - z^2)\, \alpha_p + (3z^2 - 2z^3) \qquad (18)$$

$$\cong 6\alpha_p z + 3z^2$$

after a large number of missions have elapsed. Typically, $\lambda = 100 \times 10^{-6}$ for a one hour mission. For a commercial jet aircraft with a service life of 60,000 hours the approximation of (18) is valid since the latent failure time constant is 10,000 hours. If 0.1×10^{-6} of the 0.23×10^{-6} goal for commercial aircraft is allocated to computers then, in order to meet the flight safety reliability goal, it is necessary that

$$6\,\alpha_p \times 10^{-4} + 3 \times 10^{-8} \leq 0.1 \times 10^{-6}.$$

Solving for α_p yields

$$\alpha_p \leq 0.0001166$$

i.e., the preflight test coverage must be better than 0.9998 (i.e., 99.98% of all failures must be detected). Equivalently, the failure rate of the untested equipment must be less than 1.166×10^{-8} per hour.

In practice the expression for $P(L_N)$ of (14) can be simplified considerably. If

$$\alpha_p << 1$$

then we may use the approximation

$$\frac{1}{P(q_N)} \cong 1.$$

Substitution into (14) yields

$$P(L_N) = P(Q_N \cdot a_N) + P(Q_N \cdot b_N) + P(Q_N \cdot c_N) + P(Q_N \cdot d_N) \quad (19)$$

approximately.

Henceforth, in order to distinguish between inflight and preflight test deficiencies, the former will be denoted by α_i and the latter by α_p.

At this point we summarize the successive development of the reliability model. For this purpose consider the triplex configuration of the previous example except that each channel is self-monitored inflight.

Model #1

The simplest model is based on the following assumptions:

(1) 100% inflight coverage

(2) No latent failures at the start of each mission

(3) No nuisance alarms.

Accordingly, the probability and loss of system is

$$Z^3 = 10^{-12} T^3, \text{ approximately.}$$

Model #2

This model is based on assumption (2) and (3). Then, the probability of loss of system is

$$Z^3 + 6\alpha_i Z^2 = 10^{-12}T^3 + 6\alpha_i x10^{-8} \ x \ T^2, \text{ approximately.}$$

Model #3

In this model only assumption (3) is retained. As a consequence, the probability of loss of system is

$$Z^3 + 6\alpha_i Z^2 + 6\alpha_p \left[1-\bar{e}^{.0001 \ (N-1)T}\right] Z =$$

$$10^{-12}T^3 + 6\alpha_i x10^{-8}T^2 + 6\alpha_p \left[1-\bar{e}^{.0001(N-1)T}\right] T \ x \ 10^{-4}$$

approximately.

A comparision of these models indicate that the successive additional terms could easily dominate the preceding terms. Thus, a flight safety reliability estimate based on model #1 or even model #2 could be excessively optimistic.

4. Alternate Measures of Flight Safety Reliability

In the presence of latent failures the probability, $P(L_N)$, is a function of mission duration and number of elapsed missions. In this case there is an ambiguity in the meaning of flight safety reliability since the probability of a safe mission is time dependent. Several options are available:

a. Require that

$$\frac{P(L_N)}{T} \le \text{flight safety reliability goal for all } N.$$

This is a valid criterion for a commercial aircraft whose service life is well in excess of the latent failure time constants of the system LRU's.

b. Require that

$$\frac{P(L_N)}{T} \le \text{flight safety reliability goal for } NT = \text{service life of the airplane.}$$

This criterion insures that $\frac{P(L_N)}{T}$ will never be less than the goal.

While sufficient, the criterion is not necessary in order to meet the reliability goals as estimated from field data.

c. Require that some "average" value of $\frac{P(L_N)}{T}$ be less than the flight safety reliability goal.

The average (mean failure rate) is defined to reflect the way in which reliability estimates are obtained from field data, i.e., the number of aircraft losses divided by the number of flight hours of the sample.

Options 1 and 3 will be used in the tradeoff studies to follow. An expression for the mean failure rate will now be derived.

Define

S_N = Event the airplane failed sometime during the first N missions.

Thus

$P(\bar{S}_{K-1})$ = probability that the airplane survived the first K-1 missions, $P(\bar{S}_o) = 1$.

$P(L_K) = P(S_K \mid \bar{S}_{K-1})$

$P(L_K)\, P(\bar{S}_{K-1})$ = probability of loss of the airplane during the Kth mission

$P(\bar{S}_{K-1}) = (1-q_1)(1-q_2)\dots(1-q_{K-1})$

where

$q_K = P(L_K)$, $q_o = 0$.

Accordingly, the mean time to first failure for a single airplane is

$$MTFF = \sum_{K=1}^{\infty} KT\, q_K(1-q_1)\,(1-q_2)\dots(1-q_{K-1}) \qquad (20)$$

Observe that if

$q_K = q$ = constant

then

$MTFF = \frac{T}{q}$, as expected.

The **MTFF** (or its reciprocal) is not a particularly desirable criterion of flight safety because a) it requires a very large number of computations to evaluate and b) a typical MTFF greatly exceeds the service life of the airplane and c) it bears little resemblance to the way in which reliability estimates are obtained from field data.

d. Mean Failure Rate

An alternate measure, the mean failure rate, is defined as follows: Define R as the ratio of airplane losses to total flight time in a sample of n airplanes. Thus

$$R = \frac{\sum_{K=1}^{N} n_K}{\sum_{K=1}^{N} KT\, n_K + NT\,(n - \sum_{K=1}^{N} n_K)} \qquad (21)$$

where n_K = number of airplanes lost during the Kth mission.

$\sum_{K=1}^{N} KT\, n_K$ = total flight time of all airplanes which failed during the service life.

$NT = SL$ = service life

T = duration of mission

$(n - \sum_{K=1}^{N} n_K)$ = number of airplanes which reached the end of the service life.

Because the events n_i and n_K, $i \neq K$, are independent, the expected value of R is the ratio of expected values. Thus,

$$E(R) = \frac{\sum_{K=1}^{N} E(n_K)}{\sum_{K=1}^{N} KTE(n_K) + NT\left(n - \sum_{K=1}^{N} E(n_K)\right)} \qquad (22)$$

We interpret the Kth mission as a Bernoulli trial with

p_K = probability of loss of airplane.

Therefore, the average number of losses during the Kth mission is

$$E(n_K) = n\, p_K \qquad (23)$$

and the average flight time is

$$E(KT\ n_K) = KT\ n\ p_K. \qquad (24)$$

Substituting (23) and (24) into (22) yields

$$E(R) = \frac{\sum_{K=1}^{N} n\ p_K}{\sum_{K=1}^{N} KT\ n\ p_K + NT\left(n - \sum_{K=1}^{N} n\ p_K\right)} \qquad (25)$$

$$= \frac{\sum_{K=1}^{N} p_K}{\sum_{K=1}^{N} KT\ p_K + NT\left(1 - \sum_{K=1}^{N} p_K\right)}$$

We define

MFR (Mean Failure Rate) = E(R).

<u>Example</u>

For the case when $P(L_K) = q =$ constant, we certainly expect that

$$\frac{1}{MTFF} = MFR$$

where MTFF is computed according to (20)

From (20)

$$MTFF = \sum_{K=1}^{\infty} KT\ q(1-q)^{K-1} = \frac{T}{q}.$$

Also $\sum_{K=1}^{N} p_K = \sum_{K=1}^{N} q(1-q)^{K-1} = 1-(1-q)^N$

and $\sum_{K=1}^{N} KT\, p_K = \sum_{K=1}^{N} KT\, q(1-q)^{K-1} = \frac{T}{q}\left[1-(1-q)^N\right] - NT\,(1-q)^N$

Substituting these expressions into (25) yields

MFR = q/T, as expected.

Equation (25) can be simplified by observing that the number of airplane losses is small compared with the numbers of airplanes involved. As a consequence the total flight time may be approximated by n NT. Thus,

$$MFR \cong \frac{\sum_{K=1}^{N} p_K}{NT} .$$

5. Periodic Tests

Flight safety reliability goals may impose severe requirements on preflight test coverage. It will be shown that some configurations require coverages in excess of 99.9%. Unfortunately, preflight test is also subject to operational requirements which limit test time, test equipment and accessibility to system components. As a consequence, the coverage attained may be less than required. A poor initial preflight coverage can be effectively improved by administering an additional and more complete test at longer periodic intervals. For purposes of this discussion this periodic test is assumed to have 100% coverage in order to simplify the computations. The effects of periodic testing can be seen in Figure 1. The dashed curve shows the probability of a latent failure versus NT for a channel failure rate of $z = 300 \times 10^{-6}$. The solid curve is the resultant failure rate with periodic testing where N_p is the number of mission between periodic tests.

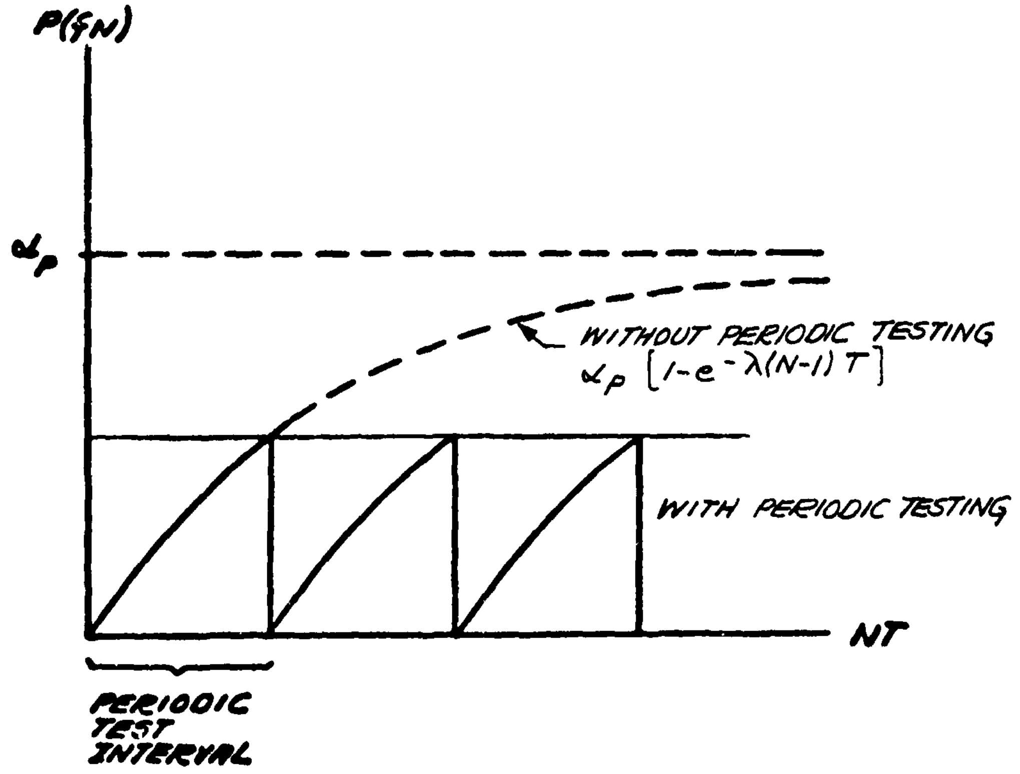


Figure 1. Effect of periodic testing with 100% coverage

6. Effects of Failures of the Test Device and Disengage Logic

There are two general classes of failures which affect system operation: the active failure which is a failure in the command chain and the failure of the test device or disengage logic, either of which prevents disengagement of the failed channel. The effects of the latter type failures depend upon the configuration. In the case of a self test procedure a failure of the test only impairs the test coverage in the failed channel. The effects of these failures are relatively straightforward and will be discussed presently. The situation is more complicated with comparison monitoring where a monitor failure could impair coverage in two channels. The difference is illustrated in the following example.

Example

The configuration is dual and fail passive. (i.e., to trim) As a consequence, loss of system occurs in the event of either channel failing undetected. If both channels are self monitored then this event is

$$E_s = F_1\bar{A}_1 + F_2\bar{A}_2$$

or

$$E_c = (F_1 + F_2)\,\bar{A}$$

with a single comparator between channels. Thus, if the channel #1 test fails

$$E_s = F_1 + F_2\bar{A}_2$$

and if the comparator fails

$$E_c = F_1 + F_2.$$

The difference could be significant.

However, if failures of the test are ruled out, then

$$\begin{aligned} P(E_s) &= \alpha_1 z_1 + \alpha_2 z_2 - \alpha_1 \alpha_2 z_1 z_2 \\ &= 2\,\alpha z - \alpha^2 z^2 \end{aligned}$$

and

$$P(E_c) = \alpha(2z - z^2) = 2\,\alpha z - \alpha z^2 .$$

Clearly, the difference is insignificant in this case. For purposes of this discussion there is no distinction made between test failures and disengage failures since they both prevent removal of the failure. Let

F_t = Event of failure of the test during the mission

A = Event of an alarm of the LRU

z_t = $P(F_t)$

α = Test deficiency with respect to the LRU

α_t = Preflight test deficiency with respect to the test device

F = Event of failure of the LRU during the mission

z = $P(\Gamma)$

Then

$$P(F\bar{A}) = P(F\bar{A}|\bar{F}_t)\ P(\bar{F}_t) + P(F\bar{A}|F_t)\ P(F_t)$$

$$P(F\bar{A}|\bar{F}_t) = \alpha z$$

$$P(F\bar{A}|F_t) = z$$

In this last expression it is assumed that any failure of the test results in total loss of coverage. Accordingly,

$$P(F\bar{A}) = (\alpha + z_t - \alpha z_t)z$$

Thus, the test deficiency is effectively increased from

α to $\alpha + z_t - \alpha z_t$.

In general, the probability of loss of test will be a function of elapsed mission due to latent failures. In this case z_t is replaced by

$$z_t + \alpha_t (1-z_t) \left[1-(1-z_t)^{N-1}\right] .$$

For large N the probability of a failure in the test is

$$z_t + \alpha_t - \alpha_t z_t$$

and the deficiency is

$$\alpha + z_t + \alpha_t, \text{ approximately.}$$

For a typical fail-safe comparator

$$z_t = 1.55 \times 10^{-6}$$

and

$$\alpha_t = 0, \text{ approximately,}$$

and since α is typically larger than .01 the effects of failures of the LRU test can be neglected. It should be noted that z_t, as given above, does not include single point failures, as might, for example, occur in the power supply and hence could affect all comparators.

SECTION 4

DESCRIPTION OF CANDIDATE REDUNDANT CONFIGURATIONS

In the next section detailed tradeoffs will be presented for several versions of triplex and quadruplex configurations. In this section several basic redundant configurations will be presented together with ground rules governing failure effects and those properties of secondary actuators and signal selection devices that are pertinent to the tradeoff studies.

1. Secondary Actuators

A detailed description of force-summed redundant secondary actuators is given in Appendix IV. For purposes of the tradeoffs the following properties are sufficient:

Dual Actuators

The output is the mid-value of the two commands and a hypothetical zero command.

Triplex Actuators

The output is the mid-value of the three commands.

Quadruplex Actuators

The output is the mid-value of the four commands and a hypothetical zero command. Upon detection and disengagement of a failed quadruplex actuator the configuration reverts to a triplex arrangement.

2. Signal Selection Device (SSD)

The signal selection device is a majority device. If an input to the SSD fails and is detected then that signal is disqualified and the SSD proceeds as a majority device with the remaining signals. The SSD output is considered to have failed if and only if

a. the last signal input fails or

b. there are at least as many failed (and not disqualified) inputs as non-failed inputs

Incidentally, these rules of failure effects also apply to the secondary actuators.

No distinction is made between passive and non-passive failures of the system. In practice it is, of course, desirable that the airplane fail to a trim condition following loss of system.

Failure Status Events

In the absence of nuisance alarms the four events $F\bar{A}$, FA, $\bar{F}A$, and $\overline{FA}$ associated with each LRU reduce to $F\bar{A}$, FA and F, where F is the event of an inflight failure of the LRU. A similar set of events is defined for latent failures except that fA is a vacuous event. Each of the three events is associated with an integer:

$f\bar{A}$, $F\bar{A}$ → 1

fA, FA → 2

f, $\bar{F}$ → 3

Combinations of latent and inflight failures of an LRU combine to form composite failure events according to the following table:

TABLE 2. COMPOSITE FAILURE EVENTS FOR AN LRU

		INFLIGHT	
LATENT	1	2	3
1	1	2*	1
2			
3	1	2	3

*This event could have designated "1" for worst case.

According to the table a latent failure followed by an undetected inflight failure is an undetected failure. Also, a latent failure followed by a detected failure is considered to be a detected failure.

If X, Y, Z designate the composite failure events of the three inputs to a triplex voter (SSD) then the voter fails for the following combinations: of X, Y and Z:

(1, 1, 1) and all combinations

(1, 1, 3) and allcombinations

(1, 2, 2) and all combinations

(1, 2, 3) and all combinations

(2, 2, 2) and all combinations

These rules are in accordance with the rules already established for SSD's. Observe that a detected failure effectively disqualifies that input to the SSD. A similar set of combinations are defined for the quadruplex SSD. Of these, only a few are enumerated:

(1, 1, 3, 3) and all combinations

(2, 2, 1, 3) and all combinations

(2, 2, 2, 1) and all combinations etc.

When nuisance alarms are allowed, the status events FA and $\bar{F}$A have the same effect* as a detected failure. Therefore, both events are associated with a "2" type status event where

$$P(FA + \bar{F}A) = P(A) = \left(\frac{1-\alpha}{1-\beta}\right) z.$$

The "3" type status event becomes $\overline{FA}$ where

$$P(\bar{F}\bar{A}) = 1 - \left(\frac{1-\alpha\beta}{1-\beta}\right) z.$$

The effects of nuisance alarms on flight safety reliability will be established in the tradeoffs.

3. Effects of Mission Duration, T

In the absence of latent failures the probability of loss of aircraft depends only upon mission duration, T. In this case

$$P(L_N) = \text{constant}.$$

* Here we overlook the fact that loss of a triplex system, for example, due to three nuisance alarms does not represent loss of the airplane if the pilot has reset capability.

It will be shown that the dominant* failure combination in the triplex arrangement is the pair

$$F_i \cdot \bar{A}_i \cdot F_j$$

where i and j are channel designations and i ≠ j. Therefore

$$P(L_N) \sim P(F_i \bar{A}_i F_j) = \alpha z^2$$

(where "$\sim$" denotes "proportional to") and the loss rate is

$$\frac{P(L_N)}{T} \sim \frac{\alpha z^2}{T} \sim T$$

since $z \sim T$.

In the quadruplex arrangement the dominant failure combination is

$$F_i \bar{A}_i \; F_j \bar{A}_j \text{ and, hence,}$$

$$P(L_N) \sim P(F_i \bar{A}_i F_j \bar{A}_j) = \alpha^2 z^2$$

and

$$\frac{P(L_N)}{T} \sim \frac{\alpha^2 z^2}{T} \sim T.$$

In both the triplex and quadruplex configurations the loss rate is then proportional to mission time.

When latent failures are present the situation is quite different because the latent failure combinations dominate for most of the service life. In the triplex configuration the dominant failure event is

$$f_i \; F_j \text{ and,hence,}$$

$$P(L_N) \sim P(f_i F_j) = \alpha_p (1-e^{-\lambda t}) \; z$$

and $\frac{P(L_N)}{T}$ is independent(approximately) of T.

Similarly, the dominant failure combination of the quadruplex configuration is

$$f_i \; F_j \bar{A}_j \text{ and, hence,}$$

* Excluding single point failures.

$$P(L_N) \sim P(f_i F_j \bar{A}_j) = \alpha_p(1-e^{-\lambda t})\, \alpha z$$

and $\frac{P(L_N)}{T}$ is,again, independent of T.

In the event that primary actuator failures are the dominant failures then $P(L_N) \sim P(F_{Actuator}) \sim T$

and $\frac{P(L_N)}{T}$ is independent of T.

4. Self Tested Versus Comparison Monitored Configurations

In the tradeoff studies the only distinction made between an inflight self tested and comparison monitored system is that the comparison monitored system requires at least two good channels for non-failed operation. Thus, a failure combination such as (2, 2, 2, 3) would represent a failed system if the configuration were quadruplex.

Self tested channels are only used in the dual and triplex configurations. This approach is justified because the added benefits of self test tend to be negligible in the quadruplex system compared with more dominating failures such as single point, latent and inflight undetected.

According to the ground rules already established a comparison monitored triplex system does not provide any advantages over an unmonitored (i.e., inflight) system. The major benefits of comparison monitoring in the triplex system are

a. First failure does not propagate to the surface and

b. Second failure following a detected first failure results in a passive failure of the airplane.

c. Pilot is warned of failed channel. He then has the option of aborting the mission (a factor which effectively increases flight safety reliability).

However, it has already been assumed that the force summed actuators will prevent an undetected failure from propagating to the surface, whether detected or not, and no distinction was made between passive and non-passive loss of system. In practice, of course, this is an important consideration; but it was not a factor in the tradeoffs. If good inflight coverage is required, a completely self tested channel is difficult to achieve without a significant increase in cost of extra hardware or software in the form of servo models, self-tested sensors, performance monitors, reasonableness tests, sensor stimuli, etc. However, the cost depends upon the coverage required and it is this basic requirement that will be determined in the tradeoffs.

5. Triplex Versus Quadruplex

Before proceeding to a description of the configurations, there are several aspects of the triplex versus quadruplex tradeoff which deserve a separate discussion.

a. With a force summed servo arrangement two undetected failures in a quad configuration could result in a passive failure of the airplane (provided that trim is maintained). In a triplex configuration two undetected failures could result in a non-passive failure of the airplane. The quad configuration has a clear advantage in this respect.

b. There is one feature of the quadruplex comparison monitored configuration which has significant implications regarding the benefits of that arrangement. In the triplex, self-test configuration the dominant failure combinations have the form

$$F_i\bar{A}_iF_j,\ f_iF_j$$

where f and F denote latent and inflight failures, respectively. Thus, an undetected failure followed by any failure could result in loss of system. In the quad configuration the dominant failure combinations are

$$F_i\bar{A}_iF_j\bar{A}_j,\ f_iF_j\bar{A}_j.$$

Thus, two undetected failures could result in loss of system. If comparison monitoring is used exclusively, then there is a possibility that an undetected failure in one channel will impair coverage of subsequent inflight failures in the remaining channels. Taking the worst case, if subsequent inflight coverage is zero following an undetected failure, then the dominant failure combinations of the quad comparison monitored configuration are

$$F_i\bar{A}_i\ F_j \text{ and } f_iF_j.$$

Comparing these events with those of the triplex arrangement it can be seen that the quad configuration provides no benefits over the triplex unless inflight coverage is significantly better, as it must be in order to compensate for the larger number of combinations of the form $F_j\bar{A}_jF_j$ in the quad arrangement. If preflight test coverages are the same in both configurations then the latent terms could become dominant. Again, because there are more such combinations in the quad configuration the triplex would provide greater flight safety.

As a consequence of these observations it is assumed that comparison monitoring is always augmented by other techniques of inflight testing in order to insure a minimum impairment of coverage following an undetected failure. In the tradeoffs to follow it is assumed that coverage of subsequent failures is not significantly impaired following an undetected failure in a quad channel.

6. LRU Failure Rates

As indicated in Appendix I, the following LRU failure rates are assumed:

Primary Actuator (Pitch, Yaw, Roll)	$= 0.5 \times 10^{-6}$
Secondary Actuator (Pitch, Roll, Yaw)	$= 100 \times 10^{-6}$
Accelerometer (Pitch, Yaw)	$= 20 \times 10^{-6}$
Rate Gyro (Pitch, Roll, Yaw)	$= 25 \times 10^{-6}$
Stick Force Sensors (Pitch, Roll, Yaw)	$= 5 \times 10^{-6}$
Digital Computer	$= 120 \times 10^{-6}$

The secondary actuator failure rate does not include the hydraulic supply which could double the indicated failure rate.

7. Dual Configuration

Although the emphasis of the study is on triplex and quad configurations, the dual configuration will be discussed, briefly, for purposes of comparison. In order to simplify the computation it is assumed that the digital computers are cross strapped and the sensor failure rates are zero. Both channels are self tested. The event of loss of system for a secondary actuator or a digital computer is

$$E = F_1F_2 + F_1\bar{A}_1 + F_2\bar{A}_2$$

and

$P(E) = z^2 + \alpha z + \alpha z$, approximately,
where $P(F_1) = P(F_2) = z$.

Digital Computer

$$P(E) = (120 \times 10^{-6})^2 + 2\alpha\,(120 \times 10^{-6})$$

$$= 240\,\alpha \times 10^{-6}\text{, approximately.}$$

Secondary Actuators

$$P(E) = (100 \times 10^{-6})^2 \times 2\alpha\,(100 \times 10^{-6})$$

$$= 200\,\alpha \times 10^{-6}\text{, approximately.}$$

Combining three sets of secondary and primary actuators yields, for the probability of loss of system in one hour,

$$840\,\alpha \times 10^{-6} + 1.5 \times 10^{-6}\text{, approximately.}$$

In order to meet the goal of 3.0×10^{-6} we require

$$840\,\alpha \times 10^{-6} + 1.5 \times 10^{-6} < 3.0 \times 10^{-6}$$

or $$\alpha < \frac{1.5}{840} = .0010$$

i.e., 99.9% of all inflight failures must be detected.

In addition to this high inflight coverage requirement, failures must be detected rapidly since it must be presumed that the airplane is out of control (but passive) until the failed channel is detected and removed.

8. Triplex Configuration

The basic inflight, self tested triplex configurations are shown in Figures 2 and 3 with no cross strapping and full cross strapping, respectively. The cross strapping is ideal in that there are no failure probabilities associated with cross strapping. The effective locations of the cross straps are indicated by boxes labelled "V". Details of these signal selection devices are contained in Appendix VI, If the voting of sensors in Configurations 1 and 2 is performed in computer software and the cross strapping of signals is done digitally through intercomputer data buses, a computer failure could cause simultaneous failures of the monitoring and cross strapping. If monitoring of the secondary actuators is performed by the digital computers via data links between the servos and computers, and cross-strapping of the computer outputs is performed by the same or similar data links, data link and interface component failures as well as computer failures could fail monitoring and cross

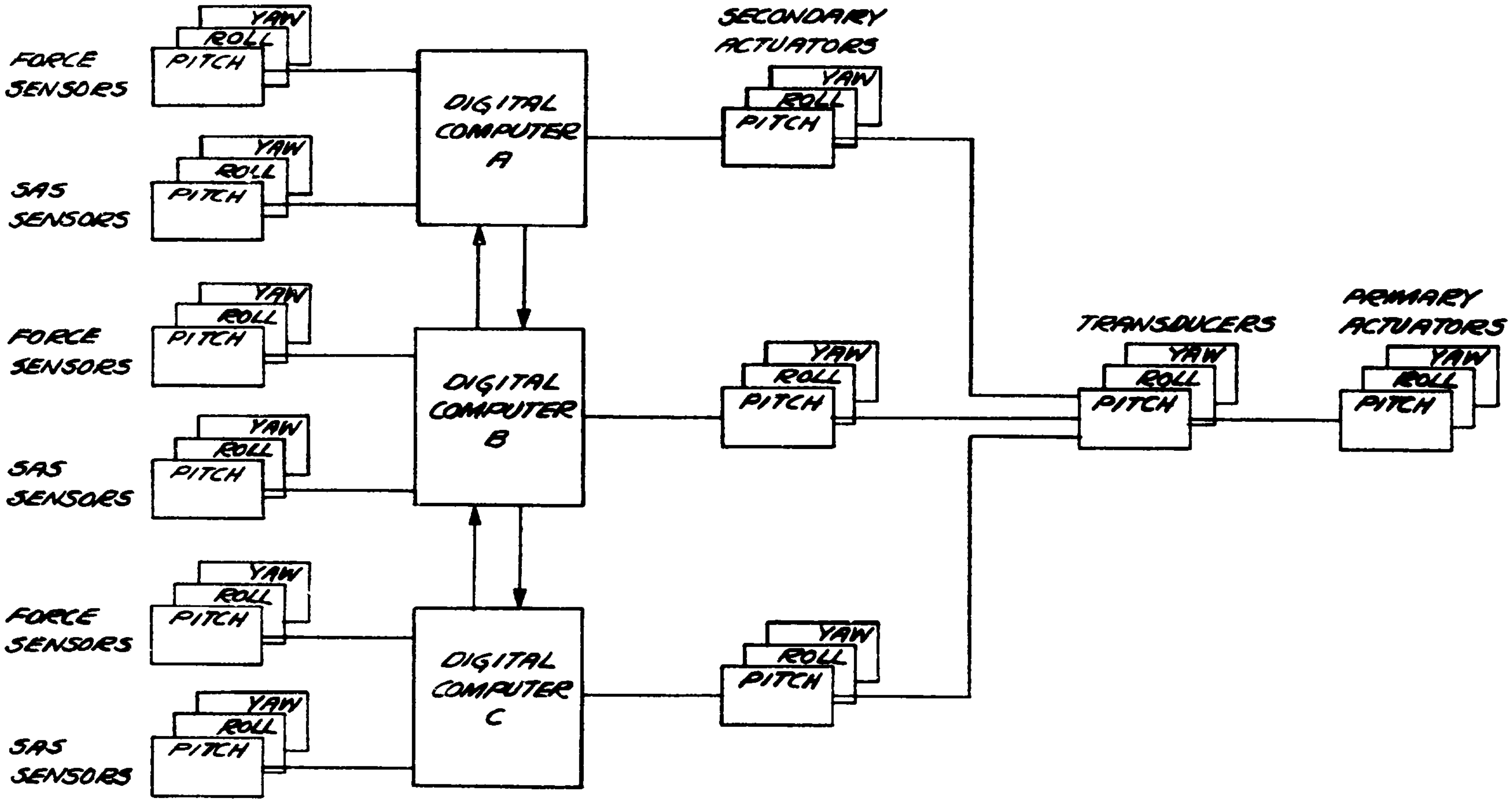


Figure 2. Triplex, inflight, self tested configuration no cross strapping configuration 1

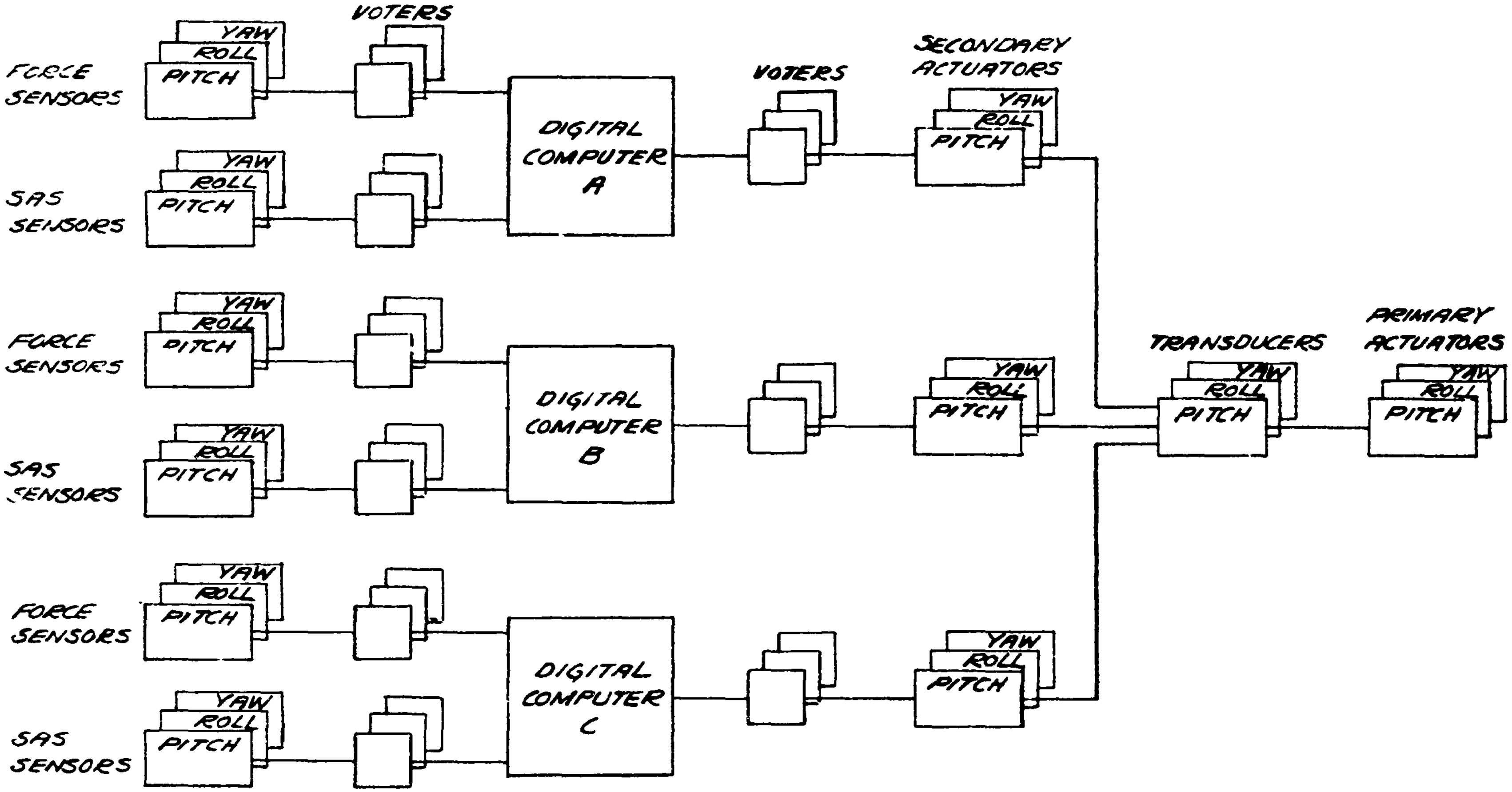


Figure 3. Triplex, inflight, self tested configuration full cross strapping configuration 2

strapping simultaneously. Such considerations complicate the analysis of any actual system and tend to obscure the basic potentialities of the redundant system. Ideally, in a well-designed system, the failure rates of any auxiliary cross strapping and monitoring components should be considerably less than those of the components in the main signal chains. The same is true of any logic and automatic disengagement features that might be required to insure operation after one or two failures. Of course, great care must be exercised to insure that no single failure with a probability approaching the flight safety goal can cause complete loss of the system. In the present trade studies, all auxiliary components including voters are assumed to have zero failure probabilities. In order to obtain the added reliability benefits of cross strapping the cross straps at the output of the digital computers must be dedicated devices controlled by dedicated logic.

9. Quadruplex Configurations

The basic quadruplex configurations are shown in Figures 4 and 5 with no cross strapping and full cross strapping, respectively. The quadruplex configurations are "comparison monitored" as defined previously. Explicit techniques of cross channel monitoring are discussed in Appendix VI and in Reference 1 and 5.

10. Triplex with Back-Up Configuration

From a previous discussion of the relative merits of the triplex versus quadruplex configuration, it is apparent that the added reliability improvement of the quad arrangement is not commensurate with what would be expected from the extra channel of redundancy. Essentially, this is due to the even number of channels which require inflight monitoring in order to realize the advantage of redundancy.

The basic triplex with back-up configurations are shown in Figures 6 and 7 with no cross strapping and full cross strapping, respectively. For purposes of the tradeoffs the back-up channel is assumed to be identical to the other channels. In practice, however, the back-up electronics would be analog with the minimal get-home-and-land capability. As a consequence, the back-up channel requires no inflight testing and can be thoroughly tested in preflight test. In the tradeoffs the back-up channel is not tested inflight and its preflight coverage is assumed to be the same as the other channels.

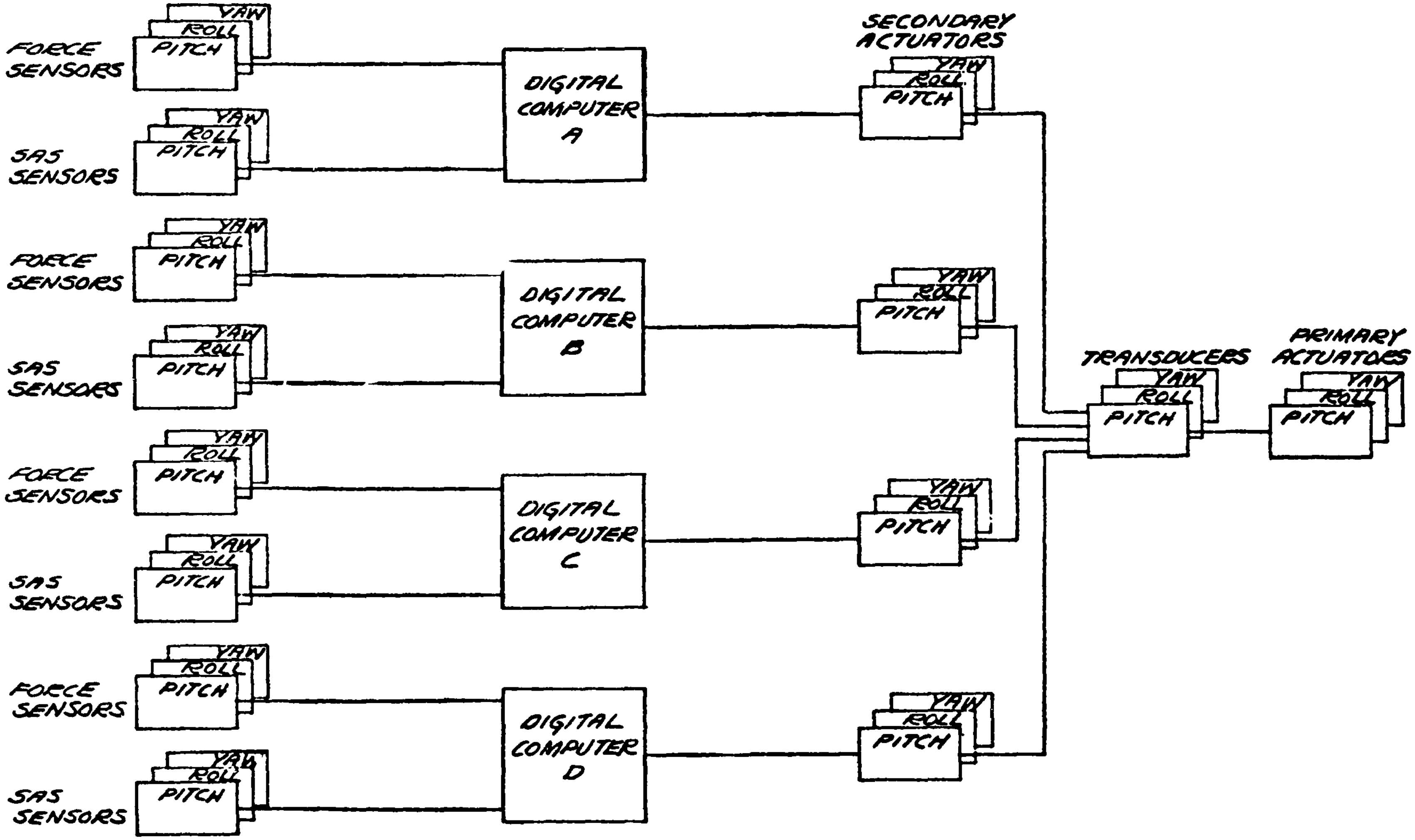


Figure 4. Quadruplex configuration 1 no cross strapping

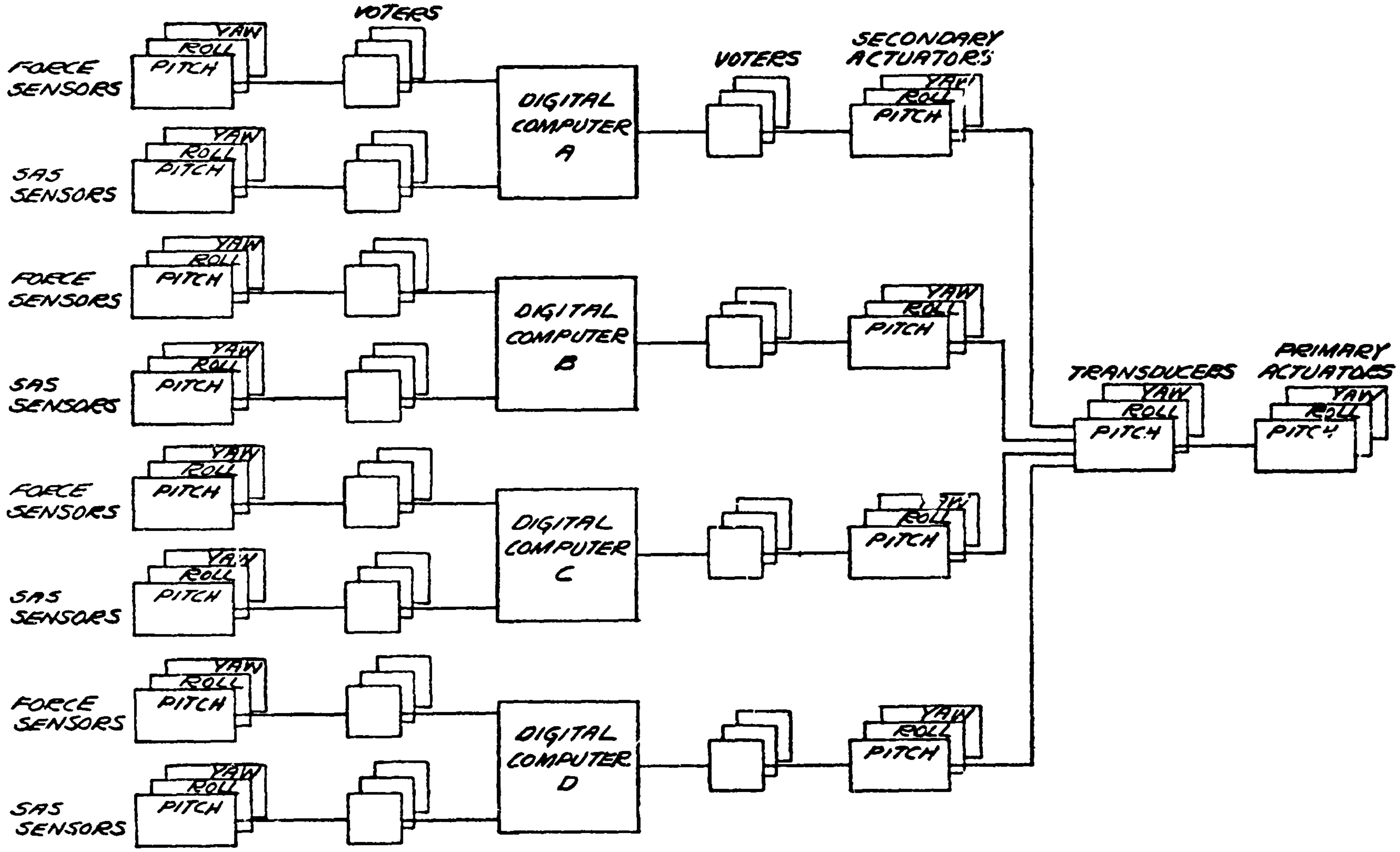


Figure 5. Quadruplex configuration 2 full cross strapping

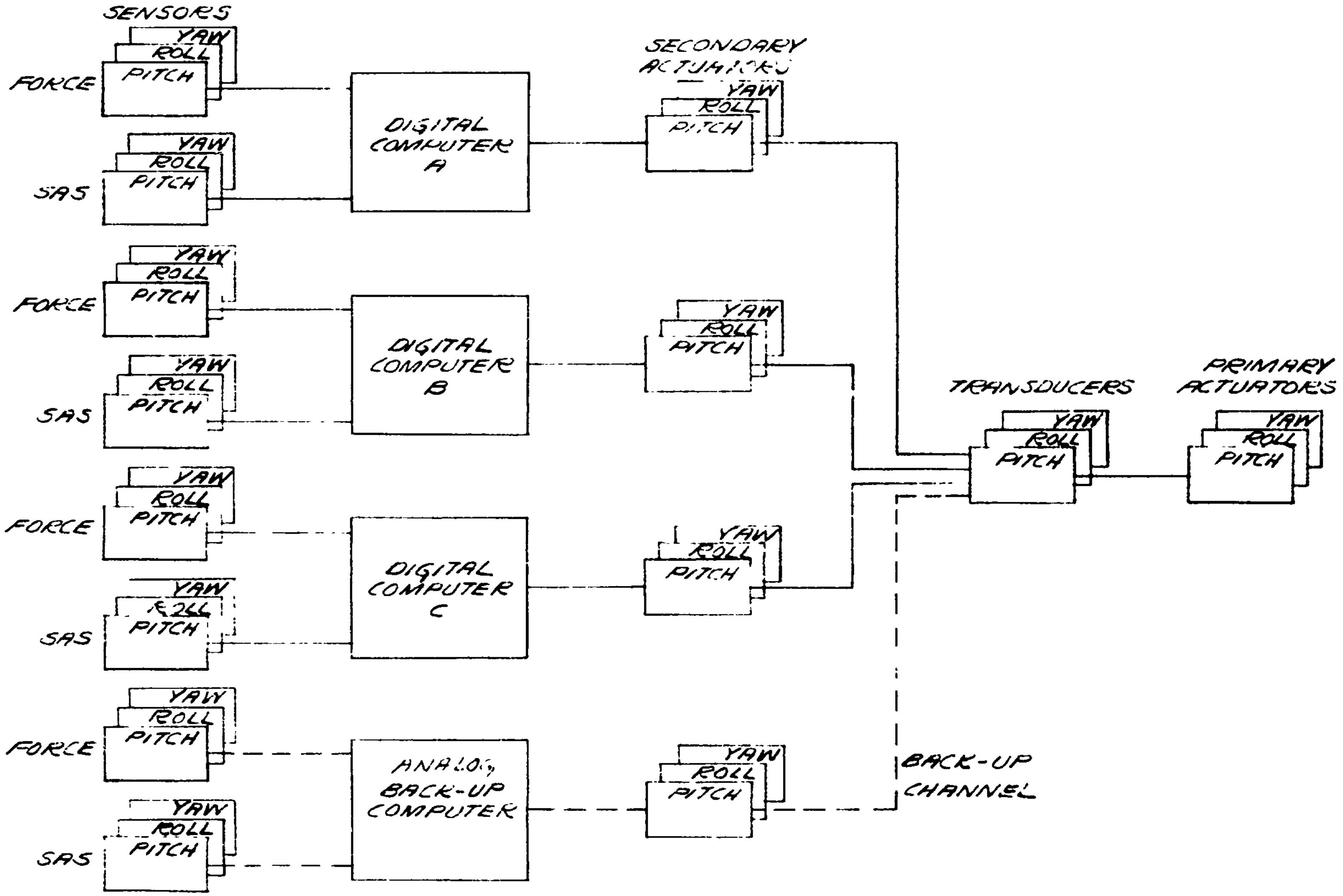


Figure 6. Triplex with back-up configuration 1 no cross strapping

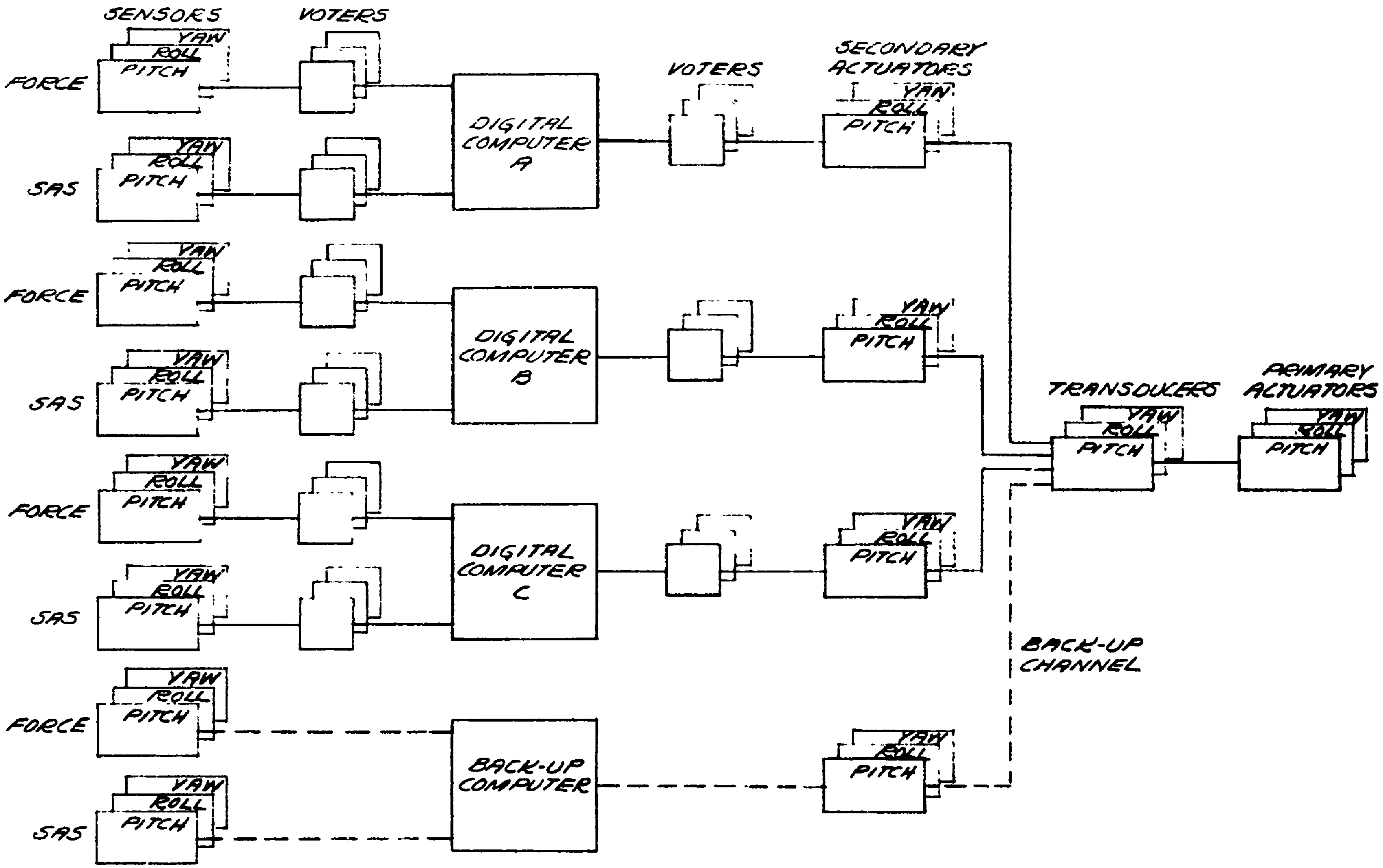


Figure 7. Triplex with back-up configuration 2 full cross strapping

a. Disengage/Engage Strategy

Upon detection of the first failure, the failed channel will automatically disengage. An alternative is to annunciate the failure and let the pilot manually disengage the failed channel. In any case the strategy for a first failure is not critical. This is a consequence of our assumption that an undetected failed channel will result in little or no degradation in performance because of the mechanical voting of the actuators. In the event of a second detected failure, the triplex, in-line channels will be automatically disengaged and the back-up channel engaged. If the second failure is not detected, we make the assumption that the pilot can recognize loss of control and manually engage the back-up before serious damage occurs. It is difficult to envision how a back-up channel can be used to any advantage if it is assumed that the pilot either cannot recognize loss of control or cannot manually engage the back-up in time to avert serious damage. This would imply that any two failures of the inline channels, one of which is undetected, may result in loss of the airplane. The back-up configuration, under these conditions, would compare unfavorably with a straight quadruplex configuration where loss of control requires two, undetected failures, or three detected failures. While the back-up channel loses its effectiveness if the assumption is invalid, the validity of this assumption remains, nevertheless, an open question.

In previous configurations we took the conservative position and equated loss of control with loss of the airplane, i.e. the airplane failed to a non-trim condition. We now modify this position and distinguish between passive and non-passive states of the airplane following loss of control. Table 3 summarizes the effects of loss of control as a function of detected, undetected, passive and non-passive failure sequences in a triplex configuration. The table entries were obtained assuming a force-summed servo model. From the table it can be seen that, of the 16 possible failure sequences, 14 result in passive loss of control. Only when the first failure is undetected and non-passive and is followed by a second non-passive failure does loss of control result in a non-passive state of the airplane. Accordingly, our original assumption can be restated as follows:

In a FBW primary control system,

(1) the pilot can recognize passive loss of control and manually engage the back-up channel in time to avert serious damage to the airplane, and

(2) the event of an undetected, non-passive first failure followed by a non-passive second failure is remote or if not remote, the pilot will recognize the failure and manually engage the back-up channel in time to avert serious damage. This latter presumption is justified on the grounds that the back-up configuration presents the clear and unique alternative of engaging the back-up channel upon the occurrence of the second failure. There is no time wasted in determining which of the remaining channels are non-failed as is the case with the quadruplex configuration. No distinction is made between passive and non-passive failures following loss of the system. In practice it is, of course, desirable that the airplane fail to a trim condition following loss of system.

TABLE 3. RESULTANT AIRCRAFT STATES FOLLOWING LOSS OF CONTROL IN A TRIPLEX CONFIGURATION

1st Failure Detected	1st Failure Undetected	2nd Failure Detected	2nd Failure Undetected	Effect on Aircraft
P		P		P
P			P	P
P		NP		P
P			NP	P
NP		P		P
NP			P	P
NP		NP		P
NP			NP	P
	P	P		P
	P		P P	P
	P	NP		P
	P		NP	P
	NP	P		P
	NP		P	P
	NP	NP		P and NP Transient
	NP		NP	NP

P = Passive Failure
NP = Non-Passive Failure

As a direct consequence, the triplex, in-line channel performance is assumed to be independent of inflight failure detection capability and loss of the airplane occurs only if two of the triplex channels fail followed by a failure of the back-up channel.

Although inflight monitoring may not be required for improved flight safety (e.g., the loss of two channels may be sufficiently improbable) it should be included, in practice, to appraise the pilot of system status so that he may abort the mission, if desired. If automatic disengagement of the triplex system is allowed then nuisance alarms could degrade flight safety reliability.

The dominant failure combinations of the back-up configuration are

$$F_iF_j(f_B + F_B),\ f_iF_j(f_B + F_B)$$

where the subscript "B" denotes back-up channel. Observe that inflight testing is not required for improved reliability. The benefits of the back-up configuration can be seen by comparing its dominant failure combinations with those of the triplex and quad arrangements, i.e.,

$$F_i\bar{A}_iF_j,\ f_iF_j \qquad \text{(Triplex)}$$

$$F_i\bar{A}_i\ F_j\bar{A}_j,\ f_iF_j\bar{A}_j \qquad \text{(Quad)}$$

Test Coverage

In order to simplify the computations all LRU's are assumed to have the same inflight and preflight test coverage (i.e., $1-\alpha_i$ and $1-\alpha_p$, respectively) and the same nuisance alarm sensitivity, β .

Loss of Airplane

In the tradeoffs loss of airplane is equivalent to loss of at least one axis. In a cross strapped configuration this will occur whenever the output of a signal selection device (including secondary actuators) fails.

11. Aborts

It has been established from field data that the abort rate of fighter aircraft due to failures of the PFCS is several orders of magnitude greater than the loss rate (e.g., 420×10^{-6} compared with 3.8×10^{-6} for the F-4). Although there is an element of arbitrariness in any definition of abort the following abort strategy appears to be reasonable:

A mission is presumed to be aborted when:

Triplex

A single LRU alarms in any axis. This includes sensors, computers and secondary actuators.

Quad

Two LRU's supplying inputs to any signal selection device, in any axis, alarm.

Triplex with Back-Up

The pilot switches to the back-up channel.

Calculated Abort Rates

Following the prescribed strategies abort rates are calculated, approximately, for each of the candidate configurations.

Triplex, Configurations 1 and 2

$$\text{Abort Rate} = \frac{1-\alpha}{1-\beta} \times 1650 \times 10^{-6} \text{ aborts/flight hour}$$

Quadruplex, Configuration 1 (Worst Case)

$$\text{Abort Rate} = \left(\frac{1-\alpha}{1-\beta}\right)^2 \times 1.13 \times 10^{-6} \text{ aborts/flight hour}$$

Triplex with Back-Up, Configuration 1 (Worst Case)

$$\text{Abort Rate} = \frac{1-\alpha}{1-\beta} \times \frac{1.13}{2} \times 10^{-6} \text{ aborts/flight hour}$$

In arriving at this last result we took the conservative approach and assumed that one of the channels was disengaged due to a nuisance alarm indication.

From these results it can be seen that the abort rate of the triplex configuration is about 4 times that of the F-4, assuming no nuisance alarms. If only one alarm out of every two is a nuisance alarm (i.e., $\beta = 1/2$) then the abort rate is 8 times that of the F-4. The abort rates of the quad and back-up configuration are several orders of magnitude less than that of the F-4.

SECTION 5

TRADEOFF OF REDUNDANT CONFIGURATIONS

All configurations are evaluated for a one hour mission. Loss of airplane is defined as a failure of at least one of three axes. The effects of mission duration have already been discussed in Section 4 where it was concluded that, because of the dominance of latent failure probabilities, $P(L_N)$ and MFR tend to be independent of mission time. It will be T shown. subsequently, that the dominance of single point failures, particularly the primary actuators, tends to equalize the relative differences between configurations. For this reason each configuration is evaluated for two primary actuators with failure rates of 0 and 0.5 x 10^{-6}/flight hour/axis, respectively.

1. Tradeoff Parameters Identified

Configurations are evaluated in terms of the following parameters:

$P(L_\infty)$ versus $1-\alpha_i$; $1-\alpha_p = 1.0$

$P(L_\infty)$ is the steady state value of $P(L_N)$. As indicated previously this parameter is a valid criterion for a commercial aircraft whose service life* is well in excess of the effective latent failure time constant.

$P(L_\infty)$ versus $1-\alpha_p$; $1-\alpha_i = .95$

This graph shows the sensitivity to preflight test coverage assuming an inflight test coverage of 95%. The inflight test coverage was selected because it is achievable without being prohibitive in terms of extra hardware, memory or real time. In any case the results are not especially sensitive to this parameter.

$P(L_N)$ versus Mission Time; $1-\alpha_i = .95$; $1-\alpha_p = .999$

This parameter versus time shows the effective latent failure time constant and the resultant degradation of flight safety reliability with time. The maximum time shown is 5000 hours since this value is approximately the service life of a typical fighter aircraft. The dashed horizontal lines are the steady state values of $P(L_N)$. Observe that preflight test coverage is 99.9%. Preflight coverage greater than 99.9% may be extremely difficult to achieve.

* A typical service life is 60,000 hours.

MFR versus 1- α_i; 1- α_p = .999; SL = 5000 hours

MFR versus 1- α_p; 1- α_i = .95; SL = 5000 hours

These graphs show the sensitivity of mean failure rate to inflight and preflight test coverage, respectively. The mean failure rate is calculated for a service life of 5000 hours.

MFR versus N_p; 1- α_i = .95, 1- α_p = .999, SL = 5000 hours

This graph shows the improvement in mean failure rate as a function of the number of missions between periodic tests of 100% coverage. The parameter, N_p, denotes the number of missions between periodic tests.

2. Results

Figures 8, 9

From these figures it can be seen that, assuming a 100% preflight test coverage, all configurations result in acceptable flight safety reliability for a wide range of inflight test coverages. The equalizing effect oi the single point primary actuator failures can be seen by comparing the two figures. For inflight test coverage of the order of 0.95 all configurations are compatible with the commercial transport flight safety goal of 0.23 x 10^{-6}/hour in the sense that flight safety reliability will be determined primarily by single point failures.

Figures 10, 11

The degrading effects of non-perfect preflight test coverage can be seen in these figures where it has been assumed that inflight test coverage is 0.95. Several conclusions may be inferred from these figures:

a. Cross-strapping improves incremental* flight safety reliability by a factor of 10 whereas with perfect preflight test coverage, the improvement is a factor of 3 or 4.

b. The triplex configuration is most sensitive to latent failures and the triplex with back-up configuration is the least sensitive. Figure 10 is summarized in Table 4.

*Incremental = excludes primary actuator failure rates.

TABLE 4. INCREMENTAL P (L_∞) VERSUS PREFLIGHT TEST COVERAGE (FIGURE 10)

	$1-\alpha_p=.99$	$1-\alpha_p=.999$	$1-\alpha_p=.9999$
*Triplex #1	202.75×10^{-6}	22.617×10^{-6}	2.337×10^{-6}
*Triplex #2	33.53×10^{-6}	3.427×10^{-6}	$.3558\times10^{-6}$
*Quad #1	19.462×10^{-6}	2.259×10^{-6}	$.2323\times10^{-6}$
*Quad #2	3.326×10^{-6}	$.342\times10^{-6}$	$.0350\times10^{-6}$
Triplex #1 With Back-Up	10.7×10^{-6}	$.13\times10^{-6}$	$.00237\times10^{-6}$
Triplex #2 With Back-Up	2.3588×10^{-6}	$.0267\times10^{-6}$	$.00735\times10^{-6}$

*$1-\alpha_i = .95$

Figures 12, 13

The degradation of flight safety reliability with time can be seen in these figures where preflight test coverage is assumed to be 0.999 for all configurations. Observe that there is a considerable difference between $P(L_\infty)$ and $P(L_K)$ when KT = 5000 hours. This is due to the small effective latent failure time constant of the overall system. Figure 12 is summarized in Table 5.

TABLE 5. INCREMENTAL P (L_K) AT 5000 HOURS VERSUS PREFLIGHT TEST COVERAGE = .999 (FIGURE 12)

*Triplex #1	4.65×10^{-6}
*Triplex #2	1.135×10^{-6}
*Quad #1	$.4644\times10^{-6}$
*Quad #2	$.113 \times 10^{-6}$
Triplex #1 with Back-Up	$.0080\times10^{-6}$
Triplex #2 with Back-Up	$.0027\times10^{-6}$

*$1-\alpha_i = .95$

As in all figures the triplex with back-up configuration provides superior reliability performance.

Figures 14, 15

These figures show the insensitivity of mean failure rate to inflight test coverage for all configurations. The triplex with back-up configurations are not shown since inflight test coverage is assumed to be 0.

Figures 16, 17

A comparison of incremental $P(L_\infty)$, MFR and $P(L_K)$ at 5000 hours for all configurations is given in Table 6. Preflight test coverage is .999 and inflight test coverage, where applicable, is 0.95.

TABLE 6. INCREMENTAL $P(L_\infty)$, $P(L_K)$ AT 5000 HOURS, MFR WITH PREFLIGHT TEST COVERAGE = .999 (FIGURE 16)

	Incremental $P(L_\infty)$	Incremental $P(L_K)$ at 5000 Hours	Incremental MFR
*Triplex #1	$22.617\text{x}10^{-6}$	$4.65 \text{ x } 10^{-6}$	$2.5\text{x}10^{-6}$
*Triplex #2	$3.427\text{x}10^{-6}$	$1.135\text{x}10^{-6}$	$.62\text{x}10^{-6}$
*Quad #1	$2.259\text{x}10^{-6}$	$.464\text{x}10^{-6}$	$.25\text{x}10^{-6}$
*Quad #2	$.342 \text{ x } 10^{-6}$	$.113\text{x}10^{-6}$	$.0615\text{x}10^{-6}$
Triplex #1 with Back-UP	$.13\text{x}10^{-6}$	$.0080\text{x}10^{-6}$	$.0034\text{x}10^{-6}$
Triplex #2 With Back-Up	$.0267\text{x}10^{-6}$	$.0027\text{x}10^{-6}$	$.00175\text{x}10^{-6}$

* $1- \alpha_i = .95$

From the table and the figures it can be seen that MFR is a less conservative estimate of flight safety reliability than either $P(L_\infty)$ or $P(L_K)$ at 5000 hours.

Figures 18, 19

These figures show the effective improvement in mean failure rate with periodic (100% coverage) testing. A comparison of incremental MFR for all configurations is given in Table 7. Preflight test coverage is 0.999 and inflight test coverage, where applicable, is 0.95.

TABLE 7. MFR VERSUS PERIODIC TESTING WITH PREFLIGHT TEST COVERAGE = .999 (FIGURE 18)

	MFR No Periodic Test	MFR $N_p = 1000$	MFR $N_p = 500$
*Triplex #1	2.5×10^{-6}	$.61\times10^{-6}$	$.335\times10^{-6}$
*Triplex #2	$.62\times10^{-6}$	$.152\times10^{-6}$	$.0839\times10^{-6}$
*Quad #1	$.25\times10^{-6}$	$.031\times10^{-6}$	$.053\times10^{-6}$
*Quad #2	$.0615\times10^{-6}$	$.0132\times10^{-6}$	$.0077\times10^{-6}$
Triplex #1 with Back-Up	$.0034\times10^{-6}$	$.00054\times10^{-6}$	$.000336\times10^{-6}$
Triplex #2 with Back-Up	$.00117\times10^{-6}$	$.000019\times10^{-6}$	$.000117\times10^{-6}$

* $1-\alpha_i = .95$

It can be seen from the table that a periodic test at an interval as large as 1000 hours results in a considerable improvement in flight safety reliability.

3. Conclusions

a. In all configurations the benefits of redundancy tend to be negated by the dominating influence of latent and single point failures.

b. Cross-strapping can provide a significant improvement in flight safety reliability* primarily because of the dominance of latent failure probabilities.

*As defined by any of the several criteria proposed.

c. Because of the dominance of latent failures flight safety reliability is relatively independent of a given mission time.

d. A triplex configuration augmented by a back-up channel is less sensitive to the effects of latent failures than the straight triplex or quadruplex configurations.

e. Preflight test coverage requirements depend upon configuration and flight safety goals. The requirements can differ significantly depending upon the definition of flight safety reliability and whether or not periodic testing is employed. As an indication of the possible variation Table 8 shows the preflight test coverage required to meet an incremental flight safety goal of 1.0 x 10^{-6} for the Triplex #2 configuration.

TABLE 8. PREFLIGHT TEST COVERAGE REQUIRED TO ACHIEVE INCREMENTAL FLIGHT SAFETY RELIABILITY GOAL OF 1.0 x 10^{-6} WITH INFLIGHT TEST COVERAGE = .95

	$P(L_\infty)$	$P(L_K)$ at 5000 Hours	MFR	MFR With N_p=1000 Hrs.
Triplex #2	.9997	.9992	.9984	.9928

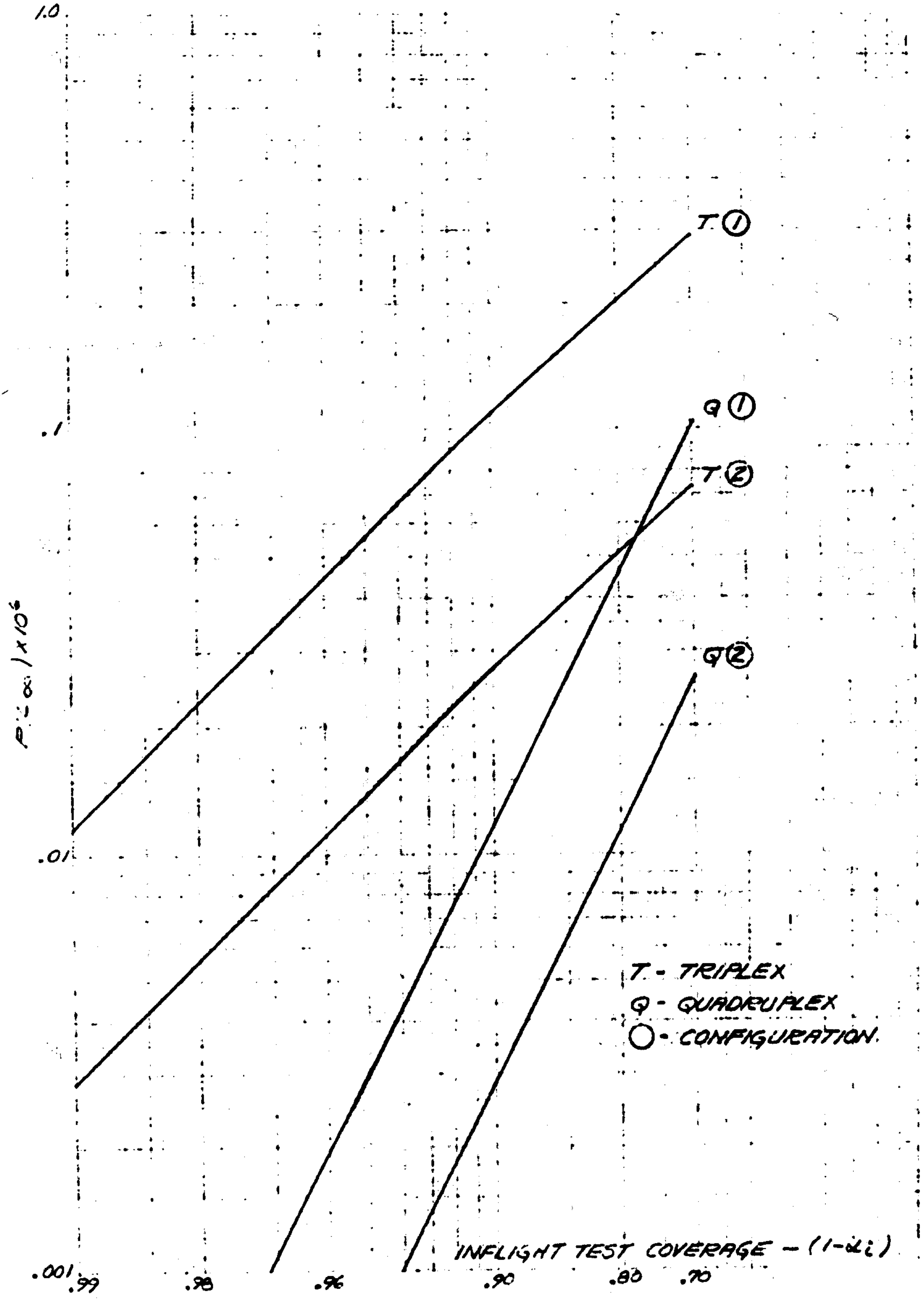


Figure 8. $P(L\infty)$ Versus $(1-\alpha_i)$; $1 - \alpha p = 1.0$
Primary actuator failure rate = 0

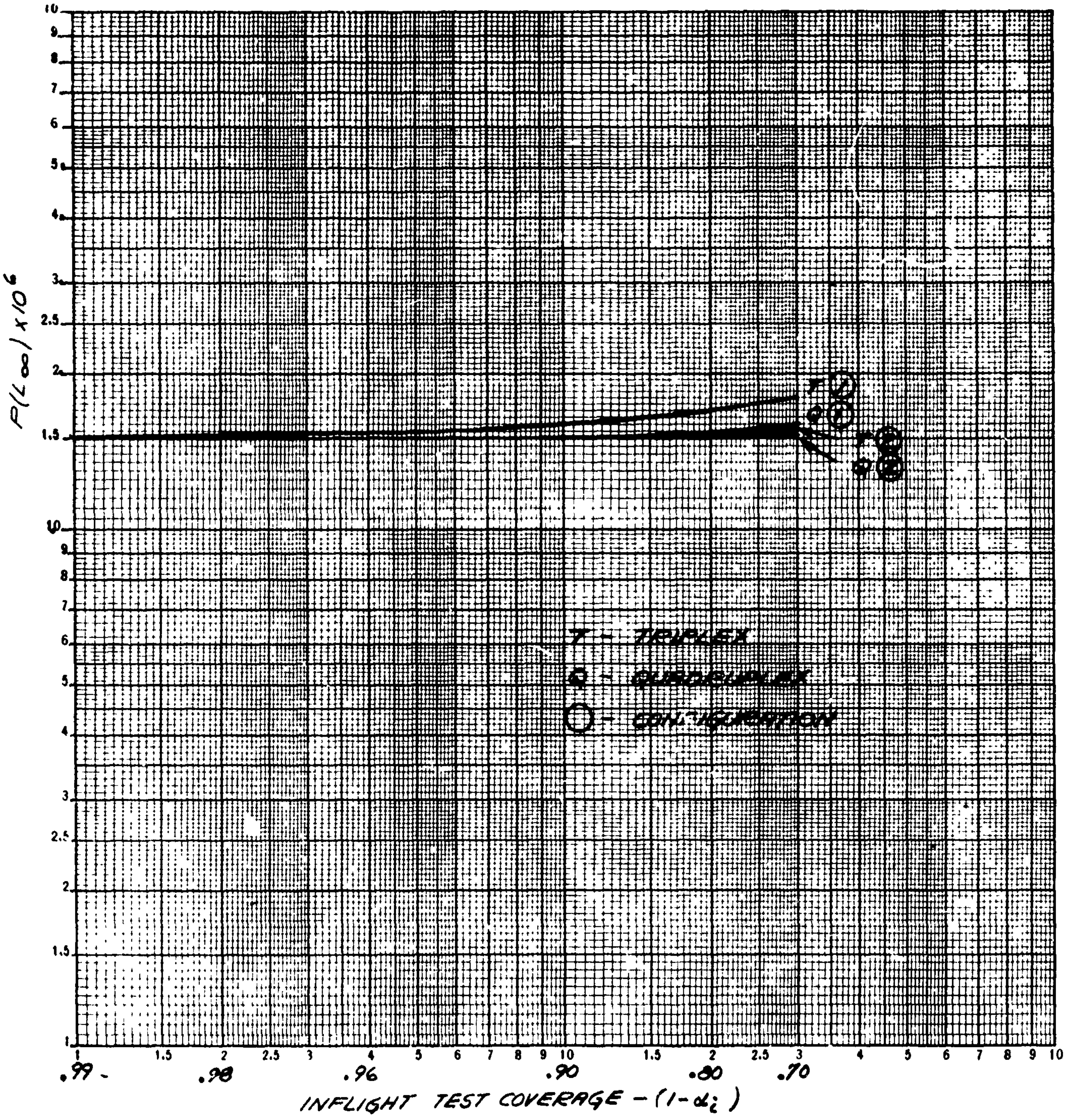


Figure 9. $P(L_\infty)$ Versus $(1-\alpha_i)$; $1-\alpha_p = 1.0$
Primary failure rate = .5 x 10-6

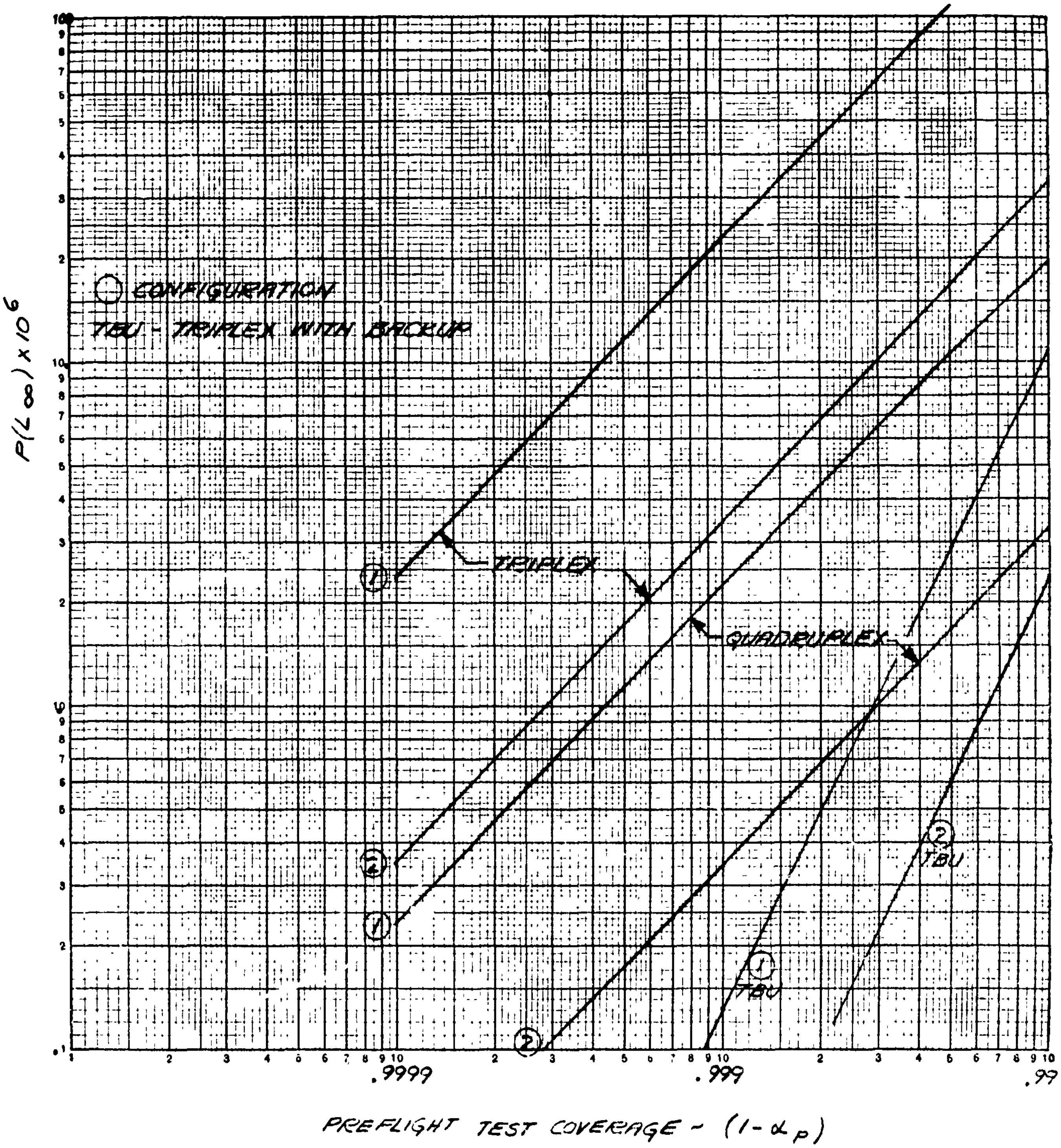


Figure 10. P(L ∞) Versus (1- αp); 1- αi = .95
Primary actuator failure rate = 0

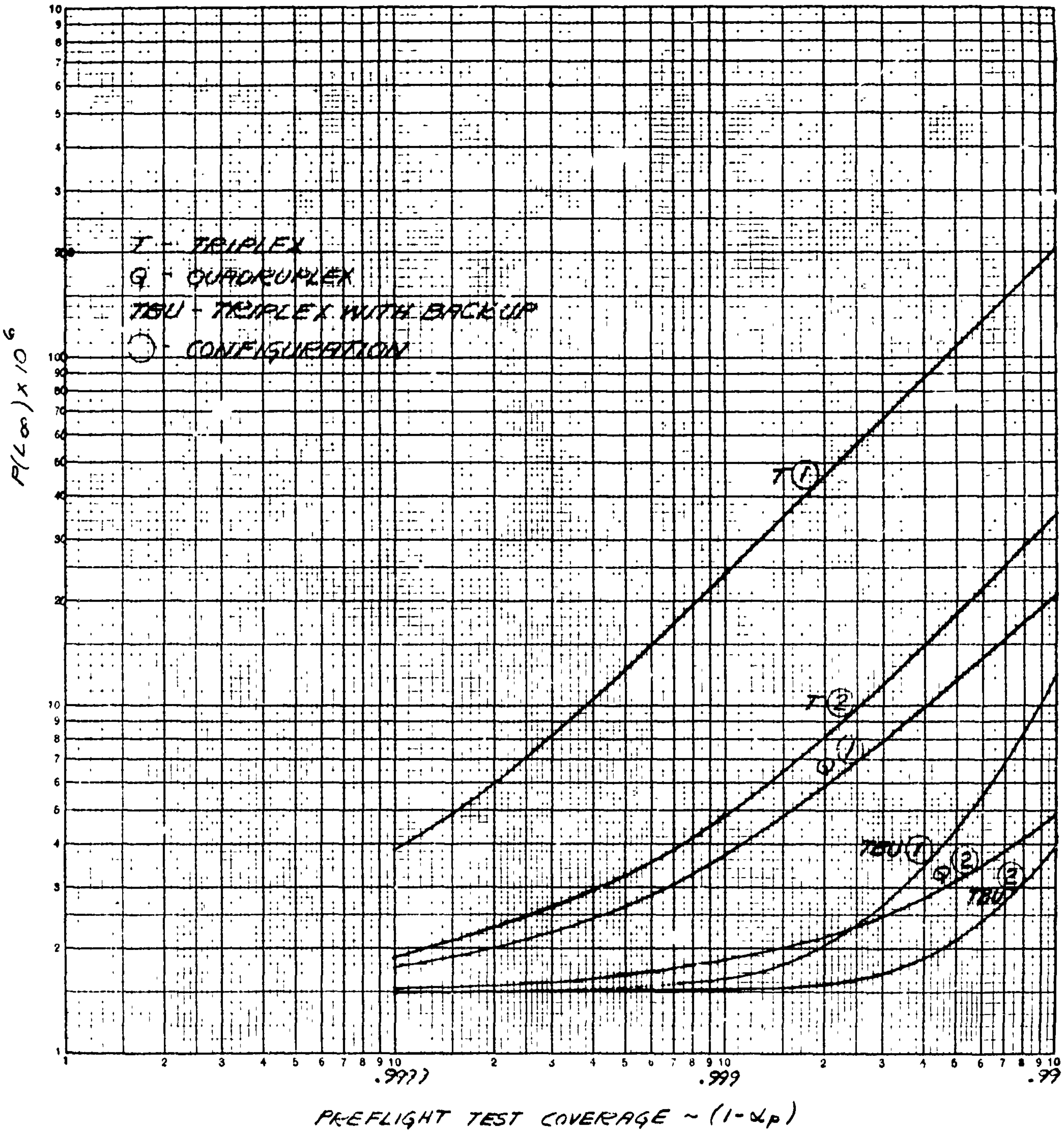


Figure 11. P(L ∞) Versus (1- αp); 1- αi = .95
Primary actuator failure rate = .5 x 10^{-6}

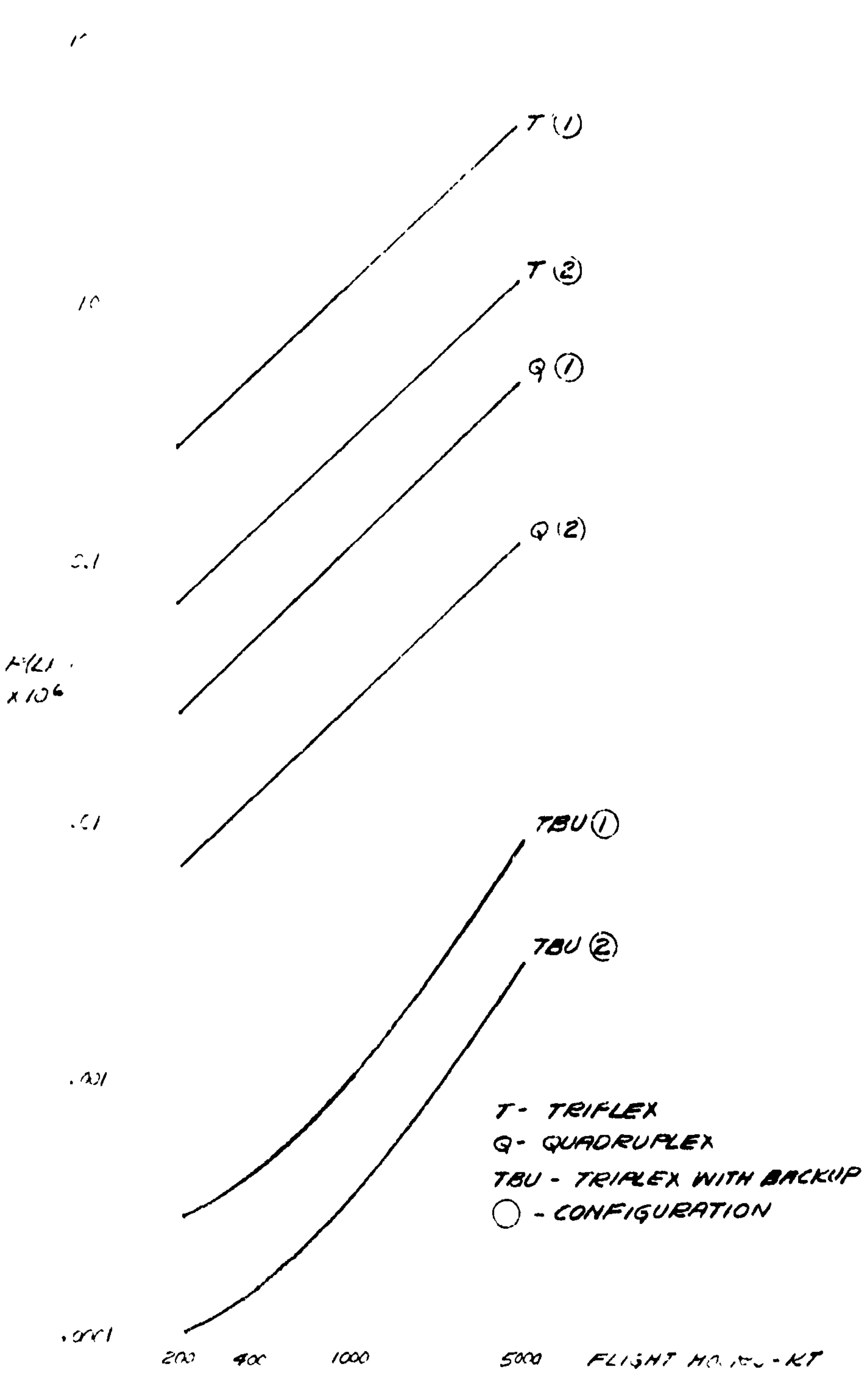


Figure 12. $P(I_K)$ Versus KT; $1- \alpha_i = .95$; $1= \alpha_p = .99$
Primary actuator failure rate = 0

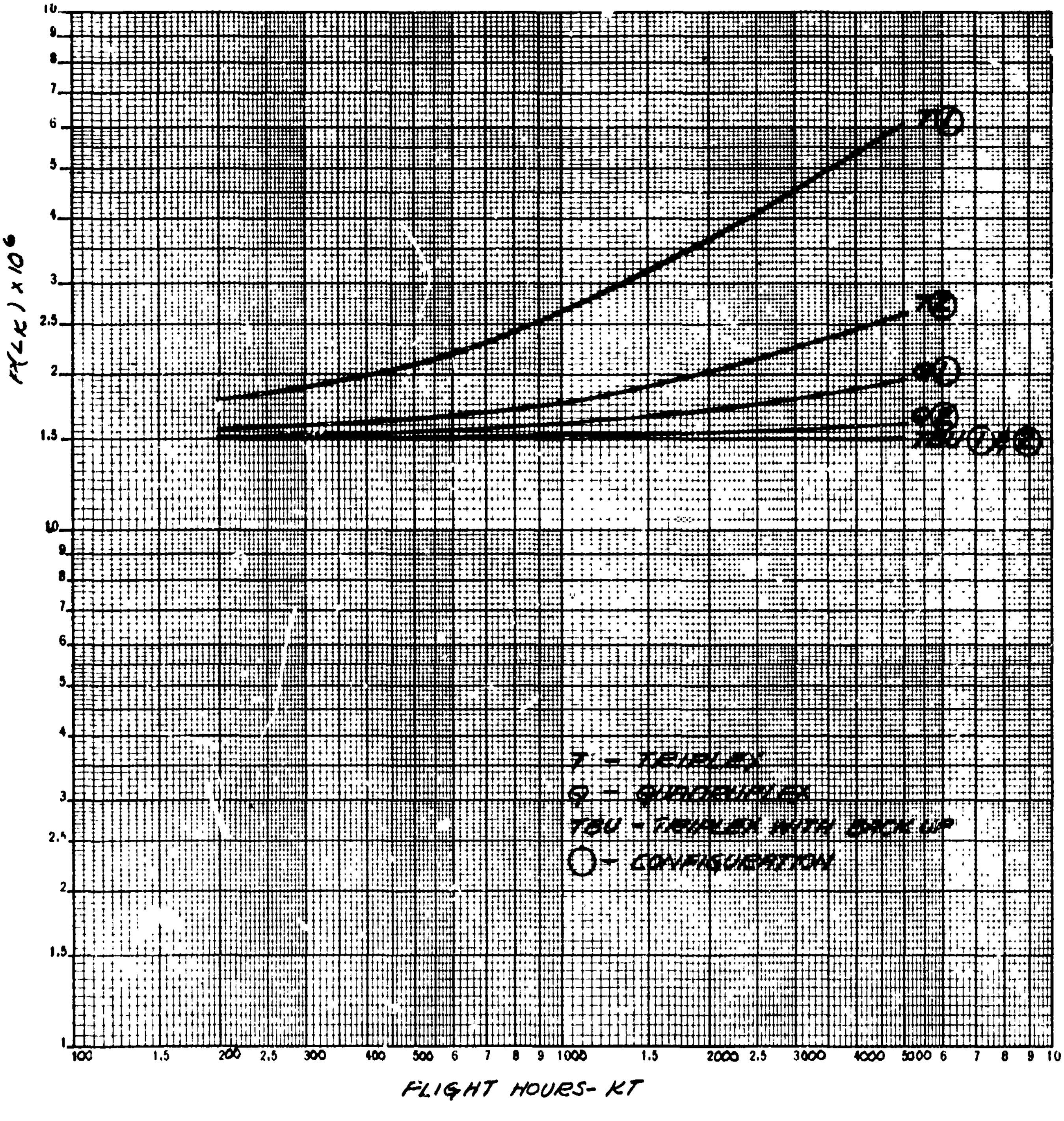


Figure 13. P(L_K) Versus KT; 1- α_i = .95; 1- α = .999
Primary actuator failure rate = .5 x 10^{-6}

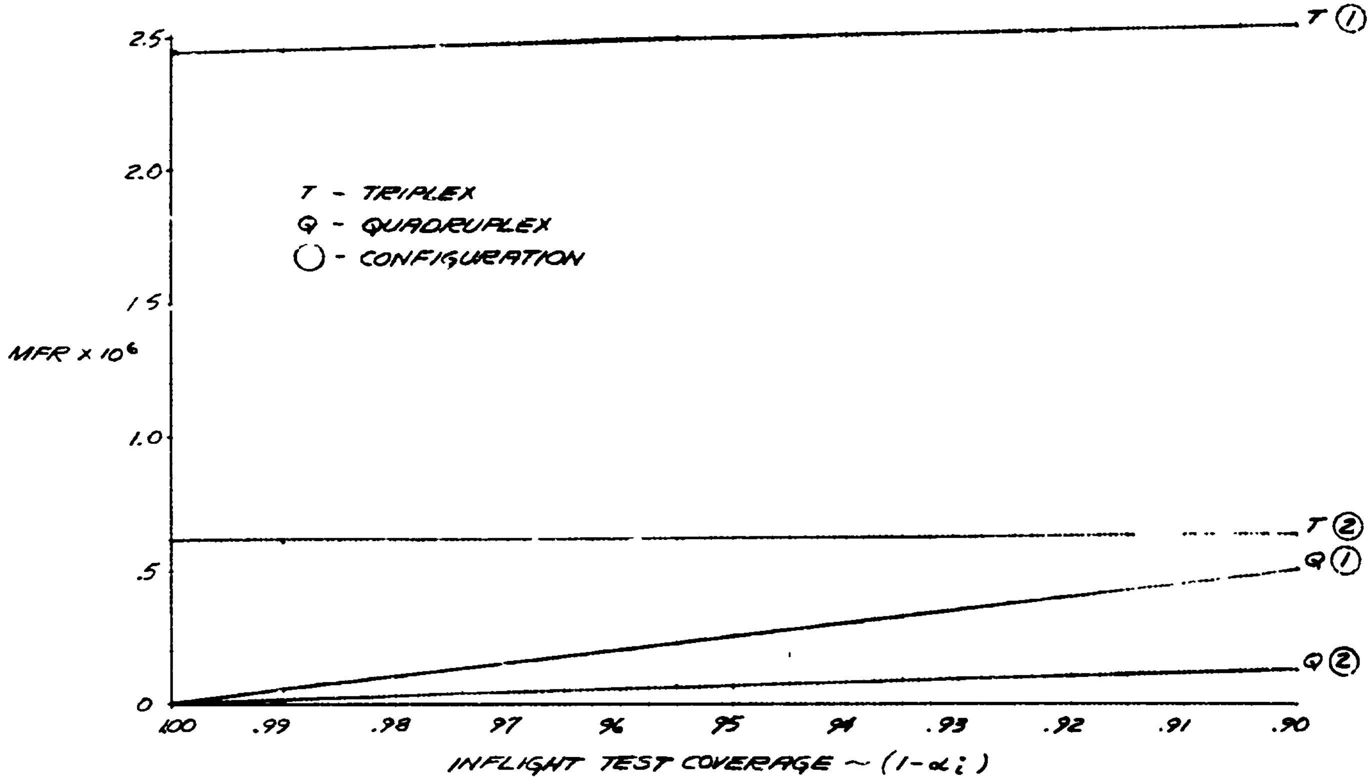


Figure 14. MFR Versus $1-\alpha_i$; $1-\alpha_p$ = .999
Primary actuator failure rate = 0

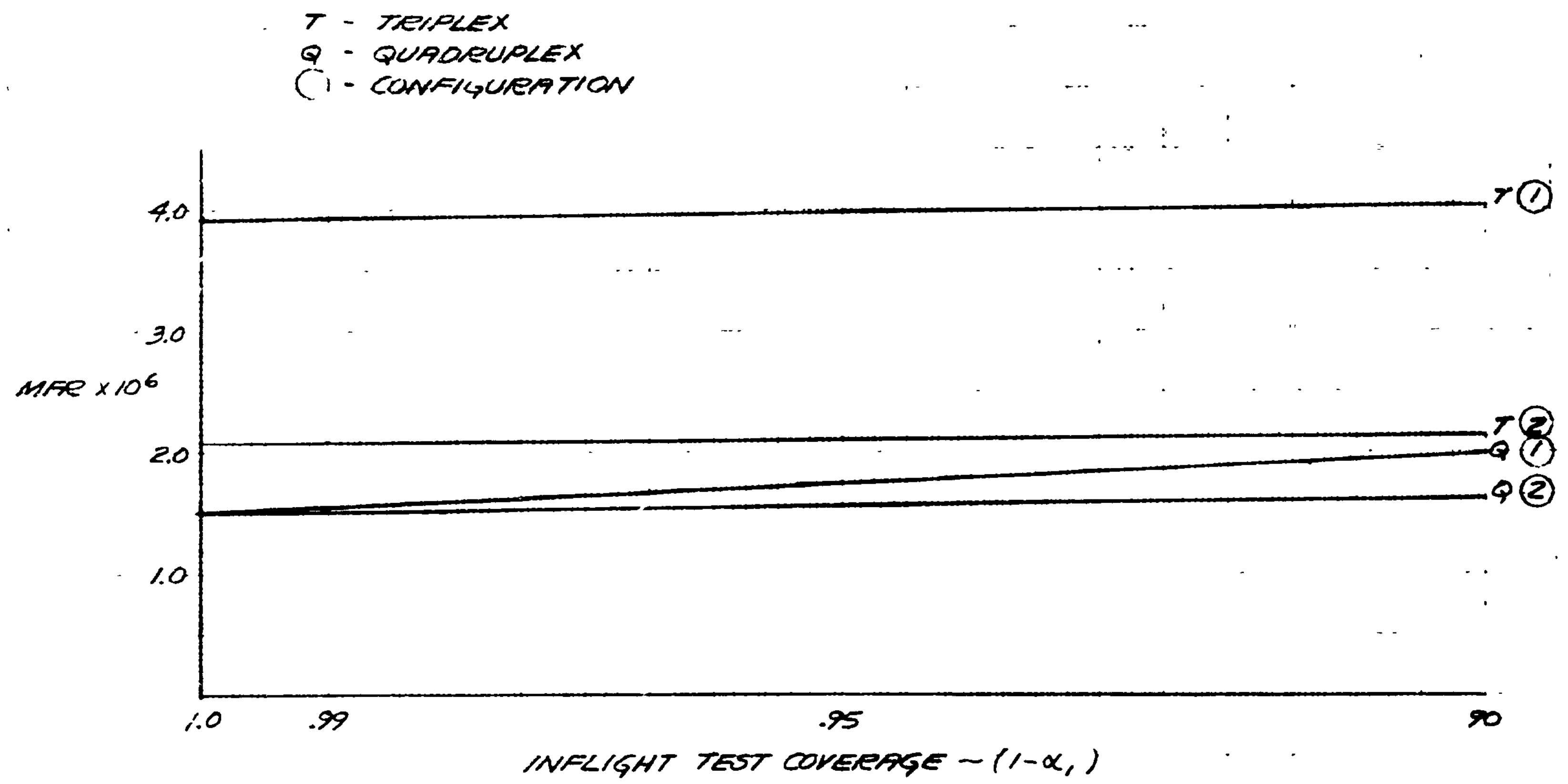


Figure 15. MFR Versus 1 - α_i; 1- α p = .999
Primary actuator failure rate =
.5 x 10^{-6}

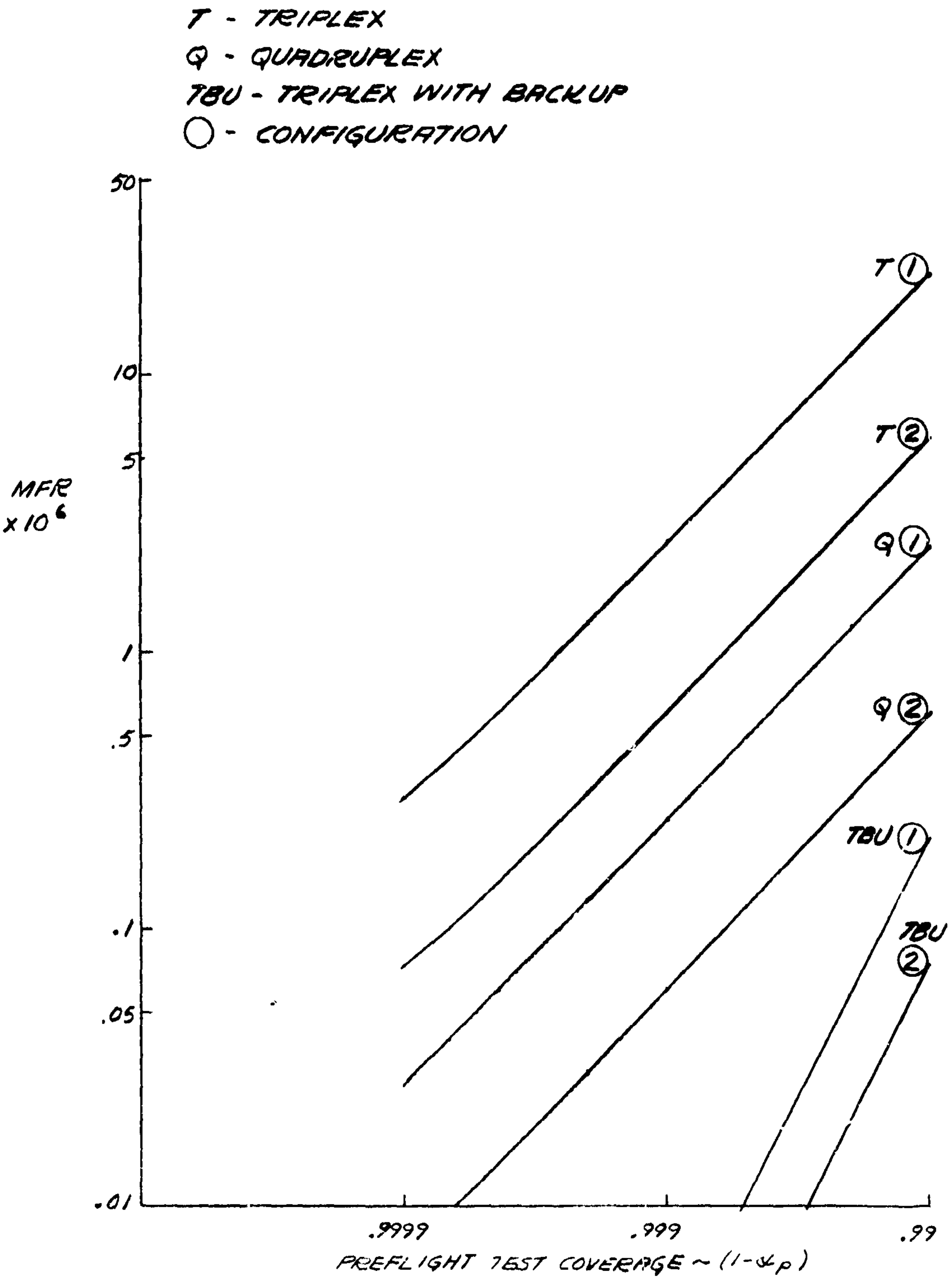


Figure 16. MFR Versus 1 - α_p; 1- α_i = .95 Primary actuator failure rate = 0

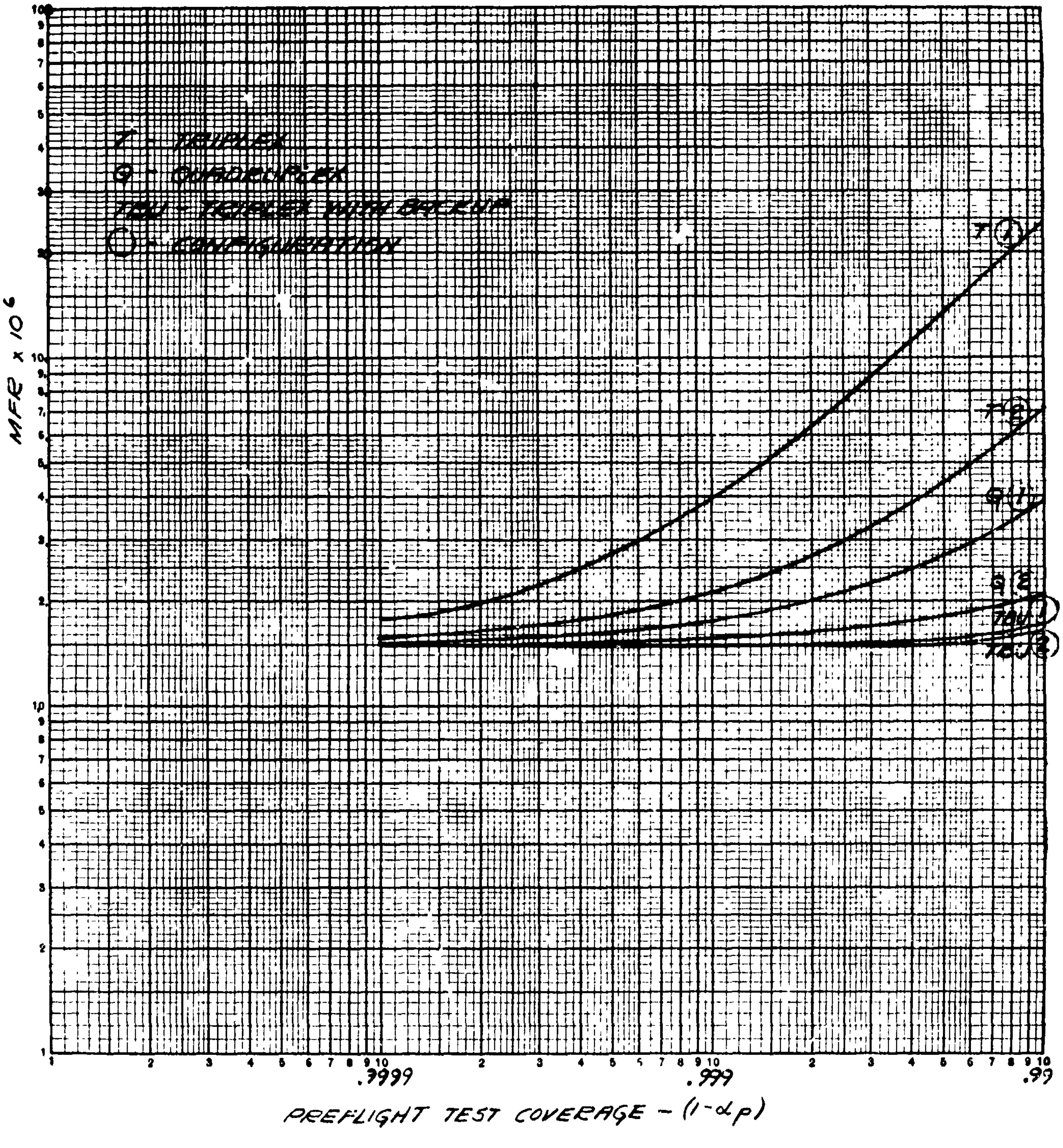


Figure 17. MFR Versus 1- αp; 1- α_i = .95
Primary actuator failure rate = .5 x 10^{-6}

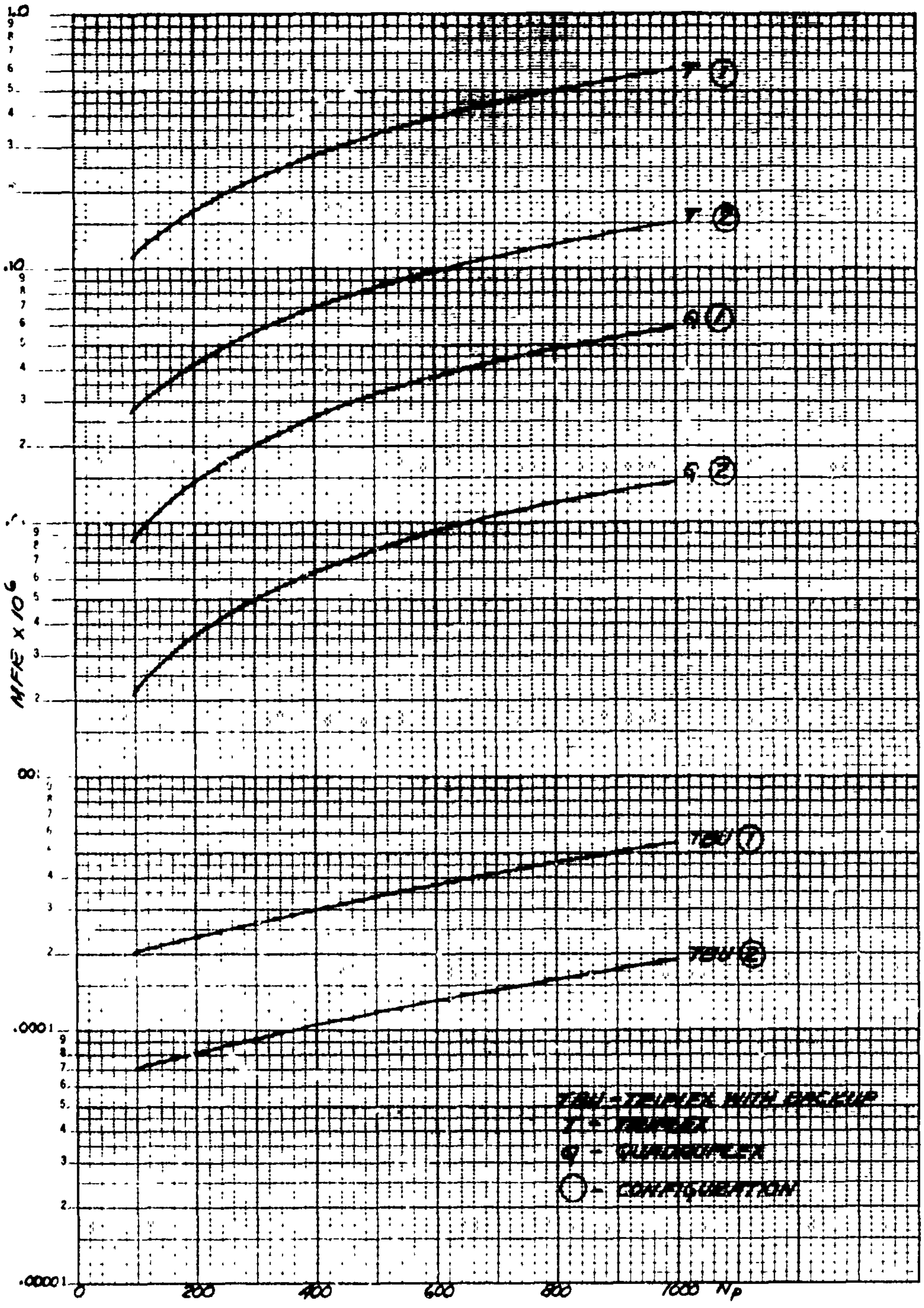


Figure 18. MFR Versus N_p; 1- α i = .95; 1-α_p = .999
Primary actuator failure rate = 0

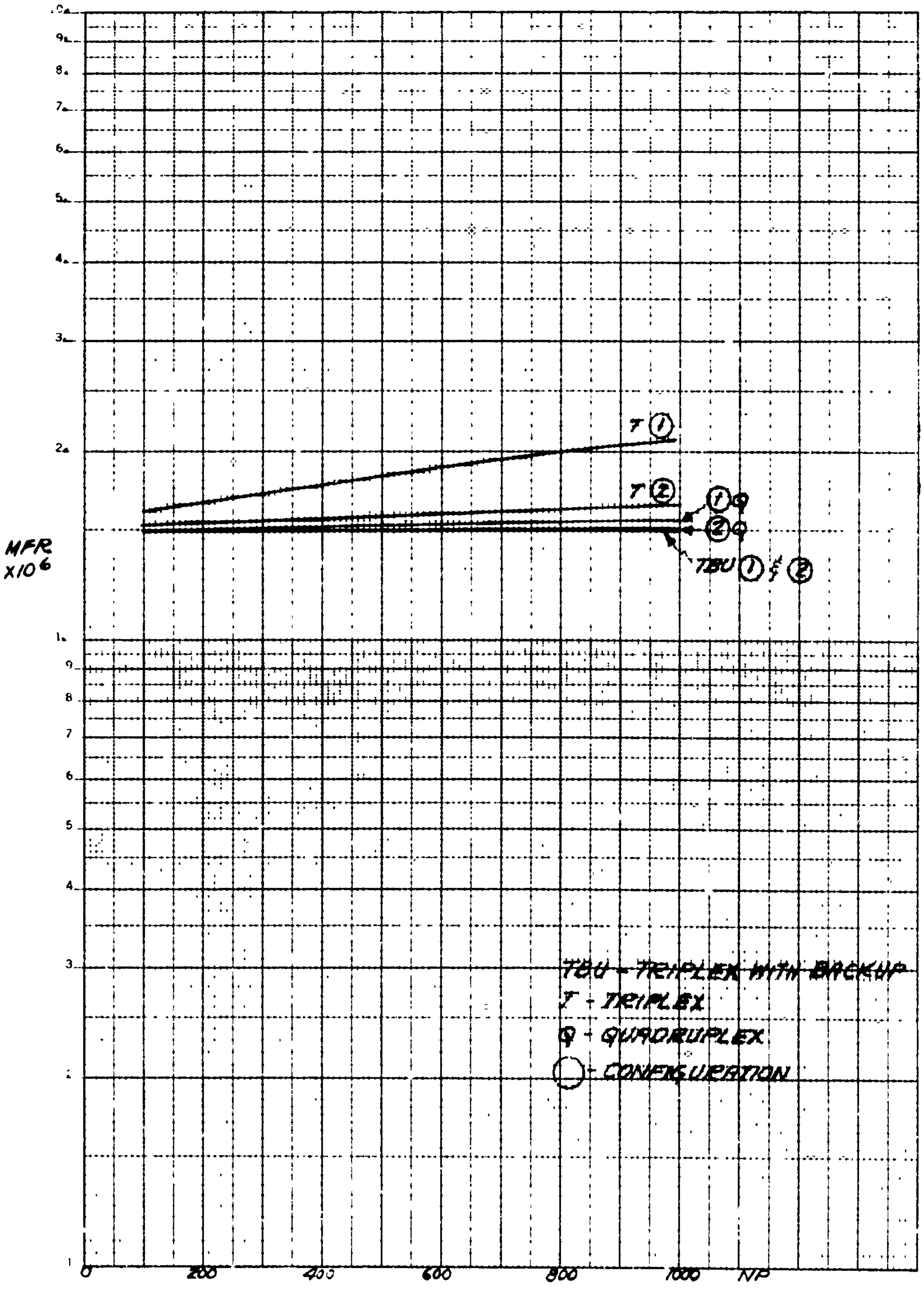


Figure 19. MFR Versus N_p; 1- αi = .95; 1- αp = .999
Primary actuator failure rate = .5 x 10^{-6}

SECTION 6

APPLICATION TO THE 680-J SURVIVABLE FLIGHT CONTROL SYSTEM 680-J SURVIVABILITY AIRPLANE (F-4)

1. Ground Rules

The 680-J Program incorporated five configurations:

- Present F-4 System
- Phase I: Simplex
- Phase IIA: FBW with Mechanical Back-Up
- Phase IIB: Same as IIA with Mechanical Back-Up Removed
- Phase IIC: Survivable Flight Control System With FBW

Phases IIA, IIB, and IIC are quadruplex configurations, with IIC representing the "ultimate" in mission reliability. The major difference between IIB and IIC is that in IIC the secondary and primary stabilator actuators are combined into a single package (called the SSAP, i.e., Survivable Stabilator Actuator Package). Hydraulic and Electrical power supplies are the same in both phases. As a point of comparison the failure rate for single point failures of the SSAP of IIC is 0.26 x 10^{-6}/hour whereas the corresponding failure rate for the primary actuator (stabilator) of IIB is 1.0 x 10^{-6}/hour (Ref. 3, Table V, page 39). Because the 680-J Program never reached the IIC phase it was decided to use the IIB phase for the Applications Study.

a. Phase IIB

For purposes of this study, we can characterize IIB as follows:

(1) Separate mechanical trim actuator

(2) One stabilator actuator

(3) FBW

(4) Lateral and directional axes are redundant in the sense that only one must function in order to return and land the airplane (Ref. 4, page 27). Thus, flight safety reliability is determined primarily by the catastrophic failure rate of the longitudinal axis.

(5) Longitudinal axis flight safety reliability only is being considered.

(6) Phase IIB is shown in Figure 20, which includes electrical and hydraulic supplies.

(7) Component Failure Rates used in the study are:

(a)	Primary Actuator	= 1.0×10^{-6}/hour
(b)	Secondary Actuator, Channel 1	= 188×10^{-6}/hour*
	Secondary Actuator, Channel 2	= 278×10^{-6}/hour*
	Secondary Actuator, Channel 3**	= 301×10^{-6}/hour*
	Secondary Actuator, Channel 4	= 188×10^{-6}/hour*
(c)	Digital Computer	= 120×10^{-6}/hour
(d)	Normal Accelerometer	= 8.1×10^{-6}/hour
	Pitch Rate Gyro	= 3.8×10^{-6}/hour
	Stick Force Sensor	= 7.8×10^{-6}/hour

All failure rates are those of IIB except that we have substituted a digital computer for the IIB pitch computer (failure rate = 25×10^{-6}/hour).

b. Two quadruplex and two triplex configurations were considered:

(1) Quadruplex with Comparison Monitoring

(a) No voting, (b) Fully voted (Figures 23 and 24, respectively)

*Includes Hydraulic Supplies
**Omitted in Triplex

(2) Triplex with Self-Tested Channels

(a) No voting, (b) Fully voted
(Figures 21 and 22, respectively)

Although it does not correspond to any 680-J configuration it was decided to duplicate all computations using a reduced stabilator actuator failure rate of 0.25 x 10^{-6} per hour. The resultant mission reliability represents a realistic goal and corresponds, at least approximately, to what can be attained in Phase IIC.

Flight Safety Reliability Goals

Estimates of catastrophic failures of the primary flight control system of the F-4 airplane are summarized as follows:

For carrier-based F-4's:

6.6 x 10^{-6} failures/hour.

Estimate obtained by the Air Force for non-carrier F-4's:

3.8 x 10^{-6} failures/hour.

Calculated for standard F-4's:

1.145 x 10^{-6} failures/hour.

From these estimates we may conclude that a calculated FBW F-4 failure rate should not greatly exceed 1.145 x 10^{-6} failures/flight hour.

2. Results

Figures 25, 27, 29, 31, 33, 35, refer to the 680J, Phase IIB configuration as defined in Figure 20. Figures 26, 28, 30, 32, 34, and 36 are the corresponding figures except that the stabilator actuator failure rate has been reduced from 1.0 x 10^{-6} to 0.25 x 10^{-6} failures/hour.

Figures 25, 26. $P(L_\infty)$ Versus $(1-\alpha_i)$; $(1-\alpha_p) = 1.0$

These figures substantiate an earlier conclusion that probability of loss of system is not strongly dependent on in-flight test coverage, at least among the values selected. As a design objective, which appears to be attainable, we will, henceforth, assume that inflight test coverage is 95%.

Figures 27, 28. $P(L_\infty)$ Versus $(1-\alpha_p)$; $(1-\alpha_i) = .95$

These are important figures because $P(L_\infty)$ = MFR for any airplane with a long service life. From Figure 27, we may conclude that Triplex (1) is unacceptable. Comparing Triplex (2) with Quad (2) indicates that Quad (2) requires an order of magnitude less in preflight test coverage e.g., 99% in Quad (2) and 99.9% in Triplex (2) to achieve the same $P(L_\infty)$. Since it will be extremely difficult to achieve a 99.9% preflight test coverage (and to prove that it has been achieved), Quad (2) is the recommended configuration. At this stage in the development of FBW systems, we believe that the additional safety is well worth the extra cost and complexity of the quad configuration.

Figures 29, 30, $P(L_K)$ Versus KT; $(1-\alpha_i) = .95$, $(1-\alpha_p) = .999$

These figures show the degradation of flight safety reliability with time. Again, the quadruplex configurations are superior.

Figures 31, 32 MFR Versus $(1-\alpha_i)$; $(1-\alpha_p) = .999$

As in Figures 25 and 26, these figures show that inflight test coverage does not strongly influence mean failure rate, at least for the coverages selected.

Figures 33, 34 MFR Versus $(1-\alpha_p)$; $(1-\alpha_i) = .95$

Referring to Figure 33, it can be seen that Quad (1) and Quad (2) are both acceptable with a preflight coverage of 99.9% and Quad (2) is probably acceptable with a coverage of 99.0%. With the improved actuator, Quad(2), with a coverage of 99.9%, results in an MFR of approximately 1.0×10^{-6} failures/hour. The Triplex (2), on the other hand, shows almost no improvement between the existing and improved actuators with a coverage of 99.0%.

Figures 35, 36. MFR Versus N_p; $(1-\alpha_i) = .95$, $(1-\alpha_p) = .999$

These figures show that even a relatively modest periodic test can provide a significant improvement in MFR for all configurations. The dashed lines correspond to the MFR values of the respective configurations with no periodic testing.

3. Conclusions

a. With perfect preflight test coverage and a relatively modest inflight test coverage both the triplex and quadruplex configurations yield acceptable flight safety reliability.

b. The triplex configuration tends to be more sensitive to latent failures than the quad configuration. In the triplex, a latent failure followed by a failure in another channel, whether detected or not, could result in loss of the airplane. In the quad arrangement loss of the airplane requires two undetected failures.

c. The triplex configuration requires a preflight test coverage of .999, or better, in order to meet the reliability goals. The quad configuration requires a coverage between .99 and .999.

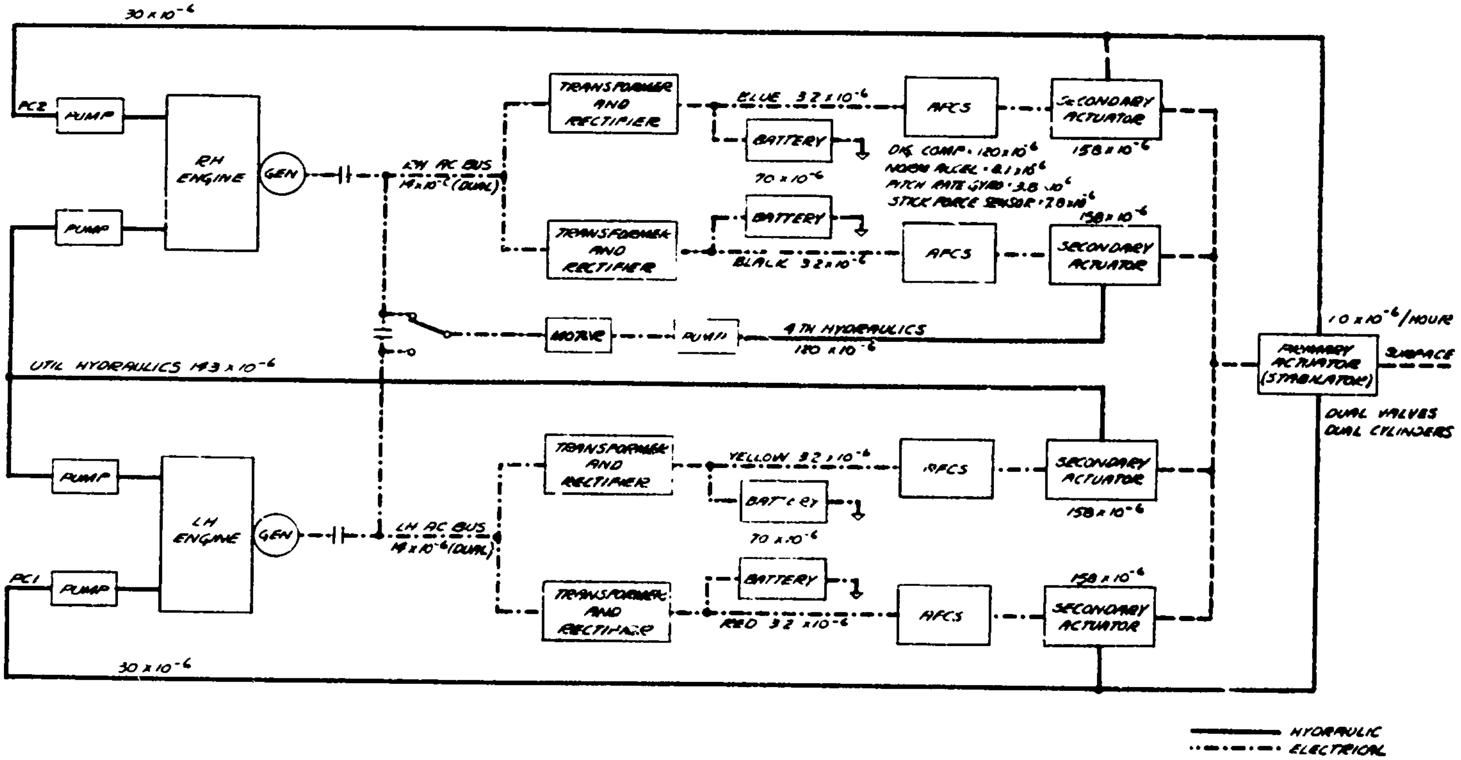


Figure 20. 680-J Survivable flight control system (F-4) pitch channel, phase IIB, FBW

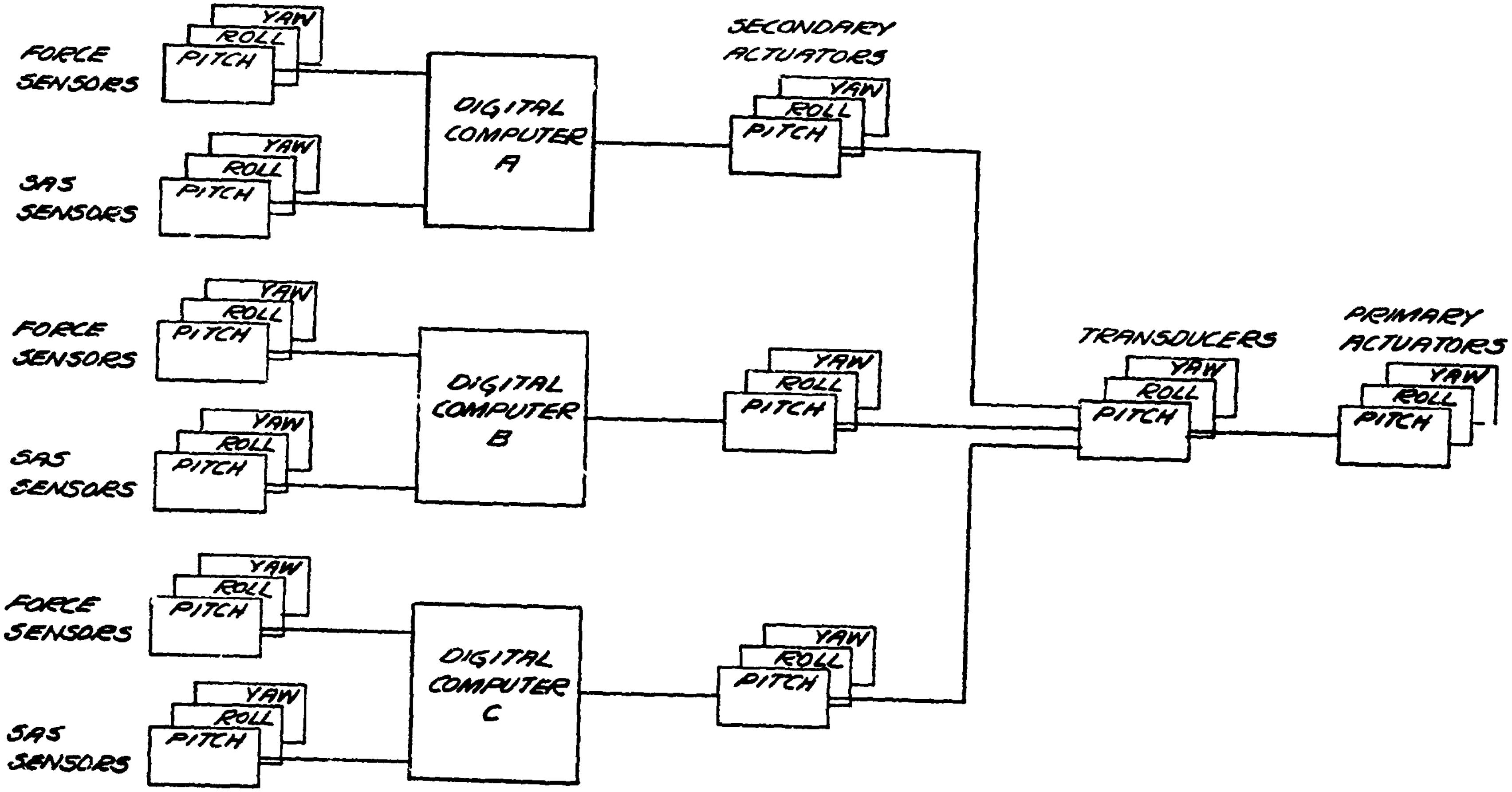


Figure 21. Triplex configuration 1

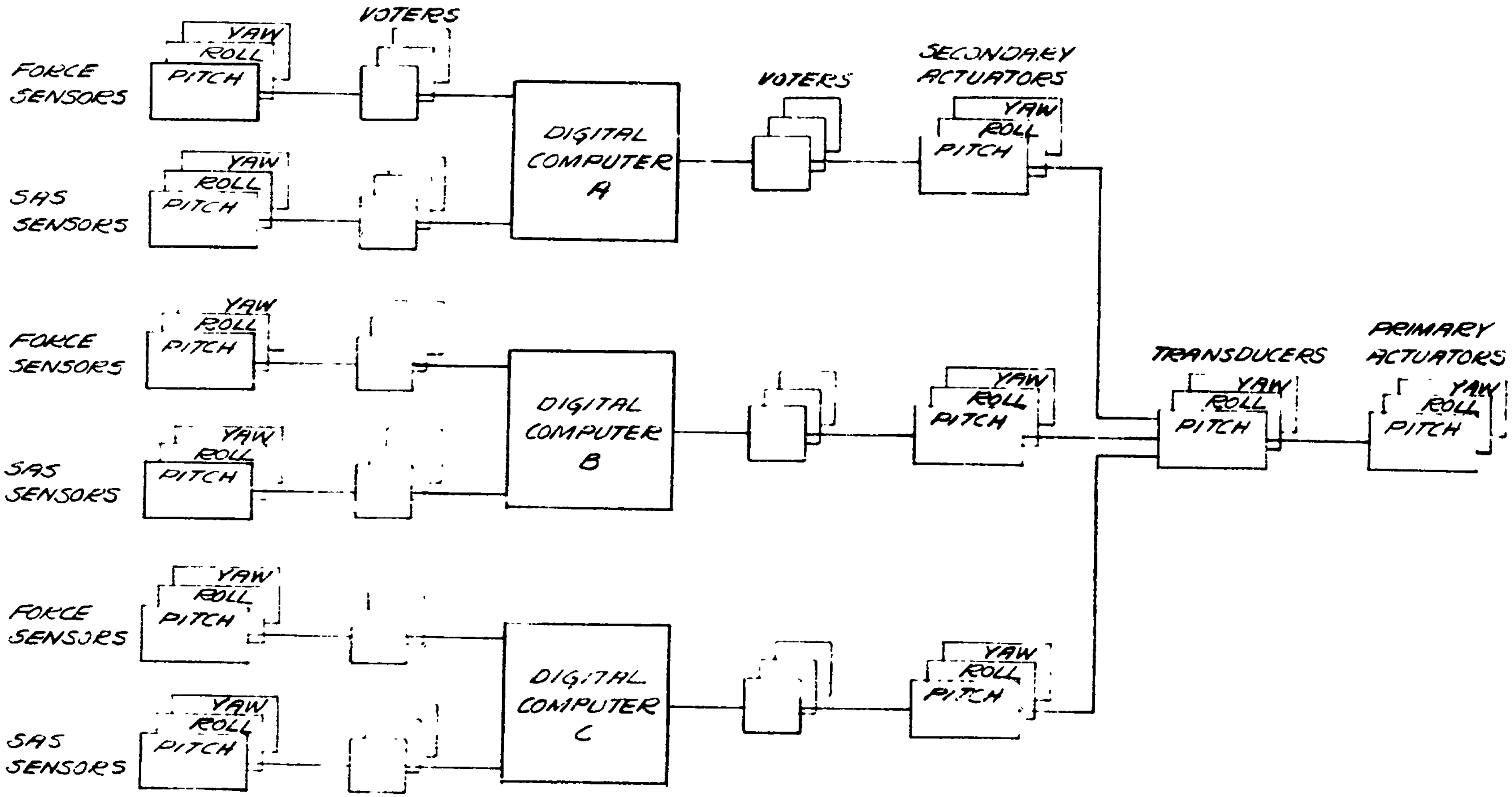


Figure 22. Triplex configuration 2

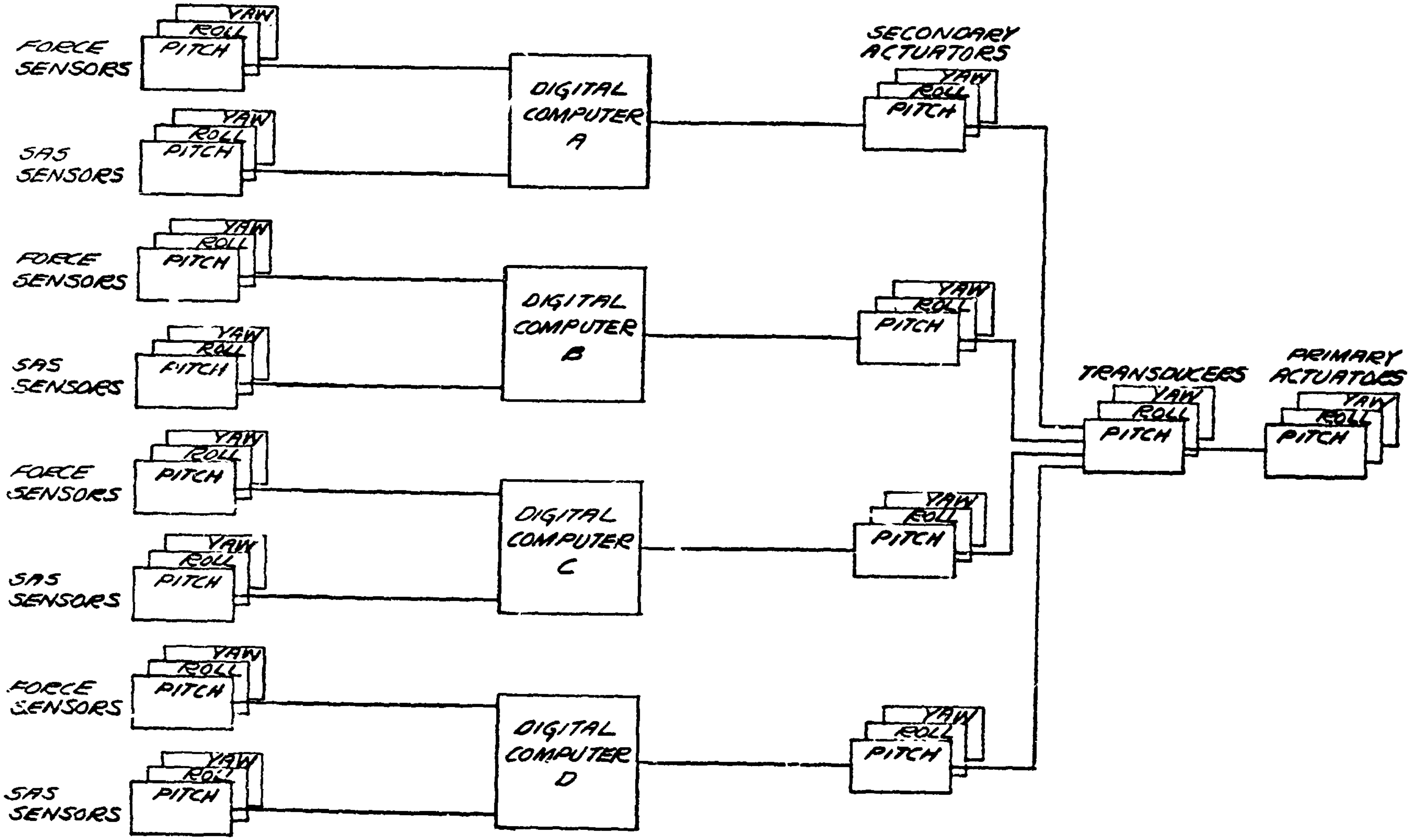


Figure 23. Quadruplex configuration 1

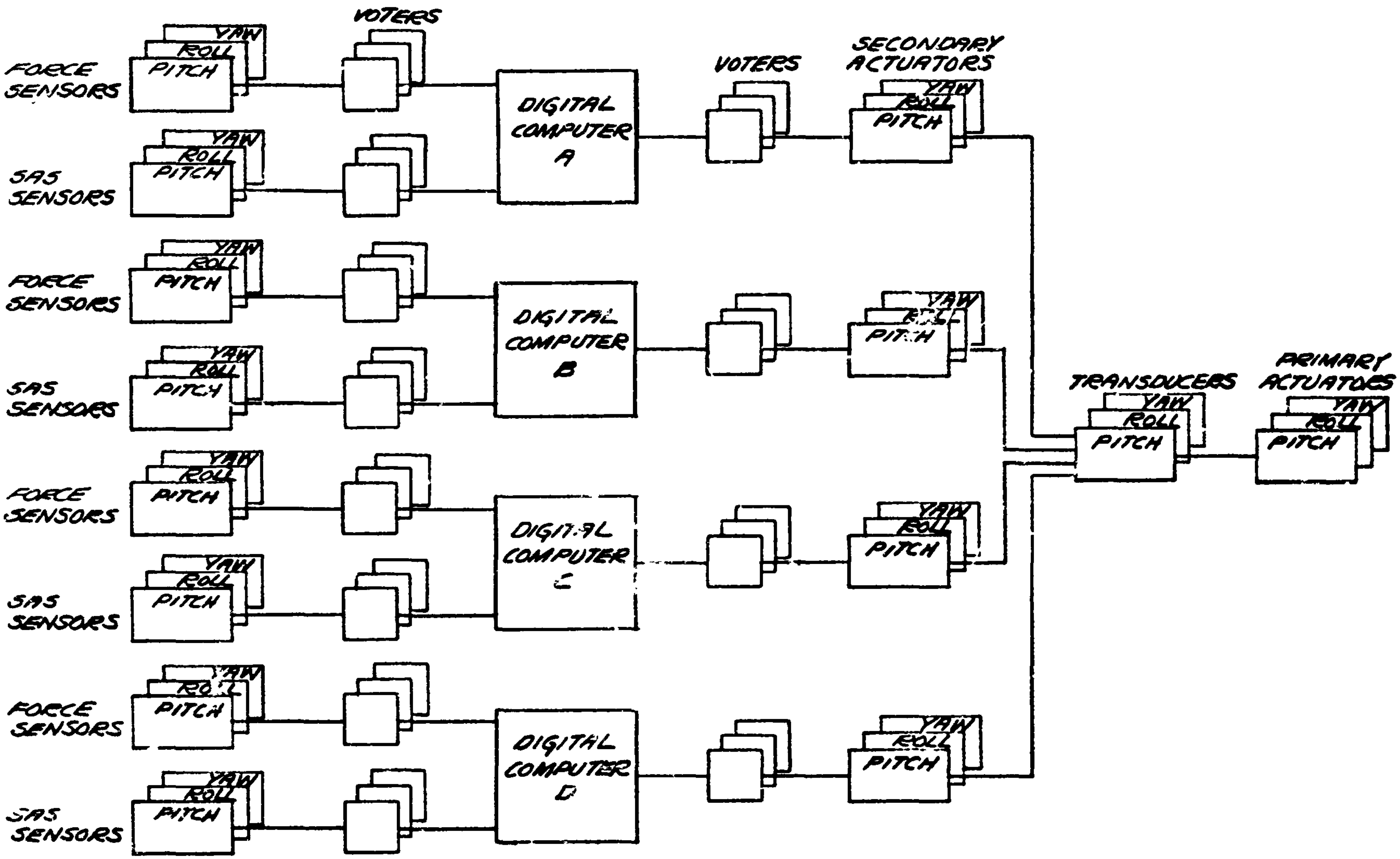


Figure 24. Quadruplex configuration 2

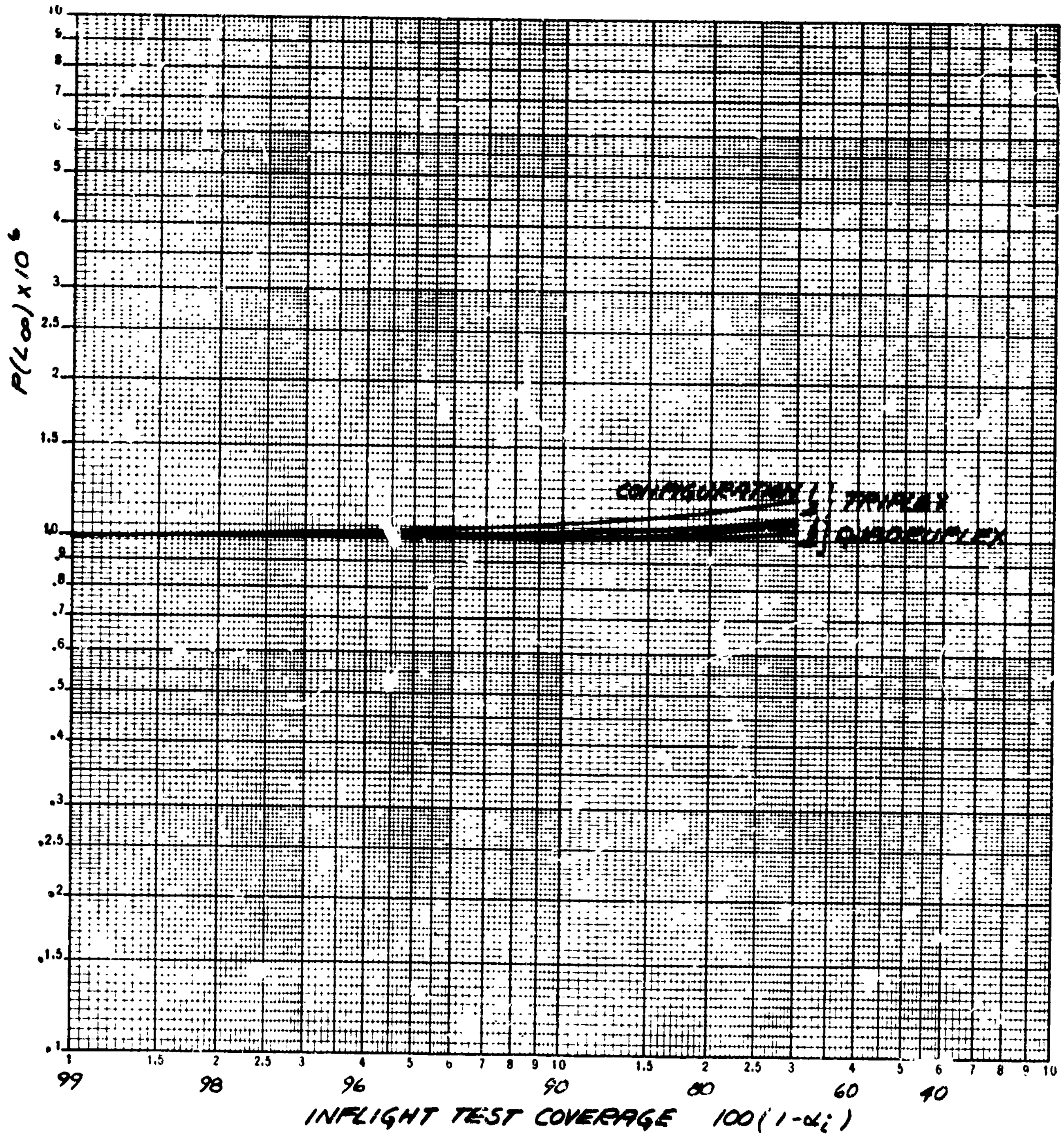


Figure 25. 680-J Airplane IIB P(L ∞) versus 1- α_i; 1- α_p = 1.0

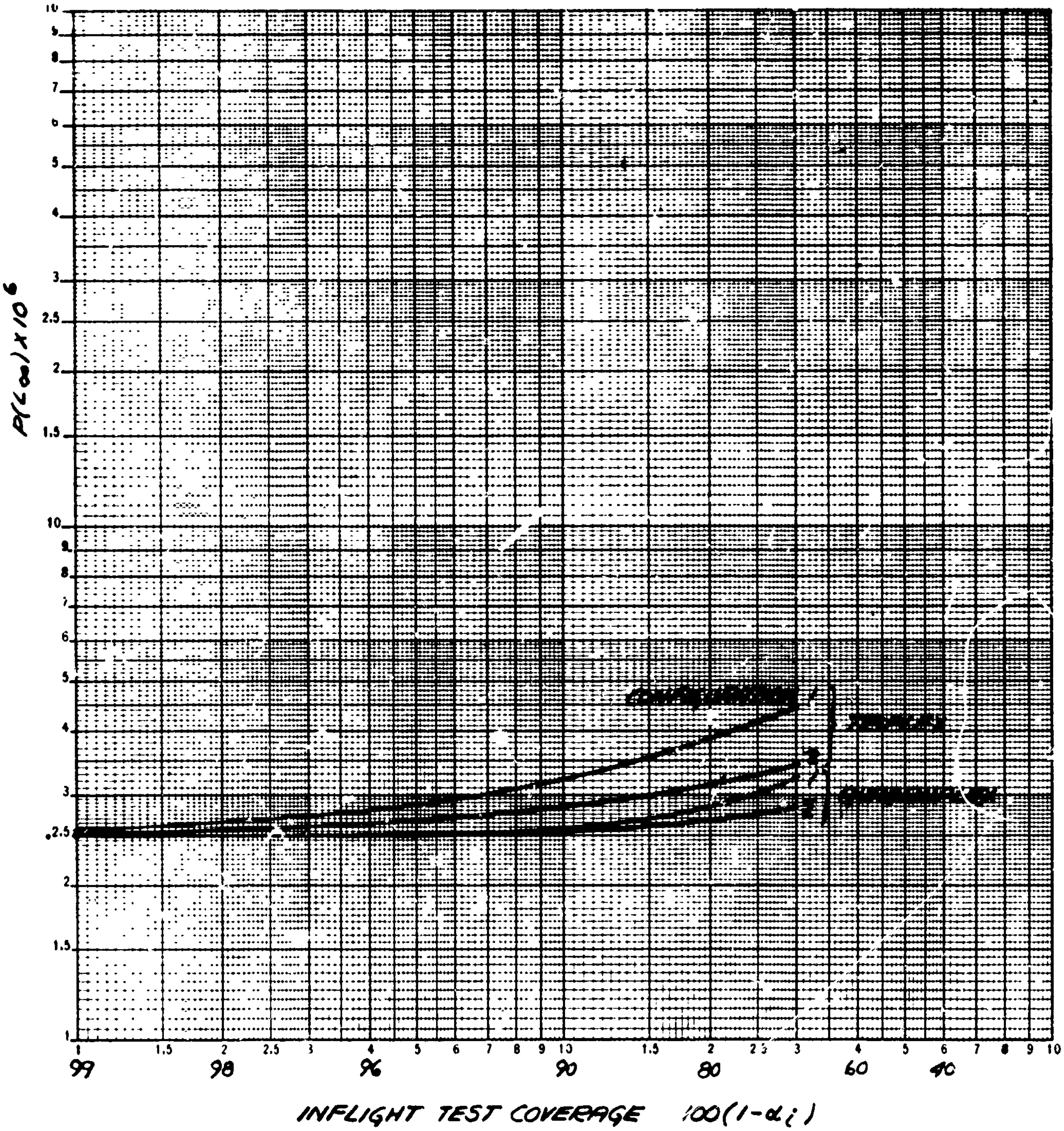


Figure 26. 680-J Airplane IIB P (L ∞) versus 1- α_i; 1- α_p = 1.0 primary actuator failure rate = .25 x 10-6

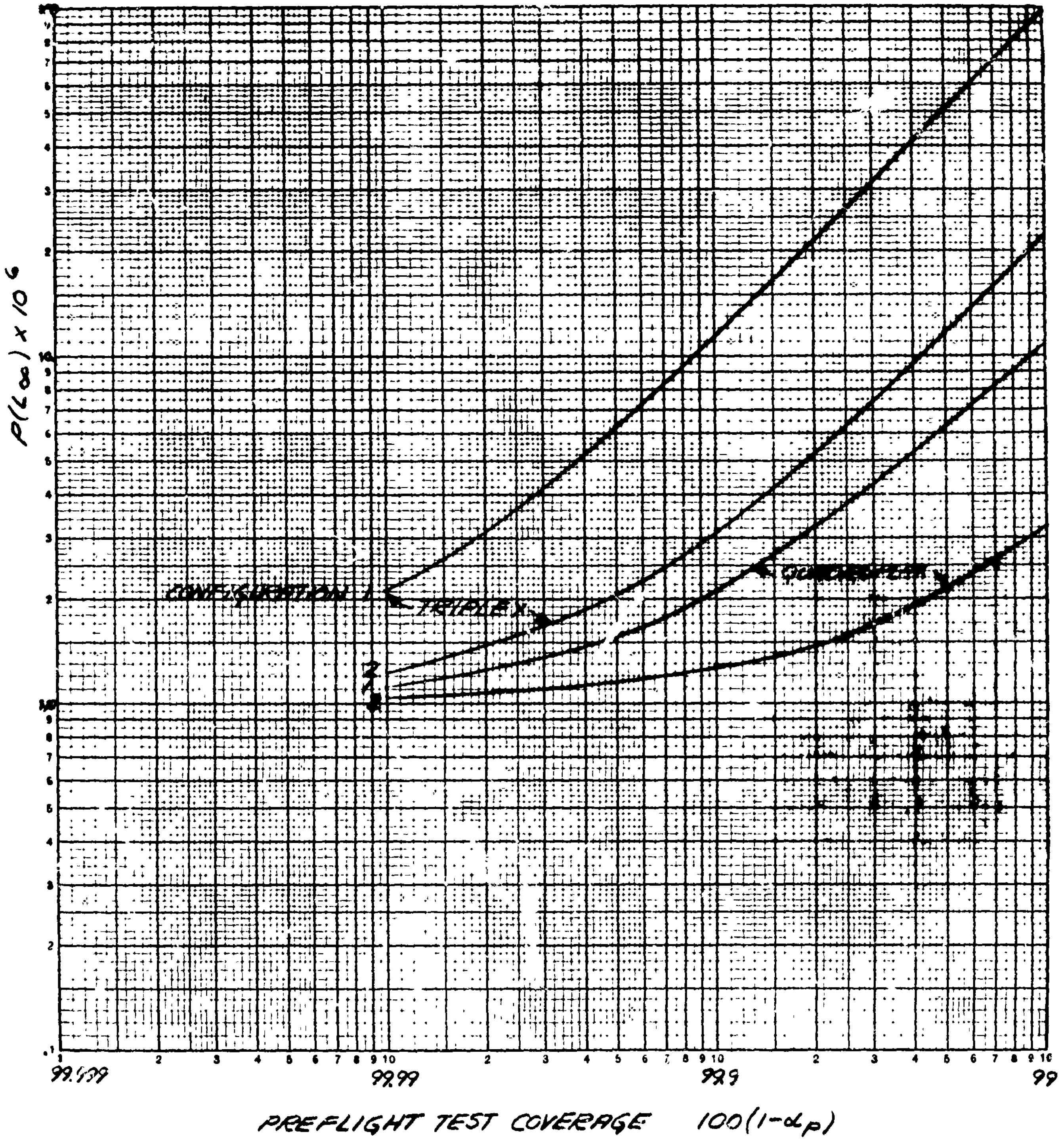


Figure 27. 680-J Airplane :IB P(L ∞) versus 1- α_p; 1- α_i = .95

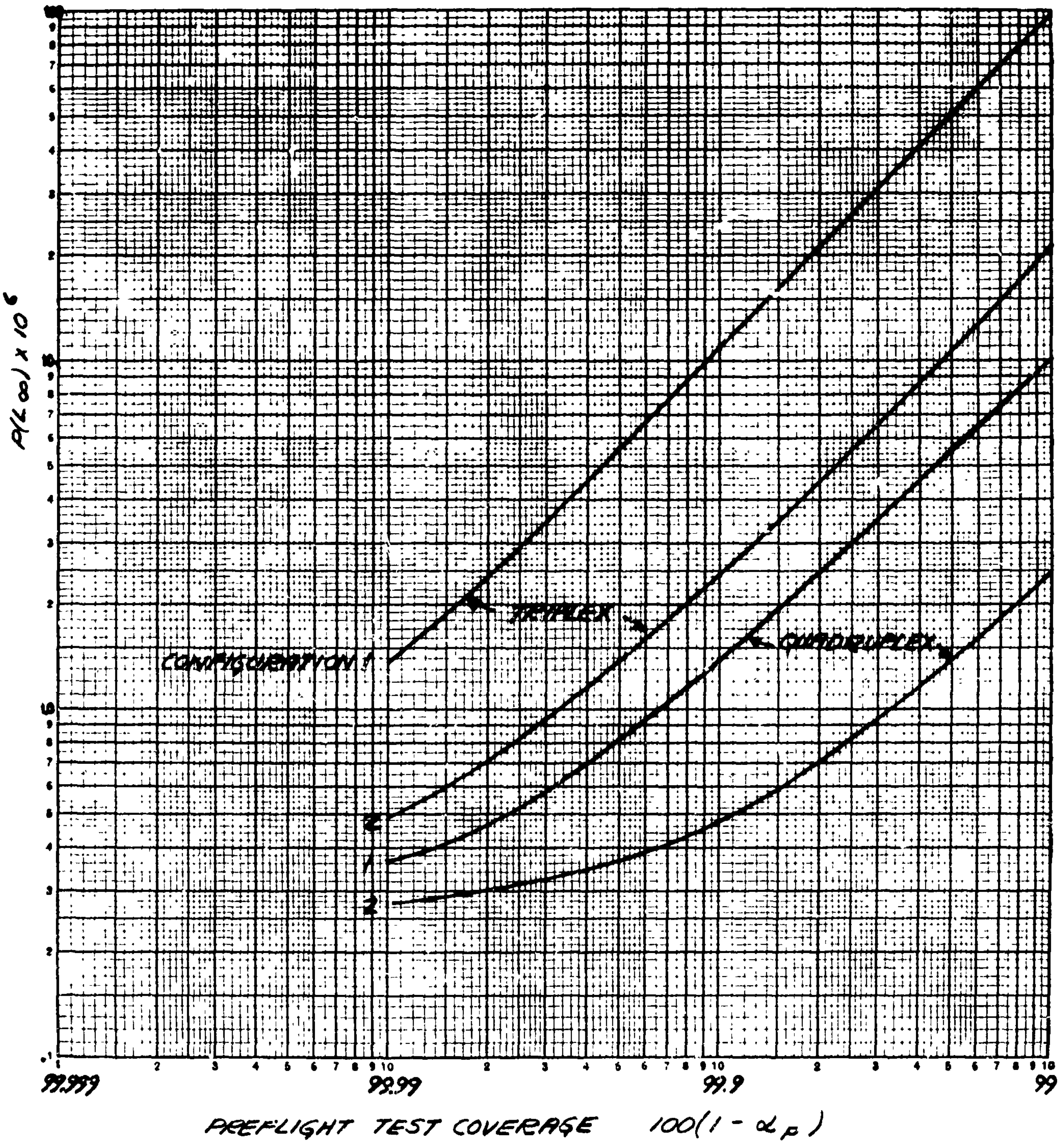


Figure 28. 680-J Airplane IIB $P(L_\infty)$ versus $1-\alpha_p$; $1-\alpha_i = .95$
Primary actuator failure rate = $.25 \times 10^{-6}$

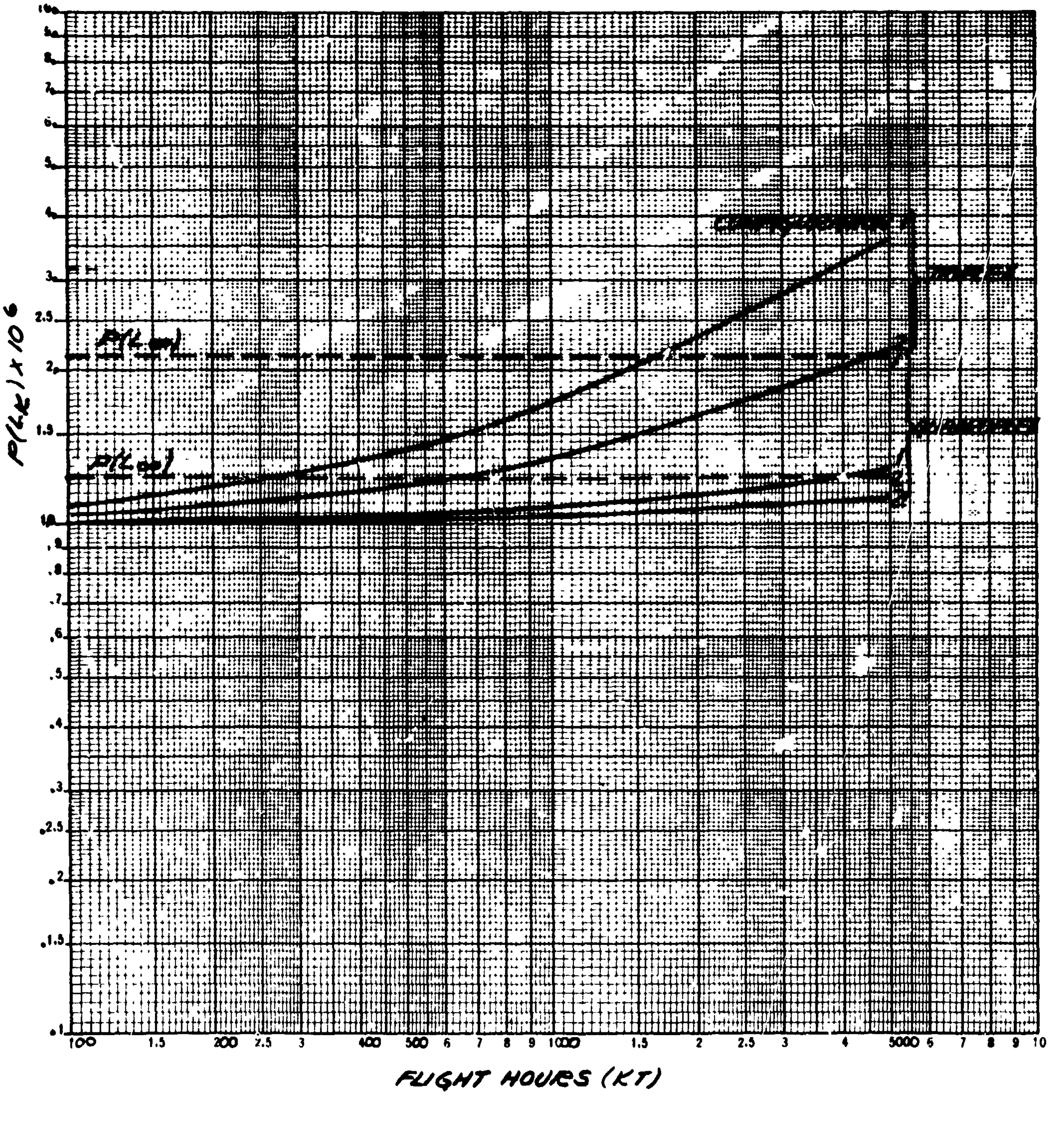


Figure 29. 680-J Airplane IIB $P(L_K)$ versus KT; $1-\alpha_i = .95$; $1-\alpha_p = .999$

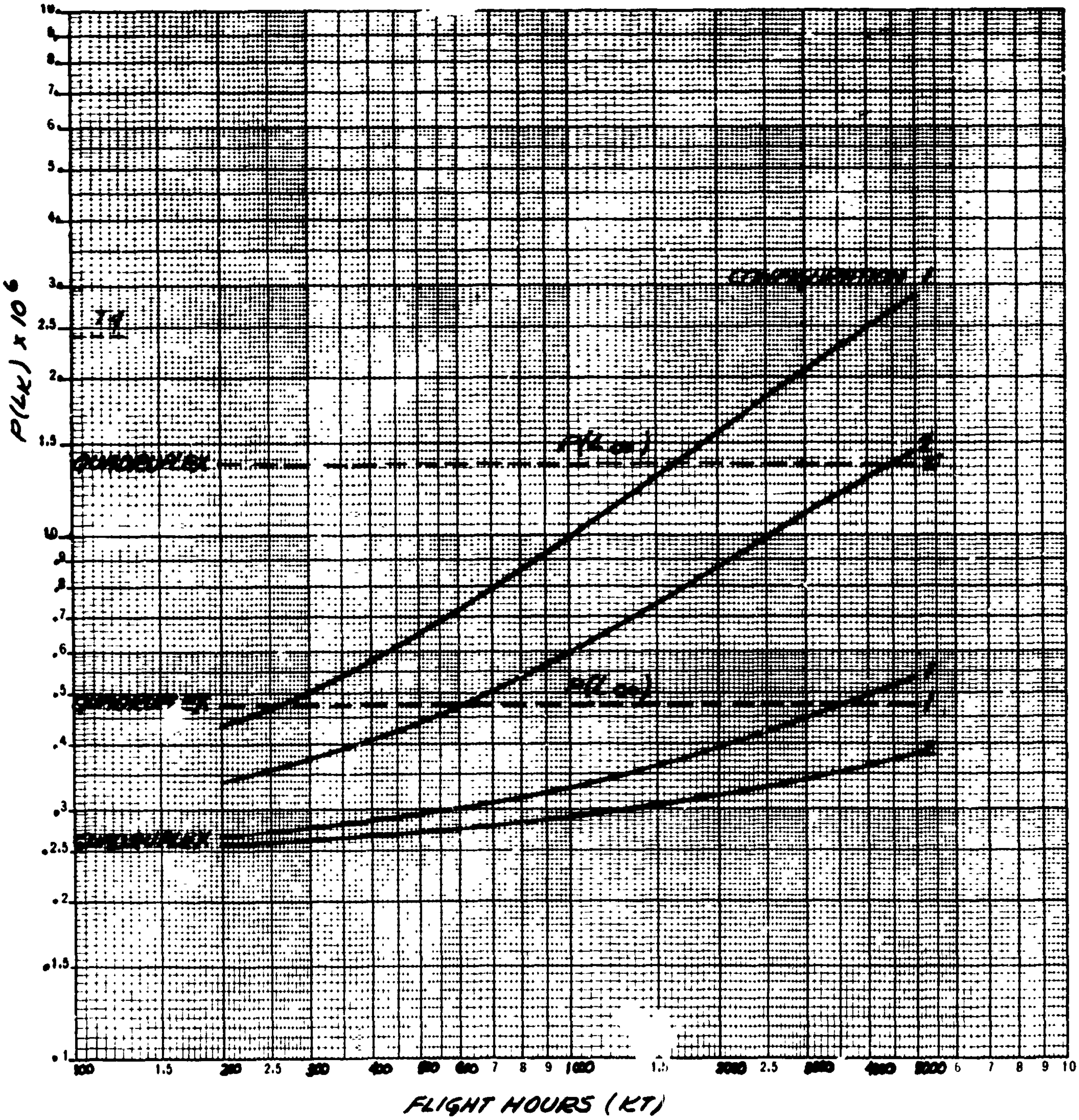


Figure 30. 680-J Airplane IIB $P(L_K)$ versus KT; $1-\ \sigma_i = .95$; $1 = \alpha_p = .999$ primary actuator failure rate = $.25 \times 10{-}6$

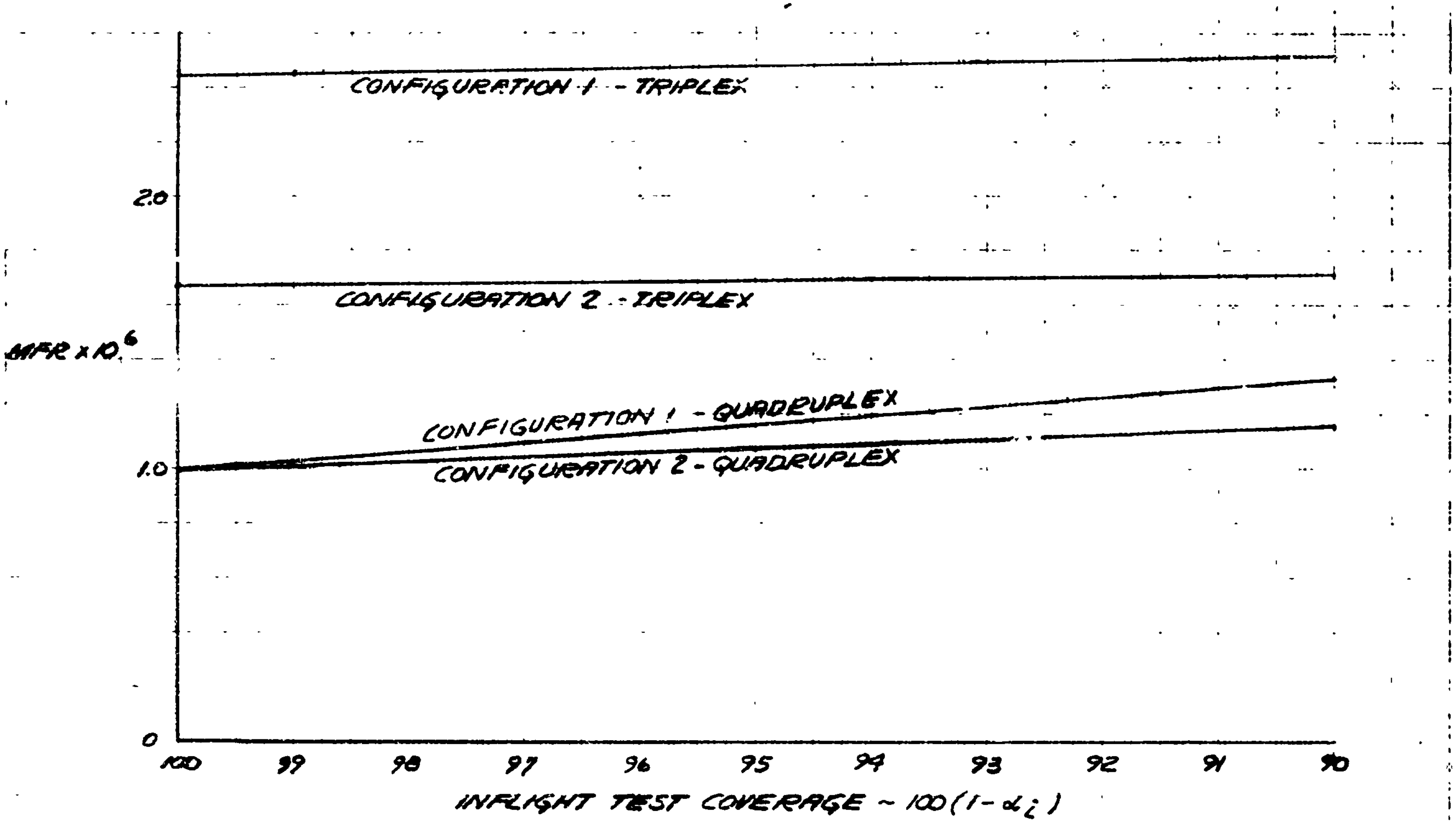


Figure 31. 680-J Airplane IIB MFR versus 1- α_i; 1- α_p = .999

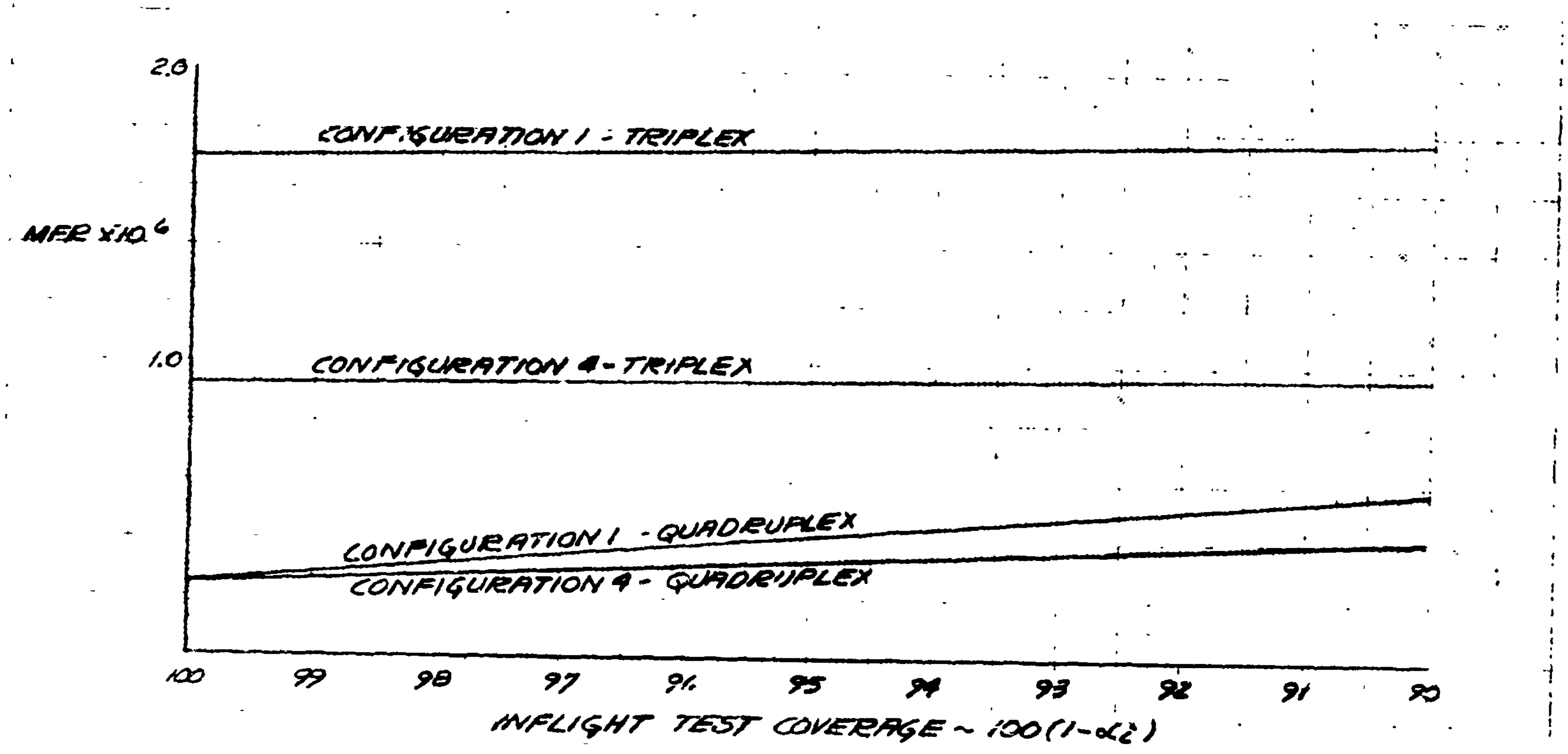


Figure 32. 680-J Airplane IIB MFR versus $1-\alpha_i$; $1-\alpha_p = .999$
Primary actuator failure rate = $.25 \times 10^{-6}$

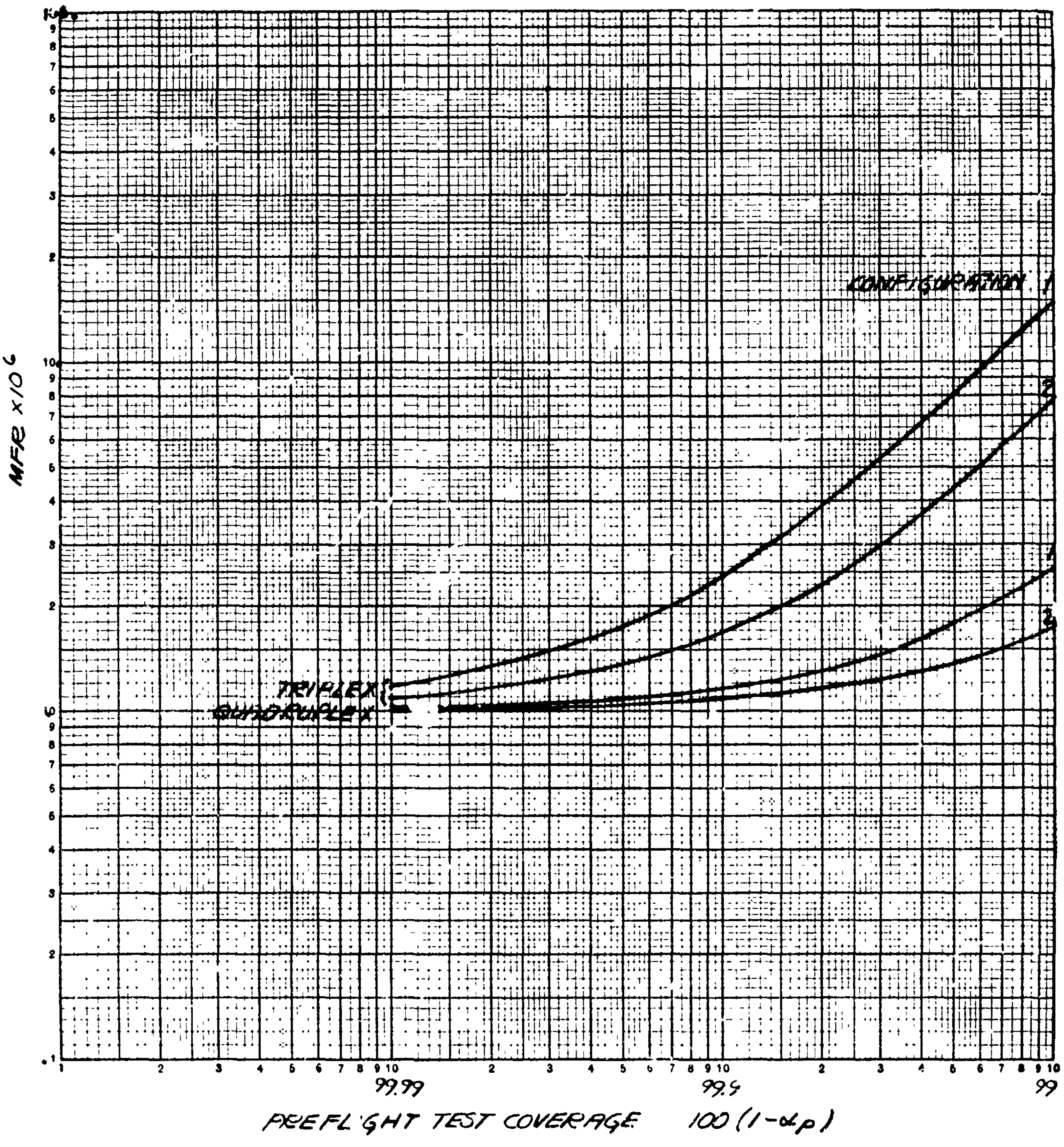


Figure 33. 680-J Airplane IIB MFR versus $1-\alpha_p$; $1-\alpha_i = .95$

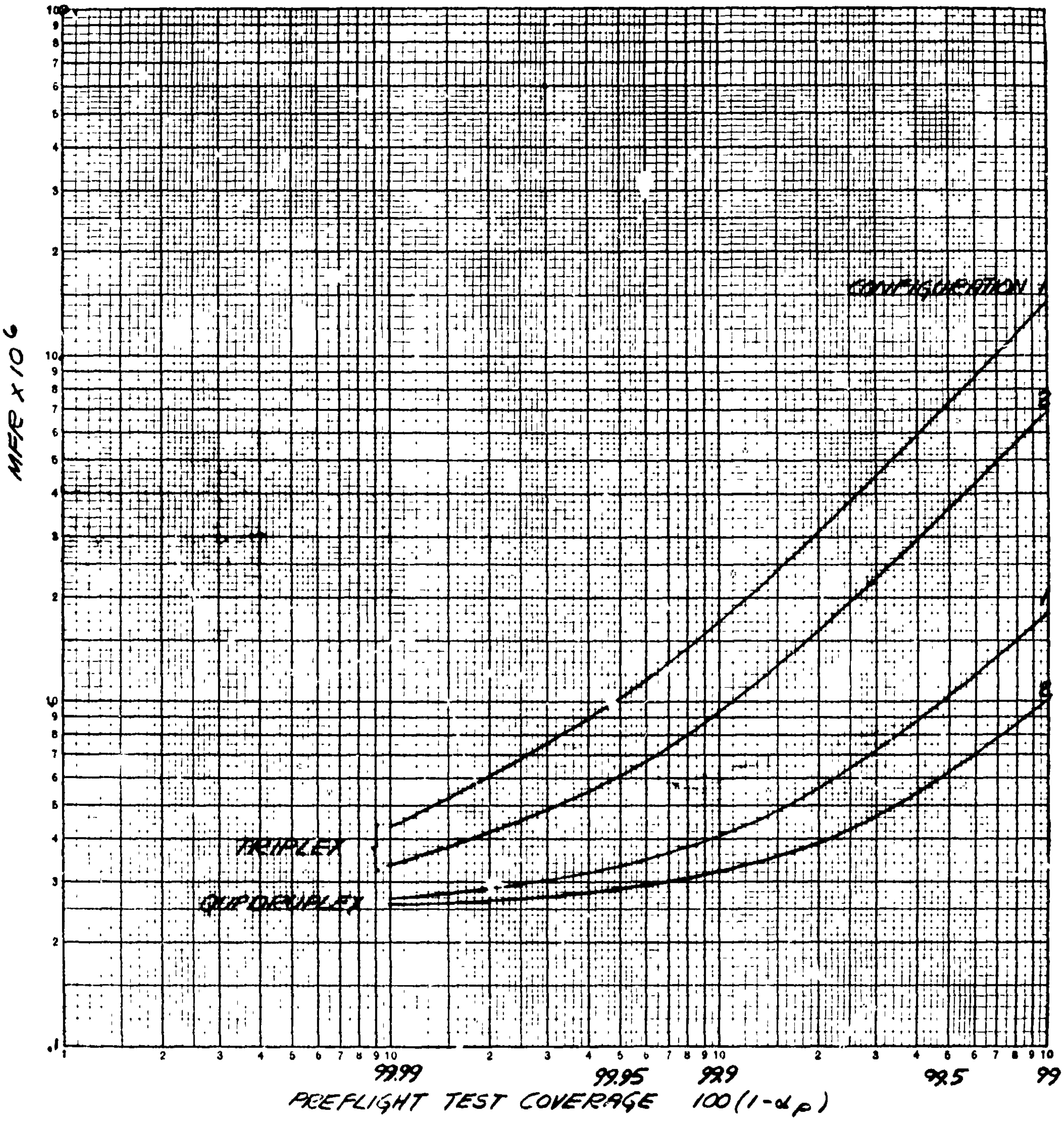


Figure 34. 680-J Airplane IIB MFR versus 1- α_p; 1- α_i = .95
Primary actuator failure rate = .25 x 10^{-6}

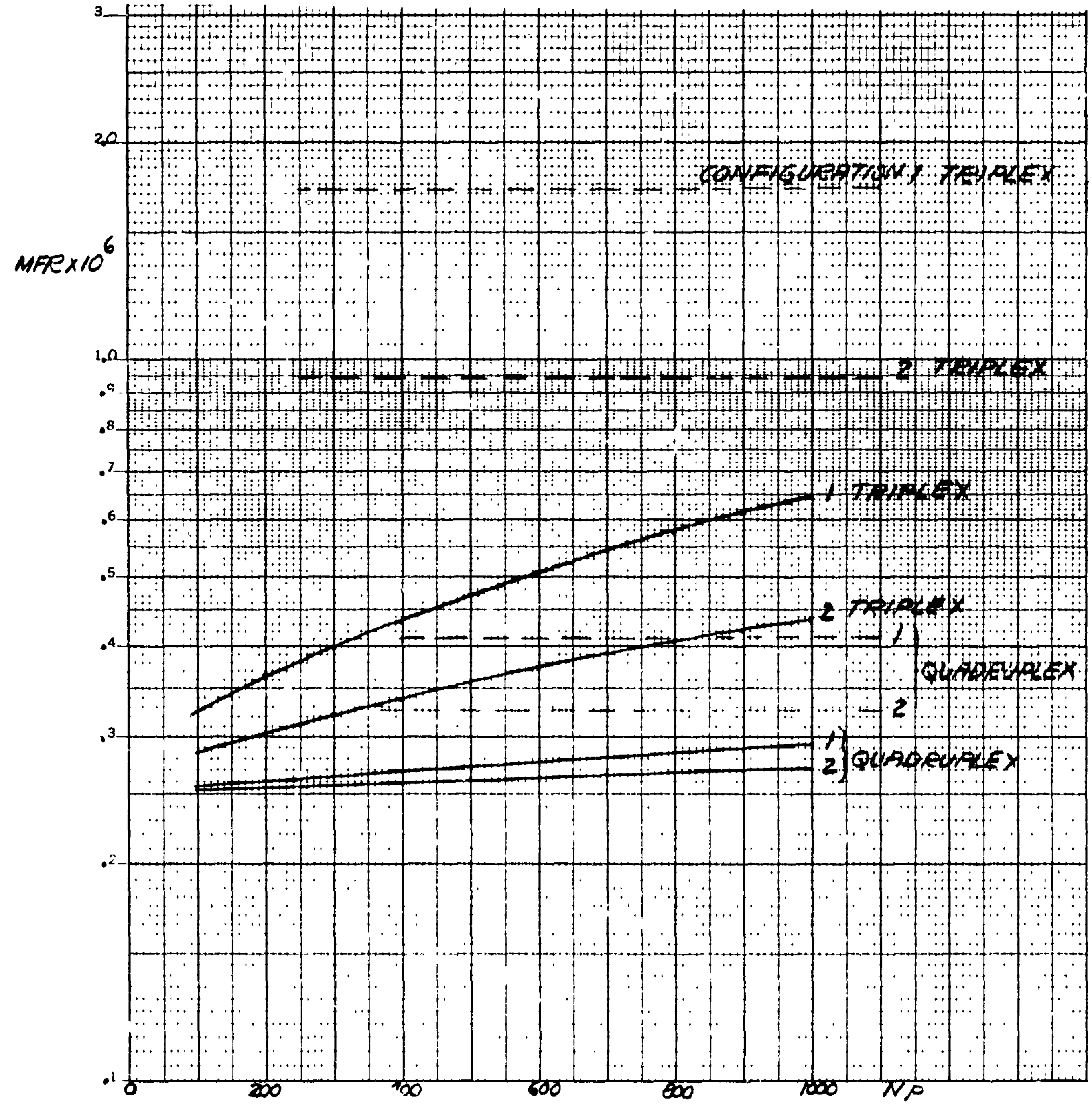


Figure 35. 680-J Airplane IIB MFR versus N_p; 1- α_i = .95; 1- α_p = .999

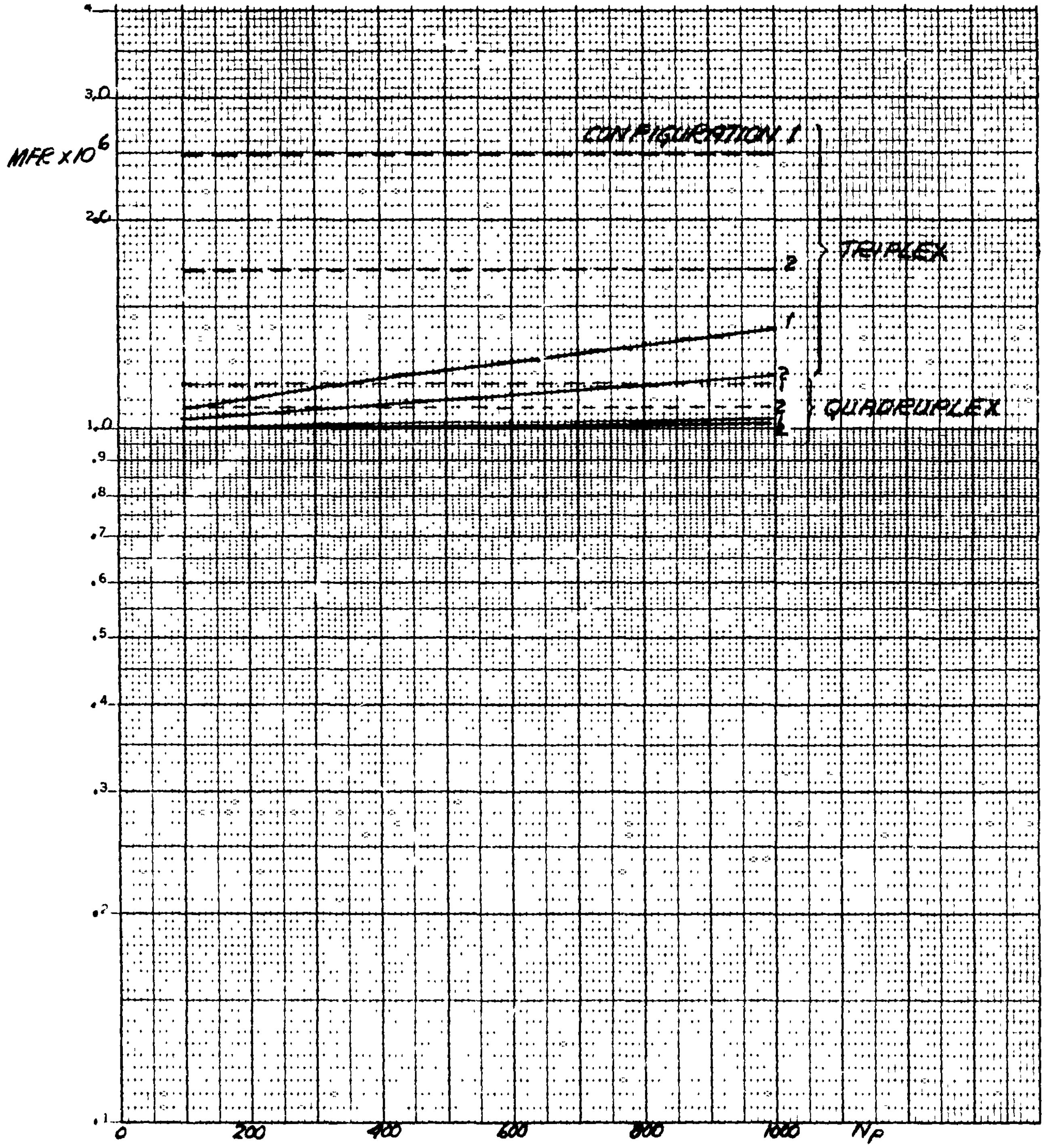


Figure 36. 680-J Airplane IIB MFR versus N_p; 1- α_i = .95; 1- α_p = .999 Primary actuator failure rate = .25 x 10^{-6}

SECTION 7

DIGITAL VERSUS ANALOG IMPLEMENTATION

1. Digital Computer Advantages

The digital computer has several potential advantages in the redundancy application:

a. It can provide superior test coverage and test effectivity as compared with present analog built-in-test (BIT). The coverage of a typical BIT ranges between 85 and 95% with a ratio of BIT hardware to total system hardware (by volume) of about 20% to 25%. In the DC-10 Autoland System (dual-dual), for example, BIT hardware comprises 22% of the total system. Typically, a digital computer self test program requires between 500 to 1500 words of memory. In a triplex redundant configuration this would comprise between 1% and 4% of the total computer (and I/O) hardware, respectively.

b. It eliminates tolerance accumulation normally contributed by the analog control computer.

c. Can provide sophisticated signal selection algorithms, reasonableness testing and performance monitoring far beyond what an analog system can yield with practical implementation.

d. With serial intercomputer links it requires fewer interconnecting wires for cross-channel comparison monitoring, if that form of monitoring is required.

e. Can be used in a variety of hybrid configurations e.g., off-line, digital outer loops/analog inner loops, etc.

2. Digital Computer Disadvantages

On the other hand a digital implementation has several disadvantages in the redundancy application:

a. Failure modes and effects tend to be difficult to characterize and some failures may be extremely difficult to detect using only a software self test program. Failure detection coverage requirements could dictate redundancy of internal computer components.

b. A digital computer implementation is susceptible to generic software failures. These failures, being common to more than one channel, could seriously degrade flight safety reliability. Eliminating or minimizing the probability of this type of failure requires rigorous software control and extensive testing of the prototype system. Dissimilar programs or a dissimilar back-up channel should be seriously considered in a FBW application. Paradoxically, the capability to make changes in the program quickly, with little or no impact on hardware, could be nullified by the degree of system testing that must accompany the change.

SECTION 8

RECOMMENDATIONS FOR MIL-F-9490

1. General Comments

The currently proposed revision to MIL-F-9490D, dated March 1974, prepared by The Boeing Company, still lacks detailed requirements that will ensure both the designer and the user, a means to achieve a redundant flight control system of the required safety and failure survivability for a given application. On the basis of this study, the recommendation is made to include in the next revision of MIL-F-9490 requirements for the following control system parameters.

a. In-flight monitoring

b. Pre-flight tests

c. Periodic maintenance tests

d. Validation requirements for (a) - (c)

It is also recommended that paragraph 3.2.4.3.2 be expanded with respect to input/output growth requirements as detailed below.

2. Test Requirements

The following paragraphs should be added to paragraph 3.1.3.2 of MIL-F-4990.

3.1.3.2.2 Redundancy Validation

For any flight critical mode, that is, any operational configuration wherein loss of the flight control system can reasonably be expected to lead to a degradation of the FCS operational state below level II, as defined in this document, the FCS specification shall include a test validation procedure and an analytical verification procedure, as appropriate, for the following system parameters:

In-flight monitor coverage

Pre-flight test coverage

Periodic maintenance test coverage

3.1.3.2.3 Redundancy Configuration

The selection of the redundancy configuration, including levels of redundancy and voting techniques, shall be based on meeting mission success and safety requirements and shall be validated by appropriate analyses.

3.1.3.2.4 In-flight Monitor Coverage

The FCS specification shall specify that adequate in-flight and pre-flight test coverage must be demonstrated. This coverage must be consistent with the probability of mission success safety requirements and the selected system configuration. Failure rates to be used in the analysis must be approved by the procuring agency.

3.1.3.2.5 Periodic Maintenance Testing

The FCS specification shall insure that periodic maintenance testing is accomplished at intervals that are consistent with the required mission success probability. The FGS specification shall develop criteria for the confidence level required in maintenance testing.

In addition, the section on digital implementation (paragraph 3.2.4.3.2) should be expanded to include the following:

3.2.4.3.2.4 I/O Capability

At the time of acceptance of the first production airplane, it is required that the digital computer I/O section contain a minimum of 10% of unused input and output lines, to take care of additional requirements over the life of the production airplane without the necessity of adding I/O hardware.

SECTION 9

CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE ACTION

1. Conclusions

The following conclusions are based on the results of the study:

a. A master plan for achieving mission and flight safety reliability goals should be an integral part of the design and synthesis of a redundant flight control system. The plan should include:

(1) A statement of mission and flight safety reliability goals. A commitment to a goal forces the designer to view the contribution of each component in the perspective of the whole system and leads to a practicable and fair allocation of failure rates. A criterion which considers only the electronics contribution to total reliability could lead to unnecessary, inconsistent and costly refinements.

(2) Allocation of failure rates - Failure rates should be allocated to all system components based on what is necessary and what is achievable.

(3) Statement of failure detection requirements - The objectives of inflight and preflight failure detection should be explicit. They should include the extent to which inflight and preflight detection coverage contributes to the attainment of the reliability goals. Inflight and preflight test coverage requirements should be allocated to all system components.

Signal selection devices should be identified and justified with regard to purpose; i.e., cross-strapping, improved failure detection, common outputs, etc.

(4) Failure detection validation procedure - Having established coverage goals and procedures to attain these goals it is necessary to validate the claimed coverages. Numbers of samples, accuracy and confidence factors specifications should be a part of the validation procedure.

b. Preflight test coverage is a critical parameter of flight safety reliability. In a triplex configuration controlling a flight critical mode the required preflight test coverage could exceed 99.9%. In a quadruplex configuration the corresponding coverage requirement could exceed 99% and possible even 99.9% depending upon inflight test strategy and its degradation in the presence of undetected failures. Inflight test coverage is much less critical.

c. The potential increased flight safety indicated by redundant control channels may represent an insignificant improvement in overall system reliability due to the dominance of single point failures of primary actuators, linkages, etc.

d. The use of a dissimilar backup channel in any flight critical configuration should be seriously considered. Advantages of the back-up channel are:

(1) Eliminates prime sources of common mode failures such as (a) generic software and (b) generic hardware failures or design defects.

(2) If the backup channel is designed for get-home-and-land capability only, then it may be relatively simple and thus can have its operational integrity more completely verified by testing preflight. Inflight monitoring or testing of the backup channel may not be necessary for improved reliability.

(3) A triplex configuration augmented by a backup channel is less susceptible to latent failures than a straight triplex or quadruplex configuration. In a triplex configuration a latent failure in one channel followed by an inflight failure in one of the two good channels could result in loss of the airplane. In a quadruplex configuration, a latent failure followed by an undetected inflight failure of one of the remaining channels could result in loss of the airplane. In the triplex-with-backup configuration loss of the airplane can occur only if two of the three triplex channels fail, one of which may be due to a latent failure, followed by a failure of the backup channel.

e. The use of time shared digital devices does not necessarily provide greater failure detection ability. Fault diagnosis of digital sequential devices can be a formidable undertaking in terms of the number of tests required in order to exercise all inputs, states, transition paths and outputs.

A self test procedure to detect all failures through recognition of all possible failure modes appears to be impracticable since the number of inputs required is prohibitive for even the simplest devices. A possible alternative would be to:

(1) enumerate the known failure modes of each device and their relative frequency of occurrence. While some failure modes will remain unknown, it can be presumed that the relative frequency of the unknown failure modes is sufficiently small to permit the attainment of the required coverage.

(2) Design the test procedure to diagnose those failure modes whose total relative frequency exceeds the coverage required.

This alternative approach requires a very precise knowledge of the failure modes of each device and their relative frequency of occurrences. This knowledge, however, does not appear to exist for many of the new MSI and LSI devices.

f. The benefits (increased system reliability) of cross strapping sensors should be carefully considered. When the sensor set is small or highly reliable, compared to the other system components, the benefits of cross-strapping are negligible. However, sensor cross-strapping can provide significant insensitivity to a greater number of latent failures particularly when the service life of the aircraft greatly exceeds 5000 hours.

g. The use of a separate trim system (or a separate trim card) in the FBW application should be considered. If trim is supplied by the flight critical digital controller then loss of the system will result not only in loss of the airplane but may also preclude a safe ejection. Furthermore, loss of a quadruplex system could result when only two channels have failed. Without a separate trim, there may be no time available for pilot determination and selection of one of the remaining good channels.

h.* The use of multiplexed data links can provide a significant reductior in numbers of wires and permits a standardization of interfaces. Multiplexing does not appear, at the preset time, to offer any weight advantage or improvement in reliability.

i. Cross-strapping or intercomputer communications of any kind are potential sources of common mode failures. Intercomputer data links are particularly susceptible because of the questionable buffering properties of normal digital gates. Shorts or even failures to ground could propagate through several levels of gates to the memory or data busses, resulting in an avalanche of failures throughout the computer.

j. With digital controllers more easily providing a common input to force summed actuators, failure transients are potentially reducible to acceptable levels. As a consequence, inflight failure detection may not be required for improved reliability or, reduction of failure transients (although it may be required for other reasons), if required, may incorporate a large time delay.

2. Recommendations for Future Action

a. Develop procedures and methods of validating the self test capabilities of airborne digital computers.

b. Information regarding failure modes and associated failure rates should be obtained to provide guide lines for modeling failed devices. Design and validation of a digital computer self test procedure requires knowledge of the failure modes of digital devices. While the vast majority of failures in digital microcircuits appear at the device terminals as frozen (or "stuck-at") signals, a low probability type of failure can occur and can be identified as "data dependent" failures. With this class of failures, internal logic is changed such that the device outputs no longer represent the design logic response, i.e., for some inputs or input sequences, the output is wrong - thus the failure is "data dependent". These failures can be particularly insidious in MSI and LSI where complex logic functions are performed.

Although these failures are in the minority, when self-test efficiencies of 99+% are required, they become of interest. Development of practical self-test and measurement of self-test efficiency requires that the frequency of occurrence of these failures be known and categorized by symptom. Modeling failed devices, in a practical way, requires this information.

*This conclusion is supported in Appendix VIII.

c. Develop procedures and methods of software verification. Such a procedure must exercise a large number of internal states and state transition paths. The procedure must be capable of practical implementation with a minimum of dependence on manual supervision.

SECTION 10

REFERENCES

1. Helfinstine, R. F., Montague, L. L., Seller, G. L., RELIABILITY AND REDUNDANCY STUDY FOR ELECTRONIC FLIGHT CONTROL SYSTEMS, Honeywell Document No. 21718-FP, Honeywell Ind., July, 1972.

2. Hendrick, R. C., Bailey, A. J., Edinger, L. D., DESIGN CRITERIA FOR HIGH-AUTHORITY CLOSED-LOOP PRIMARY FLIGHT CONTROL SYSTEMS, Technical Report AFFDL-TR-71-78, Honeywell Inc.. August, 1972.

3. Hooker, D. S., Kisslinger, R. L., Smith, G. R. Smyth, M.S., SURVIVABLE FLIGHT CONTROL SYSTEM INTERIM REPORT NO. 1 STUDIES, ANALYSES AND APPROACH, Technical Report AFFDL-TR-71-20, McDonnell Douglas Aircraft Co., May, 1971.

4. Amies, G. E., Clark, C. Jones, C. L., Smyth, M. S., SURVIVABLE FLIGHT CONTROL SYSTEM INTERIM REPORT NO. 1 STUDIES, ANALYSIS AND APPROACH, Supplement - 3, Technical Report AFFDL-TR-71-20, McDonnell Douglas Aircraft Co., May, 1971.

5. Tomlinson, L. R., SST LONGITUDINAL CONTROL SYSTEM DESIGN AND DESIGN PROCESSES, Final Report Task 4, Boeing Commercial Airplane Co., June, 1973, Prepared for FAA, Supersonic Transport Office, Washington, D. C.

6. Tomlinson, L. R., CONTROL SYSTEM DESIGN CONSIDERATIONS FOR A LONGITUDINALLY UNSTABLE SUPERSONIC TRANSPORT, Journal of Aircraft, Vol. 10, No. 10, October 1973.

7. Stengel, R. F., SOME EFFECTS OF BIAS ERRORS IN REDUNDANT FLIGHT CONTROL SYSTEMS, Journal of Aircraft, Vol. 10, No. 3, March 1973.

8. Moore, E. F., GEDANKEN-EXPERIMENTS ON SEQUENTIAL MACHINES, Automata Studies, Annals of Mathematics Studies No. 34, pp. 129-153, Princeton University Press, New Jersey, 1956.

9. Miller, R. E., SWITCHING THEORY, Vol. II, John Wiley and Sons, New York, 1965.

10. Wood, P. E., SWITCHING THEORY, McGraw-Hill, 1968.

11. "Feasibility Study for an Advanced Digital Flight Control System (Digiflic)", Lear Siegler, Inc., October, 1972.

12. "Navy Digital Flight Control System Development", Honeywell, Inc. December 1972.

13. MIL-F-8785B (ASG), "Flying Qualities of Piloted Airplanes", August, 1969

14. Technical Report AFFDL-TR-69-72, "Background Information and User Guide for MIL-F-8785B (ASG), 'Military Specification-Flying Qualities of Piloted Airplanes'", August 1969.

15. Technical Note No. 92, "Air Worthiness Requirements for Automatic Landing", Air Registration Board (U. K.), December 1966.

16. Kaman Aircraft Corporation Report G-166, "Self-Contained Electronic Flight Control System, Report No. 4", October 1961.

17. Sutton, M. L., Soderlund G. M., "The Application of Dedicated Processors to Digital Fly-By-Wire Flight Control Systems", NAECON '73 RECORD, Apr. 1 1973.

18. MIL-F-9490C (DRAFT), "Flight Control Systems - Design, Installation and Test of Piloted Aircraft, General Specification For", March 1974.

19. Technical Report AFFDL-TR-74-, "Background Information and User's Guide for MIL-F-9490", Boeing Company, March 1974.

20. Federal Aviation Regulations, Vol. II, Part 37, "Technical Standard Order Authorizations", Paragraph 37.119.

21. SAE Proposal Amendment to TSO-C9, Letter GA-3519, 1972.

22. Technical Report AFFDL-TR-71-134, "Validation of Flying Qualities Requirements of MIL-F-8785B (ASG)", Northrop Corporation, September 1971.

23. Papoulis, A., "Probability, Random Variables, and Stochastic Processes", McGraw-Hill, New York, 1965.

24. Bender, M. A. Gaabo, R. J., Smith, F. L., "Digital Flight Control Systems for Tactical Fighters", Technical Report AFFDL-TR-73-Vol. II, Interim Report, Honeywell, Inc., July, 1973.

25. Feller, W., "Probability Theory and Its Applications", Vol. I, Wiley, New York, 1950.

APPENDIX I

BASELINE FBW SYSTEM

1. Definition of Single Thread FBW System

In order to preserve generality and insure the most widespread applicability of the results of this study the single thread FBW flight control system has been configured to include only the direct control modes and those required to achieve desired handling qualities (i.e., CAS/SAS modes). Block diagrams & these basic modes are shown in Figures I-1, I-2 and I-3, In practice, however, certain outer-loop modes such as:

Localizer and runway align

Glide slope

Flare

Ground roll-out

Approach power compensation

Autothrottle

could have surviability requirements similar to FBW. For completeness, memory and real time estimates for these modes are also included as well as the failure rate and typical sensors.

Memory and Real Time Requirements

The memory and real time required for each mode is indicated in each figure and in Table I-1.

TABLE I-1
MEMORY AND REAL TIME REQUIREMENTS

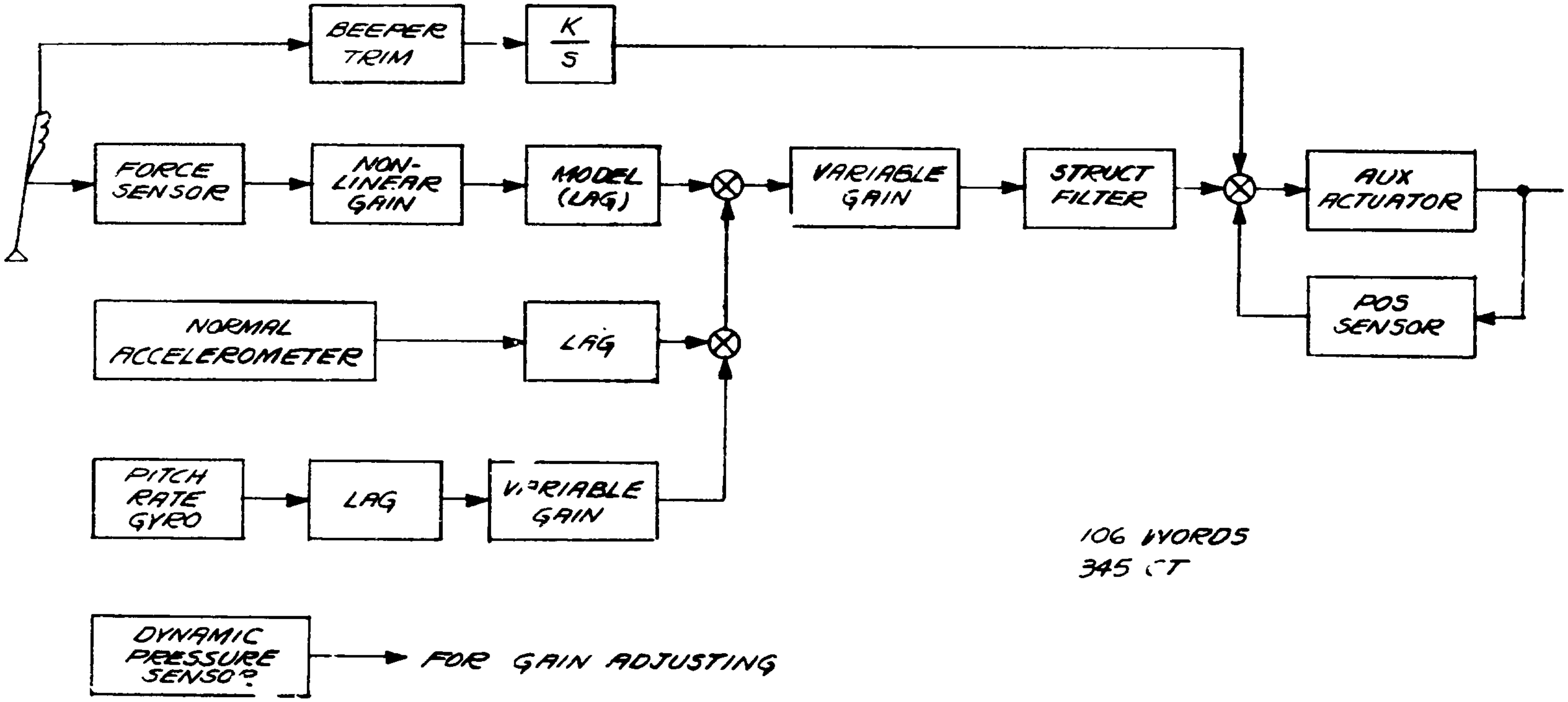
Mode	Words	Memory Cycle Time (CT)
Pitch Axis	106	345
Roll Axis	54	145
Yaw Axis	138	502
Speed Hold	67	166
APC	73	195
Loc Trk/Align	141	410
Glide Slope Track	165	435
Yaw Damper	32	89
Runway Align/GR	79	207
Glide Slope/Flare	134	335

The Autoland requirements are based on the Autoland modes as implemented in currently available equipment. It is assumed that:

- 2 cycle times = 1 add time
- 1 multiply = 6 cycle times = 3 add times
- 1 cycle time = 1 microsecond
- Inputs and outputs are executed using DMA. Thus, A/D and D/A conversion time is excluded from the estimates.

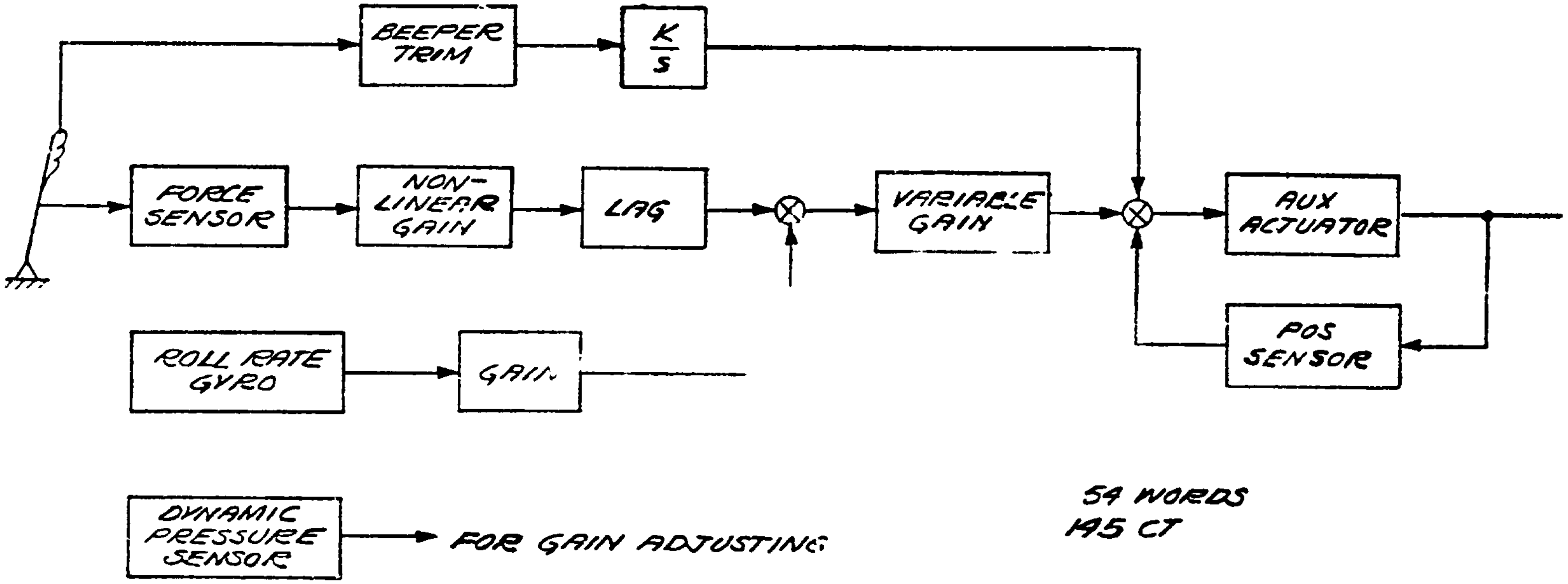
Sampling rates for inner loop control are taken at approximately 40 samples/second and at approximately 10 samples/second for outer loop and autothrottle control. Based on 40 samples/second, the total real time required for inner loop pitch, roll, and yaw axis control is 39.68 milliseconds, or 39.68% of real time. The real time requirements of the outer control loops plus the speed control loops is 73.48 milliseconds if the same sampling rate assumed (a very conservative assumption). Thus, the total real time requirement is of the order of 12%. However, to this estimate must be added the requirements for

- executive subroutines
- modal logic
- monitoring, intercomputer communication, and signal selection
- inflight and preflight test
- generation of annunciation signals

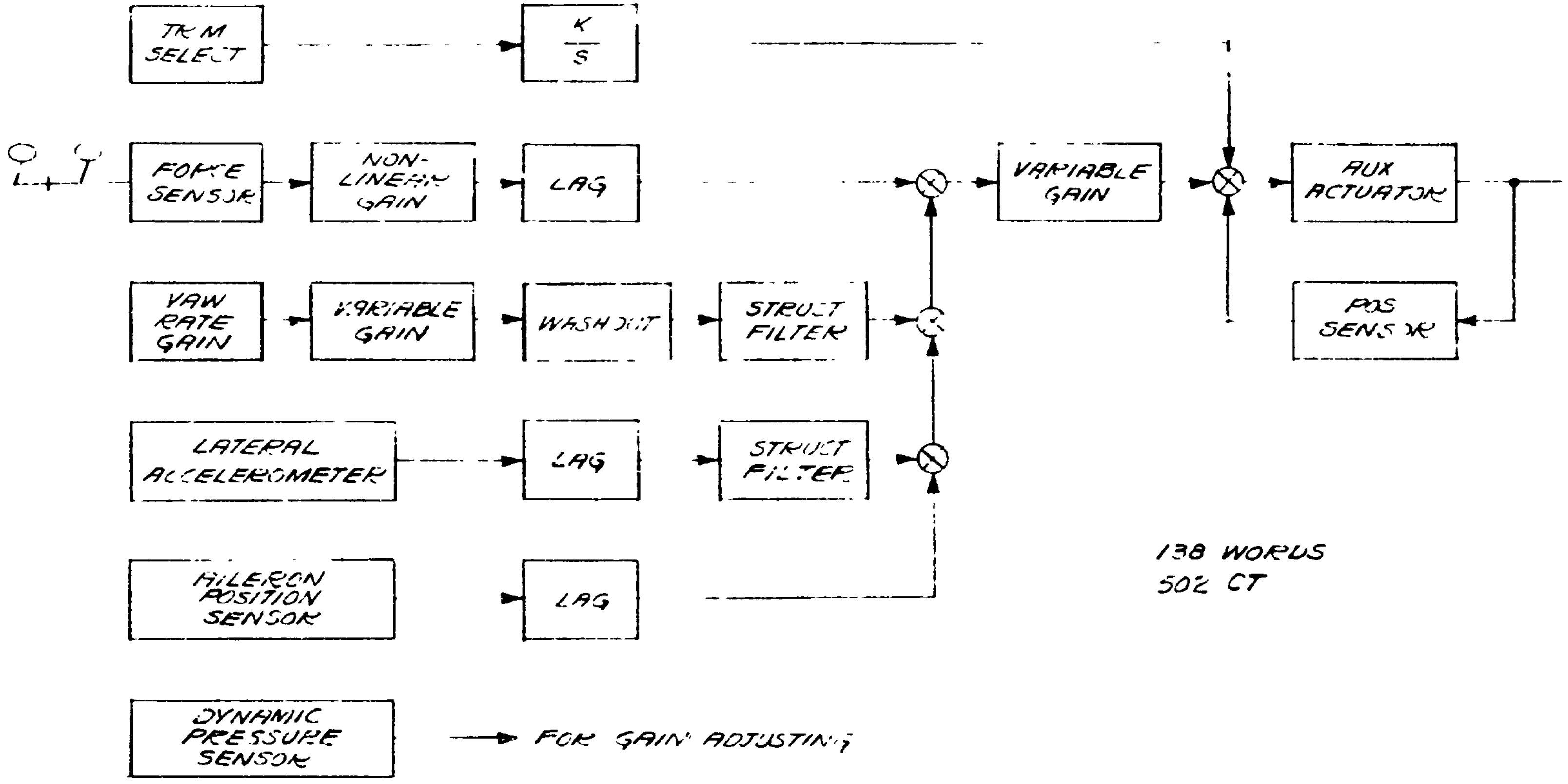


106 WORDS
345 CT

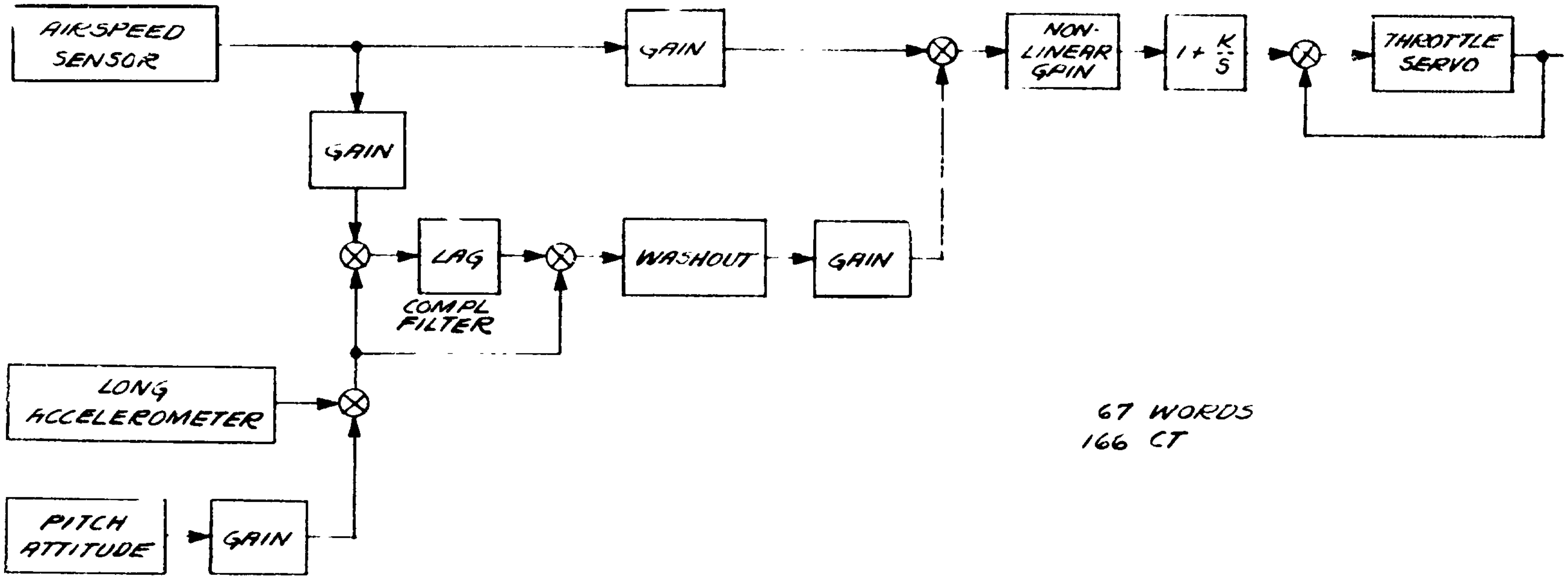
PITCH AXIS CONTROL
FIGURE I-1



ROLL AXIS CONTROL
FIGURE I-2

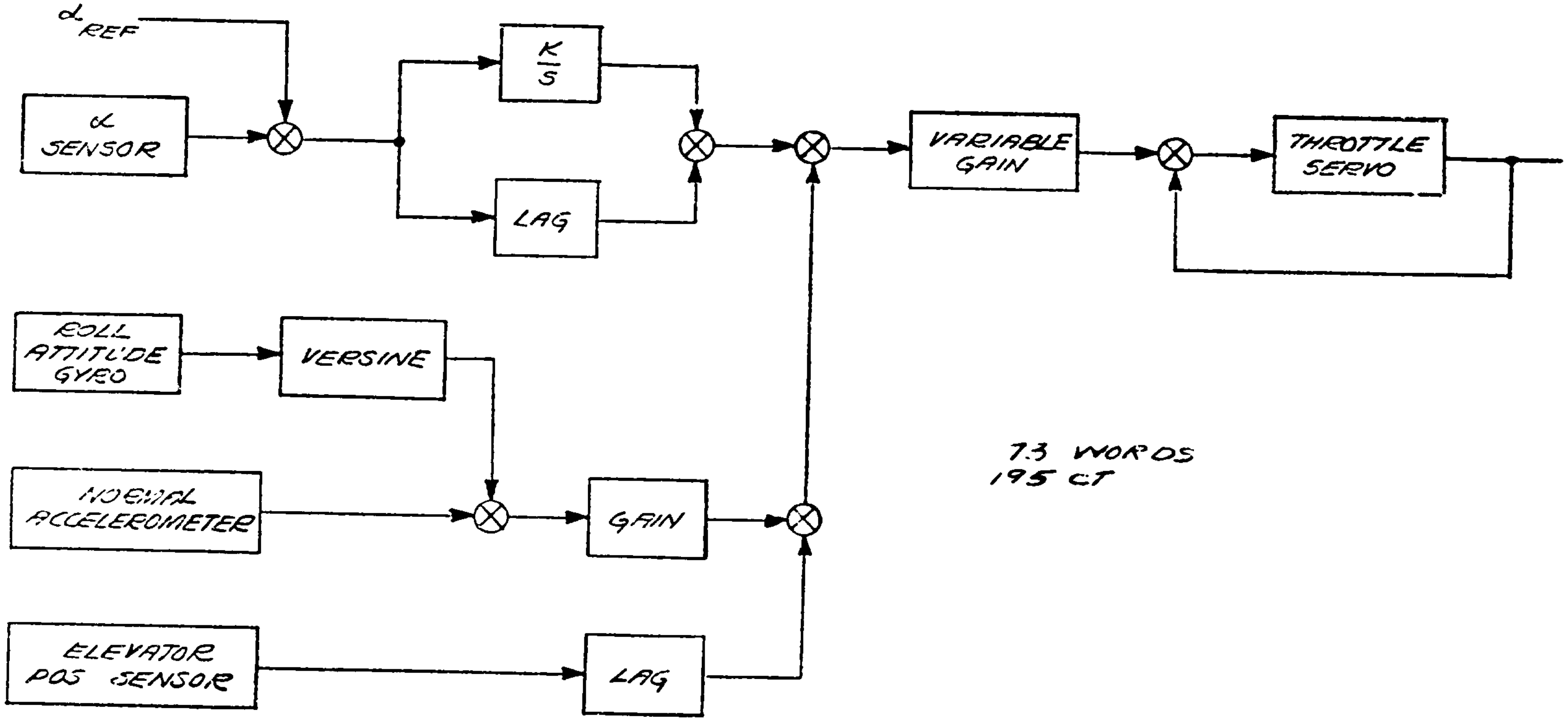


YAW AXIS CONTROL
FIGURE I-3



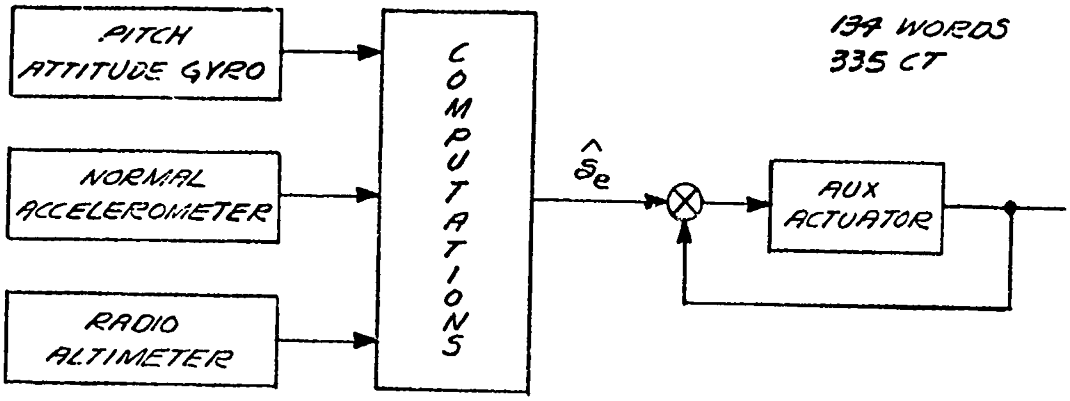
67 WORDS
166 CT

AUTOTHROTTLE (AIRSPEED HOLD MODE)
FIGURE I-4

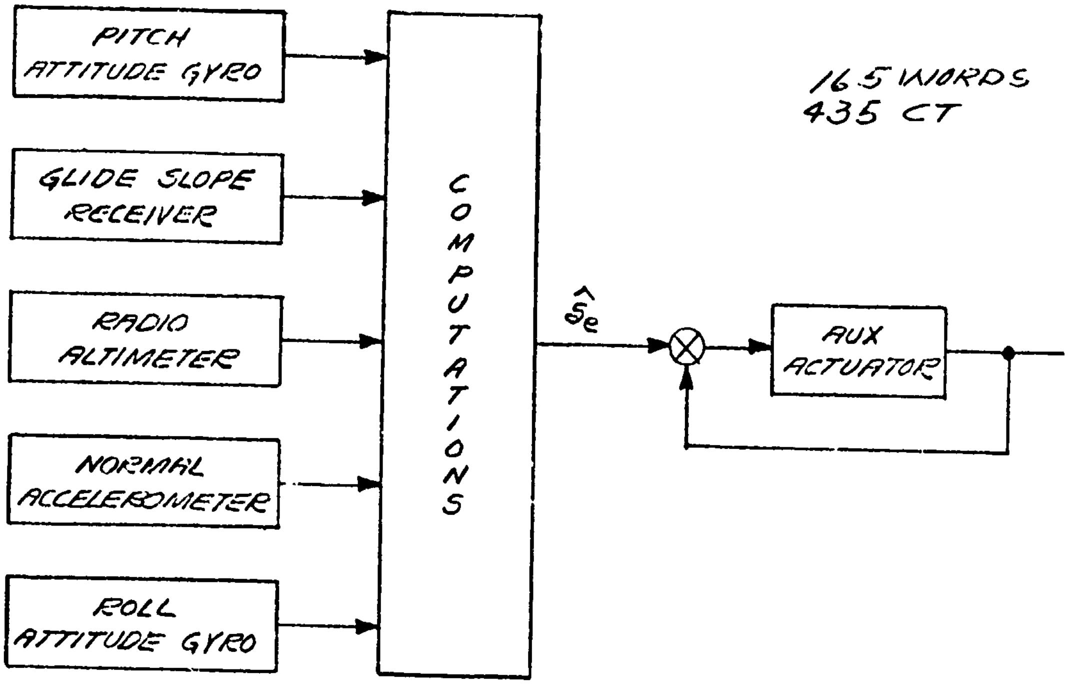


7.3 WORDS
195 CT

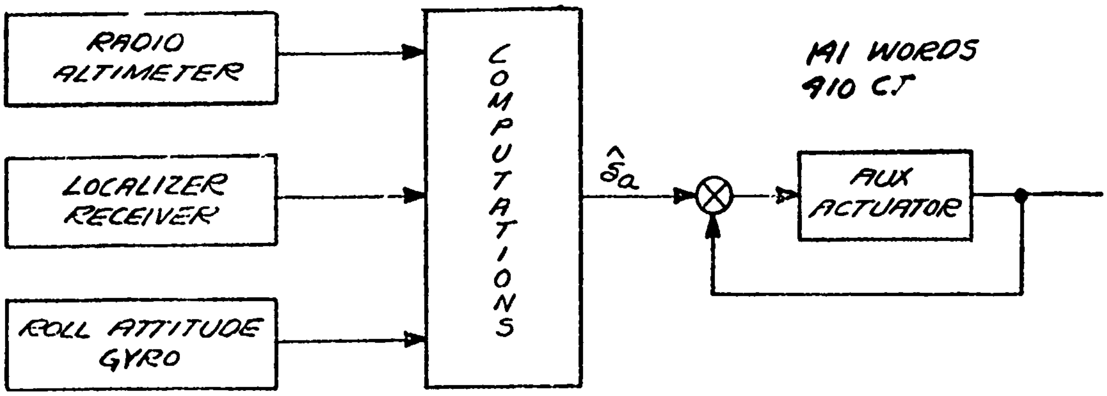
APPROACH PO' R COMPENSATOR
FIGURE I-5



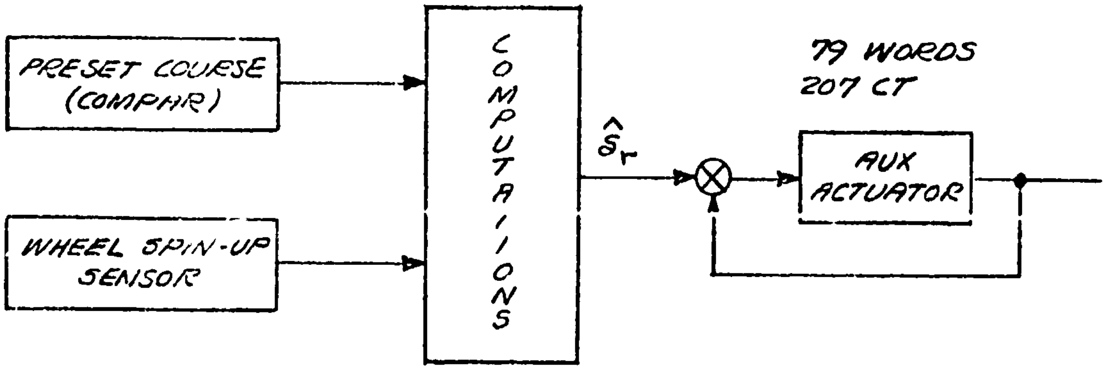
GLIDE SLOPE FLARE
FIGURE I-6



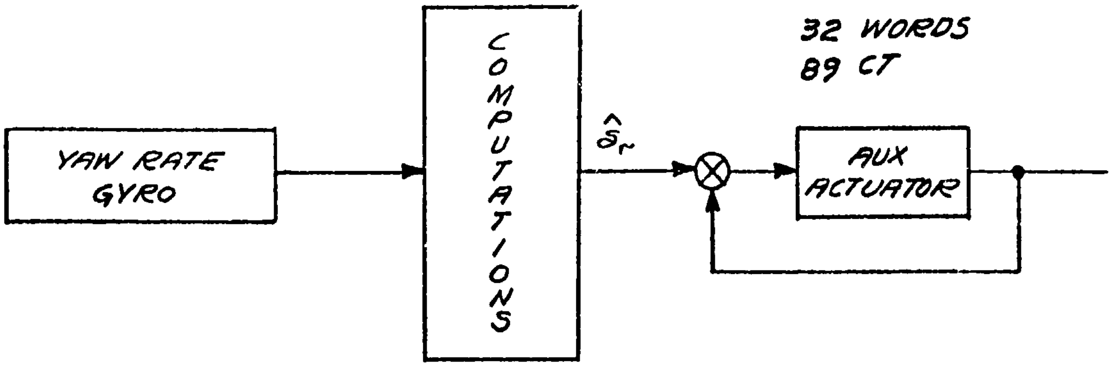
GLIDE SLOPE TRACK
FIGURE I-7



LOCALIZER TRACK/ALIGN
FIGURE I-8



RUNWAY ALIGN/GROUND ROLL
FIGURE I-9



YAW DAMPER
FIGURE I-10

The total effect of these requirements is not expected to increase the above real time requirement by more than 50%. It may be concluded that the real time requirements of digital flight control systems are well within the capabilities of present day digital computer technology.

2. Failure Rates of Basic Components

The failure ɪates that have been used for the basic components of the system reflect currently available technology. Specific component reliability references are listed where apprcpriate.

Primary Actuator: .25 x 10^{-6}/hour or 3.0 x 10^{-6}/hour

The lower of these numbers is specified in reference 3, where, however, it refers to single point failures in a four actuator package. The higher of the numbers is a conservative estimate of today's technology. In any event, while this number exerts a dominant influence on the total achievable system failure rate (see Section 6), it enters the relative evaluation of redundant configurations, representing as it does a single point failure, only frcm the point of view of whether the relative failure contributions of improved redundancy management configurations are significant in the light of this system limit.

Secondary Actuator: 100 x 10^{-6}/hour

Actuators currently available suffer from relatively high failure rates. The failure rate of 100 x 10^{-6}/hour does not include loss of associated hydraulics.

Tables I-2 and I-3 list the failure rates of other FBW system components that were used in this program. Most of the rates are standard and have been used in many FMEA's and certification programs. The failure rate for the digital computer is believed to be applicable for the 1975-6 time period.

TABLE I-2

FBW PFCS

I/O SIGNAL CHARACTERISTICS

Sensed Signal	Range	Form	Failure Rate Per 10^6 Hours
Pitch Stick Force	± 10 lbs.	26 VAC 400 Hz	5
Roll Stick Force	± 10 lbs.	26 VAC 400 Hz	5
Pedal Force	+ 10 lbs.	26 VAC 400 Hz	5
Pitch Rate	± 60°/sec.	26 VAC 400 Hz	25
Roll Rate	± 300°/sec.	26 VAC 400 Hz	25
Yaw Rate	± 60°/sec.	26 VAC 400 Hz	25
Normal Acceleration	± 10 g	± 10 vdc	20
Lateral Acceleration	± 1 g	± 10 vdc	20
Longitudinal Acceleration	± 1 g	± 10 vdc	20
Angle of Attack	± 50°	26 VAC 400 Hz	
Pitch Attitude	± 60°	3 Wire Synchro	100
Roll Attitude	± 360°	3 Wire Synchro	100
Elevator Servo Position LVDT	± 20°	26 VAC 400 Hz	3
Aileron Servo Position LVDT	± 30°	26 VAC 400 Hz	3
Rudder Servo Position LVDT	± 30°	26 VAC 400 Hz	3
Throttle Position LVDT	--	26 VAC 400 Hz	3
Dynamic Pressure	1800 lb./ft.^{2}	Serial Binary	20
Auxiliary (Secondary) Actuator			100
CPU + 8 K Core			70
P/S for CPU, 8K Core, I/O			30
Basic I/O + I/O Control			20
Power Actuator (includes control linkages)			.25 → 3.0

TABLE I-3

AUTOLAND SYSTEM

I/O SIGNAL CHARACTERISTICS

Sensed Signal	Range	Form	Failure Rate Per 10^6 Hours
Pitch Attitude Gyro	± 60°	26 VAC 400 Hz	127
Glide Slope Receiver	± 3°	± 10 vdc	65
Radio Altimeter	0-2500'	± 10 vdc	176
Roll Attitude Gyro	± 360°	26 VAC 400 Hz	127
Normal Accelerometer	± 10 g	± 10 vdc	20
Yaw Rate Gyro	± 40°/sec.	26 VAC 400 Hz	50
Preset Course	0 -- 360°	3 Wire Synchro	100
Localizer Receiver	± 5°	± 10 vdc	58
CADC	0 → 100 K h 100 → 1500 kts	± 10 vdc	352
Wheel Spin Up	---	---	8
S3A Throttle Servo	---	---	27
Electrical Power System	---	---	
AC Bus	---	---	300
DC Bus	---	---	8

APPENDIX II

Failure Performance Requirements

1. Existing Sources of Failure Performance Requirements

Failure in a flight control system may be defined as any event internal to the system that, if not compensated for, would lead to an unacceptable performance change in the aircraft. There does not appear to be any official document, issued by a cognizant U. S. military or civil agency, which defines, in a comprehensive manner, flight control system performance in the event of failures*. One reason for this is, no doubt, that "acceptable performance" depends on the application and detailed specifications are therefore best left to procurement specifications. Some aspects of failure performance are, however, discussed in MIL-F-8785B (ASG), FAA Advisory Circulars, and Air Registration Board (UK) Technical Notes. Pertinent comments from these sources are as follows:

a. MIL-F-8785B (ASG)

Paragraph 3.1.10.1, Requirements for Airplane Normal States

The minimum required flying qualities for airplane normal states are:

Within Operational Flight Envelope	Within Service Flight Envelope
Level 1	Level 2

Paragraph 3.1.10.2, Requirements for Airplane Failure States

Levels for Airplane Failure States

Probability of Encountering	Within Operational Flight Envelope	Within Service Flight Envelope
Level 2 after failure	$<10^{-2}$ per flight	---
Level 3 after failure	$<10^{-4}$ per flight	$<10^{-2}$ per flight

*This statement was made before the publication of the draft of MIL-F-9490D, March 1974.

"After failure" means "after the occurrence of one or more failures" during the longest operational mission time considered by the contractor designing the airplane. Failures are due to all causes including flight control system failures.

Paragraph 3.5.5.1 Failure Transients

With controls free, the airplane motions due to failures described in paragraph 3.5.5 shall not exceed the following limits for at least 2 seconds following the failure, as a function of the level of flying qualities after the failure transient has subsided:

Level 1 (after failure)	±0.05 g normal or lateral acceleration at the pilot's station and ±1 degree per second in roll
Level 2 (after failure)	±0.5 g at the pilot's station, ±5 degrees per second roll and the lesser of ±5 degrees sideslip or the structural limits
Level3 (after failure)	No dangerous attitude or structural limit is reached, and no dangerous alteration of the flight path results from which recovery is impossible.

Paragraph 3.5.5.2 Trim Changes due to Failures

The control forces required to maintain attitude and zero sideslip for the failures described in paragraph 3.5.5 shall not exceed the following limits for at least 5 seconds following the failure:

Elevator	-	20 pounds
Aileron	-	10 pounds
Rudder	-	50 pounds

Paragraph 3.5.6 Transfer to Alternate Control Modes

The transient motions and trim changes resulting from the intentional engagement or disengagement of any portion of the primary flight control system by the pilot shall be small and gradual enough that dangerous flying qualities never result.

Paragraph 3.5.6.1 Transients

With controls free, the transients resulting from the situations described in 3.5.6 shall not exceed the following limits for at least 2 seconds following the transfer:

Within the Operational Flight Envelope	±0.05 g normal or lateral acceleration at the pilot's station and ±1 degree per second roll
Within the Service Flight Envelope	±0.5 g at the pilot's station, ±5 degrees per second roll, and the lesser of ±5 degrees sideslip or the structural limit

These requirements apply only for Airplane Normal States.

Paragraph 3.5.6.2 Trim Changes

The control forces required to maintain attitude and zero sideslip for the situations described in paragraph 3.5.6 shall not exceed the following limits for at least 5 seconds following the transfer:

Elevator	-	20 pounds
Aileron	-	10 pounds
Rudder	-	50 pounds

These requirements apply only for Airplane Normal States.

Paragraph 3.6 Characteristics of Secondary Control Systems

Paragraph 3.6.1 Trim System

In straight flight, throughout the Operational Flight Envelope the trimming devices shall be capable of reducing the elevator, rudder, and aileron control forces to zero for Levels 1 and 2. For Level 3, the untrimmed cockpit control forces shall not exceed 10 pounds elevator, 5 pounds aileron, and 20 pounds rudder. The failures to be considered in applying the Level 2 and 3 requirements shall include trim sticking and runaway in either direction. It is permissible to meet the Level 2 and 3 requirements by providing the pilot with alternate trim mechanisms or override capability. Additional requirements on trim rate and authority are contained in MIL-F-9490 and MIL-F-18372.

It should be noted that 3.1.10.2 also specified that no (single) failure state shall degrade any flying quality outside of the Level 3 limit. It should also be noted that, for Level 3, untrimmed cockpit control forces should not exceed 10 pounds for elevator, 5 pounds for aileron and 20 pounds for the rudder. In reference 22, it is indicated that an F5 requires 48 pounds elevator, 8 pounds aileron and 19 pounds rudder in order to compensate for hardover trim. Thus, in order to insure compliance with this specification, trim is assumed to be required in order to maintain Level 3 flying qualities.

b. Air Registration Board Technical Note No. 92

The document states that the present fatal manual landing accident rate is about 1.0 x 10^{-6} accidents per landing for transport aircraft and suggests that the total fatal landing accident rate (below 200 ft. and 1/2 mile range) should not exceed 1.0 x 10^{-7} accidents per automatic landing. The document also suggests that the automatic landing abort rate should not exceed 1 abort in 20 committed landings. According to this document an abort is the termination of autoland from the time that the aircraft has been accepted for approach. A more severe criterion is the requirement that autoland be functional following a 2 hour en route flight. For the DC-10 autoland system, it is required (by the aircraft manufacturer) that the probability of a failure occurring during a 2 hour en route flight which would reduce the functional capability of autoland upon engagement should be less than 1/200.

c. Federal Aviation Regulations, Vol. II, Part 37

Paragraph 37.119

d. TSO-C9c Paragraph 4.6

"The automatic pilot design shall be such that, should a single failure (except gyro mechanical failures) occur in the system, no signal shall result which would apply to the aircraft maximum servo control forces as determined in Paragraph 4.5.2, in more than one primary and trim aerodynamic axis."

In Reference 21 the following revision is suggested:

"The system design must be such as to avoid multiaxis hardovers. If multiaxis hardovers can result from a single failure, the resultant aircraft response must be controllable by the pilot."

e. The following paragraphs have been extracted from MIL-F-9490D (DRAFT)

1.0 SCOPE AND CLASSIFICATION

1.1 Scope. This specification establishes general performance, design, development and quality assurance requirements for the flight control systems of USAF manned piloted aircraft. Flight control systems (FCS) include all components used to transmit flight control commands from the pilot or other sources to appropriate force and moment producers. Flight control commands may result in control of aircraft attitude, airspeed, flight path, aerodynamic configuration, ride, and structural modes. Among components included are the pilot's controls, dedicated displays and logic switching, system dynamic and air data sensors, signal computation, test devices, transmission devices, actuators, and signal transmission lines dedicated to flight control. Excluded are aerodynamic surfaces, engines, helicopter rotors, fire control devices, crew displays and electronics not dedicated to flight control.

1.2 Classification

1.2.1 Flight Control System (FCS) Classifications

1.2.1.1 Manual Flight Control Systems (MFCS). Combinations of electrical, mechanical and hydraulic components which transmit pilot control commands and/or generate and convey commands which augment pilot control commands, and thereby accomplish flight control functions are classified Manual Flight Control Systems. This classification includes the longitudinal, lateral-directional, lift, drag and variable geometry control systems and their associated stability augmentation, command augmentation, and performance limiting and control devices.

1.2.1.2 Automatic Flight Control Systems (AFCS). Combinations of electrical, mechanical and hydraulic components which generate and transmit automatic control commands which provide pilot assistance through automatic or semiautomatic flight path control, or which automatically control airframe response to disturbances are classified Automatic Flight Control Systems. This classification includes automatic pilots, stick or wheel steering, autothrottles and structural mode control.

1.2.2 FCS Operational State Classifications

1.2.2.1 Operational State I (Normal Operations). The normal state of flight control system performance, safety and reliability achieved. This state satisfies MIL-F-8785 or MIL-F-83300 Level 1 flying qualities requirements.

1.2.2.2 Operational State II (Restricted Operation). The state of less than normal equipment operation or performance which involves degradation or failure of only a noncritical portion of the overall Flight Control System. A moderate increase in crew workload and degradation in mission effectiveness may result from restricted choice of normally operating FCS modes available for use; however, the intended mission may be accomplished. This state satisfies at least MIL-F-8785 or MIL-F-83300 Level 2 flying qualities requirements.

1.2.2.3 Operational State III (Minimum Safe Operation). A state of degraded flight control system performance, safety or reliability which permits safe termination of precision tracking or maneuvering tasks, and safe cruise, descent, and landing at the destination of original intent or alternate but where pilot workload is excessive and/or mission effectiveness is inadequate. Phases of the intended mission involving precision tracking or maneuvering cannot be completed satisfactorily. This state satisfies at least MIL-F-8785 or MIL-F-83300 Level 3 flying qualities requirements.

1.2.2.4 Operational State IV (Controllable to an Immediate Emergency Landing). The state of degraded FCS operation at which continued safe flight is not possible; however, sufficient control remains to allow engine restart attempt(s), a controlled descent and immediate emergency landing.

1.2.2.5 Operational State V (Controllable to an Evacuable Flight Condition). The state of degraded FCS operation at which the FCS capability is limited to maneuvers required to reach a flight condition at which crew evacuation may be safely accomplished.

1.2.3 FCS Criticality Classification

1.2.3.1 Essential. A function is essential if loss of the function results in an unsafe condition and inability to maintain FCS Operational State III.

1.2.3.2 Flight Phase Essential. A function is flight phase essential if loss of the function results in an unsafe condition and inability to maintain FCS Operational State III only during specific flight phases.

1.2.3.3 Noncritical. A function is noncritical if loss of the function does not affect flight safety or result in control capability below that required for FCS Operational State III.

Classes. Airplane classes are defined using the MIL-F-8785 definitions for the following classes.

Class I — Small, light airplanes such as
- Light utility
- Primary trainer
- Light observation

Class II — Medium weight, low-to-medium maneuverability airplanes such as
- Heavy utility/search and rescue
- Light or medium transport/cargo/tanker
- Early warning/electronic countermeasures/airborne command, control, or communications relay
- Antisubmarine
- Assault transport
- Reconnaissance
- Tactical bomber
- Heavy attack
- Trainer for Class II

Class III — Large, heavy, low-to-medium maneuverability airplanes such as
- Heavy transport/cargo/tanker
- Heavy bomber
- Patrol/early warning/electronic countermeasures/airborne command, control, or communications relay
- Trainer for Class III

Class IV High-maneuverability airplanes such as
Fighter/interceptor
Attack
Tactical reconnaissance
Observation
Trainer for Class IV

Where MIL-F-83300 applies, the corresponding MIL-F-83300, Class I, II or III or IV applies.

3.1.3.10 All Weather Landing Performance Standards. "The lateral-directional control system's performance shall be such that aircraft lateral velocities normal to the runway centerline shall not cause a maximum aircraft lateral displacement greater than 75 ft. as measured to either side of the runway centerline from the outermost main landing gear of the aircraft more often than 1 in 10^6 landings."

3.1.3.2 Failure Immunity and Safety. Within the permissible flight envelope, no single failure or failure combination in the FCS, which is not extremely remote, shall result in any of the following before a pilot or safety device can react. For this specification, extremely remote (6.6) is defined as numerically equal to the maximum aircraft loss rate due to relevent FCS material failures specified in 3.1.7.

a. Flutter, divergence, or other aeroelastic instabilities within the permissible flight envelope of the aircraft, or a structural damping coefficient for any critical flutter mode below the fail-safe stability limit of MIL-F-8870.

b. Uncontrollable motions of the aircraft within its permissible flight envelope, or maneuvers which generate limit airframe loads.

c. Inability to safely land the aircraft.

d. Any asymmetric, unsynchronized, unusual operation or lack of operation of flight controls that produces operation below FCS Operational State III.

3.1.3.3.4 Failure Transients. Aircraft motions following sudden airplane system or component failures shall be such that dangerous conditions can be avoided by pilot corrective action. Transients due to failures resulting in FCS Operational States I or II within a redundant FCS shall not exceed 0.5g incremental normal or lateral acceleration at the center-of-gravity or ±10°/sec roll rate. Transients due to failures within the FCS resulting in FCS Operational State III shall not exceed 75% of limit load factor or 1.5g's, whichever is less, at the most severe flight condition.

3.1.3.9 System Test and Monitoring Provisions. Test and monitoring means shall be incorporated into the essential and flight phase essential FCS as required to meet the mission reliability requirements of 3.1.6, and the flight safety requirements of 3.1.7 and fault isolation requirements of 3.1.10.2.

3.1.3.9.1 System Test and Monitoring Analysis. The effect of undetected FCS failures taken with the probability of occurrence of such failures shall comply with the system reliability and safety requirements. The analysis verifying this requirement shall include all failures, both active and latent, and failures in all components of the system, including mechanical, electrical and hydraulic components.

3.1.3.9.2 Built-In-Test Equipment (BIT). The total maintenance aid testing, including BIT, and inflight monitoring shall provide an integrated means of fault isolation to the LRU level with a confidence factor of 90%. BIT function shall have multiple provisions to ensure they cannot be engaged in flight.

3.1.3.9.2.1 Preflight or Pre-engage BIT. Preflight or pre-engage BIT may be automatic or pilot-initiated, and includes any test sequence normally conducted prior to take-off or prior to engagement of a control to provide assurance of subsequent system safety and operability. The preflight tests shall not rely on special ground test equipment for their successful completion. Any test sequence which could disturb the normal activity of the aircraft in a given mode shall be inhibited when that mode is engaged.

3.1.3.9.2.2 Maintenance BIT. BIT shall also be provided as a postflight maintenance aid for the FCS. BIT shall be designed to avoid duplicating test features included as part of the preflight test or monitoring functions.

3.1.3.9.3 Inflight Monitoring. Continuous monitoring of equipment performance and/or cirtical flight conditions shall be provided. The monitoring shall, as a minimum, be active during essential or flight phase essential modes of operation. An analysis shall be provided showing that false monitor warnings, including the automatic or normal pilot response thereto, will not constitute a specific hazard in excess of the system reliability requirements.

3.1.6 Mission Accomplishment Reliability. The probability of mission failure per flight due to relevant material failures in the flight control system shall not exceed either a. or b. specified below. Failures in power supplies or other subsystems that do not otherwise cause mission failure shall be included where pertinent. A representative mission to which this requirement applies shall be established and defined in the FCS Specification (4.4.2).

a. Where overall A/C mission accomplishment reliability is specified by the procurement activity, $Q_{M(fcs)} \leq (1 - R_M) A_{M(fcs)}$

b. Where overall A/C mission accomplishment reliability is not specified, $Q_{M(fcs)} \leq 1 \times 10^{-3}$,

$Q_{M(fsc)}$ = Maximum acceptable mission unreliability due to relevant FCS material failures

R_M = Specified overall aircraft mission accomplishment reliability

$A_{M(fcs)}$ = Mission accomplishment allocation factor for flight control.

3.1.7 Quantitative Flight Safety The probability of aircraft loss per flight due to relevant material failures in the flight control system shall not exceed:

$$Q_{S(fcs)} \leq (1 - R_S) A_{S(fcs)}$$

where: $Q_{S(fcs)}$ = Maximum acceptable aircraft loss rate due to relevant FCS material failures.

$A_{S(fcs)}$ = Flight safety allocation factor for flight control.

R_S = Overall Aircraft Flight Safety Requirement as specified by the procuring activity.

Failures in power supplies or other subsystems that do not otherwise cause aircraft loss shall be included where pertinent. A representative mission to which this requirement applies shall be established and defined in the FCS Specification (4.4.2). If overall aircraft flight safety in terms of R_S is not specified by the procuring activity, the numerical requirements of Table III apply.

TABLE III

FCS QUANTITATIVE FLIGHT SAFETY REQUIREMENTS

		MAXIMUM AIRCRAFT LOSSES PER FLIGHT	
OVERALL A/C FLIGHT SAFETY REQUIREMENT NOT SPECIFIED BY PROCURING ACTIVITY	MIL-F-8785 CLASS III-AIRCRAFT	$Q_{S(fcs)}$	$5x10^{-7}$
	ALL ROTARY WING AIRCRAFT	$Q_{S(fcs)}$	$25x10^{-7}$
	MIL-F-8785 CLASS I, II & IV AIRCRAFT	$Q_{S(fcs)}$	$100x10^{-7}$

3.1.7.1 Reliability - All Weather Landing System. The average hazard due to the use of the all weather landing system shall be less than the risk allowed in the contractor's reliability budget for the all weather landing system. To meet the requirements of 3.1.7, the contractor shall allocate the FCS reliability budget among AWLS and other FCS. The specific risk of a hazard due to use of the landing system under an environmental limit or operational restriction shall not increase the allowed risk by a factor of more than thirty. These analyses shall provide the basis for establishing an alert height at an altitude such that, with all systems operative at the alert height, the probability of a hazard occurring during the landing is extremely remote.

3.1.7.1.1 Assessment of Average Risk of a Hazard. The average risk of a hazard due to use of the all weather landing system shall be established by a statistical analysis which includes:

a. A system failure analysis showing the effect of a failure or combination of failures on system performance and the probability of their occurrence.

b. Failure analyses showing the effect of failure or a combination of failures in systems operating concurrently with the all weather landing AFCS on aircraft performance and the probability of their occurrence.

c. The probability of the system not performing within the required levels defined in 3.1.2.10 taken in conjunction with the probability that exceedance of those performance levels will result in a hazard.

3.1.8 Survivability. FCS Operational State IV or State V shall be provided as required by the procuring activity.

3.1.9.4 Invulnerability to Onboard Failures of Other Systems and/or Equipment

a. Flight control systems shall retain FCS capability at Operational State III (minimum safe) or better after sustaining the following failures:

(1) Failure of one engine in a two-engine airplane.

(2) Failure of two engines in three-engine and four-or-more-engine airplanes.

(3) Failure of any sinqle equipment item or structural member which, in itself, does not cause degradation below State III. This includes any plausible sinqle failure of any onboard electrical or electronic equipment.

b. Flight control systems, including the associated structure and power supplies on Class III aircraft, shall be designed so that the probability of losing the capability of maintaining FCS operation to at least State IV as a result of an engine or other rotor burst is extremely remote.

c. Flight control systems, including the associated structure and power supplies on Class I, II & IV aircraft, shall be designed so that the probability of degrading FCS operation below State V as a result of an engine or other rotor burst is extremely remote.

3.1.10.2 Malfunction Detection and Fault Isolation Provisions. Means providing a high probability for detecting failures and monitoring critical performance conditions as required to isolate faults to the LRU level shall be incorporated in all flight control electrical and electronic systems required to perform essential and/or flight-phase-essential functions. These means may include cockpit instrumentation and/or built-in test equipment. For the mechanical and fluid power portions of the flight control system, provisions for the use of portable test equipment may also be incorporated as required to meet the maintenance support and operational concept of the particular weapon system.

3.2.1.4.2 FCS Warning and Status Annunication. FCS Warning and Status Annunciation shall be provided in the cockpit. Annunciation shall be designed to clearly indicate the associated degree of urgency.

a. First degree - that is: Immediate Action Required

b. Second degree - Caution: Action may be required

c. Third degree - Informational, no immediate action required.

A panel comprising means for displaying first degree annunciations shall be located within the normal eye scan range of the command pilot. A first degree warning or status indication, which applies only to a particular mode or phase of flight, shall be inhibited or designed to clearly indicate a lesser degree of urgency for all other modes or phases of flight.

3.2.1.4.2.1 Preflight Test (Bit) Status Annunciation. This display shall:

a. Indicate the progress of the preflight test.

b. Instruct the crew to provide required manual inputs.

c. Indicate lack of system readiness when failure conditions are detected.

3.2.1.4.2.1 Failure Status. Failure warnings shall be displayed to allow the crew to assess the operable status of redundant or monitored flight control systems. Automatic disengagement of an AFCS mode shall be indicated by an appropriate warning display. Manual disengagement by the crew shall not result in warning annunciation. Loss of valid signals critical to existing modes of operation for FCS or flight director shall result in appropriate warnings and/or system deactivation.

4.2.2 Reliability and Failure Mode and Effects Analyses. When required by the procuring activity, reliability and failure mode and effects analyses shall be performed to analytically demonstrate that the FCS satisfies the requirements of 3.1.6 and 3.1.7. When required by the procuring activity, the Reliability Program Plan, defined by MIL-STD-785, shall outline steps to be used to perform these analyses.

APPENDIX III

Mathematical Addenda

Given the probability model as described in Section 3 we note the following relationships:

III-1 $\alpha = P(\bar{A} | F) = \dfrac{P(F\bar{A})}{P(F)}$

III-2 $\beta = P(\bar{F} | A) = \dfrac{P(\bar{F}A)}{P(A)}$

III-3 $P(FA) + P(F\bar{A}) = P(F)$

III-4 $P(FA) + P(\bar{F}A) = P(A)$

III-5 $P(FA) + P(F\bar{A}) + P(\bar{F}A) + P(\bar{F}\bar{A}) = 1$

From (III-1)and (III-2) we obtain

III-6 $P(F\bar{A}) = \alpha P(F)$

III-7 $P(\bar{F}A) = \beta P(A)$

and from (III-3) and (III-6)

III-8 $P(FA) = (1-\alpha) P(F)$

Substituting $\beta P(A)$ for $P(\bar{F}A)$ and $(1-\alpha) P(F)$ for $P(FA)$ in (III-4) and solving for $P(A)$ yields

III-9 $P(A) = \dfrac{1-\alpha}{1-\beta} P(F)$

Substituting $P(A)$ of (III-9) into (III-7) yields

III-10 $P(\bar{F}A) = \dfrac{\beta(1-\alpha)}{1-\beta} P(F).$

Finally, substituting (III-6) (III-8) and (III-10) into (III-5)

yields

III-11 $P(\bar{F}\bar{A}) = 1 - \dfrac{(1-\alpha\beta)}{1-\beta} P(F).$

Summarizing, we obtain

III-12 $P(F\bar{A}) = \alpha z$

III-13 $P(FA) = (1-\alpha)z$

III-14 $P(\bar{F}A) = \frac{\beta(1-\alpha)}{1-\beta} z$

III-15 $P(\overline{FA}) = 1 - \frac{(1-\alpha\beta)}{1-\beta} z$

III-16 $P(A) = \frac{1-\alpha}{1-\beta} z$

where $z = (P(F)$.

From (III-15) and the inequality

$$0 \leqslant P(\bar{F}\bar{A}) \leqslant 1$$

we obtain

III-17 $0 \leqslant \frac{1-\alpha}{1-\beta} z + \alpha z \leqslant 1.$

As indicated in Section 3, a small value of α is desirable for a given test but α does not, by itself, reflect the detection capabilities of a test. For example, a test could alarm after every application. Such a test would detect all failures and hence would yield $\alpha = 0$. A preferable measure of failure detection capability is the causal counterpart of α; i.e.,

$$\gamma = P(\bar{a}|f)$$

where f = event of a single, random failure and a = corresponding causal alarm. The quantity, γ, is a direct measure of failure detection capability and can be evaluated independently of frequency of nuisance alarms. The quantity, γ, is called the "test deficiency". We will show, by means of an example, that

$$\gamma \cong \alpha$$

when the mission time is sufficiently small.

Example

In this example, we assume that failures and non-causal alarms are Poisson distributed in time with rates λ_F, λ_A, respectively. Then, if T = mission time, III-18 $$P(F) = 1-e^{-\lambda_F T}$$

III-19 $$P(\bar{F}A) = e^{-\lambda_F T}\,(1-e^{-\lambda_A T})$$

III-20 $$P(F\bar{A}) = e^{-\lambda_A T}\sum_{k=1}^{\infty} e^{-\lambda_F T}\,\frac{(\lambda_F T)^k}{k!}\,\gamma^k$$

$$= e^{-\lambda_A T}\,e^{-\lambda_F T}\,(e^{-\gamma\lambda_F T}-1)$$

III-21 $$P(\bar{F}\bar{A}) = e^{-\lambda_F T}\,e^{-\lambda_A T}$$

III-22 $$P(FA) = P(F) - P(F\bar{A})$$

$$= (1-e^{-\lambda_F T}) - e^{\gamma_A T}\,e^{-\lambda_F T}\,(e^{\gamma\lambda_F T}-1)$$

III-23 $$\alpha = \frac{P(F\bar{A})}{P(F)} = \frac{e^{-\lambda_A T}\,e^{-\lambda_F T}\,(e^{\gamma\lambda_F T}-1)}{1-e^{-\lambda_F T}}$$

III-24 $$P(A) = P(FA) + P(\bar{F}A)$$

$$= 1-e^{-\lambda_F T} - e^{-\lambda_A T}\,e^{-\lambda_F T}\,(e^{\gamma\lambda_F T}) + e^{-\lambda_F T}\,(1-e^{-\lambda_A T})$$

$$= 1-e^{-\lambda_A T}\;e^{-\lambda_F T}\,e^{\gamma\lambda_F T}$$

III-25 $$\beta = \frac{P(\bar{F}A)}{P(A)} = \frac{e^{-\lambda_F T}\,(1-e^{-\lambda_A T})}{1-e^{-\lambda_A T}e^{-\lambda_F T}e^{\gamma\lambda_F T}}$$

If T is small then

III-26 $$P(F) \cong \lambda_F T$$

$$P(\bar{F}A) \cong \lambda_A T$$

$$P(F\bar{A}) \cong \gamma\,\lambda_F T$$

$$P(FA) \cong (1-\gamma)\,\lambda_F T$$

$$\alpha \cong \gamma$$

$$\beta \cong \frac{\lambda_A}{\lambda_A+\lambda_F(1-\gamma)}$$

From this example we conclude that if the mission time is sufficiently small then

$$P(F\bar{A}) \cong P\text{ (a single failure and no alarm)}$$

$$P(\bar{F}A) \cong P\text{ (a single alarm and no failure)}$$

$$P(FA) \cong P\text{ (a single failure and a causal alarm)}$$

and we may approximate γ by α .

APPENDIX IV

Redundant Secondary Actuators

It has been assumed explicitly throughout all of the tradeoff studies that the mechanical voter of the secondary actuators is a signal selection device of the mid-valve (MV) type. As a consequence, it has the following properties:

In the absence of monitoring the signal selector is a majority device. If an input fails and is detected then that signal is disqualified and the SSD proceeds as a majority device with the remaining signals. The output fails if and only if

- the last signal input fails or
- there are at least as many failed (and not disqualified) inputs as non-failed inputs.

In addition, it is assumed that no failure, detected or not, will result in damage to the airframe provided that the good signals are in the majority subsequent to the failure. Thus, a single failed channel of a triplex channel, detected or not, will not result in a transient sufficient to cause damage to the airplane. In practice, a failure transient will always result when an active failure occurs. The severity of the transient is influenced by:

- aircraft dynamics
- mode of operation (i.e., rate or acceleration feedback, etc.
- dynamical properties of the actuators

In the absence of details regarding these influences it is impossible to characterize the effects of failures whether detected or not. Nevertheless, some insight can be obtained by considering an idealized version of a mechanical SSD as implemented in several existing and proposed aircraft.

1. Force Summing Characteristics

The mechanical quadruplex arrangement is shown in Figure IV-1. It is assumed that the detent force is significantly less than the maximum force input of the servos. otherwise differential pressure feedback would affect the results.

Referring to the figure:

x_i = Secondary actuator ram displacement, i th channel

y_i = Detent output shaft displacement, i th channel

x_i-y_i = Detent displacement, i th channel

K_L = Linkage compliance or spring constant

x_o = Primary actuator valve displacement

f_d = Detent breakout force

f_i = Force exerted on summing shaft by i th ram

f_s = Total force on summing shaft exerted by rams

z_i = Alternate equalization signal, i th channel

Figure IV-2 shows an analytical block diagram of the mechanical SSD. By making the observation that

$$x_i-x_o = f_i/K_L, \quad -f_d \leq f_i \leq f_d$$

$$= f_d, \quad x_i-x_o \geq \frac{f_d}{K_L} = a$$

$$= -f_d, \quad x_i-x_o \leq -\frac{f_d}{K_L} = -a$$

we can represent the SSD as shown in Figure IV-3. From the figure it can be seen that the mechanical SSD is a limited averaging device for a soft spring and becomes an MV SSD when the linkage compliance, K_L, is infinite (or if $K_F = 0$). A conventional representation of this device is shown in Figure IV-4 and an electrical implementation, in Figure IV-5.

2. Normal Performance

Establishing the dynamical performance of redundant actuators is a difficult and involved procedure and one which is beyond the scope of this study. However, assuming ideal operation, we know from past experience that the voting action of the mechanical transducer exhibits an undesirable threshold effect when the number of signal inputs is even. An example of this effect is shown in Figure IV-6 where the differences between channels are caused by differences in commands or in bias differences in follow-up signals. In many applications the threshold will induce a limit cycle oscillation. The threshold can be eliminated or reduced by equalizing the actuator (see Appendix VI) or by insuring that all signal inputs are the same.

3. Failure Effects and Transients

Most failures can be classified in one of the following categories:

- Step (usually a hardover)
- Slowover
- Passive (usually a null failure)
- Oscillatory
- Dynamic (i.e., gains, time constants, etc.)

In this section we are primarily interested in the response of the actuators to hardovers, null and oscillatory failures. Because a quadruplex arrangement reduces to a triplex arrangement after a detected failure the transient effects of failures can be estimated for both arrangements by considering the following sequences of failures:

- 1st failure undetected
  2nd failure undetected
- 1st failure detected
  2nd failure undetected
- 1st failure undetected
  2nd failure detected
- 1st failure detected
  2nd failure detected

Figures IV-7 through IV-12 show the effects of these failure sequences in the quadruplex and triplex configurations. In all cases the failures were chosen to exhibit the most severe transient effects. Referring to the figures

x_i = i th channel input

x_o = MV SSD output.

In the representation of the output it is assumed that

a. $x_i = x + d_i$

b. $x_i > 0$

c. $d_1 < d_2 < d_3 < d_4$

where x = nominal signal

and d_i = fixed offset, i th channel.

From the figures it can be seen that

- The transient which accompanies the first failure is determined by the channel offsets.
- Transients due to disengaged failures can be more severe than the transient which accompanied the failure. But in no case is the disengage transient more than twice the amplitude of the failure transient.
- Loss of control does not always result from two undetected failures in a triplex or quadruplex SSD. Loss of control depends upon direction of the two failures.
- In the quadruplex SSD two hardover failures in the same direction result in passive loss of control.
- In a triplex SSD two hardover failures in the same direction result in a non-passive loss of control.
- Extrapolating to passive failures, loss of control in the triplex and quadruplex configurations is always passive if at least one of the undetected failures is passive (fails to null).

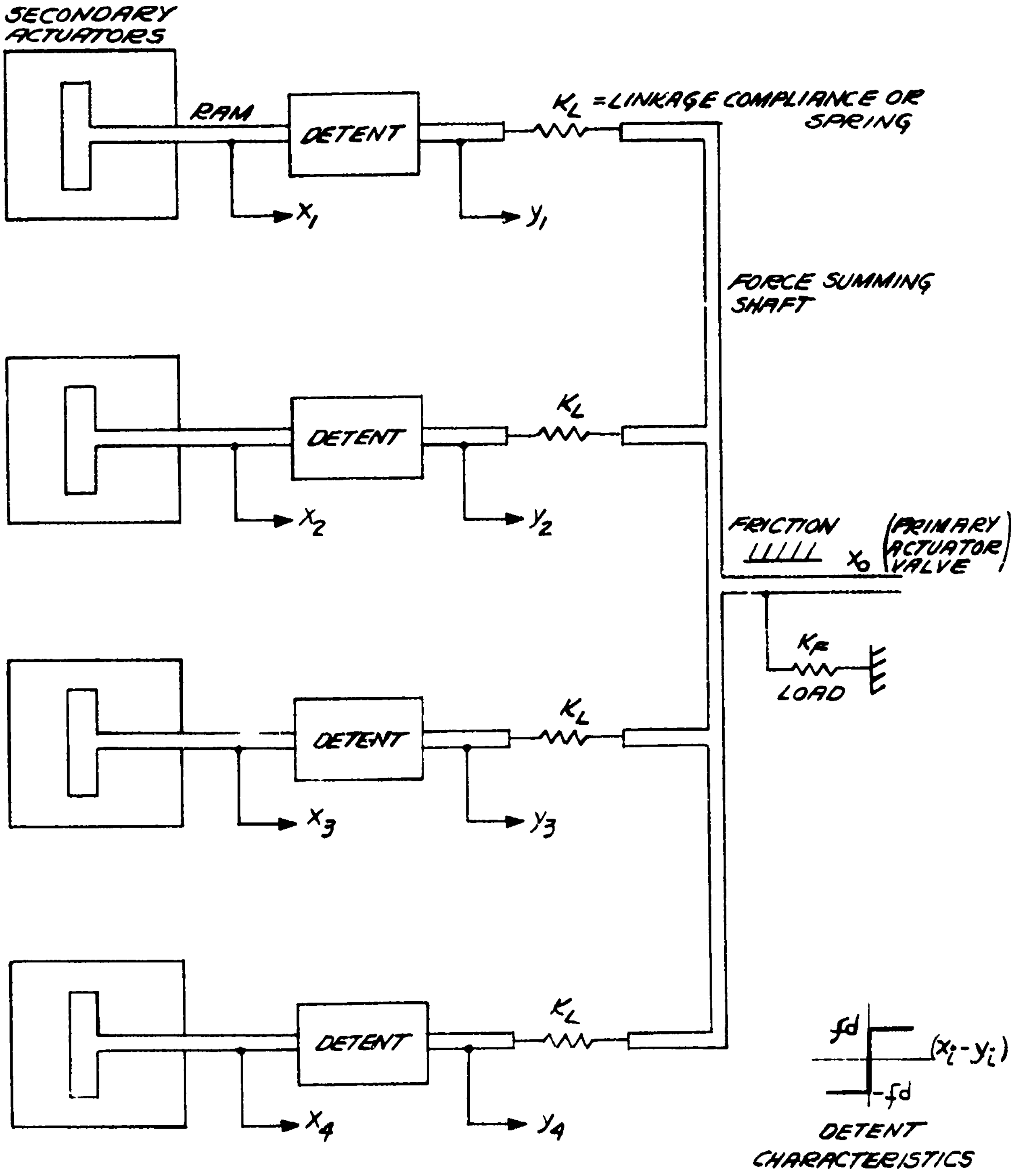
Transients due to servo failures can be eliminated (assuming follow-up biases are neglibible) by providing a common servo command from all channels. Even with common commands, transients due to failures of the upstream units could propagate to the surface--as from the common signals from the sensor SSD's.

Oscillatory Failures

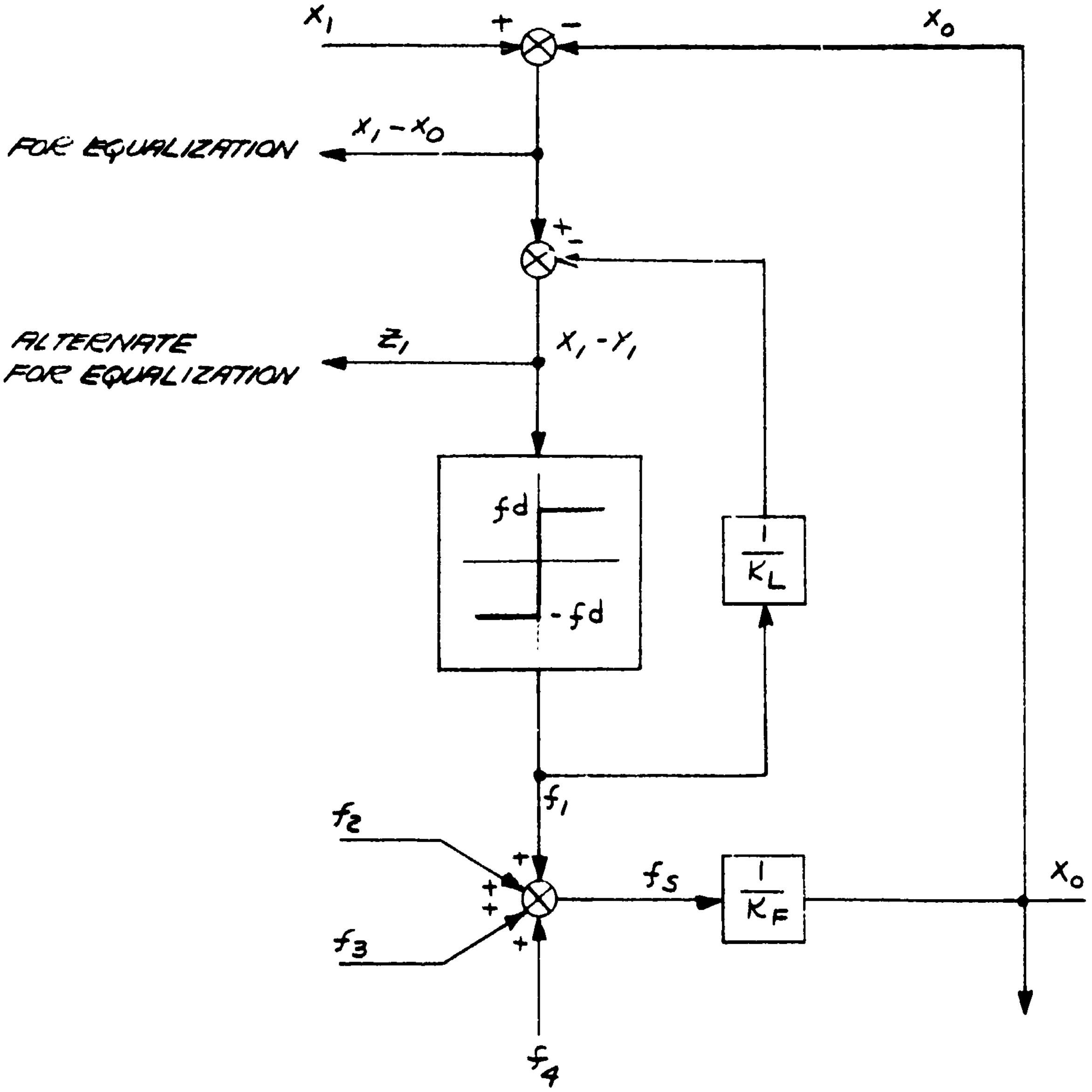
The effects of an oscillatory failure can be seen in Figure IV-13. The oscillation is propagated to the output with an amplitude determinded by the channel offsets. The frequency of the oscillation is determined by the failed component and the local area effected. The combined effect of an undetected oscillatory failure and large channel offsets could result in an undesirable airplane response.

Summary

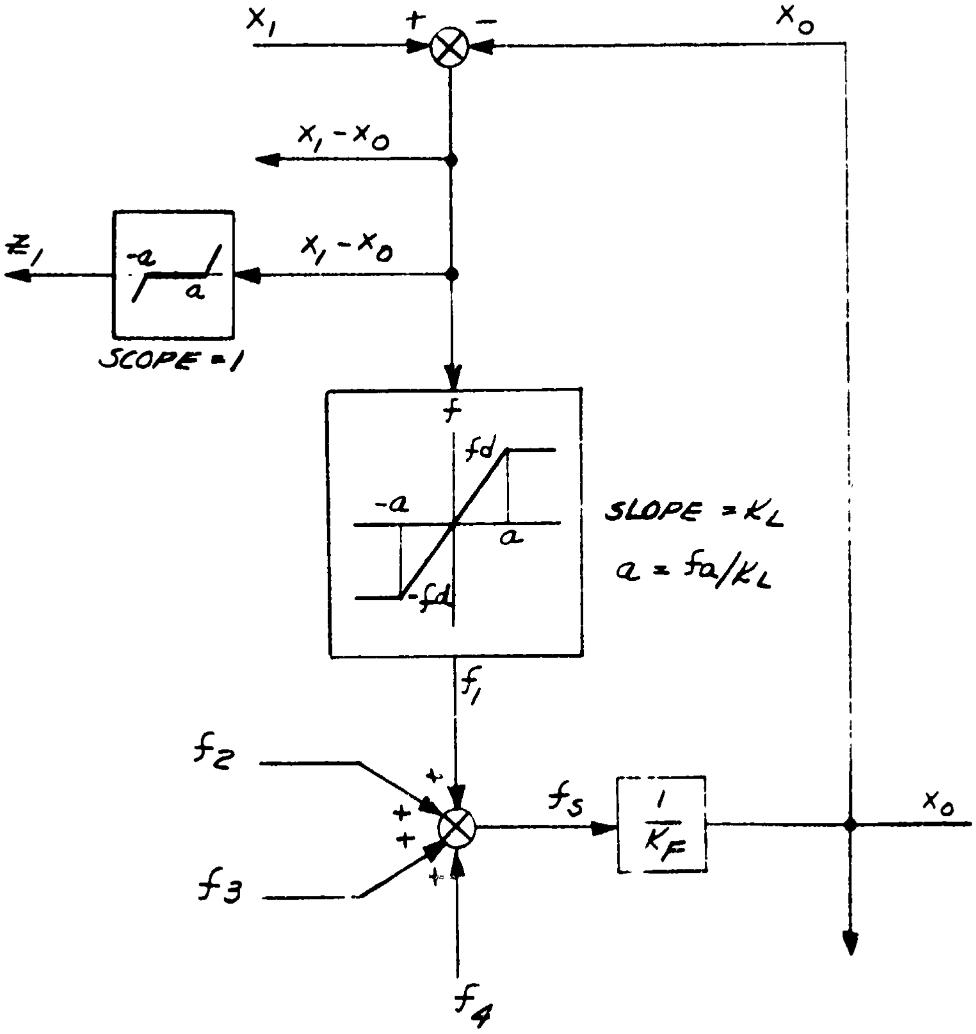
It has been shown that the effective voting properties of force summed secondary actuators approximate an ideal mid-value signal selector device with some degree of limited averaging. If channel offsets can be eliminated or reduced to acceptable levels then performance proceeds undegraded in the presence of a detected or undetected single channel failure in both the triplex and quadruplex configurations.



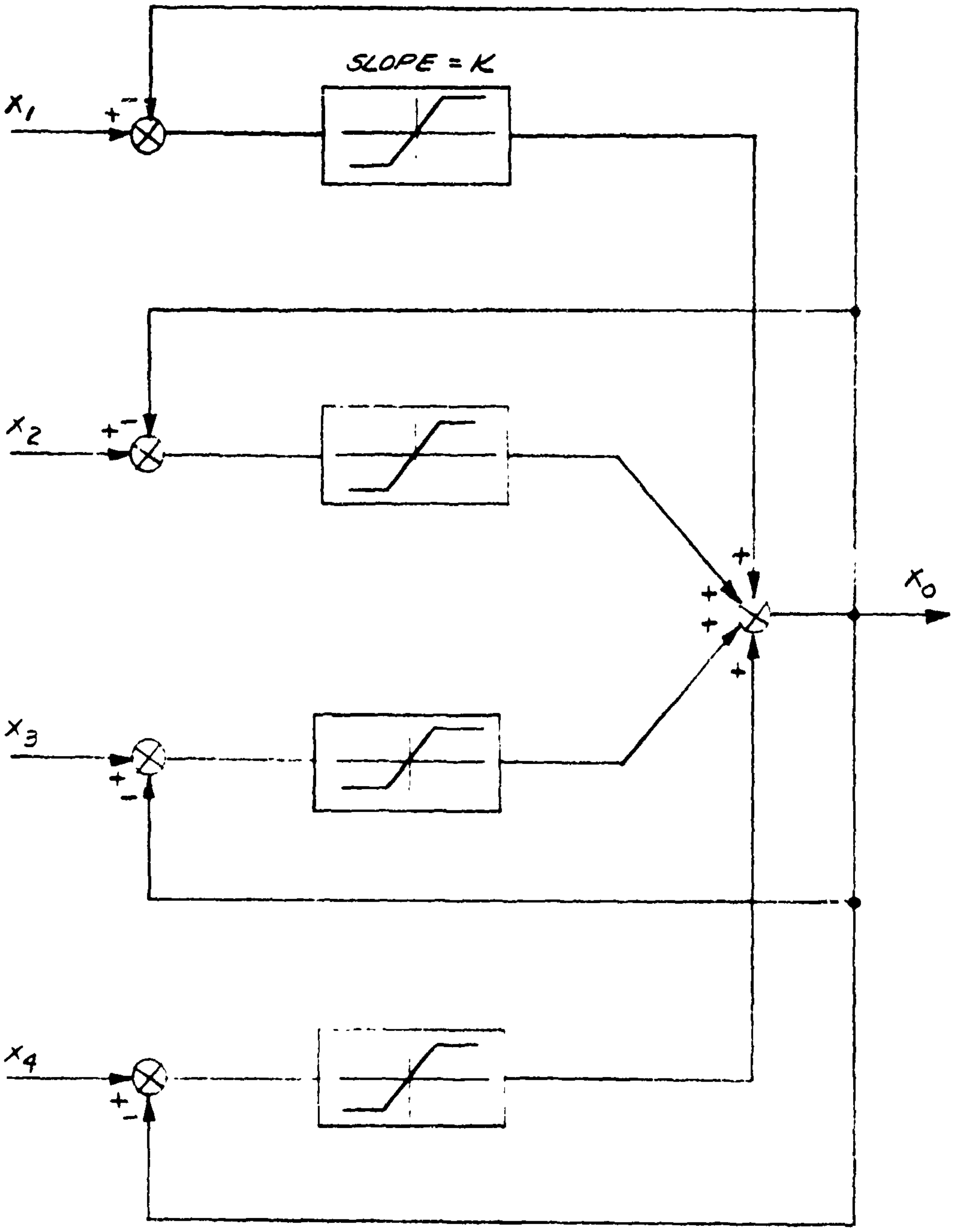
OUADRUPLEX
IDEALIZED FORCE SUMMED MECHANICAL SSD
FIGURE IV-1



ANALYTICAL BLOCK DIAGRAM
OF MECHANICAL SSD
FIGURE IV-2

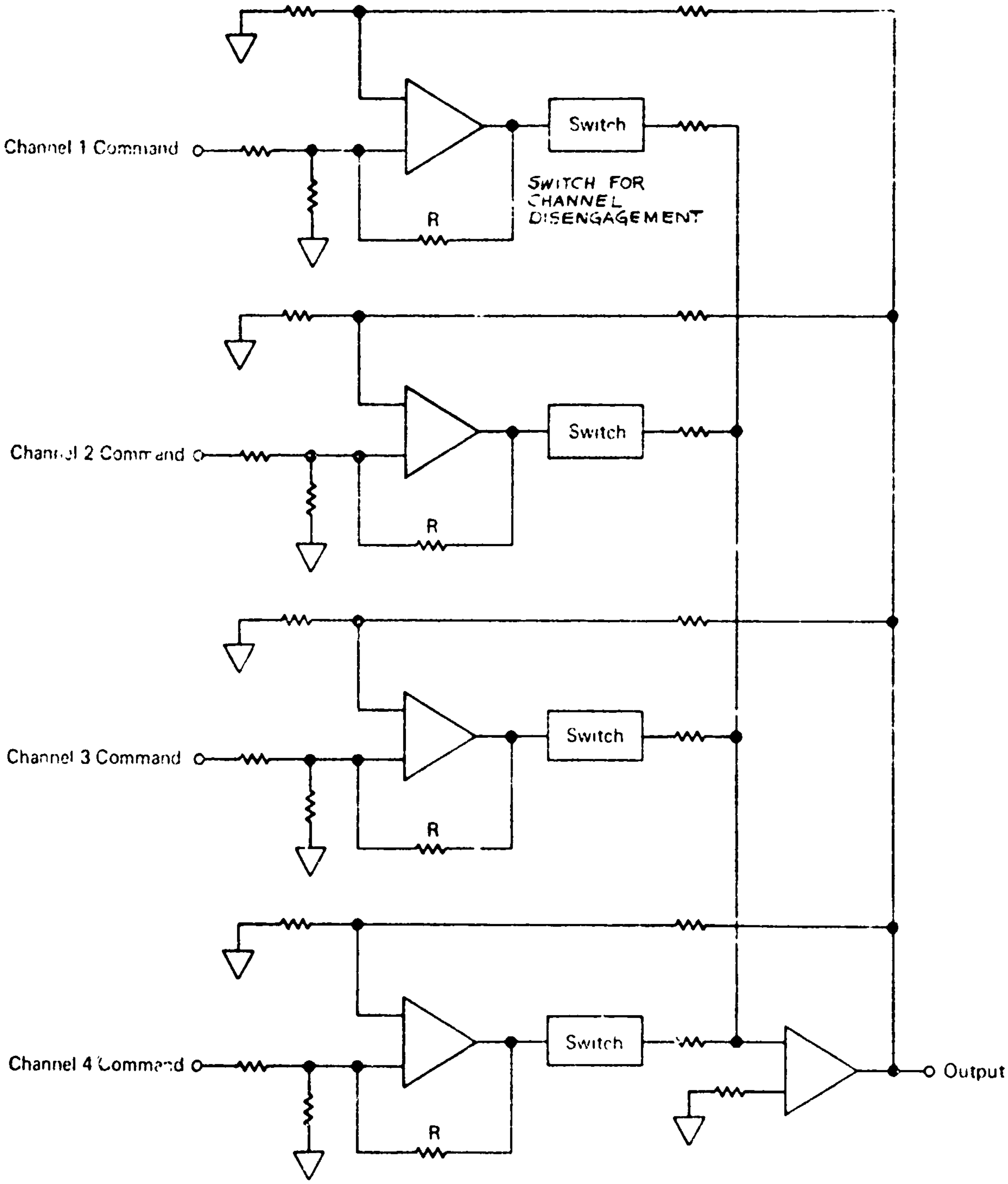


EQUIVALENT ANALYTICAL BLOCK DIAGRAM
OF MECHANICAL SSD
FIGURE IV-3

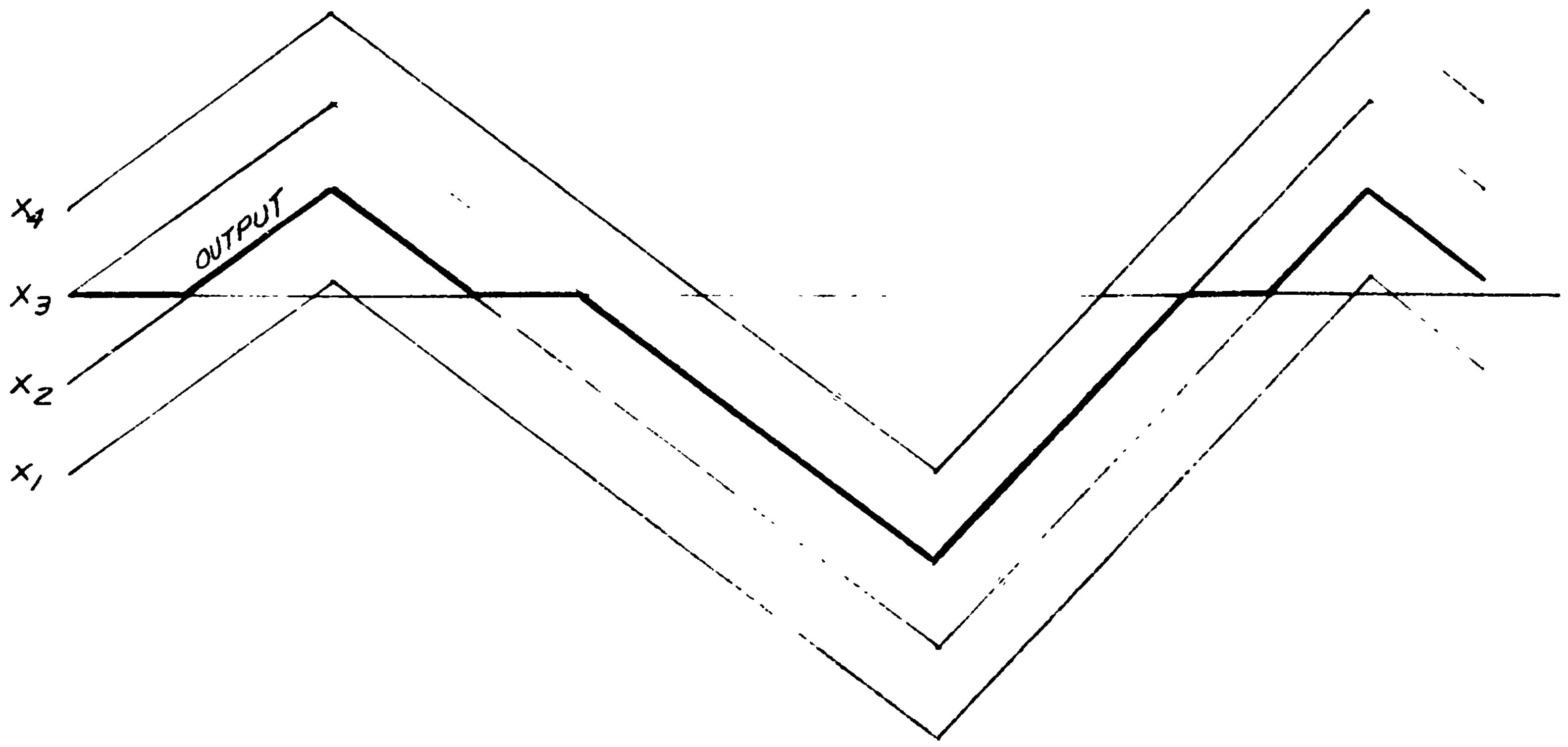


$K = \infty$ SSD = MV SELECT
$K \neq \infty$ SSD = LIMITED AVERAGING

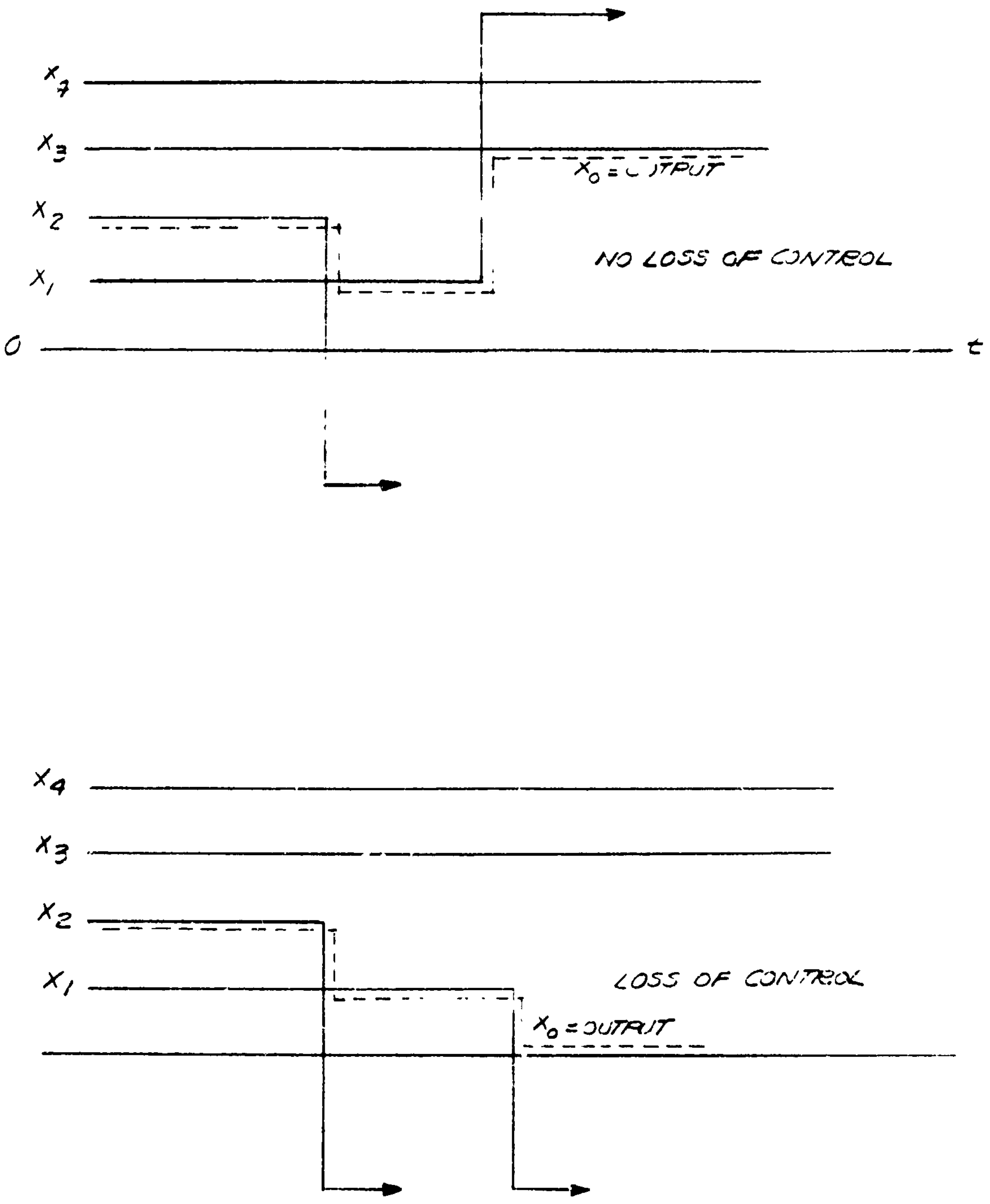
SSD PROCESS REPRESENTATION
FIGURE IV-4



SIGNAL SELECTION DEVICE
Four Channel Operational Amplifier Type
FIGURE IV-5

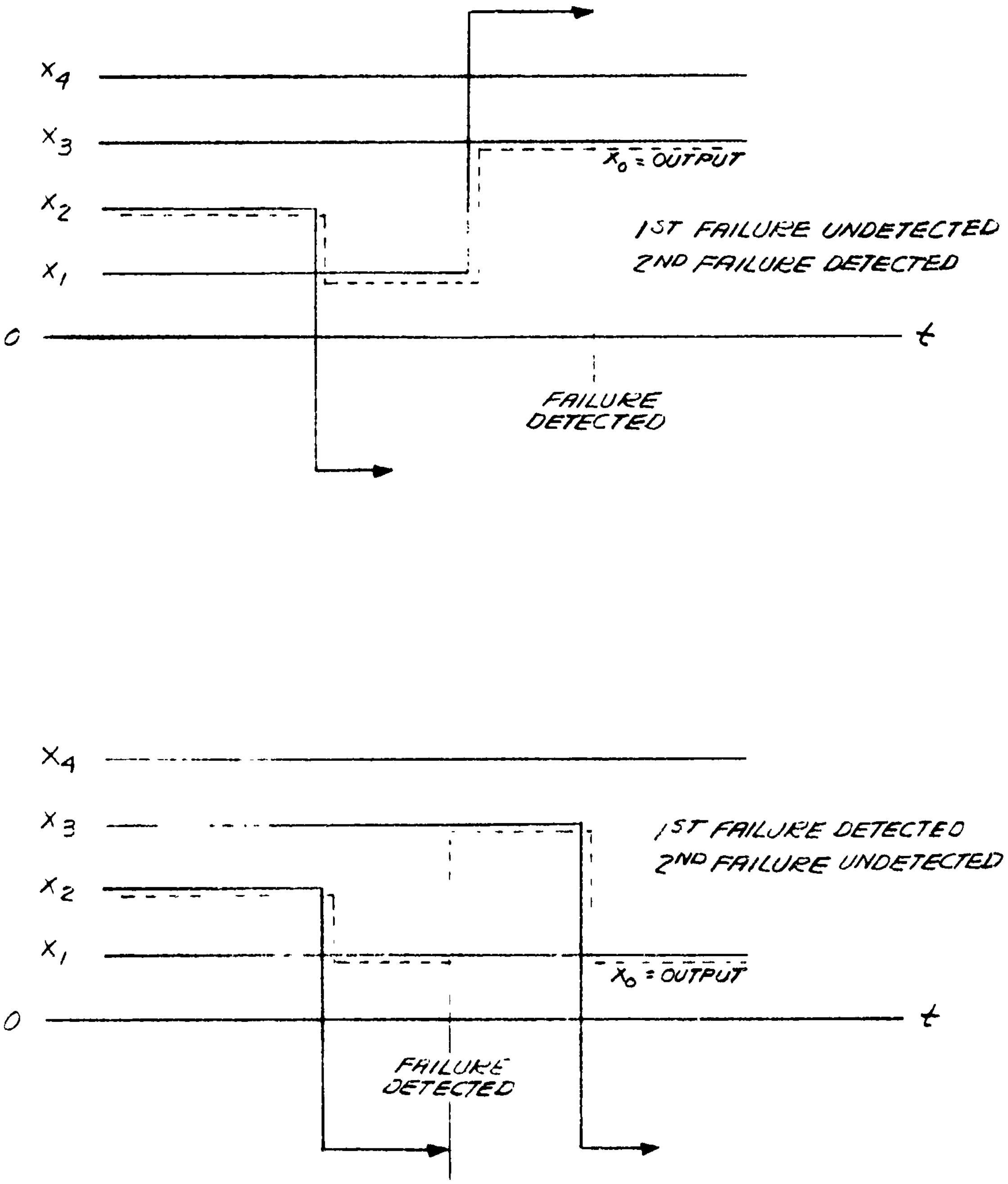


THRESHOLD CHARACTERISTICS
OF A QUADRUPLEX MV SSD
FIGURE IV-6



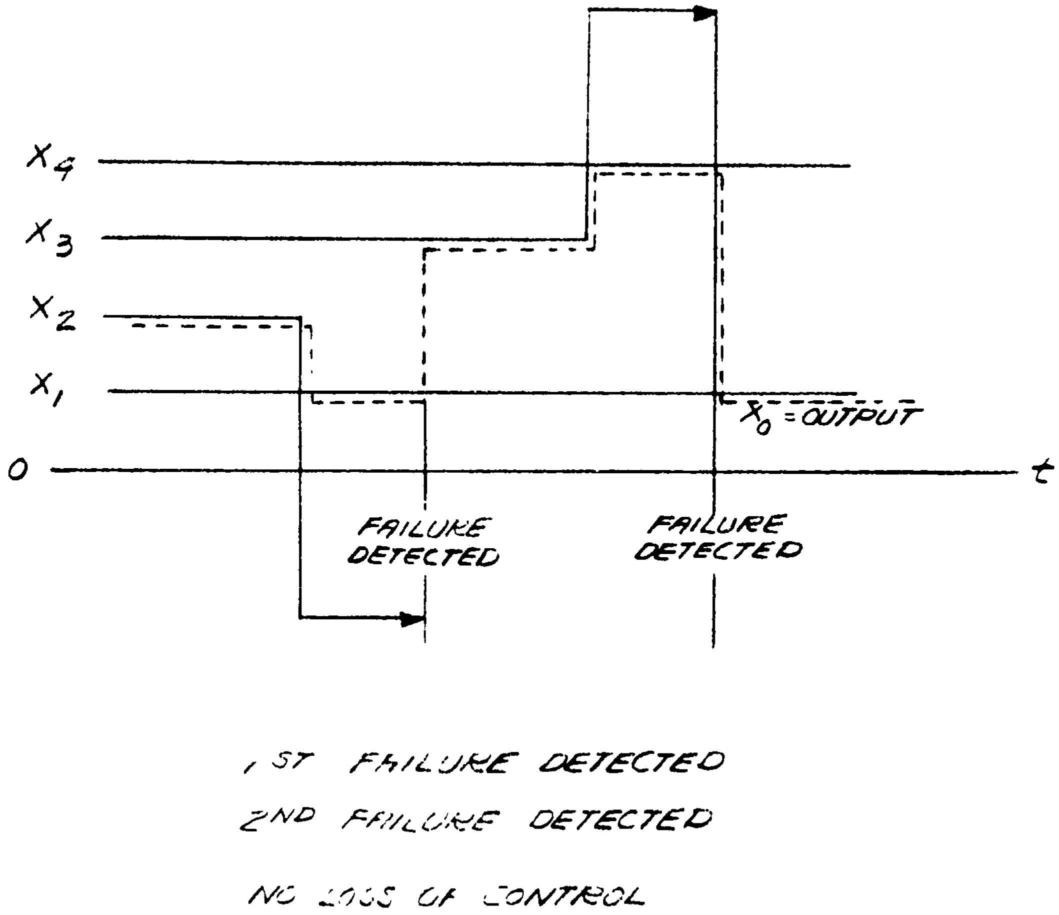
EFFECTS OF HARDOVER FAILURES
IN A QUADRUPLEX MV SSD

1st FAILURE UNDETECTED
2nd FAILURE UNDETECTED
FIGURE IV 7



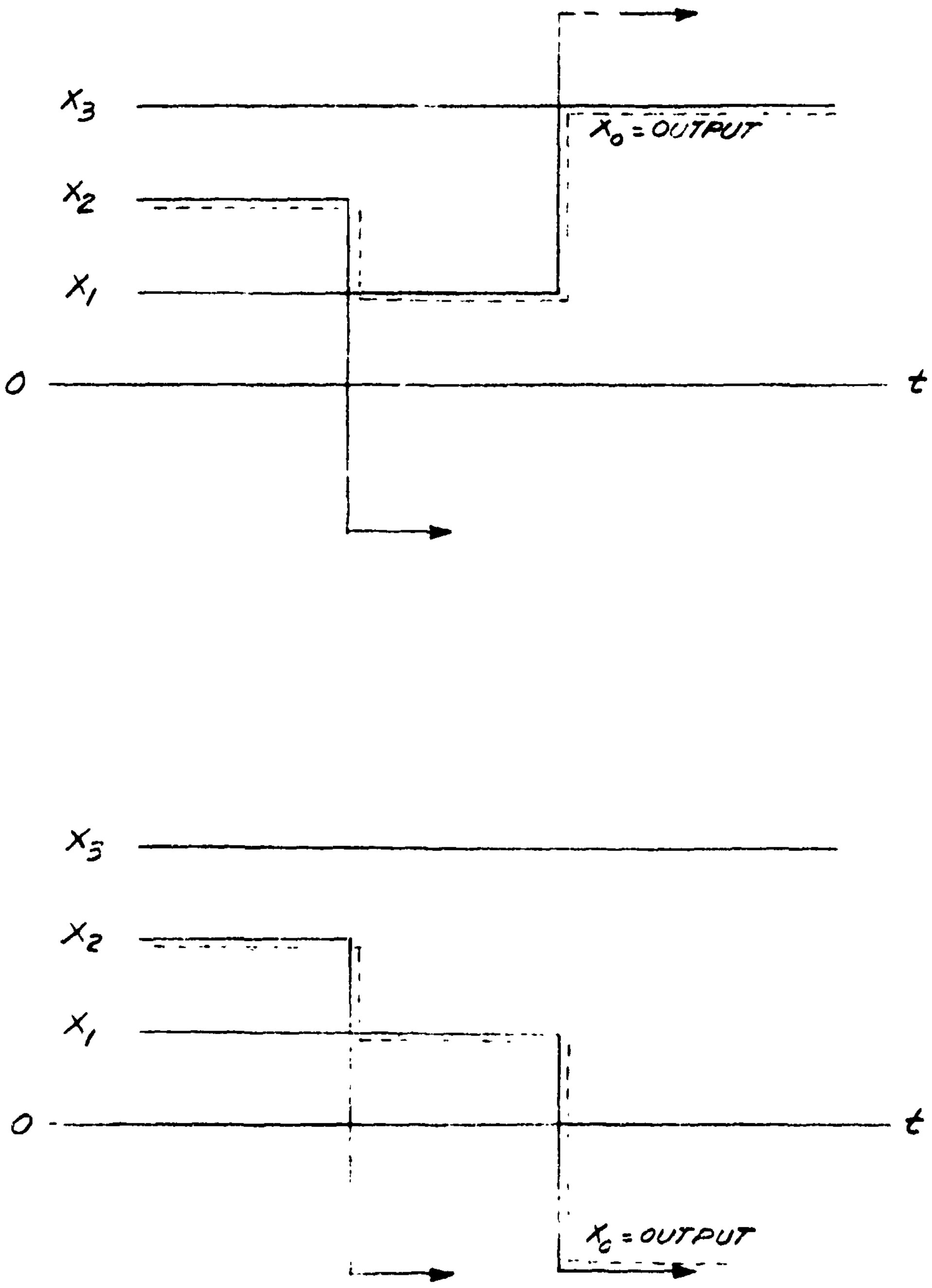
Effects of Hardover Failures in a Quadruplex MV SSD
1st Failure Undetected, 2nd Failure Detected
1st Failure Detected, 2nd Failure Undetected

FIGURE IV-8



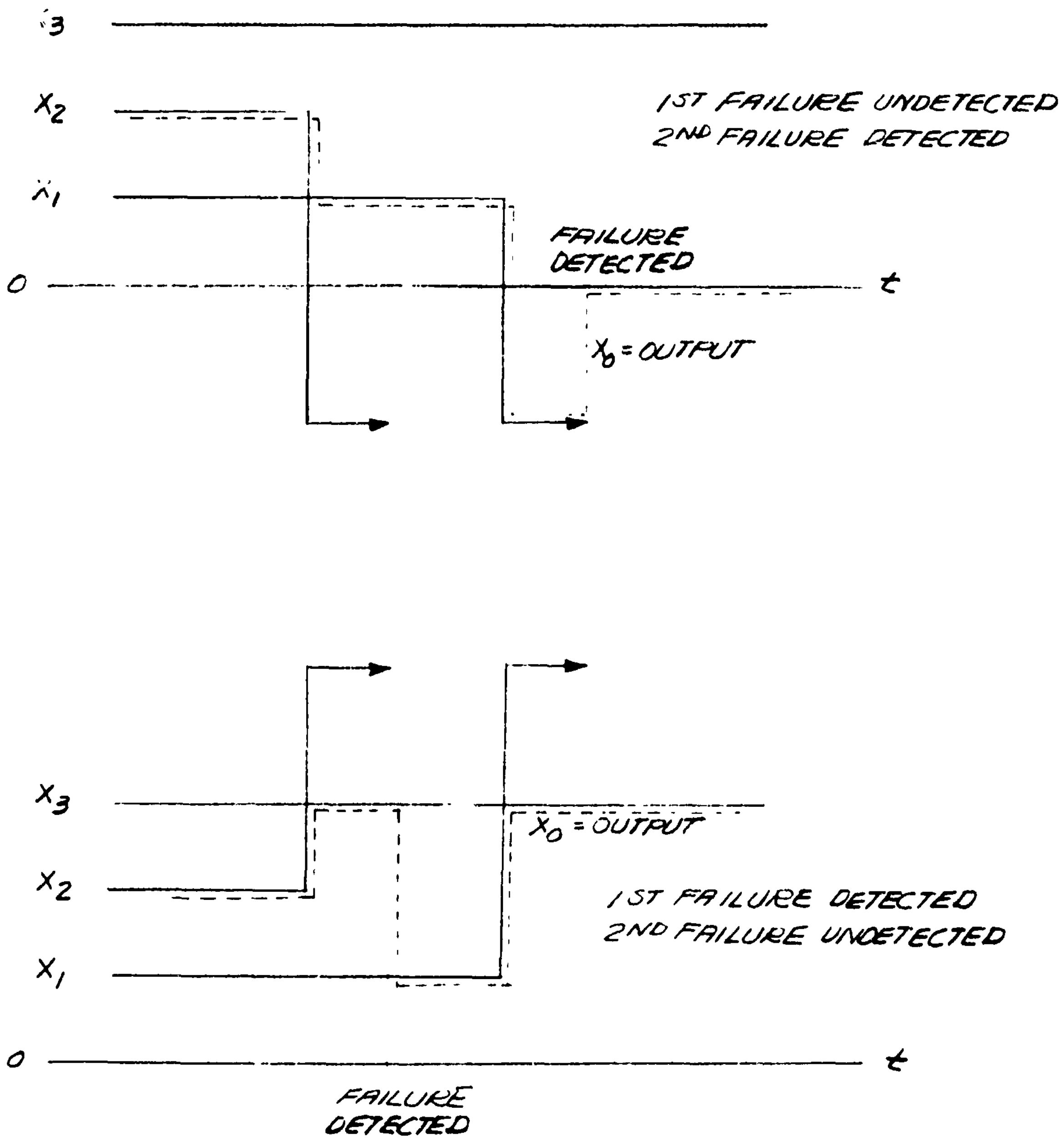
Effects of Hardover Failures in a Quadruplex MV SSD
1st Failure Detected, 2nd Failure Detected

FIGURE IV-9

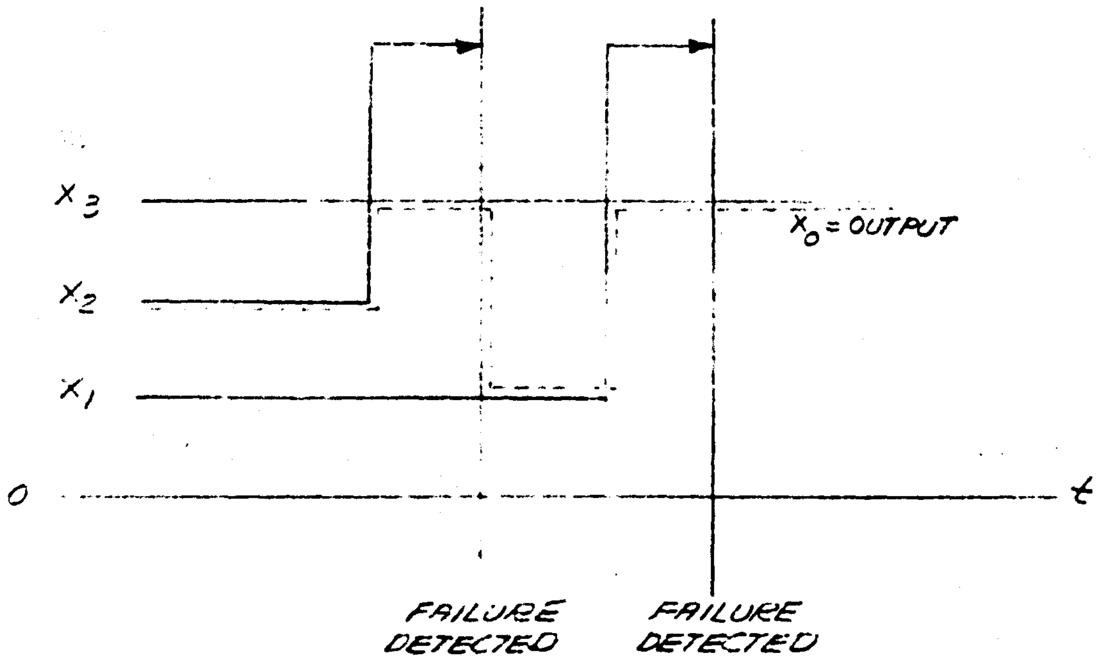


EFFECTS OF HARDOVER FAILURES
IN A TRIPLEX MV SSD

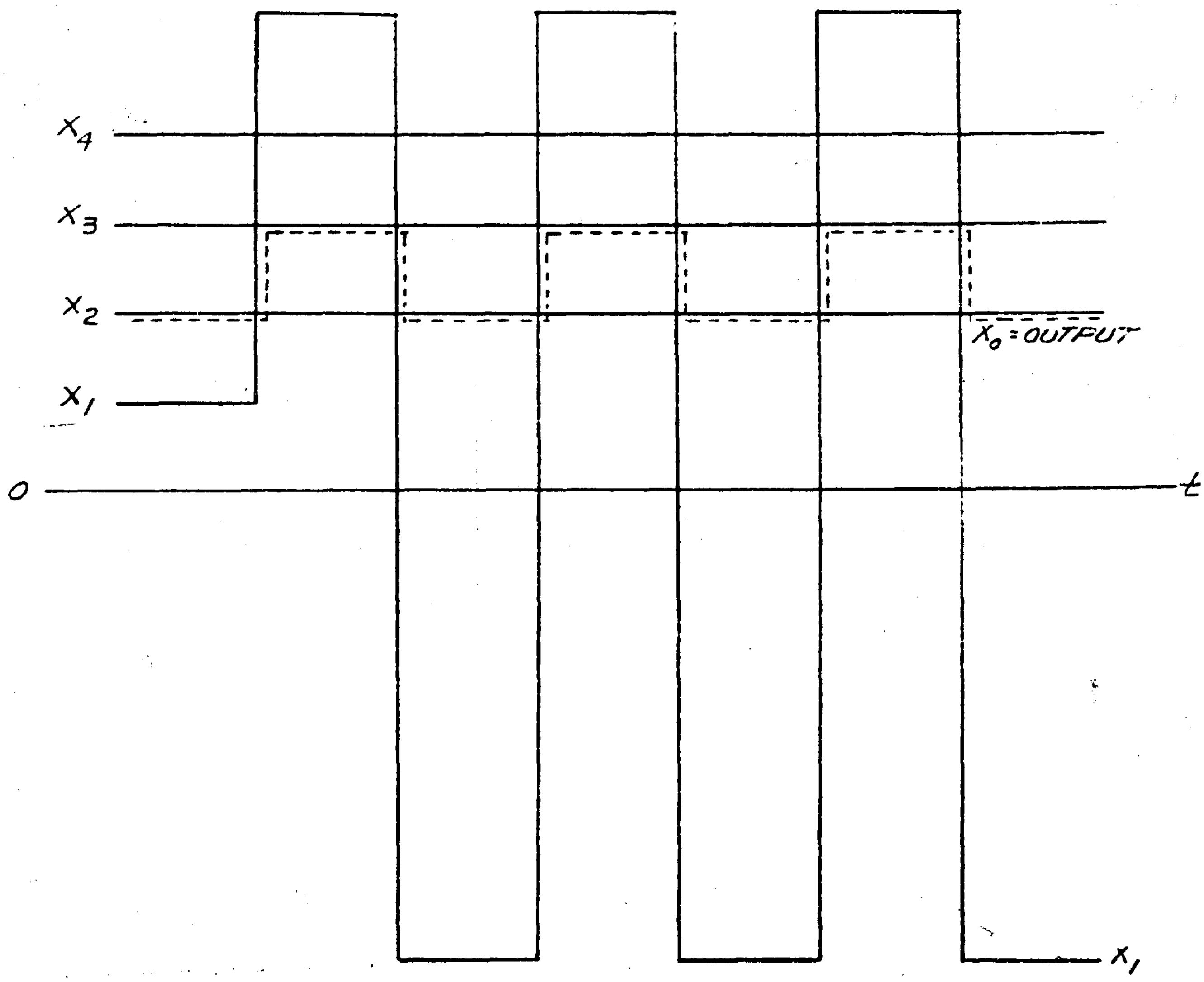
1st FAILURE UNDETECTED
2nd FAILURE UNDETECTED
FIGURE IV-10



Cffects of Hardover Failures in a Triplex
MV SSD
1st Failure Undetected, 2nd Failure Detected
1sc Failure Detected, 2nd Failure Undetected
FIGURE IV-11



Effects of Hardover Failures in a Triplex
MV SSD
1st Failure Detected, 2nd Failure Detected
FIGURE IV-12



EFFECT OF OSCILLATORY FAILURE
ON OUTPUT OF A QUADRUPLEX MV SSD
FIGURE IV-13

APPENDIX V

The Digital Computer

1. Basic Architecture and Functional Description

a. System Organization

The organization of a single thread digital automatic flight control system (AFCS) is shown in Figure V-1. The primary unit is the digital processor which will be described in detail subsequently. All of the remaining components are associated with the input/output (I/O) interface.

b. Digital Processor and I/O Organization

For purposes of the study, it is assumed that the basic digital processor is a single address minicomputer. While existing computers differ in details, they are sufficiently similar in organizational structure to justify the use of a "typical" organizational block diagram. Because it is typical and because detailed information is available, it was decided to use the organizational structure of the Bendix BDX-910 digital computer. The organizational block diagram of the computer and associated I/O is shown in Figure V-2.

c. I/O Interface

The I/O interface consists of the following components:

(1) Signal conditioners and prefilters for all dc input signals. The prefilters suppress high frequency sensor noise which, in a digital system, would otherwise "fold" into a lower frequency.

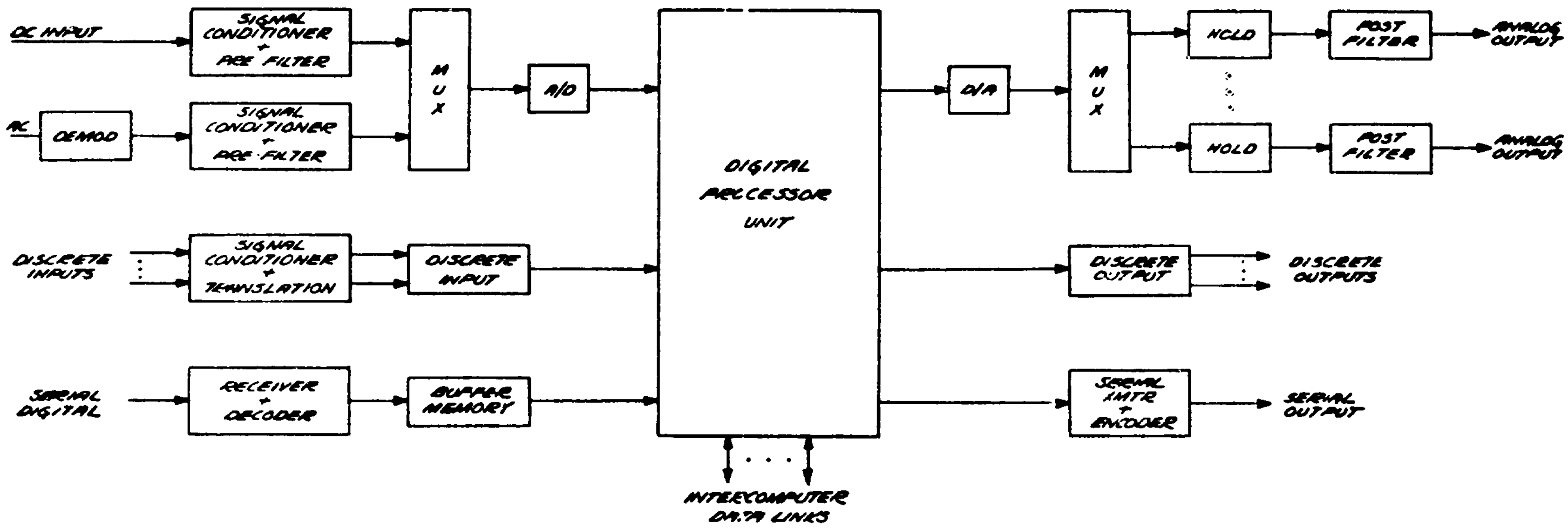
(2) Demodulators for AC inputs.

(3) Analog to digital (A/D) converters and multiplexers if the A/D is time shared.

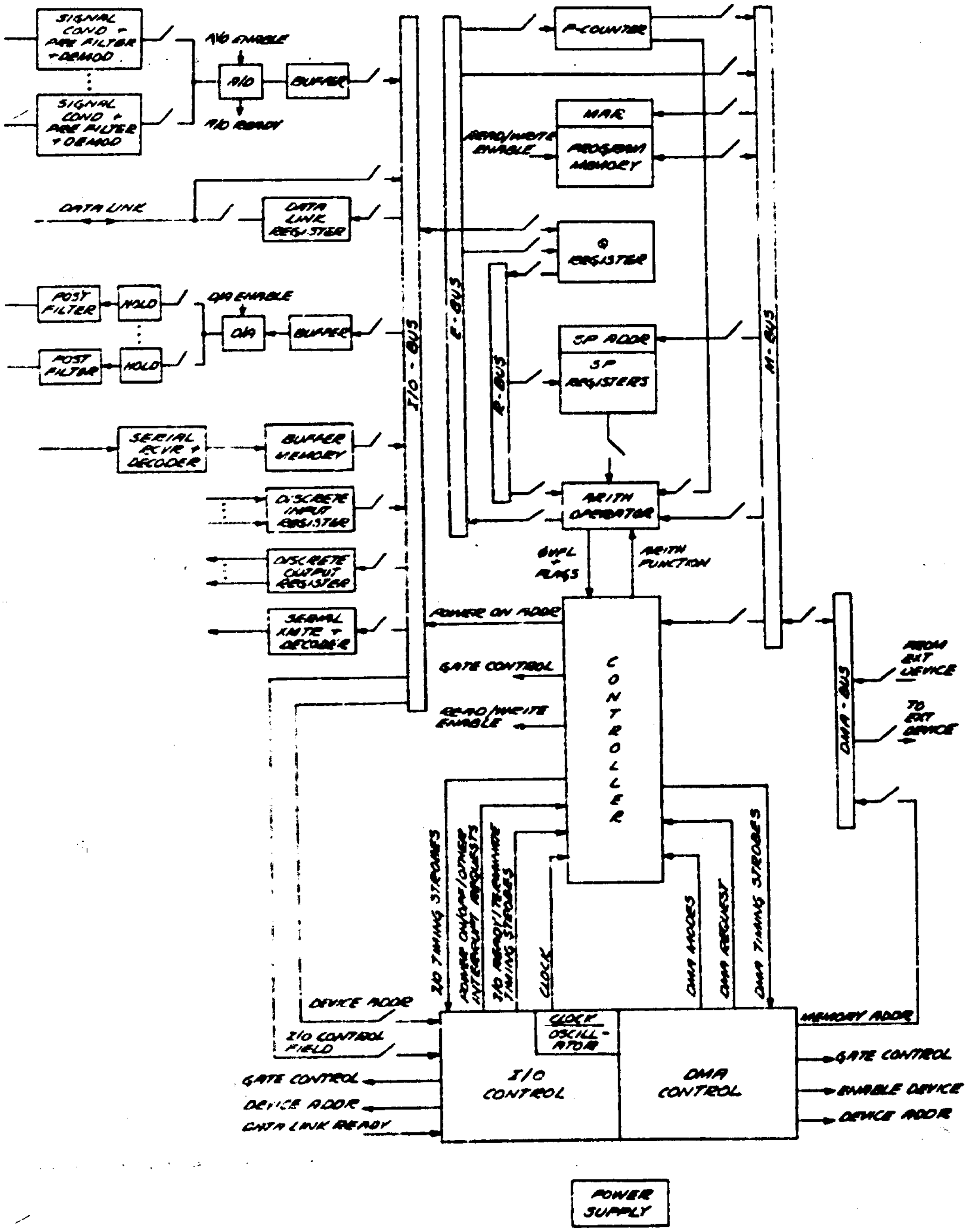
(4) Discrete input signal translators and signal conditioners.

(5) Serial receivers, decoders and buffer storage.

(6) Parallel and serial data links for communication between computers.



DIGITAL FLIGHT CONTROL SYSTEM MECHANIZATION
DIGITAL PROCESSOR I/O INTERFACE
FIGURE V-1



DIGITAL COMPUTER AND ASSOCIATED I/O
FIGURE V-2

(7) Digital to analog (D/A) converters and multiplexers if the D/A is time shared.

(3) Sample and hold circuits.

(9) Post filters - To reduce intersample ripple and frequency folding.

(10) Discrete output registers.

(11) Serial transmitters and encoders.

(12) I/O Controller - This unit controls the timing and gating of the I/O and DMA.

(13) Direct memory access controller - Although shown as a separate unit this controller is part of the I/O controller.

(14) Oscillator/Clock - This is the basic timing mechanism of the computer. It consists of a 16 MHZ oscillator and counters which yield submultiples of the oscillation frequency.

(15) Power Supply - A single power supply supplies the power for both the digital processor and I/O. In some cases the core memory has its own separate power supply.

d. <u>Digital Processor</u>

The digital processor consists of the following components:

(1) Program Counter - This register contains the address of the next instruction to be executed.

(2) Memory Address Register (MAR) - This register stores the memory address. At the issuance of an appropriate enable pulse, the word whose address is in the MAR is either read or replaced.

(3) Memory - This unit consists of between 4K and 64K 16 bit words of storage for either instructions or data. The memory is usually a core or semiconductor type. Only a portion of the semiconductor memory can be overwritten. The entire core memory can be overwritten unless the write capability is hardwire inhibited.

(4) Q-Register - This is a general purpose register used as a buffer register for I/O interfacing or writing into the scratch pad registers.

(5) Scratch Pad Address Register (SPADDR) - This register contains the address of one of the arithmetic or index registers. Its function is the same as the MAR.

(6) Scratch Pad (SP) - The scratch pad consists of the arithmetic and index registers. It is always possible to overwrite a scratch pad register.

(7) Arithmetic Operator - This unit performs the arithmetic and logic operations such as shifting, complementing, adding, subtracting, multiplying and dividing. It is also used as a simple gating register.

(8) Controller - This is the brain of the computer. It decodes each instruction and generates:

(a) Gating Signals

(b) I/O and DMA timing strobes

(c) Logic levels

It also selects appropriate arithmetic functions and enables memory read/write.

(9) Memory bus (M-bus), E-bus, R-bus - These are parallel data busses used for intercomputer transfer of data.

(10) I/O bus - This is a parallel data bus which interfaces with the I/O devices.

(11) Direct memory access (DMA) - Direct memory access is the provision to transfer external data directly into memory without requiring software control or processor time for alternate fetches and execution. DMA is considered a part of the I/O even though Figure V-2 shows separate I/O and DMA controls.

(12) DMA bus - This is a parallel data bus which interfaces with DMA devices.

(13) Intercomputer data link - In the BDX-910, the data link consists of either a parallel or serial bus and a buffer which is used exclusively for communication between computers.

e. Functional Operation

The functional operation of the computer will be described by means of an example.

(1) At 'power on' an interrupt is generated by the "power on" monitor in the I/O. This causes the controller to insert a (hardwired) starting address on the I/O bus which is then transferred to the P-counter and MAR. Thereafter the P-counter is normally incremented by the least significant bit unless otherwise instructed.

(2) Suppose the initial instruction calls for a transfer of data from a memory location to an SP-register. In this case, the instruction will contain the address of both the memory word and SP-register.

(3) The controller gates the contents of the P-counter into the MAR via the M-bus.

(4) The contents of memory (which contains the instruction) is read out and transferred to the controller via the M-bus. Simultaneously, the contents of the M-bus are gated to the Q-register via the E-bus. Those bits which reference the SP-register are gated into the SP address register.

(5) The controller next gates those bits of the instruction which reference the memory location from the Q-register to the MAR via the E and M busses. The data word is then read out of memory and gated onto the M-bus. From the M-bus, the data is transferred to the SP register, as dictated by the SP address register, via the E-bus, Q-register and R-bus.

The controller is now ready for the next instruction.

Characteristics Peculiar to Digital Systems

The two most distinguishing characteristics of a digital system are:

- The extent to which components are time shared and

- The discrete word signal and computational format.

f. Advantages of Time Sharing

(1) Permits utilization of sophisticated algorithms without a proportional increase in size, cost, weight, etc.

(2) Permits standardization of components and consequent refinement of manufacturing processes which tends to significantly increase component reliability.

(3) Facilitates, at least in principle, self checking of the time shared components.

g. Advantages of Discretization

(1) No tolerance buildup in signal chain once data has been converted.

(2) Almost total insensitivity to noise in the signal chain once data has been converted.

(3) Permits standardization of components which results in improved reliability.

(4) The discrete work format is ideal for logic computations.

The absence of tolerance buildup in the signal chain permits extremely accurate cross-channel monitoring. Any differences which do exist are the result of sensor differences or a possible out-of-synch condition of one computer cycle. Thus, with sensor monitoring excluded, the problem of nuisance alarms is practically eliminated.

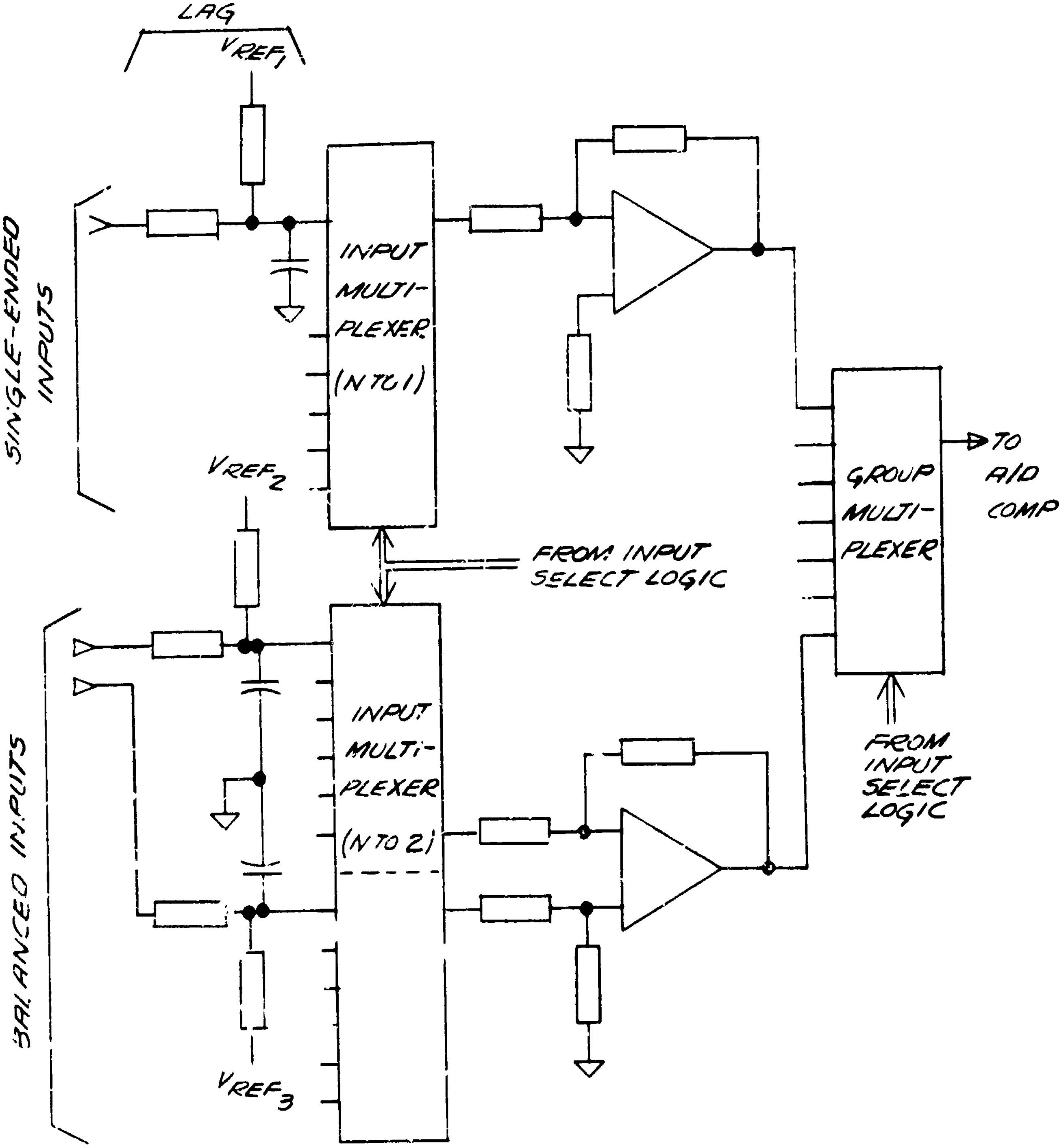
2. I/O Interface

a. Analog Input and Output Signals

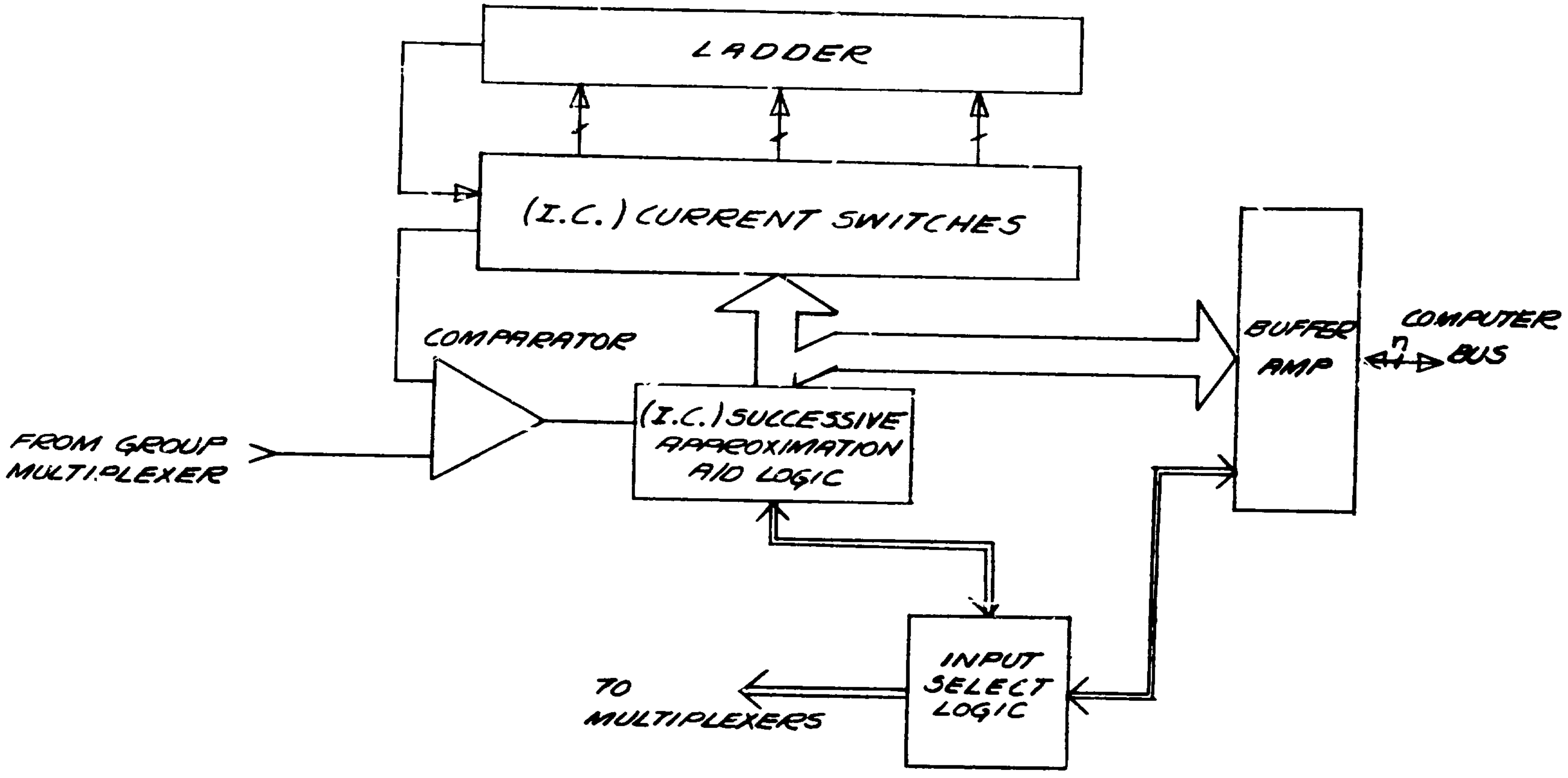
Analog input signals are first handled by passing them through a lag pre-filter as shown in Figure V-3. Note that provisions are made for both single wire and two-wire type signals. The resistance of the lag also serves as part of the scaling of the input. The input is then presented to an input multiplexer appropriate to the signal class, either one- or two-wire and, in the case of A.C. signals, strobed for peak value detection.

The outputs of the input multiplexers are then reduced to single-ended signals (if required) and gain adjusted in groups. A group multiplexer then selects the signal for conversion.

The A/D converter is a high speed successive approximation device (Figure V-4) using several LSI and hybrid microcircuits. Use of a high speed A/D makes possible the digitizing of A.C. as well as D.C. signals without additional hardware by restricting the selection of A.C. inputs to the time of the peak of the A.C. reference.



ANALOG INPUT CIRCUITS
FIGURE V-3



INPUT (A/D) CONVERTEK
FIGURE V-4

Figure V-5 shows the method of handling analog outputs. Digital output data is held in a buffer register while being converted to analog form by a high speed D/A converter. The analog output voltage is then impressed on the holding capacitor of the appropriate output channel by means of an F. E. T. demultiplexer. The demultiplexer has suitable 'ON' and 'OFF' impedances for use in this 'sample and hold' configuration. These circuits also make maximum use of large scale integration and hybrid circuit techniques.

The output of the holding capacitor is buffered by an extremely high input impedance buffer amplifier and transformed to suitable levels for the outputs. A low pass filter function is included as part of the output buffering for those outputs where the ripple component must be reduced.

b. Discrete Input and Output Signals

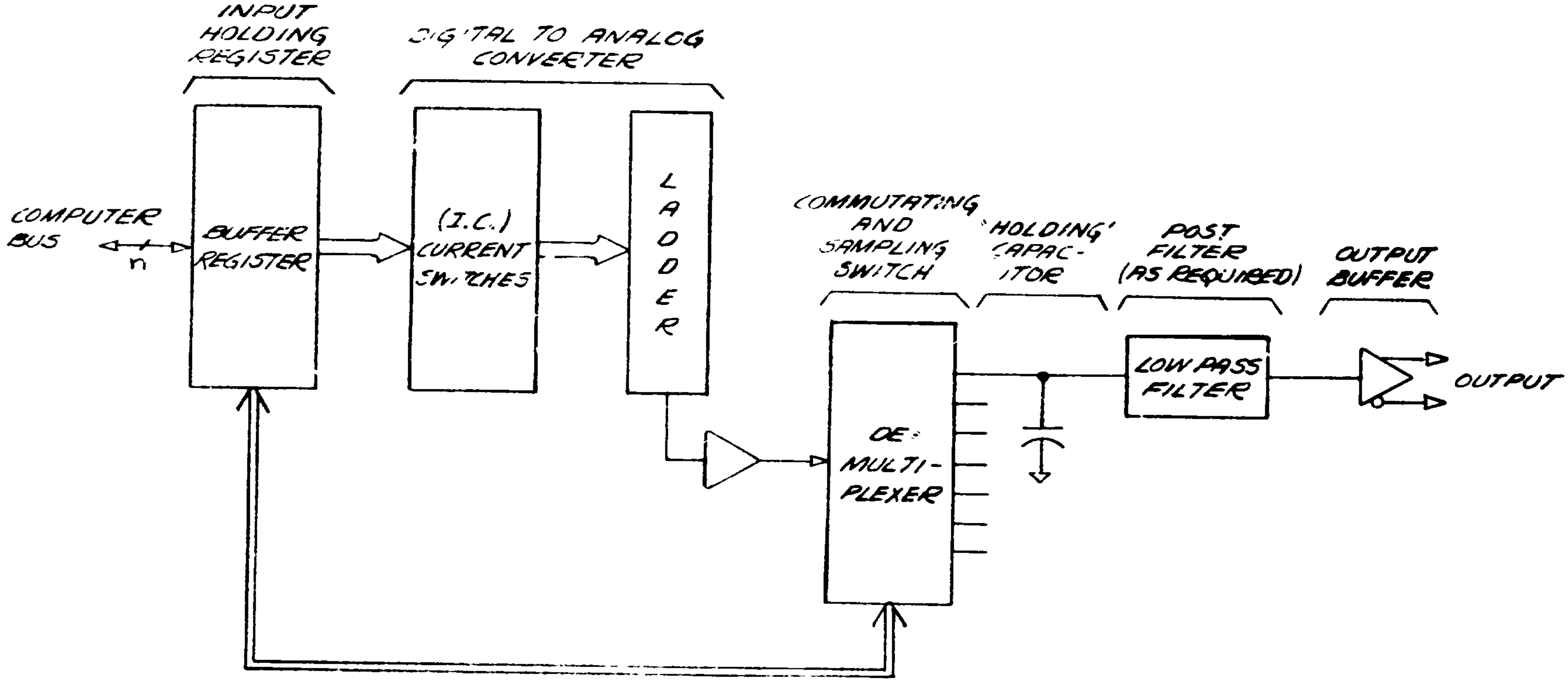
There are cases where it is advisable (by virtue of short wire runs, safety, or lack of sufficient data to be transferred) to use single 'dedicated' wires to communicate single on/off functions. A typical signal in this class could be a validity signal from a sensor, or a self-test command to a sensor.

For best noise immunity, it is recommended that discrete signals to or from points outside any package be at a 28 vdc level. These signals are readily handled on single unshielded conductors in aircraft wiring.

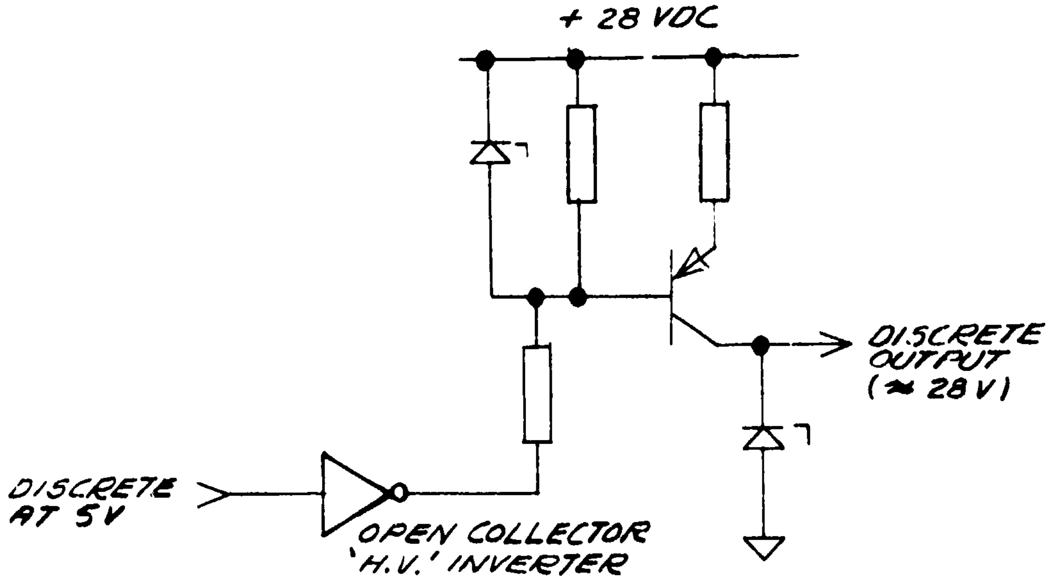
Where this is not possible, use of lower level signals (5 volts) is permissible if twisted pair (preferably shielded) wiring is used along with a balanced receiver circuit.

Figure V-6a shows a typical 28 volt discrete driver. This driver is current limited and transient suppressed as protection against 'normal' pick-up transients and faults. A matching receiver is shown in Figure V-6b. The use of a high resistance divider into the first CMOS inverter allows use of the internal clamping of the inverter for transient protection. The high input impedance of the second inverter allows large noise filter lag time constants using conveniently small capacitor values.

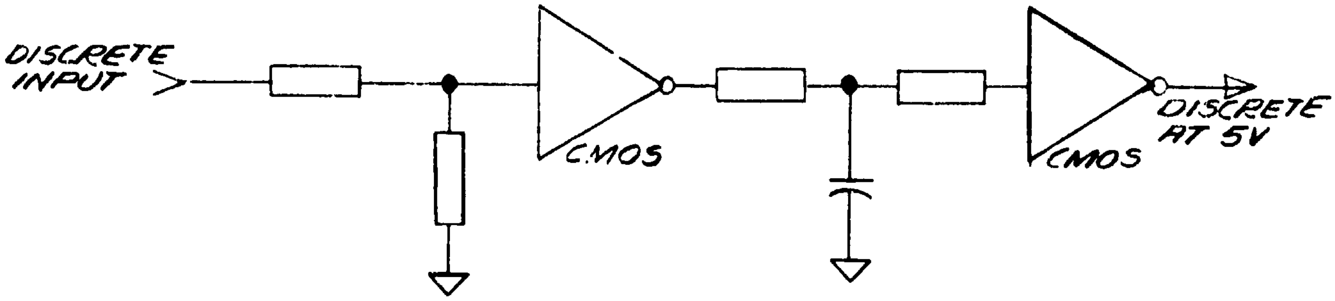
Figure V-6c shows a typical balanced line receiver for low level discretes.



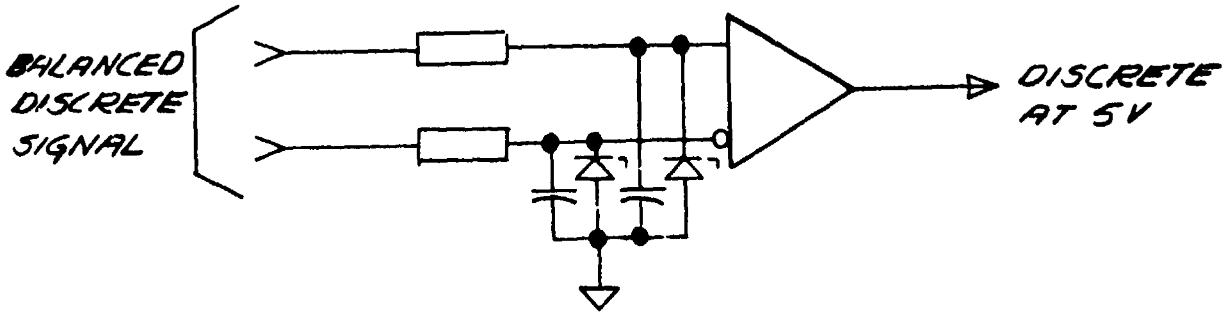
OUTPUT (D/A) CONVERTER
FIGURE V-5



Discrete Signal Output Driver
V-6a



Discrete Signal Receiver (With Noise Receiver)
V-6b



Discrete Signal Receiver (For Low Level, High Common Mode Noise Signals)
V-6c

FIGURE V-6

c. Parallel and Serial Intercomputer Data Links

Digital Data Links

Various information transfers are required between digital processors comprising the FCS and between these processors and other aircraft systems.

Data links to other aircraft systems are frequently outside the direct control of the FCS designers, being dictated by the airframe or avionics contractor. Where control is possible, it is recommended that these data links conform to the same standards as recommended in the following section describing intra-FCS digital data transmission.

Inter-Computer Data Links

In considering a system of interconnected digital computers such as might be required for a FBW PFCS, data transfer between computers offers an area of design where the function and economies of the system can be enhanced or, alternatively, seriously degraded. The following presents some of the important design considerations necessary to select a data link scheme.

Types

Binary data may be handled word-parallel or word-serial and, for each of these, bit parallel or bit serial. Word parallel effectively means separate wiring for each parameter to be transmitted and would be indicated only for extremely high data rates (perhaps 10^5 updates of the parameter per second) or very special conditions of isolation or security.

Given, then, that the parameters will be transmitted word-serial (that is to say, the parameters will be transmitted in some sequence over one channel), it becomes necessary to select bit-parallel or bit-serial (or, more conveniently stated, parallel or serial) transmission. A third possibility is called byte-serial, where portions of words are transmitted sequentially (serially) with each portion being in parallel format. This scheme will not be discussed further here because, in the context of straight binary intercomputer communication, it seems to come closer to combining disadvantages than to taking the best of each.

Considerations

The primary consideration in selecting a data link scheme is the data rate requirement, in terms of total parameters (parameters times updates per second) transmitted per second. For a parallel system, this is also the bit rate on each of the wires. The bit rate of a serial system is the data rate times the bits per word. The bit rate is important in that it acts as a constraint on the types of circuits and interbox wiring which may be used.

Hand-in-hand with the data rate is the size of the data word. As a realistic working parameter, a word of 28 bits is assumed, consisting of one marker bit, 7 address bits, 15 data bits, 3 spares and one parity bit. For the serial scheme four inter-word blank bit spaces are allowed for a total of 32 bit times.* A small change in the number of bits required should not seriously change any of the conclusions.

For the FBW PFCS application the following requirements for a single serial data link are estimated as sufficient to insure adequate capability:

Maximum Sampling Rate	100/sec
Data word length	16 bits (15 bits + sign bit)
Address word length	7 bits
Parity	1 bit
Marker	1 bit
Spares	3 bits
Blanks	4 bits
Word transfer rate	10^4/sec
Clock Rate	320 K Hz

*Transmission is assumed to be autonomous (as opposed to command/response, for example).

In selecting a data link scheme, hardware considerations come into play. The buffer registers in either serial or parallel schemes need not be too different in terms of size and cost. In terms of differences, serial schemes generally require more timing and control circuitry and probably require more sophisticated encoding and line driver and receiver equipment. On the other hand, the parallel scheme requires 16* line circuits to the serial scheme's one, although the parallel line circuits may each be somewhat simpler. A parallel link is estimated to require between two and five times the hardware of an equivalent serial link. Ship's wiring between computers is similarly impacted: Serial links requiring one somewhat more sophisticated wiring path compared to the requirement of 16 paths for parallel operation. This might trade off as one shielded twisted pair versus one 16-pair bundle. Obviously, this factor is strongly effectea by the computer locations and the resulting path lengths.

Vulnerability of the data link to noise pickup is a matter of concern, but is more a function of the electrical scheme used than of the serial/parallel selection. The serial scheme uses a somewhat wider bandwidth channel meaning somewhat greater noise sensitivity. On the other hand the serial channel is likely to be somewhat more sophisticated due to it being a single channel-every increase in complexity is multiplied by one rather than 16.

It is not contemplated that any sort of error correcting code by used in this application, due to the added channel requirements and circuitry. A simple parity bit is recommended to pick up simple hardware failures, broken wires, and similar defects.

A more serious consideration included in the term 'vulnerability' is vulnerability to failure propagation from line to computer or computer to computer. Any reasonable failure or sequence of failures associated with the line driver, line receiver, or the line itself should not cause dysfunction of other than that particular data link.

Routing

Various possibilities exist in providing data communications between computers.

* Additional lines may be required for coding and addressing.

Several factors enter into the selection of the routing, one of the more important (after providing sufficient line data handling capacity) being the consequences of a line (including driver and receiver) failure. This should not cause a loss of communication with other channels in order that failure isolation can take place without causing the dropping of a good computer due to one line failure. Similarly, a failure in one unit should not be able to disrupt communications between two other units.

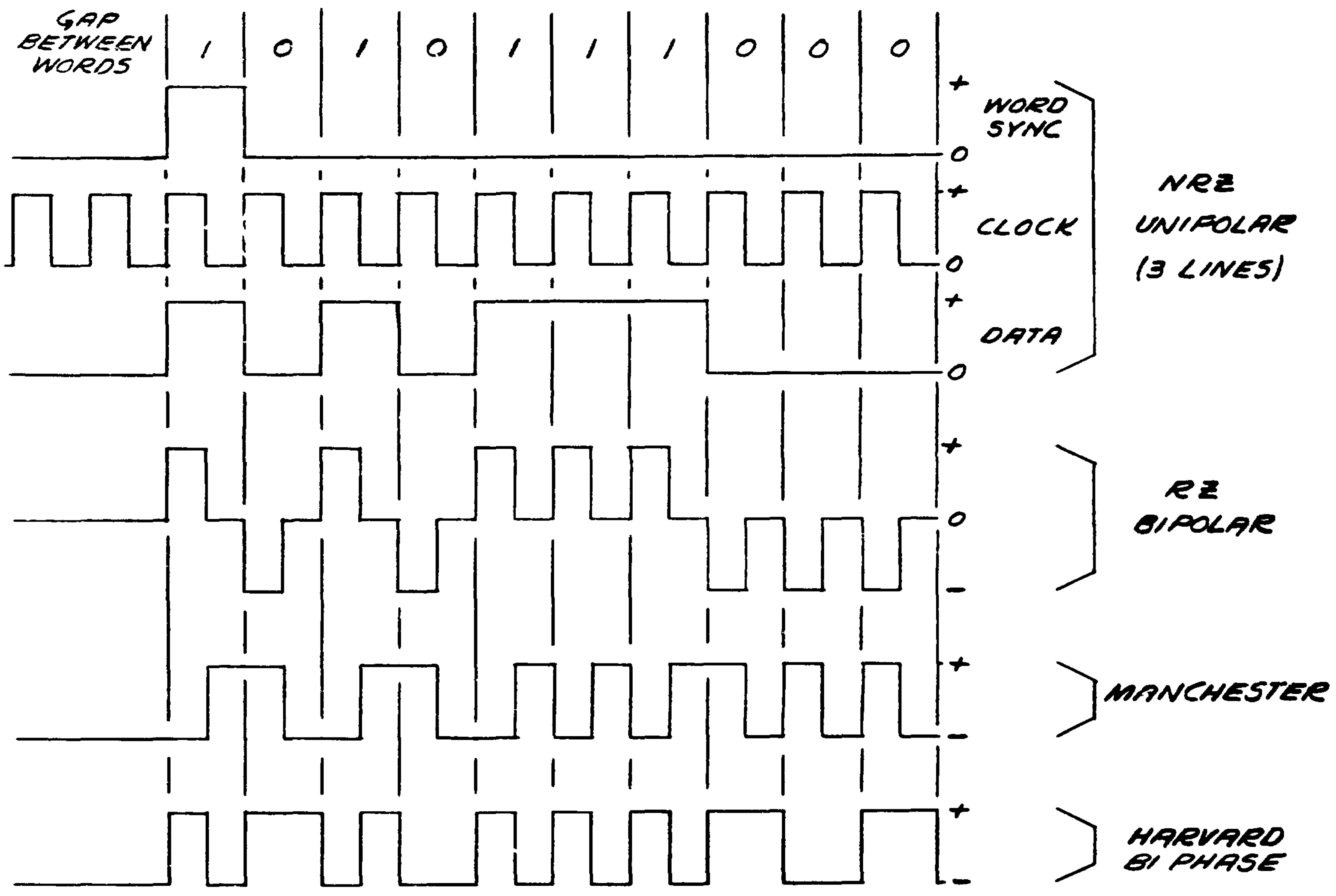
A two way channel, i.e., A to B and B to A on one set of wires, obviously saves intercomputer wiring. The line drivers and receivers are almost the same in either case, except that channel occupancy must be detected. The real difference lies in the requirement that only one computer can transmit at a time meaning that the two computers must be different in order that one is 'first'and the other 'second' within some time frame. This complicates the software required to execute a data transfer considerably and may even result in a program lock. The implications of two way links in schemes which require fully synchronous computers remain to be determined.

Encoding

Numerous codes have been used for transmitting digital data. Four of these are shown in Figure V-7.

The NRZ (Non Return to Zero) code is shown as a three line code and illustrates the three elements which must be transferred. Some type of word sync is necessary to identify the start of a word. This is shown as a pulse occurring during the first bit of the word. Similarly a clock pulse train is required to identify a bit interval. Finally, the data must be transmitted. In this simple system three separate paths are used resulting in very simple encoding/decoding circuits at the cost of extra line circuits. This simple system is specified for ARINC 561 Inertial Navigation Systems. Other versions of this system rely on clock and word synchronization between transmitter and receiver, transmitting only data. In parallel systems, word synch is not needed as a whole word is sent each bit interval while only one clock needs to be transmitted for all the bits of a word.

A completely self-clocking code is the RZ (Return to Zero) Bipolar code specified for ARINC 575 Digital Air Data Systems. In this code, a 'one' is represented by a positive voltage pulse while a zero is a negative pulse. The voltage returns to zero between pulses, while the gap between words is a zero voltage for several bit periods. This code is easily decoded as the clock may be derived by simply full-wave rectifying



SIGNALING CODES
FIGURE V-7

the input with virtually no timing problems beyond detecting the word gap. Probably the greatest disadvantage is the three voltage level structure requiring a somewhat more complex driver than would otherwise be required. Line receiver requirements are also effectively doubled.

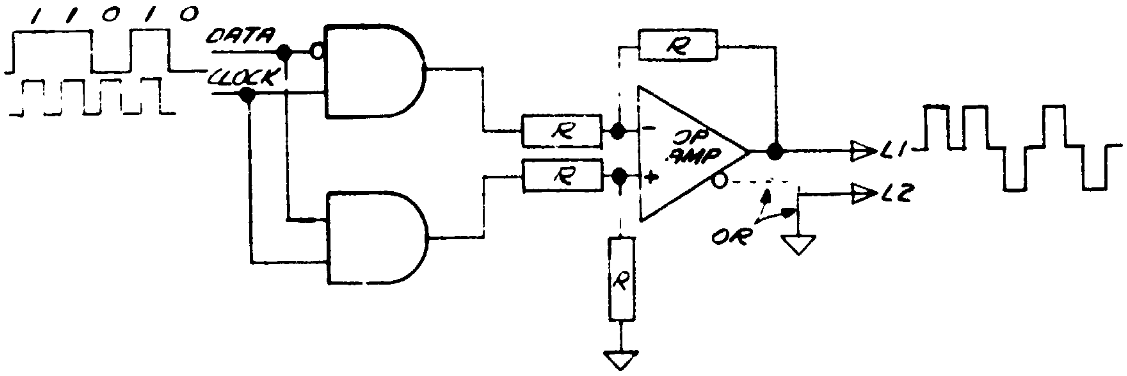
Two somewhat similar codes are the 'Manchester' code and the 'Harvard BiPhase' code. (The 'Harvard BiPhase' is specified in ARINC 573 for Airborne Integrated Data Systems). Both are two voltage level self-clocking codes which carry the information in the voltage transitions rather than in the levels per se. The Manchester code may be looked at as a periodic wave, the period of which represents one bit time. A 'zero' is encoded as an in-phase (with some reference) voltage, while a 'one' is a 180° out-of-phase (inverted) voltage. Applied to a square wave this produces the waveform shown in Figure V-7. Note two things. First, the information may be detected as the direction of the mid-period transition or as the voltage level during either the first half or the second half of the cycle (as long as we are consistent). And, second, since the input (square wave) clock has no D.C. component, the encoded waveform has no D.C. component regardless of the encoded information.

If the Manchester code is similar to the RZ bipolar code in that the '1's and '0's are encoded as opposite polarities of voltage transitions in one case and levels in the other, then the Harvard BiPhase code is similar to the NRZ Unipolar example given in that a '1' is encoded as the presence of a mid-period transition or level (respectively) while a '0' is represented as the absence of this feature. In the Harvard code a transition is added at the bit period edges to allow self-clocking. The Harvard code, like the Manchester, has no D.C. component although the Harvard code has a somewhat lower frequency component.

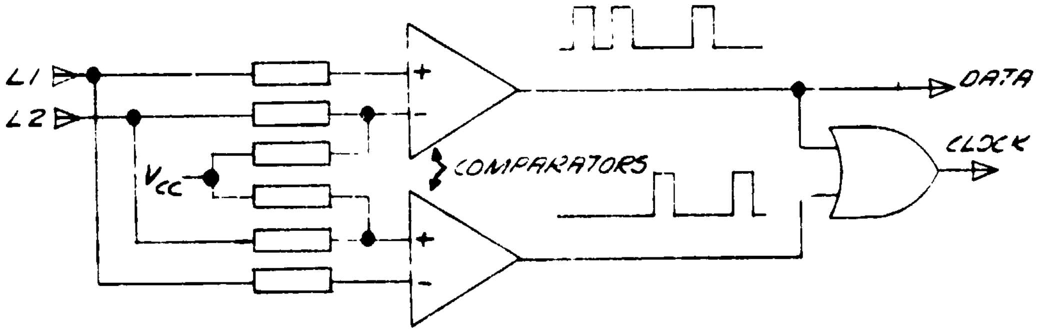
There are many methods of encoding signals which are in use today. The codes shown are typical codes which have one or more features of interest for the intercomputer data link use.

Implementation

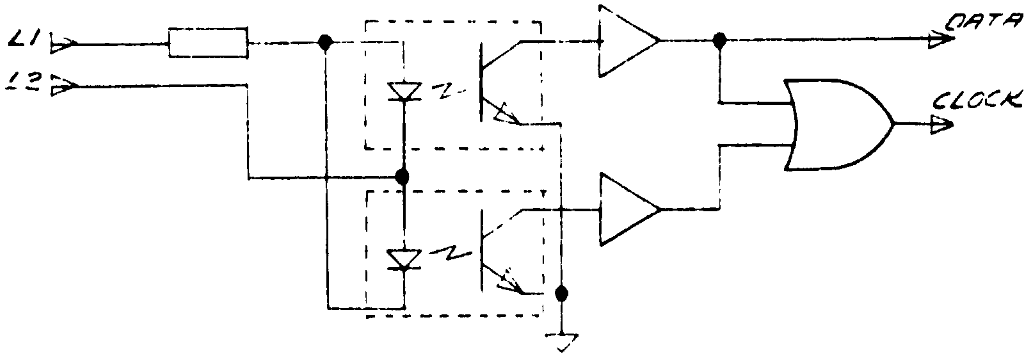
Typical implementation of an RZ Bipolar encoder and decoder is shown in Figure V-8. The encoder uses an operational amplifier to generate the bipolar output from '1' and '0' pulses provided on the inverting and noninverting inputs. Transmission of signal and transmitter ground is by shielded twisted pair to a dual differential line receiver to provide a high degree of noise immunity.



Bipolar RZ Encoder/Transmitter
V-8a



Bipolar RZ Line Receiver
V-8b



Receiver Using Optically-Coupled Isolators
V-8c

FIGURE V-8

A shortcoming of the circuits shown in their behavior under 'hot short' or other line transient conditions. The receiver probably can be made safe against propagating the fault voltage into V_{CC} or signal circuits by the use of discrete resistors and clamping diodes. An alternative is the use of a pair of optically coupled isolators on the inputs as shown in Figure V-8. Protecting the line driver is more difficult because of the low output impedance required. Microcircuit line drivers suffer from the presence of a real possibility that the chip ground lead will open first, subjecting signal and V_{CC} pins to the line transient. Suitable discrete circuits can be designed, but, of course, they increase the number of piece parts, failure rate, and cost. Note that transformer coupling cannot be used due to the D.C. component necessary in this signaling form.

A suitable encoder and decoder for Manchester code is shown in Figure V-9. A shielded twisted pair is again used as the transmission line. The amplifiers shown as the line transmitter and line receiver may be any of a number of standard opamps, line receivers and transmitters, or even (for the transmitter) standard logic elements. Isolation using transformers is an entirely feasible way of limiting fault energy transfer to a level safely handled by the line circuits.

The use of square wave signaling with stable, accurate clocks allows the use of simple single-shot timing as the receiver reclock generator (shown) and the gap detector.

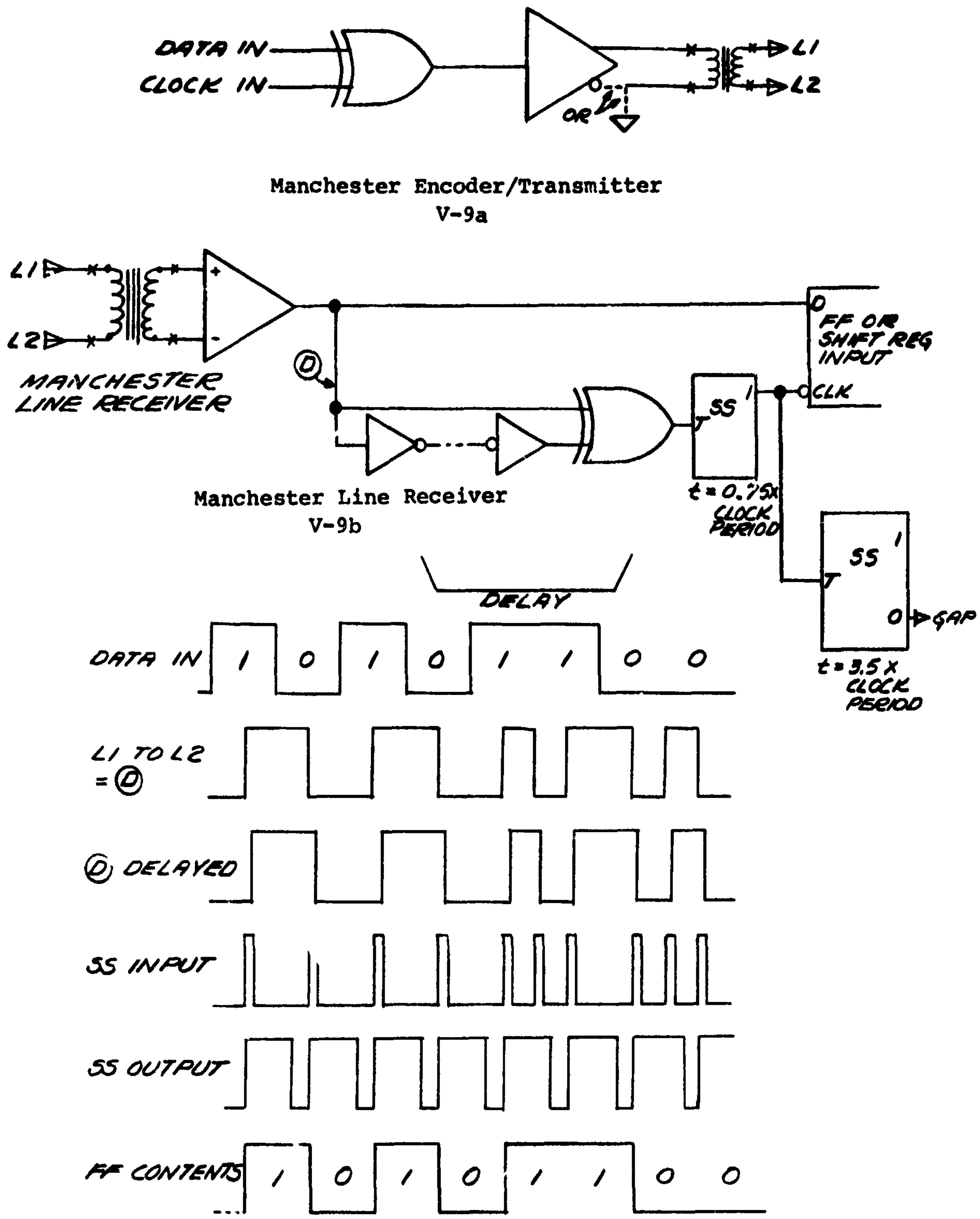
A somewhat similar system can be used for decoding the Harvard BiPhase, except that the edge detector would lock on the bit period edge transitions with the single-shot output providing a gating pulse for the bit period center transitions which signify '1'x.

d. Recommendations

Unless there is a drastic change in data link requirements, it is recommended that data be transferred between computers of the FBW PFCS by means of data links having the following characteristics:

(1) One way transmission on each path to simplify computer timing, allowing two-way simultaneous transfers.

(2) Serial data transmission with a relatively low bit clock in order of 500 kHz, simplifies hardware while allowing convenient use of techniques with desirable properties.



Manchester Encoder/Transmitter
V-9a

Manchester Line Receiver
V-9b

Manchester Code Encode/Decode Scheme
V-9c

MANCHESTER CODE
ENCODE/DECODE SCHEME
FIGURE V-9

(3) Manchester coding on the data link using a square wave 'carrier' because of the self-clocking and no D.C. component features of this code. Relatively easy to encode and decode where clock frequency is accurate and stable.

(4) Independent intercomputer data paths to the extent necessary to achieve isolation. The simplicity of the data link makes this independence practical.

(5) Transmission paths using shielded twisted pair conductors for superior EMI characteristics, both for susceptibility and emission.

(6) Differential input line receiver to minimize common mode noise pickup.

(7) Transformer isolation at transmitting end of link with either transformer or optically coupled isolators at the line receiver. This is necessary to prevent faults from propagating beyond the hardware associated with individual line circuits.

APPENDIX VI

SIGNAL SELECTION, MONITORING AND EQUALIZATION

This Appendix contains discussions of the following aspects of redundant flight control system design:

.... Signal selection devices

.... Equalization techniques

Conventional types of signal selection devices are enumerated and their relative merits and details of operation are discussed. Criteria for selection of a specific device are examined. The necessary distinctions are drawn between signal selection and monitoring. The effects of signal selection devices on performance and operation are discussed, and the effect of various types of failures is indicated for all the configurations. Application to both analog and digital flight control systems are made.

The use of equalization techniques for minimizing bias or drift errors between channels of a redundant flight control system is studied next. Mathematical criteria are derived for the stability of equalization loops, and the relative advantages of different equalization schemes are pointed out.

Signal Selection Processes

This section presents a summary of the operational objectives of signal selection,monitoring and self test in the context of redundancy management. In most flight control systems signal selection and failure detection are combined in a single device. This association is the result of a desire to minimize hardware by sharing components and does not necessarily reflect an inherent inseparability of the two processes. It will be shown, subsequently, that the objectives of signal selection do not always coincide with those of failure monitoring. As a consequence, it can be expected that both processes are less than optimal when combined in the same device.

In digital control systems the computational flexibility of the computer can accommodate a variety of approaches to the signal selection and failure detection processes without a corresponding increase in quantity and complexity of hardware. As a consequence, it is both practicable and desirable to treat signal selection and failure detection as distinct processes.

Definition of Siqnal Selection Device (SSD)

A signal selector is a device (or program or algorithm) which yields an output as a function of two or more inputs. In this general context a summing amplifier or a complementary filter are signal selection devices. The device, together with the inputs and outputs make up the signal selector process. In the context of redundancy the inputs to an SSD are, in some sense, replicates of an ideal signal. The output, however, is only required to be a "usable" replicate of the ideal input. It is not necessarily "better" than any of the inputs at least in the conventional sense of being more accurate, less noisy, more representative, etc.

Definition of Failure Detection Device

A failure detector is any device (or program or algorithm) capable of detecting and annunciating failures either of components or signals. Failure detection capability is measured in terms of the parameters α and β which were defined in Appendix III. The quantity β is a measure of the sensitivity to nuisance alarms and is defined by the conditional probability

$$\beta = P(\overline{F}|A).$$

The quantity α is a measure of the failure detection capability and is defined by

$$\alpha = P(\overline{A}|F) \text{ where}$$

F = event of a failure and A = event of an alarm.

1. Operational Objectives of Signal Selection

Improved Reliability through Cross Strapping

The SSD can improve system reliability by providing cross strapping (i.e., alternate path routing) of input signals. The SSD can in principle, provide this function without monitoring the input signals; i.e., without removing a failed signal. Every SSD is essentially a majority device if the output is independent of the failure status of the input signals.

Reduce Effects of Failures and Failure Transients

The SSD, acting as a majority device, can mask the long term effects of failures and reduce failure transients to tolerable levels. This is the primary purpose of mechanical signal selection in the secondary actuators.

Provide Common Output in All Redundant Channels

A SSD can provide a common output in all redundant channels. The effects of common outputs are:

- Improved Failure Detection in Downstream Units

  When used in conjunction with comparison - type monitoring, common outputs improve failure detection by reducing nuisance alarms in downstream units. Common outputs are especially desirable in the detection of null and slowover failures.

- Common outputs eliminate the threshold characteristics of downstream mid-value selection devices.

- Common outputs can be used to equalize redundant channels

Tolerance Reduction

A SSD can be used to obtain a "good" signal reference for (a) improved signal accuracy or as a (b) reference signal for monitoring.

2. Operational Objectives of Monitoring

In practice there are two broad categories of failure detection processes:

- Self Test

- Comparison Testing

The intended meaning of "self test" is that of a digital computer software program or any detection process which operates independently of the other channels. "Comparison testing" refers to any detection process which employs the other channels as models.

Two basic approaches to comparison testing are:

- Cross-SSD which uses the output of an SSD as the good reference signal against which all other channels are compared.

- Cross-Channel which compares each channel with the other channels and operates independently of any SSD's in the signal chain.

Typical cross-SSD and cross-channel monitors are shown in Figure VI-1, as they might appear in channel 1 of a quadruplex configuration. A tradeoff of cross-SSD versus cross-channel comparison monitoring is given in Reference 5.

Pre-Flight Failure Detection

In order to maintain the operational capability of a redundant system,failures must be detected and replaced. Reliability goals may impose severe requirements on pre-flight test efficiency.

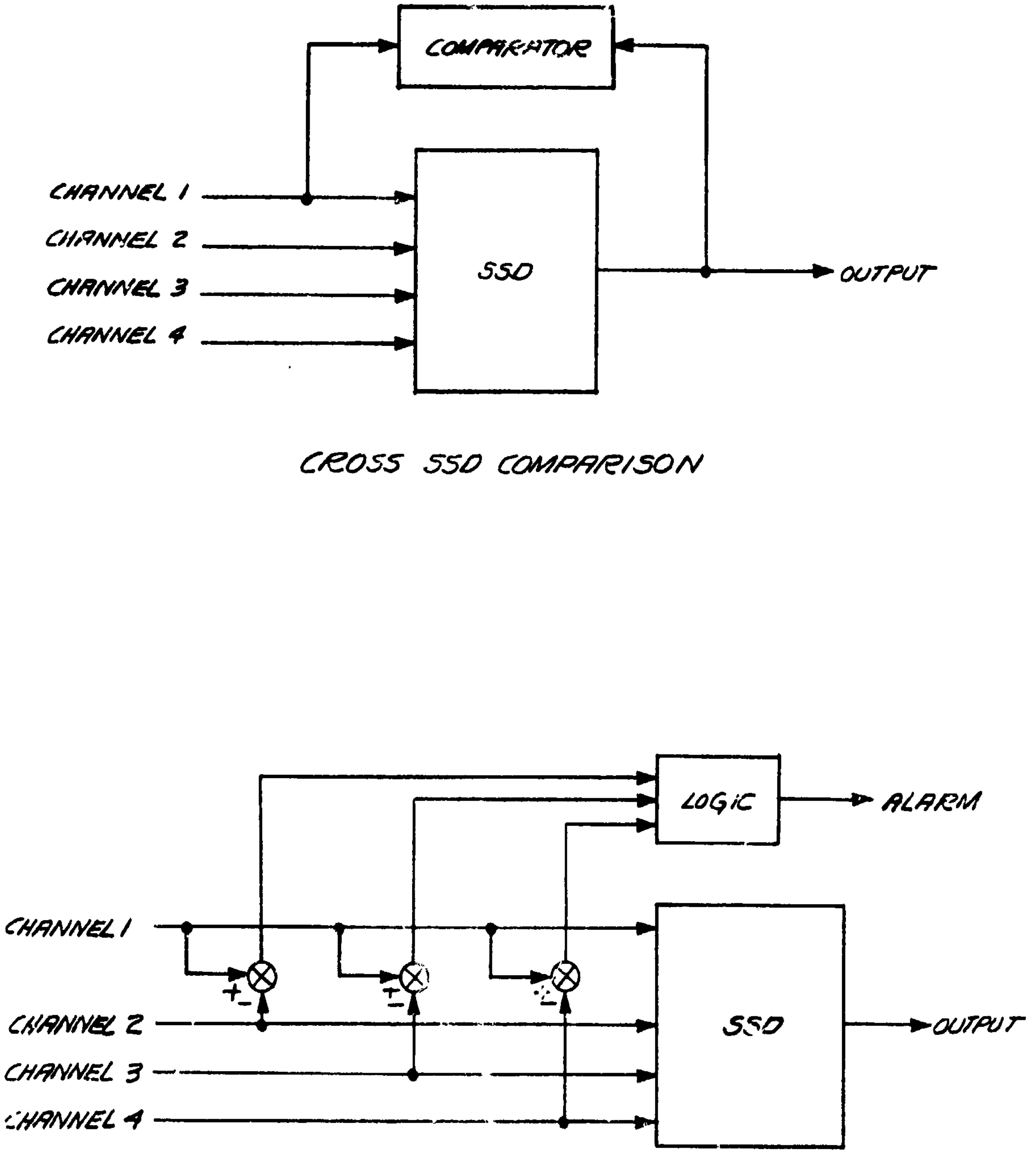
In-Flight Failure Detection

Improved Cross Strapping Benefits of the SSD

By detecting and removing a failed signal failure detection can improve the benefits of an SSD relative to its cross strapping function. As an example, without failure detection the output of an SSD with four inputs will fail after two inputs have failed. With failure detection the output could conceivably only fail after four of the inputs have failed. There is an obvious trade-off here between the probability of two failures versus the probability of those combinations of detected and undetected failures and nuisance alarms which will cause disengagement of the device.

Reduced Failure Effects and Transients

An undetected failure of an input signal to an SSD can have an undesirable effect on the output particularly in the region of small amplitudes. Oscillatory failures are especially undesirable because they can induce sustained oscillations of the output, albeit of small amplitude. In an SSD in which averaging is performed over a limited region a sustained failure will result in a reduction in gain within the region. Failure detection and disengagement can be a mixed blessing. It will be shown, subsequently, that disengagement may frequently cause a transient which is more severe than the transient which resulted at the onset of the failure.



CROSS SSD COMPARISON

CROSS CHANNEL COMPARISON

COMPARISON MONITORING TECHNIQUES

FIGURE VI-1

Supplements Pre-Flight Test

In-flight failure detection may be considered an extension of pre-flight test. By running continuously in the actual aircraft environment, in-flight failure detection can be very effective in detecting failures which, in a ground environment, would be difficult to detect.

Failure Status Annunciation

Failure annunciation is necessary if the pilot is to decide between continuing or aborting the mission.

3. Examples and Application of Signal Selection Devices

In subsequent sections we will concentrate our attention on three conventional signal selection processes:

Limited Averaging (LA SSD)

In this process the output is the average of the inputs until an input varies from the average by a predetermined distance. When this occurs the input is voted out. A typical output response is shown in Figure VI-2 where d = maximum variation from the average.

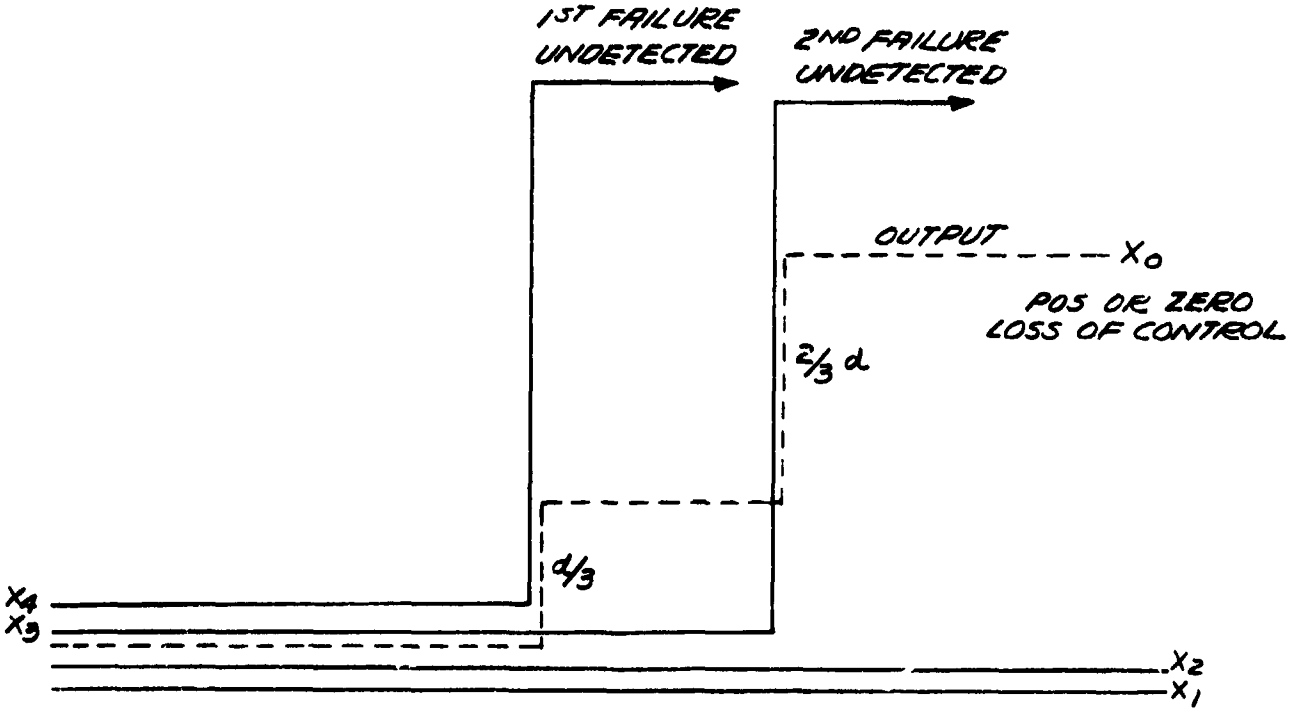
Mid-Value Selection (MV SSD)

If the number of inputs is odd then the output is the mid-value. If the number of inputs is even then the output is the mid-value of the inputs and zero.

Mux Gate

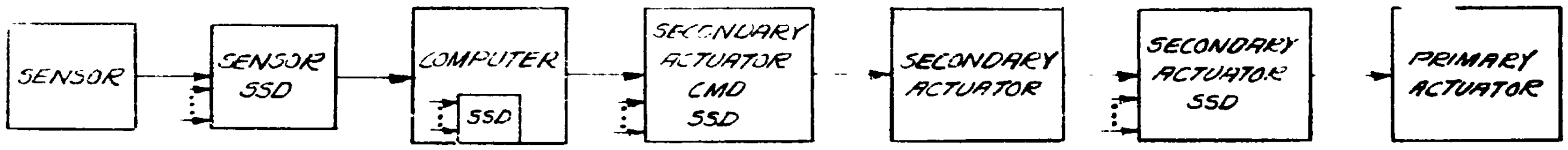
In this process one of the inputs is gated to the output. If the same input is gated to the output in all channels the process is called, "consolidation". Gating logic is activated by the failure monitor. The gating strategy depends upon the application.

Figure VI-3 shows typical placements of SSD's in the flight control application. The objectives of each SSD depend upon its location as the following summary indicates.



QUADRUPLEX LIMITED AVERAGING SSD

FIGURE VI-2



PLACEMENT OF SIGNAL SELECTION DEVICES
IN THE FLIGHT CONTROL SYSTEM

FIGURE VI-3

Objectives

Sensor SSD

- Improved reliability through cross strapping the sensors and computers. The improvement in reliability can be considerable even without the improved benefits of monitoring.

- Improved failure detection in downstream units due to common outputs in all channels. This presupposes comparison-type monitoring.

- Improved dynamic performance of downstream SSD's due to common outputs.

- Equalization of redundant channels via common outputs in all channels.

- Sensor signal selection can be performed by the digital controllers. Each computer inputs one sensor of a set and transmits the converted value to the other computers via the intercomputer data links. With this arrangement loss of a computer results in loss of its associated sensor.

Digital Computer SSD

- Equalization of Integrators - This SSD, which operates on signals which are generated in the digital computer, is used for integrator equalization in the event that the sensor SSD's do not supply a common signal in all channels. The cross strapping of the sensors could be such that small differences between channels develop due to I/O and A/D converter tolerances and biases. While such differences may have a negligible effect on the dynamics characteristics of downstream SSD's, they will, eventually, cause the channel integrators to diverge.

Actuator Command SSD

- Improved reliability through cross strapping the computers and servos. This objective requires dedicated SSD's; i.e., independent of the major failure modes of the digital computers.

- Improved failure detection in servos due to common servo commands.

- Improved dynamic performance of actuator SSD due to common commands. The dynamic performance problem, however, may be merely transferred to upstream SSD's.

Actuator SSD

- Reduced failure effects and failure transients - By appropriate selection the effects of command and servo failures are reduced. As an additional consequence, failures need not be detected immediately as they occur.

Disadvantages of Signal Selection

- Introduces undesirable dynamic characteristics particularly in the region of small amplitudes.
- Susceptible to common mode failures - As an example, a Mux Gate of the consolidated type will pass a failed signal to all channels until it is detected and gated out.
- Masks failures as seen by downstream units - A poorly designed SSD could pass a failure to all channels. An SSD of this type requires a highly efficient (and demonstrably so) in-flight test.
- Requires interchannel isolation - Because signals from all channels feed a single SSD, a common failure, even if detected, could propagate to all channels.
- Could result in disengagement of all channels due to a single failure. If the SSD is used to supply a reference signal to a monitor then a single failure could result in disengagement of all channels. Such a phenomenon has actually been observed (in Reference 5, Monitoring Avalanche, Page 145).
- Could increase tolerance build-up - A poorly designed SSD (i.e., relative to the signal noise) could cause the selected output to have a worse tolerance variation than any of the inputs.

4. Operational Characteristics of the SSD

In the context of the general definition of the signal selection process it is difficult, if not impossible, to generalize operating characteristics of an SSD. We do know, based on our experience with conventional SSD's, that two characteristics are paramount in the evaluation of a signal selection process:

- Normal dynamical performance including threshold and tolerance propagation effects
- Failure effects and transients

Normal Performance

An SSD output, which is presumed to simulate an ideal input, may exhibit dynamical characteristics which are not shared by any of the input signals.

Threshold

An MV SSD with an even number of inputs will exhibit a threshold due to bias differences between the inputs. An example of this effect is shown in Figure IV-6. In many applications, the threshold will result in a limit cycle oscillation. The threshold effect can be reduced or eliminated by introducing equalization (to be discussed subsequently) or by insuring that all signal inputs are the same. In a digital controller the introduction of a sensor SSD can insure common inputs to all downstream SSD's. However, this could transfer the threshold problem to the sensor SSD The solution here is to use an SSD which does not exhibit the threshold problem, e.g., a limited average SSD.

Tolerance Propagation

Under nominal conditions the inputs to an SSD will differ from each other and from their mean due to normal tolerance differences. As a consequence the output will differ from its mean by an amount determined by the process algorithm and the input differences. This variation of the output could have a degrading effect on performance on a single string basis and on monitoring if the SSD output is used as a reference signal to be compared with the inputs. An excessive increase in tolerance propagation will result in poor failure detection and a high rate of nuisance alarms. As an example of propagated tolerance, let us compare an MV SSD with a limited average SSD.

For simplicity, we assume three inputs u(t), v(t), w(t), whose variations about a common mean, m(t), are independent, random and stationary processes with the same statistics. Because it yields a simple solution we also assume that the processes are Gaussian.

Let

$$\sigma^2(t) = E\left((u-m)^2\right) = E\left((v-m)^2\right) = E\left((w-m)^2\right)$$

= variance of each input

z(t) = output of the SSD

E( ) = expected value.

It is shown in the Supplement to this Appendix that in an

MV SSD

E(z) = m and

$$E\left((z-m)^2\right) = \left(1 - \frac{\sqrt{3}}{\pi}\right) \sigma^2 \approx .45\,\sigma^2$$

and in a

Limited Average SSD

E(z) = m and

$$E\left((z-m)^2\right) = \frac{1}{3}\sigma^2 \approx .33\,\sigma^2.$$

Thus, we see that both the MV and LA SSD's outputs result in smaller tolerance variations than any of the input signais. Furthermore, the LA SSD is somewhat better in this respect than the MV device. However, the advantage of one or the other process depends upon the nature of the statistics of the inputs. In general, distributions which are concentrated about the mean tend to favor limited averaging whereas symmetrical distribution which are concentrated at points other than the mean tend to favor mid value selection.

Failure Effects and Transients

The steady-state and transient effects of failures depend upon aircraft dynamics, mode of operation, type of signal selection devices, etc. A detailed discussion of the effects of failures in triplex and quadruplex configurations when the signal selection device is a mid-value type has been presented in Appendix IV. A summary of failure transient effects of mv and limited averaging signal selection devices is given in Table IV-1. The table was taken from Reference 5.

TABLE IV-1

PERFORMANCE COMPARISON OF MIDVALUE VS LIMITED AVERAGING SIGNAL SELECTION PROCESSES

Failure type	Failure effects	
	MVL with perfect equalization	Limited averaging
Step	No failure transient–easily detected.	Transient equal to 1/4, 1/3, or 1/2 of averaging range–easy to detect.
Slowover	No transient–signal detected when drift exceeds detection level.	Output drifts slowly to 1/4, 1/3, or 1/2 of averaging range–detected when drift exceeds detection level.
Oscillatory	No oscillation with perfect hard vote–detectable if oscillation level is greater than detection level and if time delays on detection do not block detection.	Oscillation equal to 1/4, 1/3, or 1/2 of averaging range.
Passive	No gain change–very hard to detect without special detection provisions unless circuit activity is greater than failure detection level.	Gain about zero drops to 3/4, 2/3, or 1/2 of normal value–very hard to detect without special detection provisions unless circuit activity is greater than failure detection level.
High gain	Size of limit cycle will depend on preciseness of vote. A perfect hard vote will eliminate limit cycle. Detection may or may not be easy depending on where failure occurs.	A limit cycle is guaranteed and size or severity will depend on the averaging range–failure condition will be very evident, but determination of which channel has failed may or may not be easy

The initial configuration is quadruplex and it is assumed that (a) all previous failures were detected and removed, (b) failure detection is achieved through comparison-type monitoring and (c) channel differences have been equalized.

5. Summary of Signal Selection Processes

MV SSD

Provides cross-strapping.

May not require rapid recognition of first failure.

Failure transients determined by normal channel differences at time of failure.

Potential limit cycle oscillation around null causes by normal channel differences.

Improves tolerance propagation effects.

LA SSD

Provides cross-strapping.

Requires rapid recognition and disqualification of failures in order to reduce failure transient.

Failure transients could equal 1/4 (quad), 1/3 (triplex) or 1/2 (dual) of the averaging range.

No limit cycle due to channel differences.

Improves tolerance propagation effects. Somewhat superior in this respect than the MV SSD.

MUX GATE SSD

Provides cross-strapping.

Requires rapid recognition and disqualification of failure in order to reduce failure transients.

Failure transient could equal maximum signal level.

A latent failure followed by a detected failure could result in the failed channel supplying two channels signals.

Recommended for non-critical functions such as computers-to-displays, especially with serial transmission.

6. Common Mode Failures

One of the potential disadvantages of signal selection is the susceptibility of the resultant system to common mode failures. In a digital computer there are several sources of common mode failures associated with the signal selection process:

Signal Select Algorithm

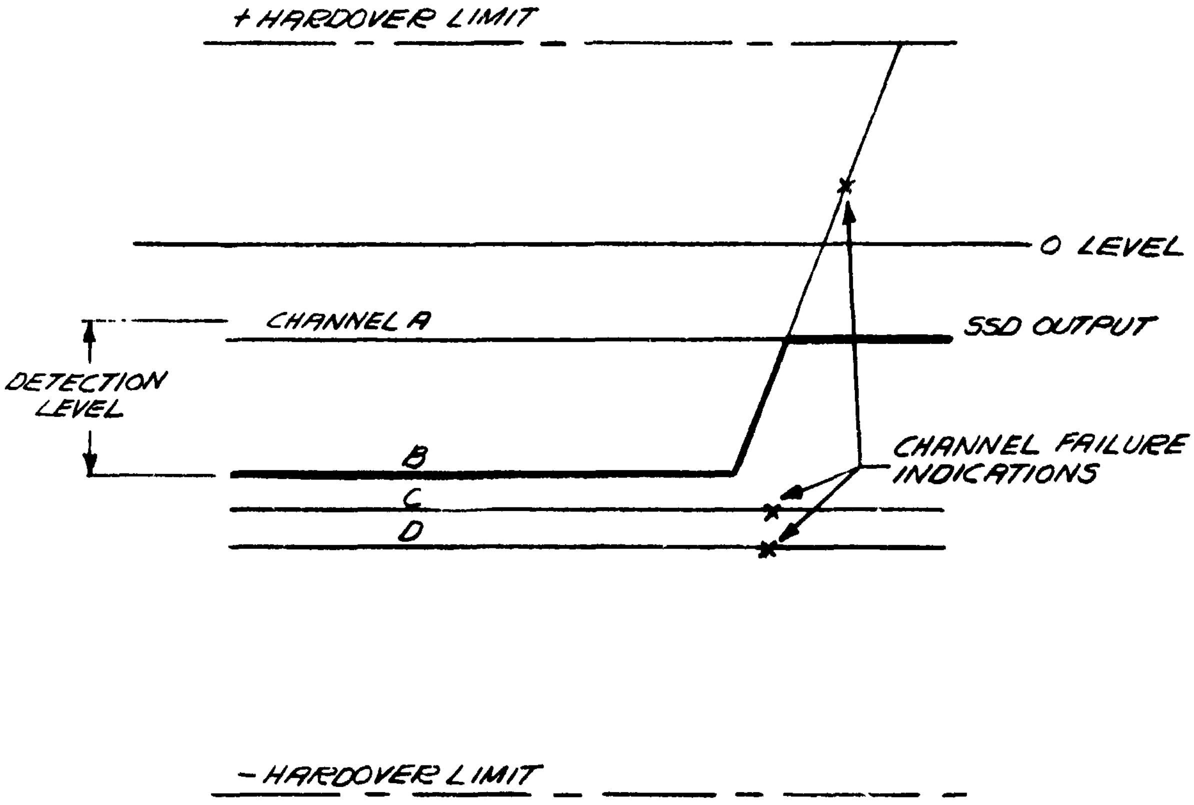
In a digital controller the complexity of a signal select algorithm may be no impediment to its implementation. However, an excessively complicated algorithm may contain a design defect which, under normal operating conditions, could propagate a common, but false, signal to all channels.

Communications Path Failure

Because signals from all channels must be transmitted to all computers a single failure of a communications path could affect all computers. Referring to Figure V of Appendix V, it can be seen that all communications paths are eventually gated onto the I/O or DMA busses from which they can be gated to almost any component in the computer. The remedy is to isolate all external paths from these busses.

Cross-SSD Monitoring

When an SSD is used to provide a reference for cross-SSD comparison monitoring it is possible to disengage all channels due to a single failure. An example of such an effect is given in Reference 5 where it is referred to as "monitoring avalanche". The sequence of events are illustrated in Figure VI-4, which is taken from Reference 5. Referring to the figure, channel A incurs excessive drift but remains within the detection level. Channel B subsequently fails in the direction indicated with the result that channels D, C and A successively indicate failures.



MONITORING AVALANCHE IN A OUADRUPLEX MV SSD

FIGURE VI-4

7. Equalization

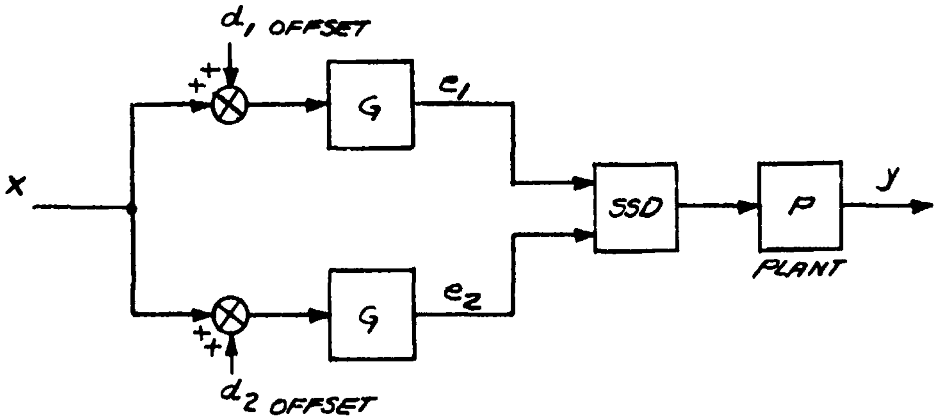
Effects of Bias Errors in Redundant Flight Control Systems

In a redundant control system and under normal operating conditions, certain internal variables may diverge while other variables are under control. This phenomena can occur even though the single string control system is stable and otherwise satisfactory. The reason is that each channel contains errors which are not the same in all channels but senses the same motion and commands the same controller as all of the other channels. These small errors excite modes which are not present in the single string system and which are uncontrollable by the single control surface. As an illustration of this effect consider the dual redundant system of Figure VI-5. From the figure it can be seen that there does not exist a control function, x(t), which can drive the state variables e_1, e_2 from an arbitrary initial condition (e.g., $e_1(0) \neq e_2(0)$) to the origin in a finite amount of time even if the offsets are identically zero. While strict controllability of e_1 and e_2 is not a requirement in most control system it is required that e_1 and e_2 remain bounded and their difference sufficiently small. This is not the case when the transfer function, G, is unstable or neutrally stable. It is always the case when G is stable. The situation can be seen more clearly in Figure VI-6. Here the single string system, $xG=e$, is controllable (i.e., in most cases of interest) but the variables e_1 and e_2 are not strictly controllable. If G is unstable or neutrally stable then e_1 or e_2 will become unbounded for arbitrary offsets. Since most flight controllers contain at least one integration (e.g., for trim, heading or attitude hold, beam error, etc.) it is necessary to devise some technique of control which will maintain internal variables within prescribed bounds when the control system is made redundant. There are three obvious techniques to accomplish this:

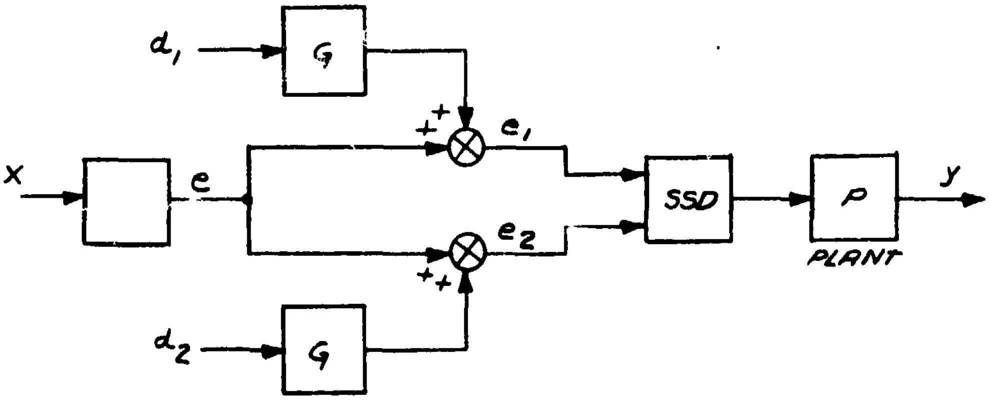
- Approximate the integrator by a lag.
- Transmit identical control signals to all channels.
- Equalize channels or integrators.

Lag Replacement

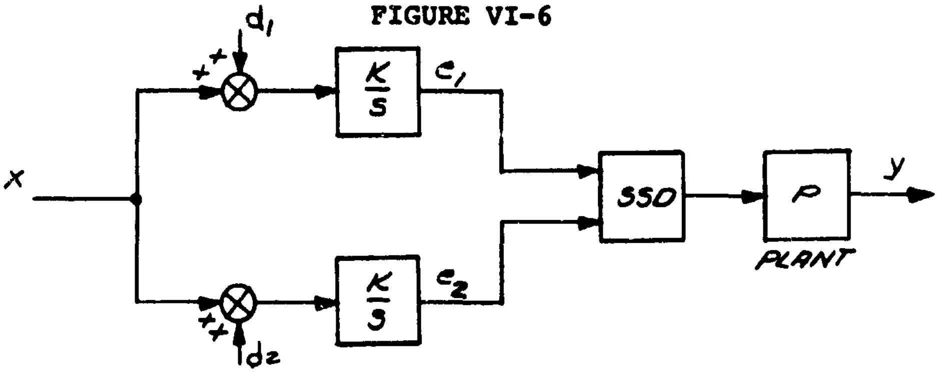
When the lag can be selected to yield satisfactory performance, this is the easiest solution.



DUAL REDUNDANT CONTROL SYSTEM
FIGURE VI-5



EQUIVALENT DUAL REDUNDANT CONTROL SYSTEM
FIGURE VI-6



DUAL REDUNDANT CONTROL SYSTEM EXHIBITING INTEGRATORS

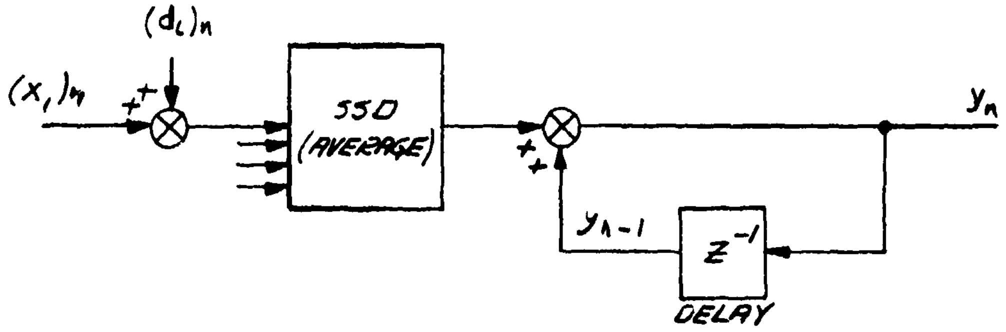
FIGURE VI-7

Common Signals in All Channels

In a digital controller it is possible and practicable to insure that the inputs to the integrators are the same in all channels. Let us assume that the sensors are cross-strapped so that each computer receives signals from all sensors. If each computer performs its own analog-to-digital conversion, then one can expect that all converted signals, in all channels, will be different due to small bias and gain errors in the converters. If the signals are inputted under DMA control, then one can even expect an additional difference between converted signals if the DMA controllers are non-synchronous. Without clock synchronization there are two approaches to insuring a common signal in all channels:

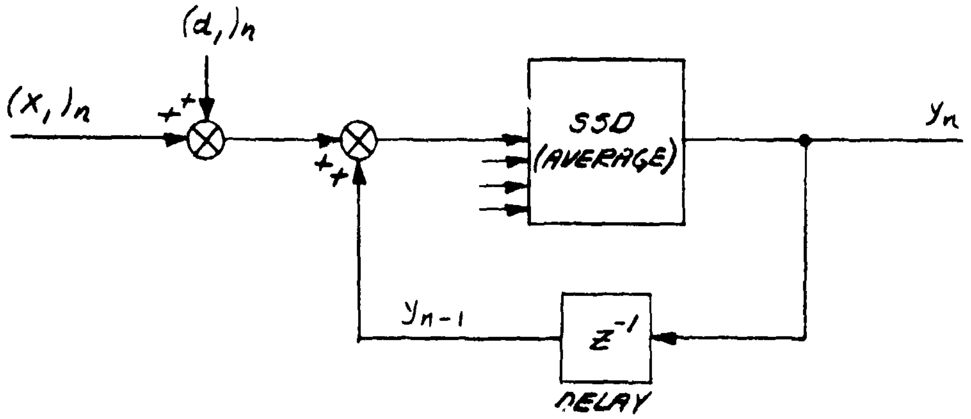
- In the normal sequences of operations each computer inputs all signals from a sensor set (assuming that we have cross-strapping) and performs a signal selection to obtain a reference output. It can be expected that this output will differ from similar outputs in other channels. At a subsequent fixed point in the program each computer transmits a code to the other computers. Upon receipt of codes from all computers each computer transmits its reference signal to all other computers via the intercomputer bus. A limited averaging selection is then performed. The resultant reference output will be the same in all computers. It should be noted that the reference signals prior to the second signal selection do not necessarily represent the sensed signal at the same instant of time due to phasing of the input sampling or to differences in the DMA clocks.

- A disadvantage of the above approach is the additional real time required to transmit all selected signals to the other computers. An alternate approach is to transmit only the integrator commands, via the intercomputer bus, and then use a limited average SSD to obtain a common input. This approach is shown in Figure VI-8. A similar approach is shown in Figure VI-9 where the integrator outputs are transmitted to the other computers. From the difference equation for each of these approaches we observe that the technique is functionally equivalent to averaging all of the inputs and transmitting the common average to all integrators.



$$\frac{x_1 + x_2 + x_3 + x_4}{4} + \frac{d_1 + d_2 + d_3 + d_4}{4} + y_{n-1} = y_n$$

STABILIZING INTEGRATOR VIA COMMON INPUTS
FIGURE VI-8



$$\frac{x_1 + x_2 + x_3 + x_4}{4} + \frac{d_1 + d_2 + d_3 + d_4}{4} + y_{n-1} = y_n$$

STABILIZING INTEGRATOR VIA COMMON OUTPUTS

FICUPE VI-9

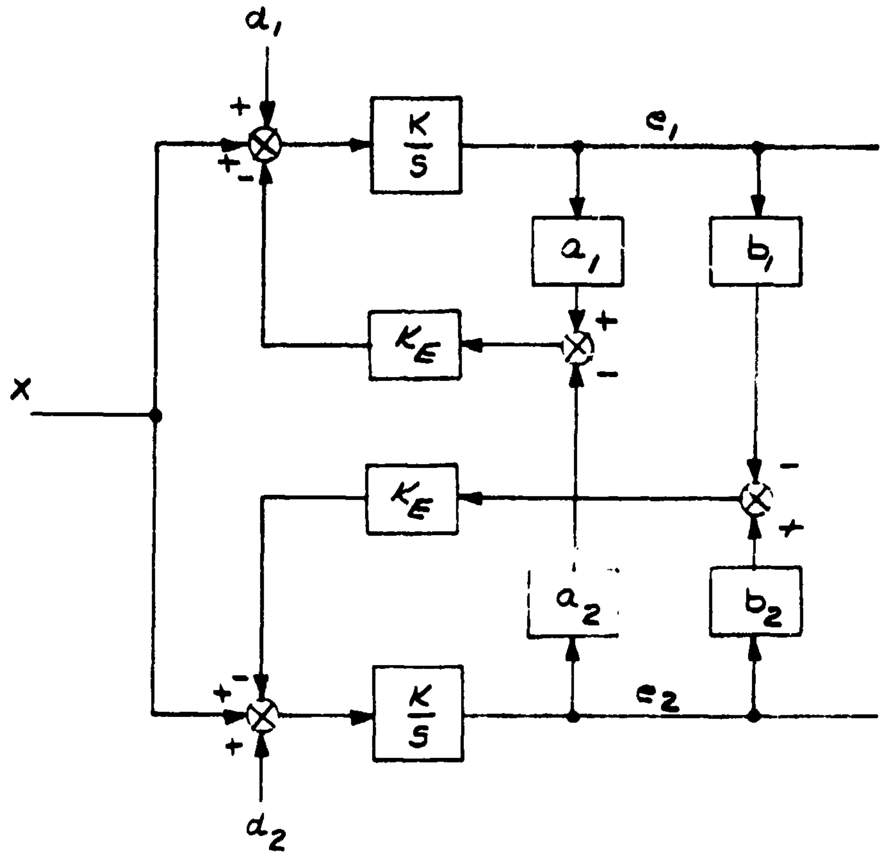
Equalization

Another technique which maintains the internal variables within prescribed bounds is "equalization". In this approach individual channel differences, or their equivalent, are fed back to all channels. This is a favorite technique in analog systems where it is difficult and impracticable to achieve common inputs in all channels due to differences in dedicated hardware and noise pick-up. Using equalization the integration of Figure VI-7 can be stabilized as shown in Figure VI-10. The extension of the same technique to a quadruplex configuration is shown in Figure VI-11.

Another approach to equalization is to utilize the difference between the output of the servo actuator signal selection device and each channel response as the equalizing feedback to that channel. This arrangement is shown in Figure VI-12. The advantage of this approach is that only one signal selection is required to service all channels.

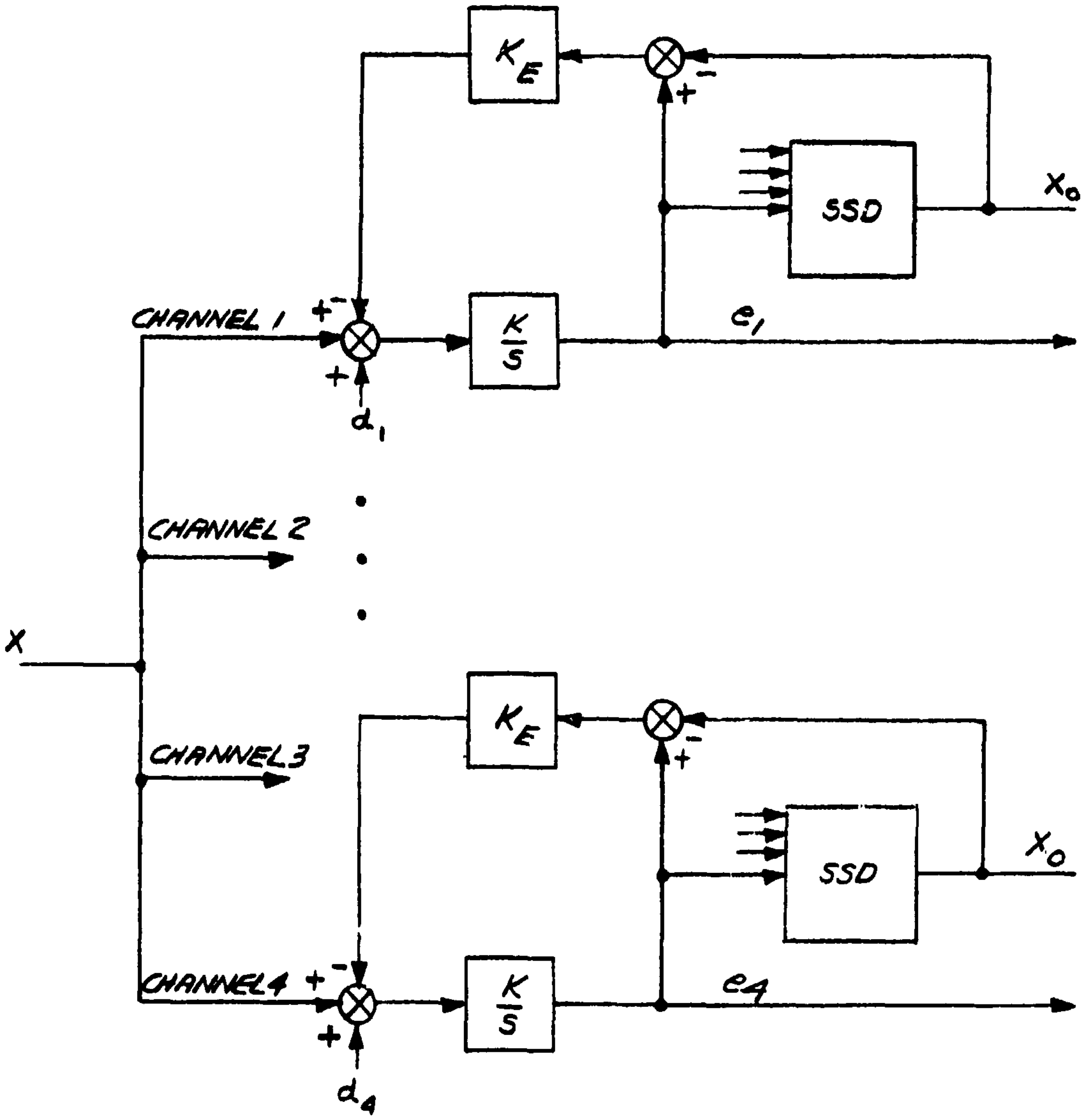
Another application of equalization is to reduce or eliminate servo command differences in order to reduce threshold and failure transients when the SSD is an MV type. A schematic version of this approach is shown in Figure VI-13. Each of these "equalizing" configurations will now be discussed in detail.

Referring to Figure VI-10, if we assume that the channel differences can be obtained without any variation in gain between channels, i.e., $1 = a_1 = a_2 = b_1 = b_2$, their equalization does, indeed, eliminate the "drift problem". From the figure we easily compute



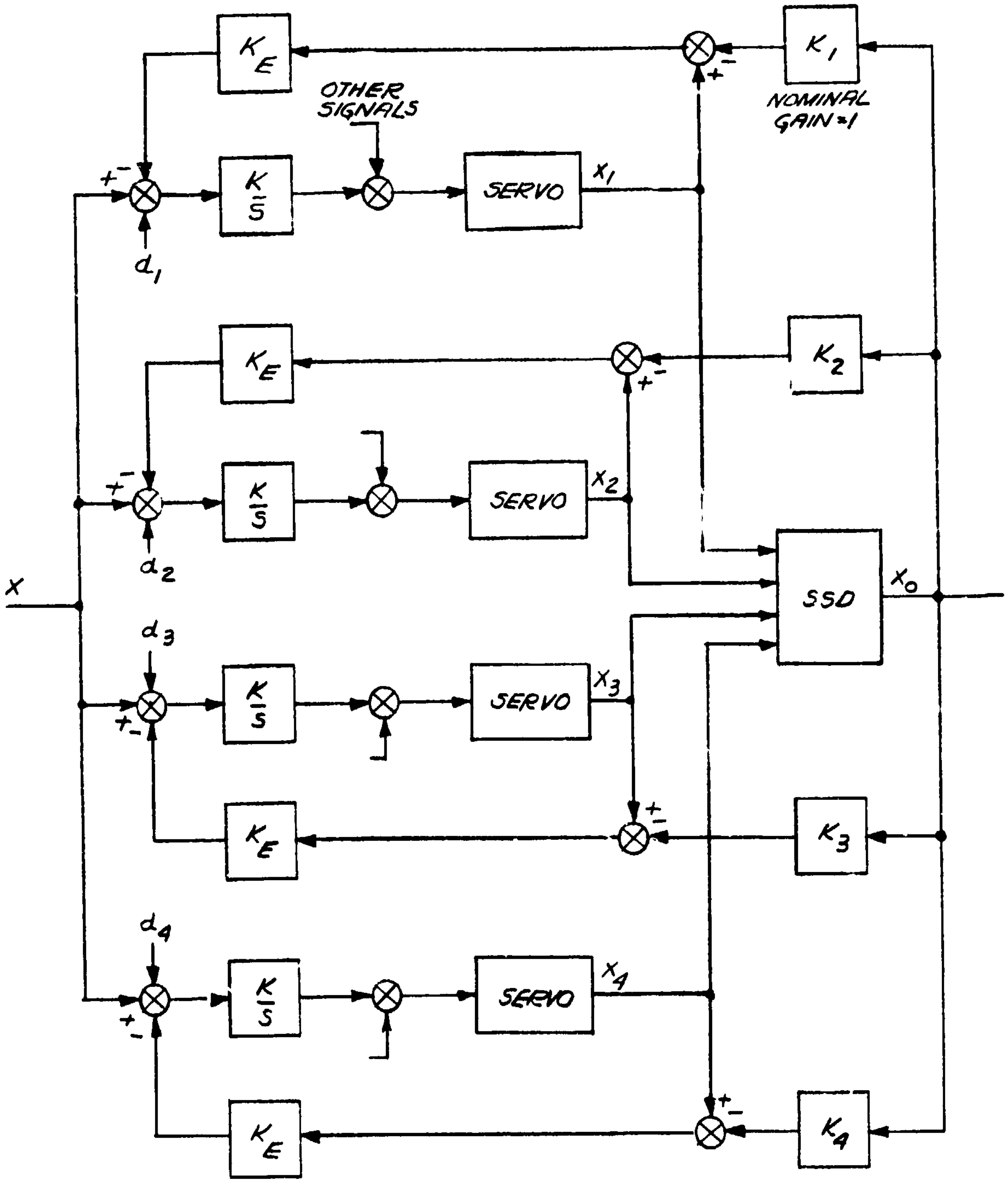
INTEGRATOR STABILIZATION VIA EQUALIZATION FOR A DUAL REDUNDANT SYSTEM

FIGURE VI-10



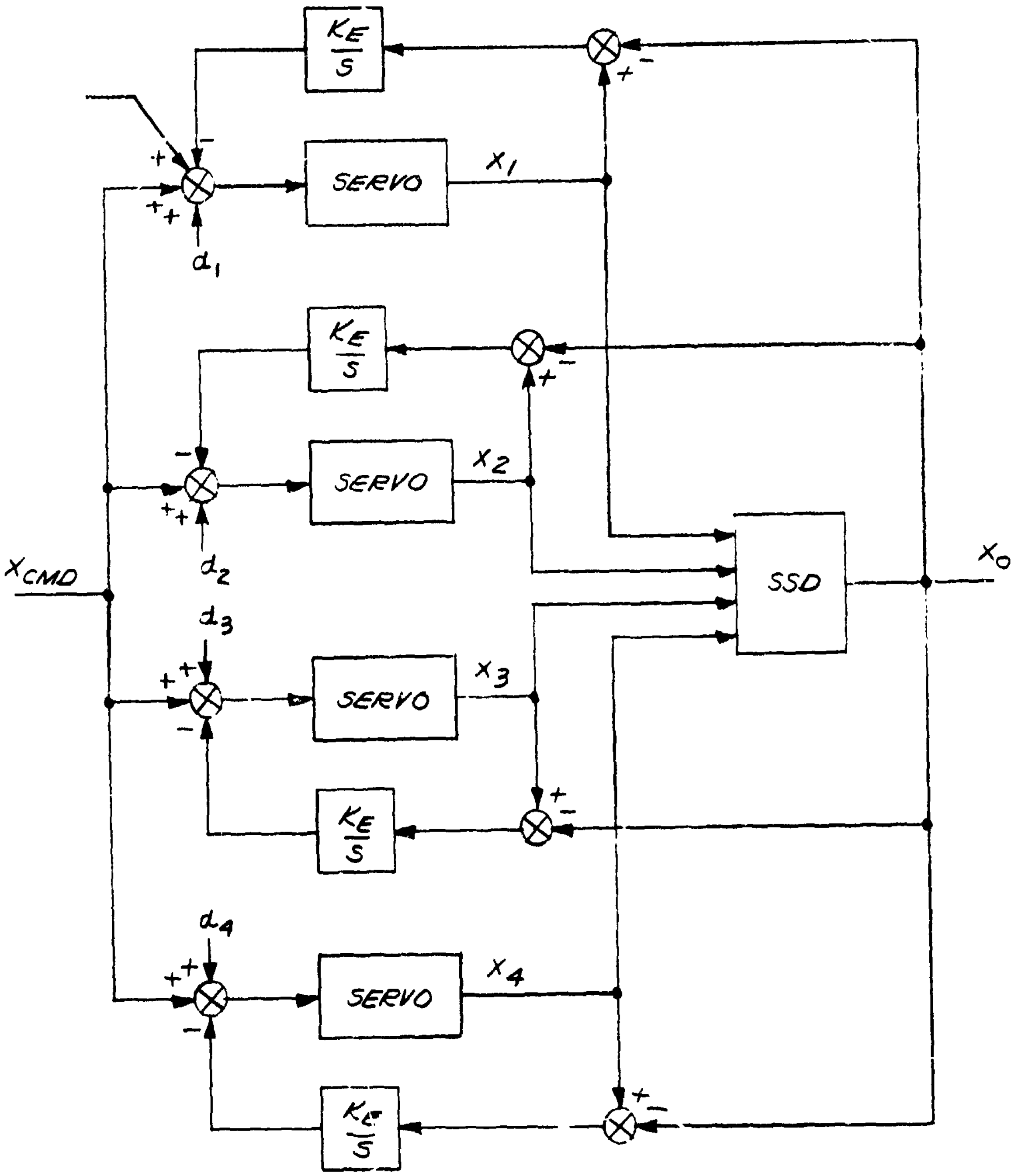
INTEGRATOR STABILIZATION VIA EQUALIZATION
FOR A QUADRUPLEX CONFIGURATION
USING INTEGRATOR OUTPUT DIFFERENCES

FIGURE VI-11



INTEGRATOR STABILIZATION VIA EQUALIZATION
FOR A QUADRUPLEX CONFIGURATION
USING SERVO DIFFERENCES

FIGURE VI-12



SERVO EOUALIZATION ''IA INTEGRATION

FIGURE VI-13

$$e_1 = \frac{K}{s}\,x + \frac{(s+K\,K_E)K\,d_1}{s(s+2K\,K_E)} + \frac{K^2K_E\,d_2}{s\,(s+2K\,K_E)}$$

$$= \frac{K}{s}\,x + \frac{\frac{1}{2}\,K\,(d_1+d_2)}{s} + \frac{\frac{1}{2}K\,(d_1-d_2)}{s+2K\,K_E}$$

$$e_2 = \frac{K}{s}\,x + \frac{(s+K\,K_E)\,K\,d_2}{s(s+2K\,K_E)} + \frac{K^2K_E d_1}{s(s+2K\,K_E)}$$

$$= \frac{K}{s}\,x + \frac{\frac{1}{2}K\,(d_1+d_2)}{s} + \frac{\frac{1}{2}K\,(d_2-d_1)}{s+2K\,K_E}.$$

Thus, $x = -\frac{1}{2}(d_1+d_2)$ will stabilize the divergence. The resultant steady-state channel difference is

$$e_1 - e_2 = \frac{1}{K_E}\left(\frac{d_1-d_2}{2}\right) .$$

In practice, however, the channel qains a_1, a_2, b_1, and b_2 are not equal. In the qeneral case the characteristic equation is

$$s^2+(a_1+b_2)K\,K_E s+K^2K_E{}^2\left[a_1(b_2-b_1)+b_1(a_1-a_2)\right]$$

and not

$$s^2 + 2K\,K_E\,s$$

as in the previous case. Thus, equalization is unstable if

$$a_1(b_2-b_1) + b_1(a_1-a_2) < 0.$$

From this simple example we see that equalization can introduce additional stability problems.

Referring to Figure VI-12 (an analogous argument applies to Figure VI-11), let us assume, first, that the nominal feedback gains are unity; i.e., $1 = K_1 = K_2 = K_3 = K_4$. We assume, throughout, that the SSD is an MV type. A necessary condition for drift stabilization requires that

$$x + d_1 - K_E\,(x_1 - x_0) = 0$$

$$x + d_2 - K_E\,(x_2 - x_0) = 0$$

$$x + d_3 - K_E\,(x_3 - x_0) = 0$$

$$x + d_4 - K_E\,(x_4 - x_0) = 0$$

Assume, without loss of generality, that

$$d_1 < d_2 < d_3 < d_4.$$

Then, from the above equalities we must have

$$x_1 - x_0 < x_2 - x_0 < x_3 - x_0 < x_4 - x_0$$

and, hence,

$$x_1 < x_2 < x_3 < x_4.$$

Since the device is a MV SSD we must have

$$x_0 = x_2 \text{ (assuming } x_0 > 0) \text{ or } 0$$

Case 1 $x_0 = 0$. Then

$$x_1 < x_2 < 0 < x_3 < x_4$$

and $x_1 = \frac{x+d_1}{K_E}$

$$x_2 = \frac{x+d_2}{K_E}$$

$$x_3 = \frac{x+d_3}{K_E}$$

$$x_4 = \frac{x+d_4}{K_E}$$

and the common input, x, can have any value such that

$$x + d_2 < 0 < x + d_3.$$

Observe that this is equivalent to a threshold about zero.

Case 2 $x_0 = x_2$. In this case

$$x = -d_2$$

will stabilize the drift. The channel differences are given by

$$x_1 - x_2 = \frac{d_1 - d_2}{K_E} < 0$$

$$x_3 - x_2 = \frac{d_3 - d_2}{K_E} > 0$$

$$x_4 - x_2 = \frac{d_4 - d_2}{K_E} > 0$$

Observe that it requires a steady-state value of x to stabilize the drifts.

Referring, again, to Figure VI-12 the equalization could become unstable if the feedback gains, K_i, are greater than unity. Such a condition could occur in practice due to tolerance variations in the sensing mechanism. There are two approaches (at least) to stabilizing the process:

- Select a nominal value of K_i less than unity and such that expected gain variations will not cause the gain to exceed unity. This technique can always be applied when the variables x_i and x_0 are individually accessible. Relative to the integrator input, x, this approach is equivalent to replacement of the integrator by a lag.

- Introduce a deadzone in the equalizing path as shown in Figure VI-14. The deadzone, ϵ , is selected in such a way that positive feedback is precluded. We now determine the minimum value of ϵ for this purpose. Consider the differences

$$x_i - K_i x_0, \quad i = 1, 2, 3, 4.$$

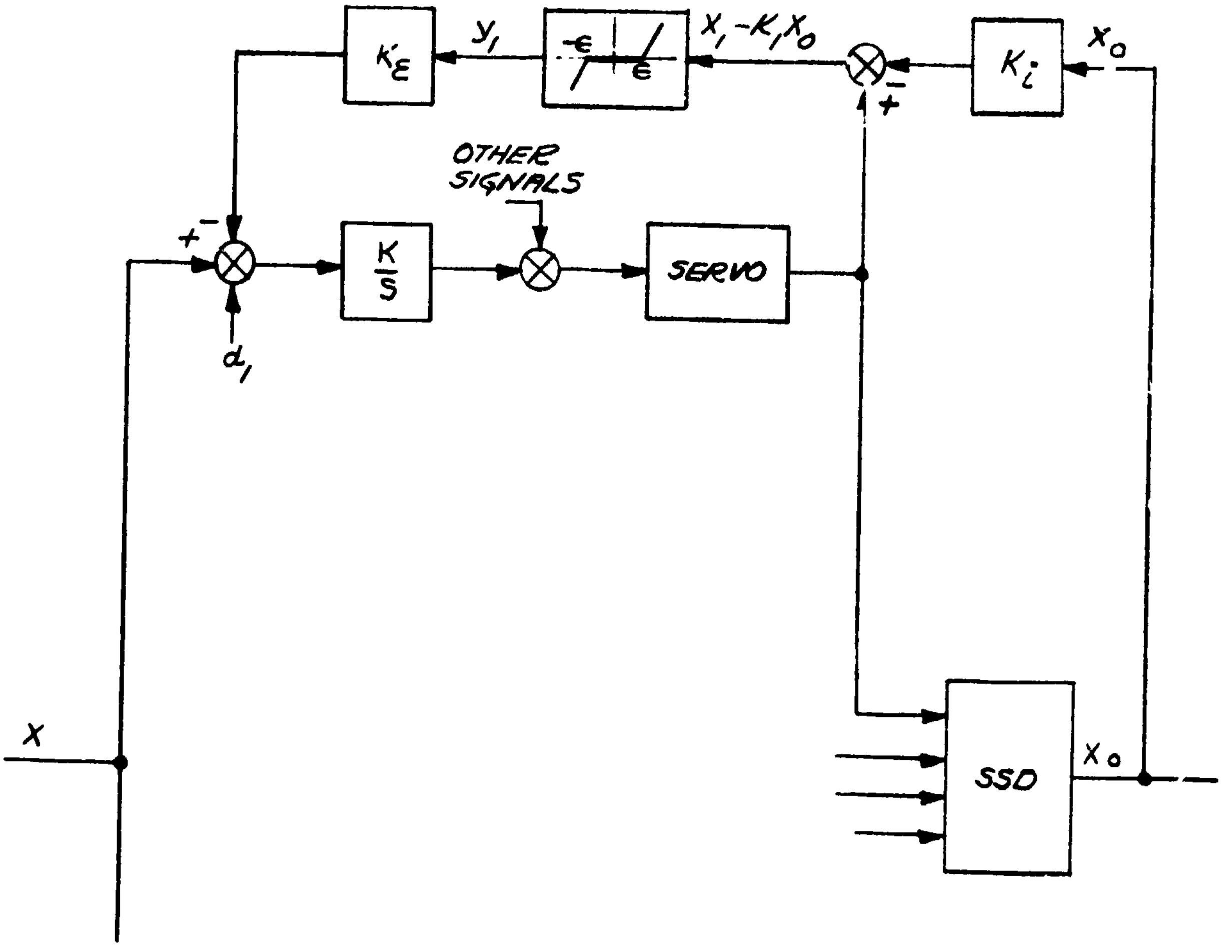
Let us suppose that K_0 is the largest of the gains K_1, K_2, K_3, K_4 and $K_0 > 1$. We vary the difference $x_i - K_0 x_0$ when $x_i = x_0$ over all values of x_0. The variation is shown in Figure VI-15. The largest positive variation, ϵ , is the minimum amplitude of the deadzone.

Let y_i = output of the deadzone in the i th channel.

We want to show that

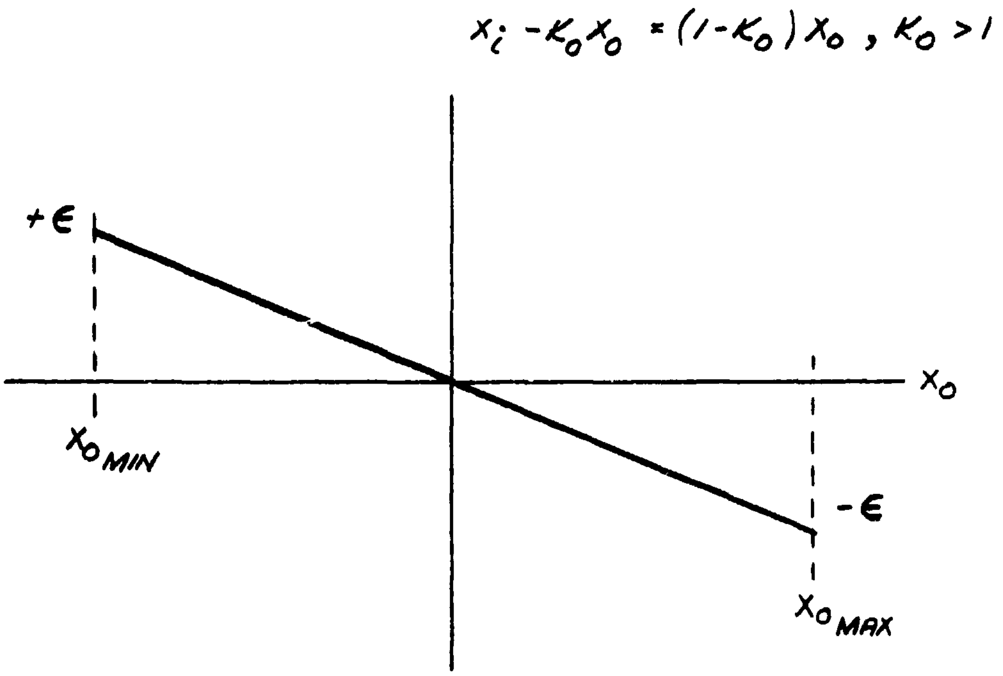
$$y_i \geq 0 \text{ when } x_i > x_0$$

and $$y_i \leq 0 \text{ when } x_i < x_0.$$



EQUALIZATION WITH DEADZONE

FIGURE VI-14



MINIMUM DEADZONE TO STABILIZE INTEGRAL EQUALIZATION

FIGURE VI-15

This will insure that no positive feedback path exists. Assume that

$$x_i > x_0. \text{ Then}$$

$$x_i - K_i x_0 = (x_i - x_0) + (1-K_i)\, x_0 > -\epsilon$$

since $x_i - x_0 > 0$ and $(1-K_i)\, x_0 \geq -\epsilon$.

Tnus $y_i \geq 0$.

Assume that

$$x_i < x_0. \text{ Then}$$

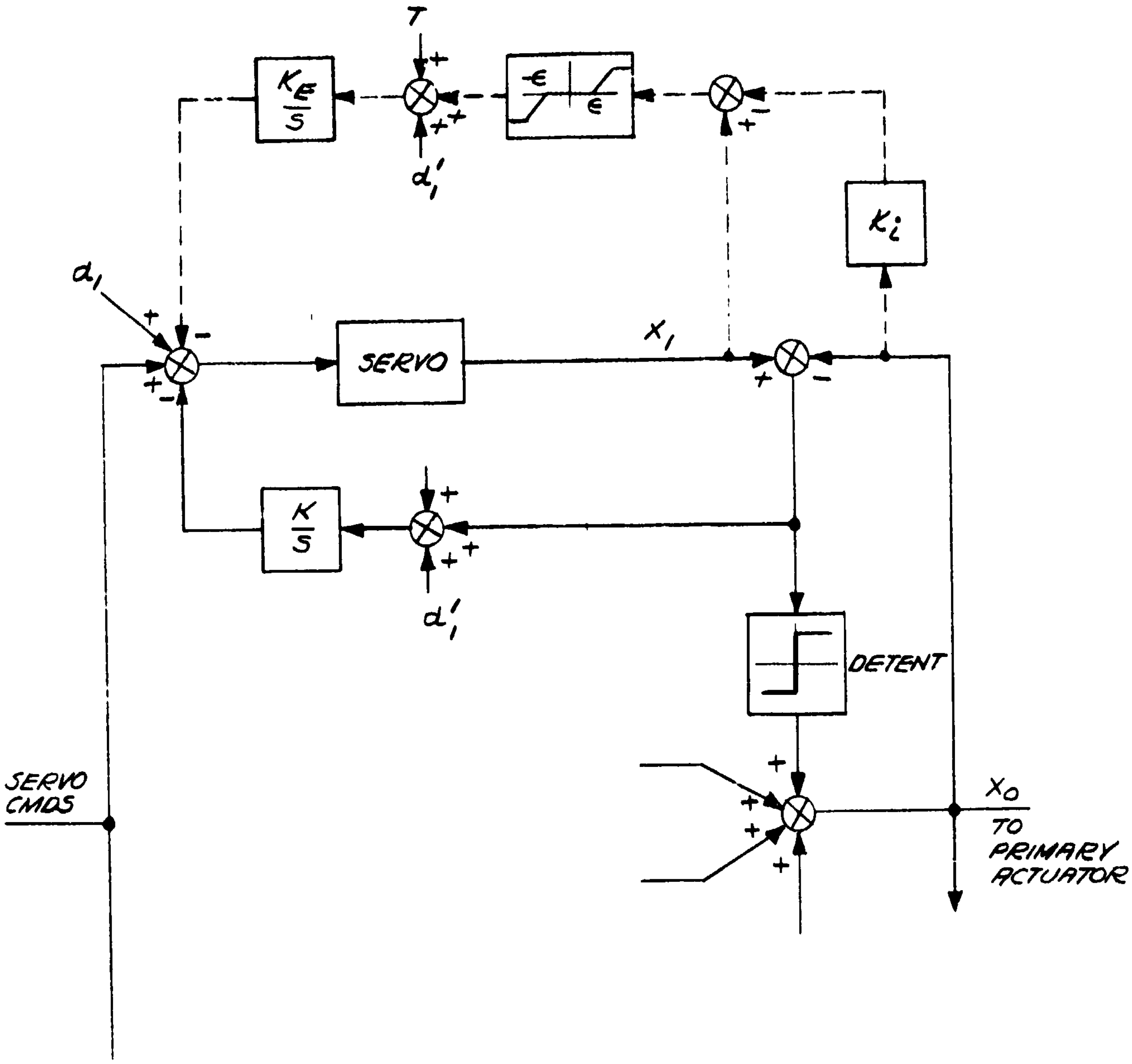
$$x_i - K_i x_0 = (x_i - x_0) + (1-K_i)\, x_0 < \epsilon$$

since $x_i - x_0 < 0$ and $(1-K_i)\, x_0 \leq \epsilon$.

Thus, $Y_i \leq 0$ and the result is established. This method of equalization can be employed when the integration is p··formed in the digital computer. The method does not require inter channel communications nor does it require common inputs to the integrators.

Servo Equalization

In Figure VI-16 we show a typical scheme for equalizing the servos which is similar to the equalization of a channel integrator, except that now an integrator is introduced for the purpose of equalization. In the figure we show two equalizing configurations, depending upon how the differences, $x_i - x_0$, are obtained. In some mechanical arrangements the difference can be measured directly. In this case we do not require a deadzone for stability. We henceforth assume that x_j and x_0



SERVO EQUALIZATION FOR A QUADRUPLEX CONFIGURATION

FIGURE VI-16

are measured separately. The same argument regarding stability applies here: if the gains, K_i, are greater than unity (the number of such gains will determine the stability after zero, one or two failures) than instability can result due to positive feedback. The remedy is the same as before: introduce a deadzone* in the feedback path.

We mention two problems in connection with integral equalization of the servos:

- The integrators will tend to drift due to internal offset or biases in the sensors or A/D converters if the integration is performed in the digital computers. The solution is to clamp the integrator in any channel in which the difference, x_i-x_0, falls in the deadzone. In general, the remaining channel differences will exceed the deadzone in order to equalize the drifts of their respective integrators.

- Because of the infinite memory of the integrator, the outputs, x_i, will eventually "walk" away from the command, x. The "walking" problem was actually observed in a simulation (Ref. 5). If the command signal, x, contains trim then "walking" can be prevented by occasionally retrimming. However, this does not prevent the integrators from eventually overloading. Furthermore, it is desirable that the integrators hold only a small percentage of the trim signal; otherwise, null failures of an integrator (after several failures) could result in a non-passive state of the airplane following loss of control.

*In some cases the equalization is limited in order to prevent equalization of slowover failures. This could inhibit detection of H/O failures if they occur upstream of the limiter.

One solution to this problem (which has not been verified in a simulation) is to bleed off the average value of the integrators on a long-time basis. The technique is illustrated in Figure VI-17. From the figure we obtain (ignoring the deadzone):

$$z = \frac{K_E}{s+k} \quad \frac{1}{4}(e_1+e_2+e_3+e_4)$$

$$\frac{y_i}{e_i} = \frac{K_E}{s} \quad \left(\frac{s+\frac{3}{4}k}{s+k}\right)$$

Thus, y_i is the integral of e_i, approximately, and the average value of the integrators, z, tends to zero, as desired.

Another approach to servo equalization is to use a lag in place of an integrator. Referring to Figure VI-13, we replace K_E/s by $\frac{K_E.}{s+\omega_o}$ To see the effects of lag equalization, we consider two cases:

Case 1 $\quad x_1 < x_2 < 0 \quad x_3 < x_4$

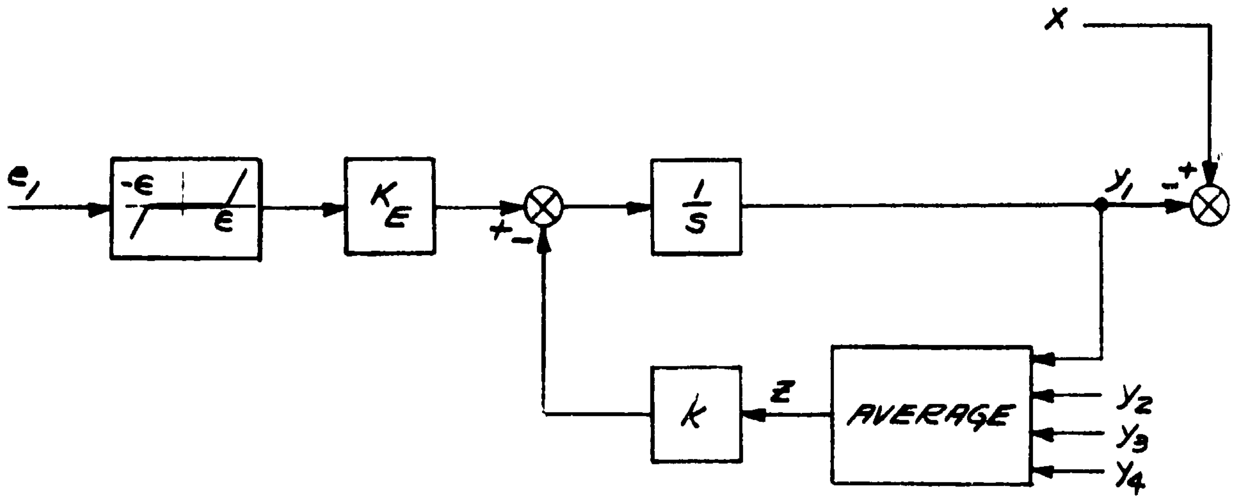
From the figure we obtain, in the steady state,

$$x + d_1 - \frac{K_E}{\omega_o} x_1 = x_1$$

$$x + d_2 - \frac{K_E}{\omega_o} x_2 = x_2$$

$$x + d_3 - \frac{K_E}{\omega_o} x_3 = x_3$$

$$x + d_4 - \frac{K_E}{\omega_o} x_4 = x_4$$



METHOD FOR PREVENTING OVERLOADING
OF THE EQUALIZING INTEGRATORS
IN A QUADRUPLEX SERVO CONFIGURATION

FIGURE VI-17

We conclude, therefore, that

$$\frac{x+d_2}{1+\frac{K_E}{\omega_o}} < 0 < \frac{x+d_3}{1+\frac{K_E}{\omega_o}} .$$

From this inequality we see that lag equalization does not reduce the steady state threshold (relative to x) about zero.

Case 2 $\qquad 0 < x_1 < x_2 < x_3 < x_4$

In this case $x_0=x_2$ and, in the steady state,

$$x + d_1 - \frac{K_E}{\omega_o} (x_1-x_2) = x_1$$

$$x + d_2 \qquad\qquad\qquad = x_2$$

$$x + d_3 - \frac{K_E}{\omega_o} (x_3-x_2) = x_3$$

$$x + d_4 - \frac{K_E}{\omega_o} (x_4-x_2) = x_4 .$$

We conclude that

$$\frac{d_4-d_2}{1+\frac{K_E}{\omega_o}} = x_4-x_2 .$$

Thus, lag equalization reduces the offsets between servo outputs, which has the effect of reducing failure transients.

Effects of Equalization

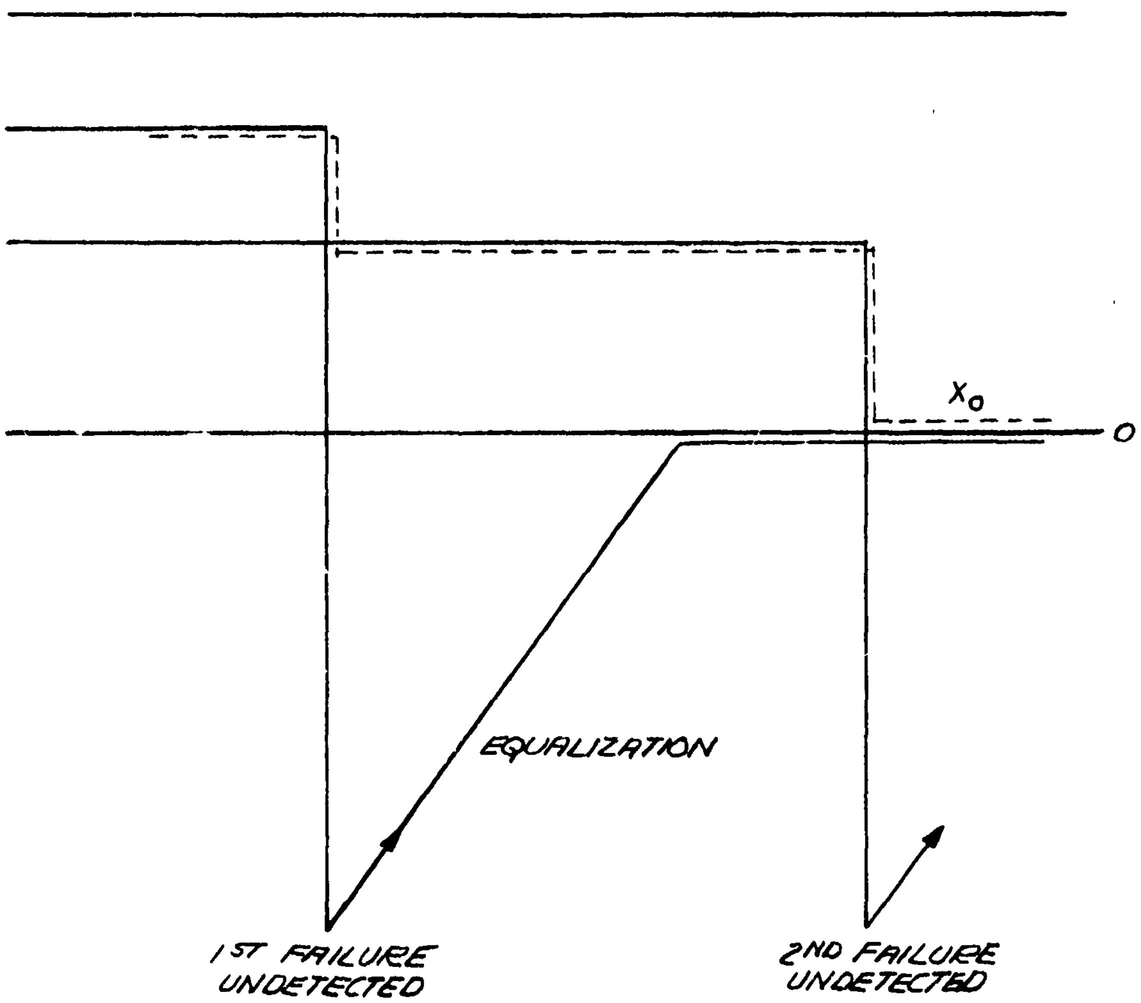
The principal advantages of equalization are:

- Eliminates integrator drift due to redundant channels.

- Reduces the steady-state differences between servo outputs. This reduces (a) the amplitude of limit cycle oscillations and (b) transients due to failures.

  An interesting feature of integral equalization is the effect it has on second failures in a triplex or quadruplex configuration. As an example, in the "triplex with back-up" configuration, a serious objection to manual selection of the back-up channel was the effect of two undetected hardover failures of the triplex system. In this situation the output would go hardover resulting in possible damage to the airplane before the pilot engaged the back-up. With integral equalization the first hardover eventually equalizes (provided that the failure is in the command), causing that servo output to maintain an average trim position. A subsequent hardover, after equalization, will cause the output to assume the good value or zero,whichever is closest to the failed signal. Thus, the result is a passive failure. This effect is shown in Figure VI-18.

The principal disadvantage of equalization is that it tends to mask failures. Thus, if in-flight failure detection is required, then equalization must be limited in order to insure that failures will be detected.



EFFECTS OF INTEGRAL EQUALIZATION
ON SECOND FAILURE TRANSIENT IN A TRIPLEX CONFIGURATION
WITH AN MV SSD
FIGURE VI-18

8. Supplement to Appendix VI

Let u(t), v(t), w(t) denote independent, random and stationary processes, with identical statistics. Let f(x), F(x) denote their probability density and cumulative distribution functions, respectively, at any instant of time. Let m denote the common mean.

Let z(t) denote the mid-value of u, v, w at time t. Then

$$\begin{aligned} z(t) &= u(t) \text{ if } (v \le u \le w) \text{ or } (w \le u \le v) \\ z(t) &= v(t) \text{ if } (u \le v \le w) \text{ or } (w \le v \le u) \\ z(t) &= w(t) \text{ if } (u \le w \le v) \text{ or } (v \le w \le u) \end{aligned}$$

Let

(1) E_{vuw} denote the event $(v \le u \le w)$

(2) E_{wuv} denote the event $(w \le u \le v)$

(6) E_{vwu} denote the event $(v \le w \le u)$

(7) E_u denote the event $(u \le x)$

(8) E_v denote the event $(v \le x)$

(9) E_w denote the event $(w \le x)$

Let G_z denote the cumulative distribution function of z. Then

$$G_z(x) = P(z \le x)$$

Since events (1), (2), ..., (6) are exhaustive it follows that

$$P(z \le x) =$$

$$P(E_{vuw} \,.\, E_u + E_{wuv} \,.\, E_u + \ldots + E_{uwv} \,.\, E_w + E_{vwu} \,.\, E_w)$$

Since the events (1), (2), 111, (6) are mutually exclusive except for the endpts

$$P(z \leq x) = P(E_{vuw} \cdot E_u) + \ldots + P(E_{vwu} \cdot E_w)$$

Because of symmetry these six probabilities are equal. Therefore, we need only compute one of them, e.g., $P(E_{uvw} \cdot E_v)$.

Observe

$$P(E_{uvw} \cdot E_v) = \iiint_R f(u)\, f(v)\, f(w)\, du\, dv\, dw$$

WHERE R is the region $(u \leq v \leq w)$ and $v \leq x$.

As an iterated integral this becomes

$$P(E_{uvw} \cdot E_v) = \int_{-\infty}^{x} f(v)\, dv \int_{-\infty}^{v} f(u)\, du \int_{v}^{+\infty} f(w)\, dw$$

Observe $F(x) = \int_{-\infty}^{x} f(u)du = 1 - \int_{x}^{\infty} f(u)du$

$$\therefore \quad \int_{v}^{+\infty} f(w)dw = 1 - F(v).$$

where $\int_{-\infty}^{v} f(u)du = F(v)$.

$$\therefore \quad P[E_{uvw} \cdot E_v] = \int_{-\infty}^{x} f(v)\, F(v)\, (1-F(v))\, dv$$

$$= \int_{-\infty}^{x} f(v)\, F(v)\, dv - \int_{-\infty}^{x} f(v)\, F^2(v)\, dv$$

A simple integration (observing that d F(v) = f(v)dv) yields

$$\int_{-\infty}^{x} F(v)\, F(v)\, dv = \frac{F^2(v)}{2} \Big|_{-\infty}^{x} = \frac{F^2(x)}{2}$$

since F(- ∞ = 0. Also

$$\int_{-\infty}^{x} f(v)\, F^2(v)\, dv = \frac{F^3(v)}{3} \Big|_{-\infty}^{x} = \frac{F^3(x)}{3}$$

Therefore, we conclude that

I $$G_z(x) = P(z \le x) = 6\left[\frac{F^2(x)}{2} - \frac{F^3(x)}{3}\right]$$

The probability density function is

II $$\frac{dG_z(x)}{dx} = g_z(x) = 6[1-F(x)]\, F(x)\, f(x)$$

since $\frac{dF(x)}{dx} = f(x)$.

Observe that if f(x) is symmetrical about the mean, m ; i.e., f(m-x) = f(m + x) then g_z is symmetrical about x = m. This follows from the identity

$$\int_{-\infty}^{m-x} f(y)dy = \int_{m+x}^{\infty} f(y)dy = 1 - \int_{-\infty}^{m+x} f(y)dy$$

i.e., F(m - x) = 1 - F(m + x).

As a consequence of this sysmmetry it follows that the expected value of z is m; i.e.,

E(z) = m.

In order to simplify the computations it will be henceforth assumed, without loss of generality, that the expected value is zero; i.e., m = 0.

In the following section the variance of z will be computed assuming that the initial process is Gaussian; i.e.,

(10) $$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-x^2}{2\sigma^2}} .$$

(11) $$F(x) = \int_{-\infty}^{x} f(y)dy,$$

Successively differentiating (10) yields

(12) $$f'(x) = -\frac{x}{\sigma^3\sqrt{2\pi}} e^{\frac{-x^2}{2\sigma^2}}$$

(13) $$f''(x) = \frac{x^2}{\sigma^5\sqrt{2\pi}} e^{\frac{-x^2}{2\sigma^2}} - \frac{1}{\sigma^3\sqrt{2\pi}} e^{\frac{-x^2}{2\sigma^2}}$$

$$= \frac{x^2}{\sigma^4} f(x) - \frac{f(x)}{\sigma^2} .$$

Observe also that

(14) $F'(x) = f(x)$, $F''(x) = f'(x)$ and $F'''(x) = f''(x)$.

The variance of z is

$$(15) \quad \sigma_z^2 = \int_{-\infty}^{+\infty} x^2 g_z(x)\, dx = \int_{-\infty}^{+\infty} 6\, x^2 [1 - F(x)]\ F(x)\, f(x)\, dx$$

$$= 6 \int_{-\infty}^{+\infty} x^2 F(x)\, f(x)\, dx - 6 \int_{-\infty}^{+\infty} x^2 F^2(x)\, f(x)\, dx.$$

Each of these two integrals will now be evaluated.

Let $I_1 = \int_{-\infty}^{+\infty} x^2 F(x)\, f(x)\, dx.$

Observe that

$$x^2 f(x) = [f''(x) + \frac{1}{\sigma^2} f(x)]\, \sigma^4$$

$$= [F'''(x) + \frac{1}{\sigma^2} F'(x)]\, \sigma^4$$

$$= \sigma^4 F'''(x) + \sigma^2 F'(x)$$

which follows from (13) and (14).

Therefore

$$I_1 = \int \sigma^4 F''' F\, dx + \int \sigma^2 F^1 F\, dx$$

Now observe that

$$\frac{d}{dx} \left(F'' F - \frac{F'^2}{2}\right) = F''' F$$

and $\frac{d}{dx} \frac{1}{2} F^2 = F'F$. Hence

$$I_1 = \sigma^4 (F'' F - \frac{F'^2}{2}) + \sigma^2 \frac{F^2}{2} \Big|_{-\infty}^{+\infty} = \frac{\sigma^2}{2}$$

since $F''(\infty) = F''(-\infty) = 0$, $F(\infty) = 1$, $F(-\infty) = 0$.

To evaluate

$I_2 = \int_{-\infty}^{+\infty} x^2 F^2(x) f(x) dx$ observe, again, that

$$x^2 f(x) = \sigma^4 F'''(x) + \sigma^2 F'(x)$$

and thus

$$I_2 = \int_{-\infty}^{+\infty} \sigma^4 F''' F^2 dx + \int_{-\infty}^{+\infty} \sigma^2 F' F^2 dx .$$

Now observe that

$$\frac{d}{dx} (F'' F^2) = F''' F^2 + 2 F F' F''$$

$$= F''' F^2 + \frac{d}{dx} (F F'^2) - (F')^3$$

and $\frac{d}{dx} (\frac{1}{3} F^3) = F' F^2$. Thus,

$$I_2 = \sigma^4 (F'' F^2 - F F'^2)\Big|_{-\infty}^{+\infty} + \sigma^4 \int_{-\infty}^{+\infty} (F')^3 \, dx + \sigma^2 \frac{F^3}{3} \Big|_{-\infty}^{+\infty}$$

$$= \sigma^4 \int_{-\infty}^{+\infty} f^3(x) \, dx + \frac{\sigma^2}{3}$$

Since $f(x) = \frac{1}{\sigma\sqrt{2\pi}} \; e^{\frac{-x^2}{2\sigma^2}}$

$$f^3(x) = \left(\frac{1}{\sigma\sqrt{2\pi}}\right)^3 e^{-\frac{3x^2}{2\sigma^2}} = \frac{1/\sqrt{3}}{(\sigma\sqrt{2\pi})^2} \; \frac{1}{\frac{\sigma}{\sqrt{3}}\sqrt{2\pi}} \; e^{-\frac{x^2}{2\left(\frac{\sigma}{\sqrt{3}}\right)^2}}$$

Therefore $\int_{-\infty}^{+\infty} f^3(x) \, dx = \frac{1/\sqrt{3}}{(\sigma\sqrt{2\pi})^2}$. Hence

$$I_2 = \frac{1}{\sqrt{3}} \left(\frac{\sigma}{\sqrt{2\pi}}\right)^2 + \frac{\sigma^2}{3} .$$

From (15) we conclude, finally that

$$\sigma_z^2 = 6 I_1 - 6 I_2 = 6 \left[\frac{\sigma^2}{2} - \frac{\sigma^2}{\sqrt{3}(2\pi)} - \frac{\sigma^2}{3} \right]$$

$$= \left(1 - \frac{\sqrt{3}}{\pi}\right) \sigma^2 .$$

When the selection is based on the averaqe of the inputs then, obviously,

$$E\left[\left(\frac{u + r + w}{3} - m\right)^2\right] = \frac{1}{3}\sigma^2$$

where m is the mean value of μ, v and w.

APPENDIX VII

SELF TEST CONSIDERATIONS

In the flight control application the major components whose failures must be detected either in-flight or in preflight are:

- Sensors
- Digital flight control computers
- Actuators
- Displays and controls
- Monitoring, testing and disengage devices
- Communications paths
- Redundant-system-associated components (e.g., SSD's, intercomputer links, etc.)

As demonstrated previously, undetected failures in these components can result in a significant reduction in mission reliability. Failures must be detected with a coverage which is consistent with the reliability goals of the system. It has been shown that failure detection requirements are a function of the redundant configuration. For some conventional configurations, a FBW mission reliability goal may require a preflight test efficiency of 99.9% if periodic, 100% testing is not employed. Such requirements are several orders of magnitude beyond what is demonstrably achievable. As a consequence, we may conjecture that:

- Methods of failure detection will have to be exceedingly more comprehensive than techniques now in use
- New methods of validating a failure detection procedure will be required

With respect to failure detection the following tasks should be an integral part of the design and synthesis of a redundant flight control system:

Statement of Mission Reliability Goal

An explicit goal forces the designer to view the contribution of each component in the perspective of the whole system and leads to a practicable and fair allocation of failure rates. The criterion of relative mission reliability can lead to unnecessary, inconsistent and costly refinements.

Allocation of Failure Rates

Failure rates should be allocated to all system components based upon (a) what is necessary and (b) what is achievable.

Statement of Failure Detection Requirements

The objectives of in-flight and pre-flight failure detection should be stated. In particular, the extent to which in-flight detection contributes to the attainment of the mission reliability goal should be made explicit. In-flight and pre-flight failure detection efficiency requirements for all system components should be explicit and should consider their effect on mission reliability.

In-Flight and Pre-Flight Failure Detection Validation

Having determined detection efficiency goals and techniques to achieve these goals it is then necessary to develop a procedure which is capable of validating the claimed efficiencies.

In summary, we may state that the three development phases of a test procedure relative to the flight control system are:

- Requirements
- Achievement
- Validation

Thus far in the study we have emphasized failure detection requirements. What is actually achievable and by what means has not been discussed at all. In the following sections we will examine some general aspects of failure detection for the purpose of exposing some of the difficulties involved in achieving near-perfect coverage. The emphasis of the discussion is on digital devices, exclusively.

1. The Sequential Machine Model

We take, as our model of the digital device, the sequential machine. The following brief description of a sequential machine can be augmented by almost any reference on the subject, and in particular, by References (8), (9), and (10).

A sequential machine is a device which accepts, at discrete instants of time, an input and simultaneously issues an output. In general the output will depend upon the past as well as the present input. This dependence on the past leads naturally to the motion of "state" which embodies the past history of the device.

A lag filter whose inputs and outputs are impulse modulated is an example of a sequential machine. In this device the inputs, outputs, and states are infinite in number. In our application, however, the inputs and outputs will be binary coded decimals of fixed length and, thus, are finite. Similarly, the number of internal state variables (usually the equivalent of flip-flop outputs) will be finite. A machine with a finite number of inputs, outputs and states is called a finite, sequential machine. We formalize the definition as follows:

Let $X = \left\{x_1, x_2, \ldots\ldots, x_m\right\}$

$= $ set of inputs

$S = \left\{S_1, S_2, \ldots\ldots, S_n\right\}$

$= $ set of internal states

$Y = \left\{y_1, y_2, \ldots\ldots, y_p\right\}$

$= $ set of outputs.

Then a finite sequential machine is the pair of functions f and g such that

$$y^i = f(x^i, S^i)$$
$$S^{i+1} = g(x^i, S^i)$$

where$x^i \epsilon X$, $y^i \epsilon Y$, $S^i \epsilon$ Sand the superscript, i, denotes the ith instant of time. When S consists of a single state the machine is called a combinatorial machine.

In general the inputs, outputs and states are vector quantities as, for example, when the input consists of the binary bits of a binary cod·d decimal. The sequential machine, as defined above, provides the option of outputting one of many output symbols from a given state, depending on the input. This type of machine is referred to as a "Mealy" model in contrast to a "Moore" model which yields an output as a function only of the state and not the input. For our purposes, however, the two models can be shown to be equivalent (Ref. (9), page 29).

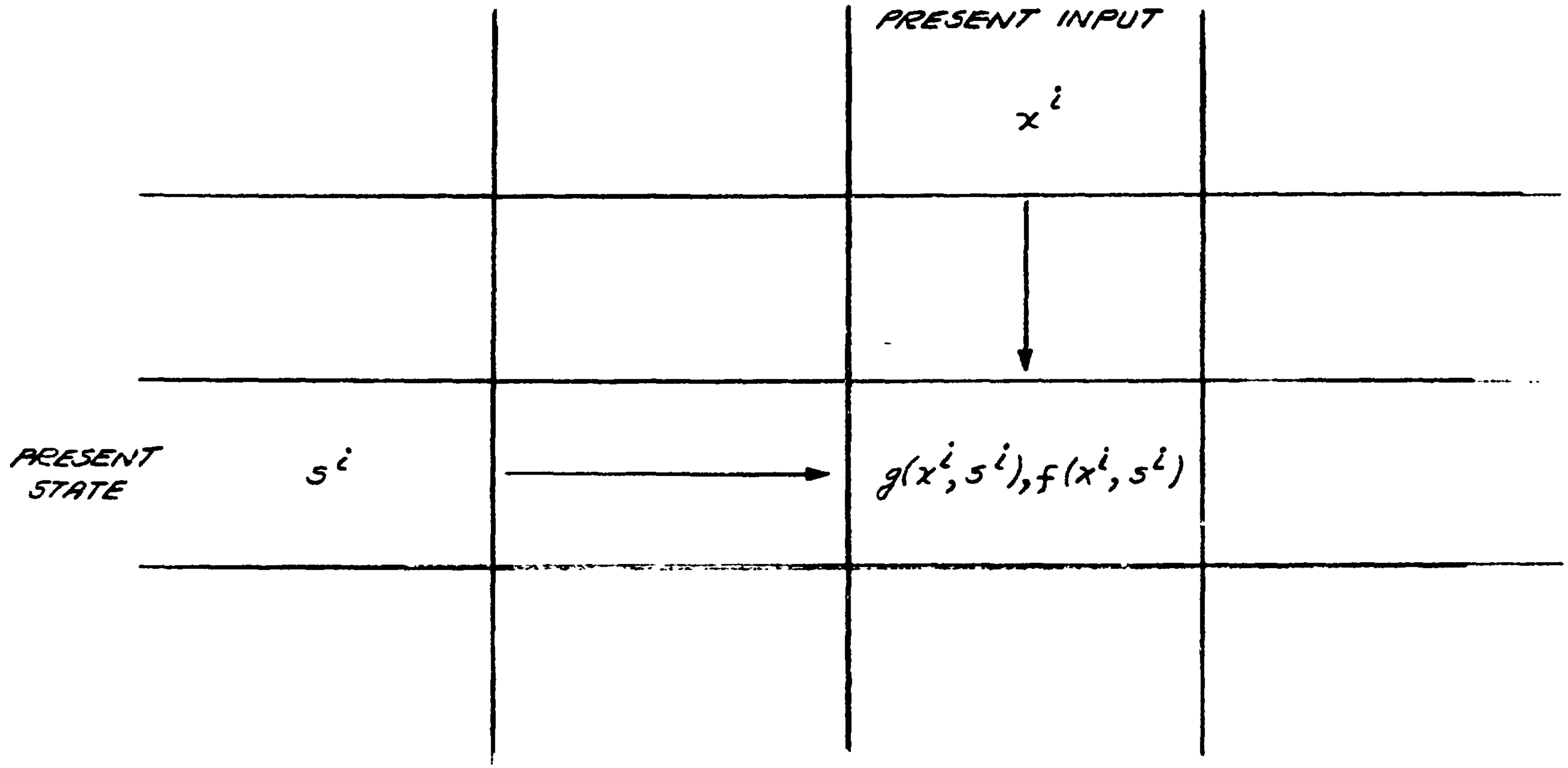
The functions f and g completely define the sequential machine. An alternate representation is by means of (a) a state table or (b) a state diagram. In a state table the row entries are the present states, the column entries, the present inputs. Each table entry consists of the next state and the output. The state table is shown in Figure VII-1. In the state diagram each state is represented by a circle. Directed arrows connecting pairs of states indicate the next state. Above each arrow is noted the present input and output. The arrows are called transition paths or branches (See Figure VII-2).

A machine with n states, m inputs, and p outputs will be called an <u>(n, m, p) machine</u>.

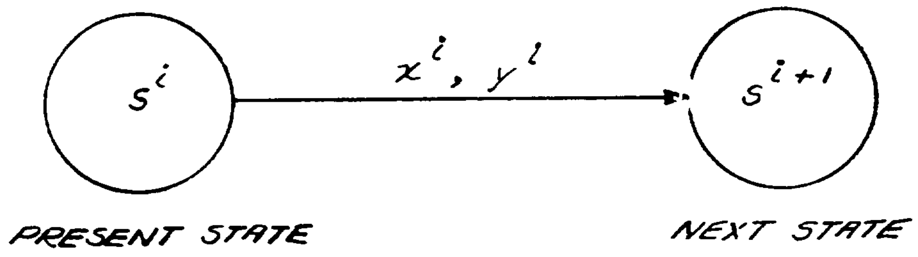
From the state table it can be seen that there are nm entries. Each entry can consist of one of n states and one of p outputs. Hence, there are $(np)^{nm}$ possible (n, m, p) machines. However, not all of these are distinct. For any machine we can obtain n! equivalent machine by simply permutting and relabeling the states. Thus, we conclude that there are, at most, $\frac{(np)^{nm}}{n!}$ distinct (n, m, p) machines.

<u>Examples of Sequential Machines</u>

The sequential machine is a convenient device for representing the operation of a digital circuit and, a fortiori, for representing the effects of failures. Before proceeding further we give the rationale for the present discussion. We are given a digital circuit which is represented by a certain (n, m, p) sequential machine, provided that the circuit has not failed. If the circuit fails then it will behave like some other sequential machine, not necessarily an (n, m, p) machine. It is then the task of the fault diagnostician to determine that the failed machine does not behave like the original machine. We place one restriction on the observer: He cannot directly observe the internal states; his diagnosis must be obtained by injecting inputs and observing the corresponding outputs. It is permissible that he have at his disposal the state table representations of as many sequential machines as are necessary.

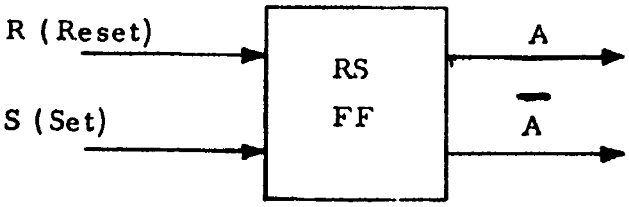


STATE TABLE
FIGURE VII-1



PORTION OF STATE DIAGRAM
FIGURE VII-2

Example 1 RS Flip-Flop



"Set" changes A=0 to A=1

"Reset" changes A=1 to A=0

States:

S_1 = (A=1, $\bar{A}$=0)

S_2 = (A=0, $\bar{A}$=1)

S_3 = (A=1, $\bar{A}$=1)

S_4 = (A=0, $\bar{A}$=0)

} For Failure Conditions Only

Outputs:

y_1 = (1,0)

y_2 = (0,1)

y_3 = (1,1)

y_4 = (0,0)

} For Failure Conditions Only

Inputs:

x_1 = (R=0, S=0)

x_2 = (R=0, S=1)

x_3 = (R=1, S=0)

x_4 = (R=1, S=1)} For Failure Conditions Only

RS FLIP-FLOP
FIGURE VII-3

It is emphasized that states S_3 and S_4, input x_4 and outputs y_3 and y_4 do not occur under nominal, non-failed conditions. They should be included in the state table, however, for fault-diagnosis purposes even though the corresponding "next of state" and "output" entries are left blank. Obviously, it is always an advantage to know how the circuit will respond under failed conditions. In particular, the response of a flip-flop to the input x_4 should always be specified. The state table and state diagram of the RS flip-flop are shown in Table VII-1 and Figure VII-4, respectively.

Example 2 Serial Binary Adder

States: S_1 = 0-carry

S_2 = 1-carry

Outputs: Y_1 = 0

y_2 = 1

Inputs: $x = (a,b)$, $a=0, 1, b=0, 1$

a = addend bit, b = augend bit.

Thus, there are four possible inputs. The state table is shown in Table VII-2 and the state diagram in Figure VII-5. A logic diagram of the serial adder is shown in Figure VII-6.

Example 3 Random Access Memory

For simplicity we assume that the RAM consists of 2, 1-bit words,

States: S = (w1, w2)

w1 = word #1 = 1 bit
w2 = word #2 = 1 bit

Thus, there are 4 states.

Outputs: y_1 = 0

y_2 = 1

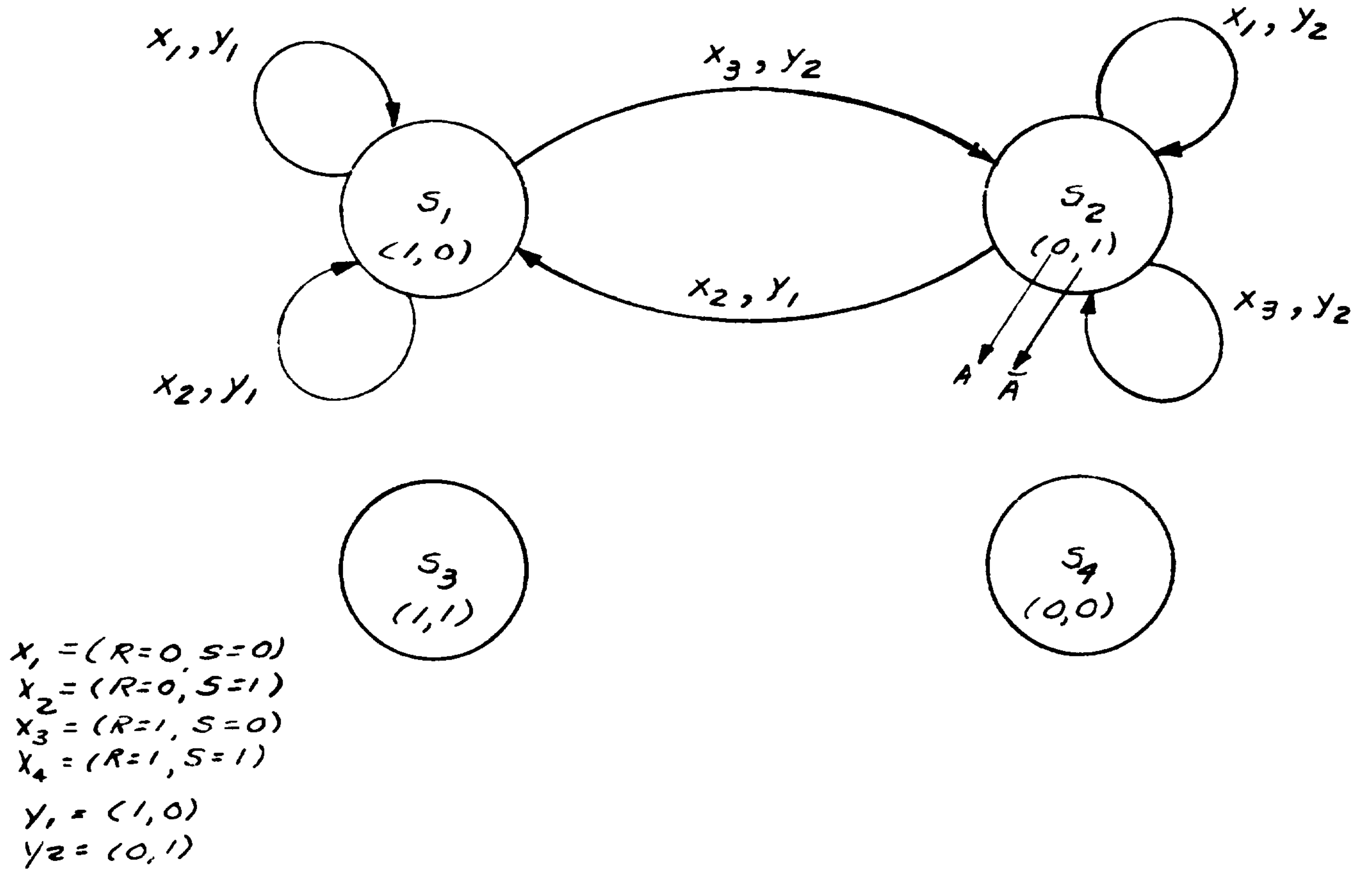
Inputs: x = (ADDR, RW, 1)
ADDR = 1-bit address
RW = 0 if read,
= 1 if write,
I = 1-bit word, to be stored.

Thus, there are 8 inputs.

INPUTS

		X_1 $(R=0, S=0)$	X_2 $(R=0, S=1)$	X_3 $(R=1, S=0)$	X_4 $(R=1, S=1)$
PRESENT STATES	$S_1 = (A=1, \bar{A}=0)$	$(A=1, \bar{A}=0)$, $(1,0)$	$(A=1, \bar{A}=0)$, $(1,0)$	$(A=0, \bar{A}=1)$, $(0,1)$	
	$S_2 = (A=0, \bar{A}=1)$	$(A=0, \bar{A}=1)$, $(0,1)$	$(A=1, \bar{A}=0)$, $(1,0;$	$(A=0, \bar{A}=1)$, $(0,1)$	
	$S_3 = (A=1, \bar{A}=1)$				
	$S_4 = (A=1, \bar{A}=0)$				

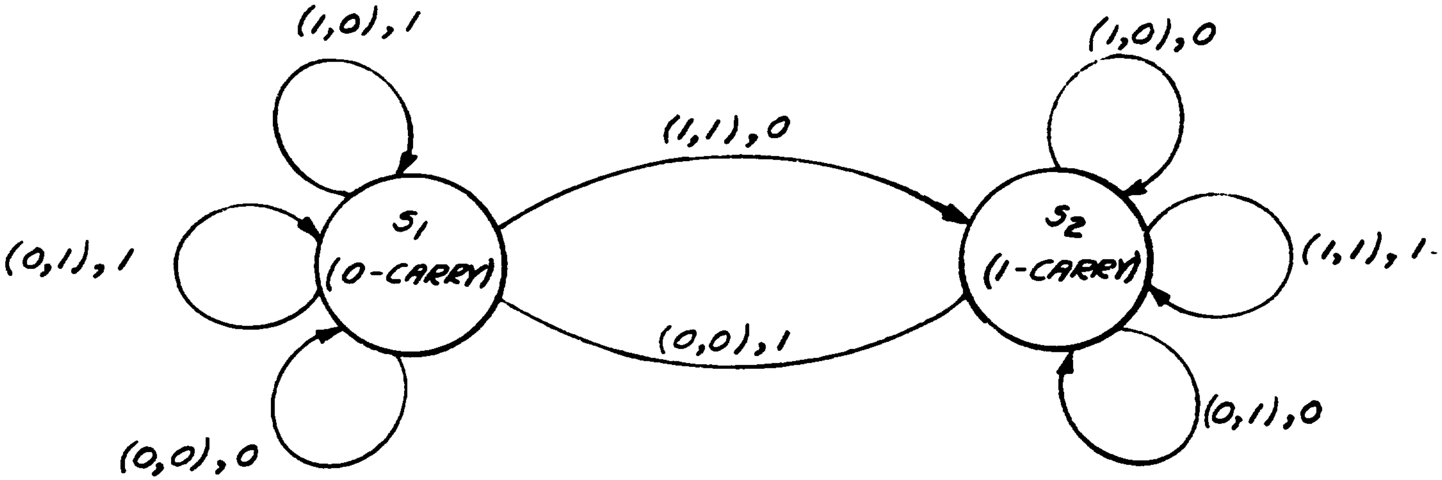
STATE TABLE FOR RS FLIP-FLOP
Table VII-1



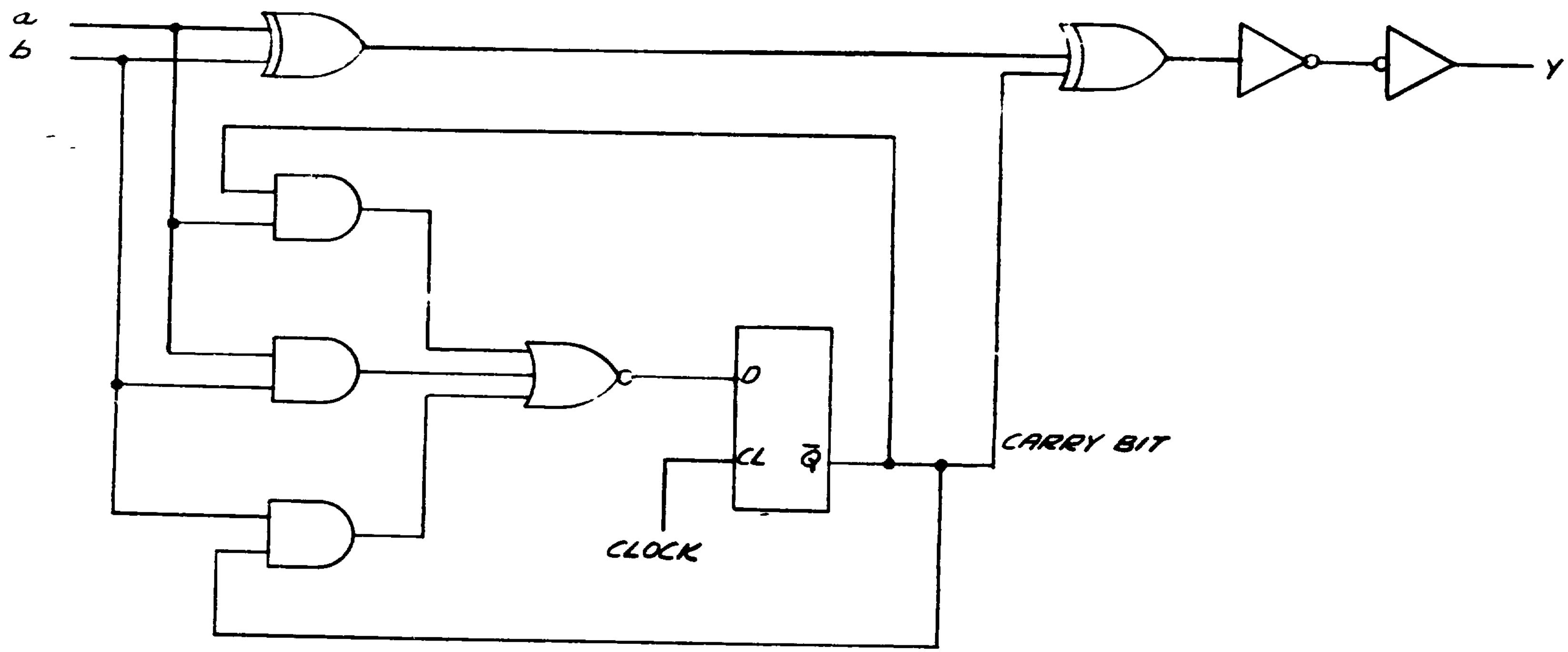
STATE DIAGRAM FOR RS FLIP-FLOP
FIGURE VII-4

		INPUTS			
		(0,0)	(0,1)	(1,0)	(1,1)
PRESENT STATES	S_1 (=0)	S_1, 0	S_1, 1	S_1, 1	S_2, 0
	S_2 (=1)	S_1, 1	S_2, 0	S_2, 0	S_2, 1

STATE TABLE FOR SERIAL BINARY ADDER
Table VII-2



STATE DIAGRAM FOR SERIAL BINARY ADDER
FIGURE VII-5



LOGIC DIAGRAM OF SERIAL BINARY ADDER
FIGURE VII-6

A portion of the state diagram is shown in Figure VII-7 for the state (0,0).

2. Representation of Failures (i.e., Failure Effects)

We start with a digital circuit which realizes a certain sequential machine, M'. In its non-failed condition the circuit is a realization of a certain (n, m, p) machine, M. Because of the possible existence of failures M' may or may not be equivalent to M. It is the task of the diagnostician to make this determination. Before proceeding further it is necessary to define the notion of "equivalence" of two machines.

Definition 1

Machine M' is equivalent to machine M if and only if

- for every state, S_i' of M' and every input sequence there exists a state, S_j, of M such that the input sequence with M' in state S_i' produces the same output sequence as the input sequence beginning with M in state, S_j, and
- for every state, S_j, of M and every input sequence there exists a state, S_i', of M' such that the input sequence with M in state S_j produces the same output sequence as the input sequence beginning with M' in state S_i'.

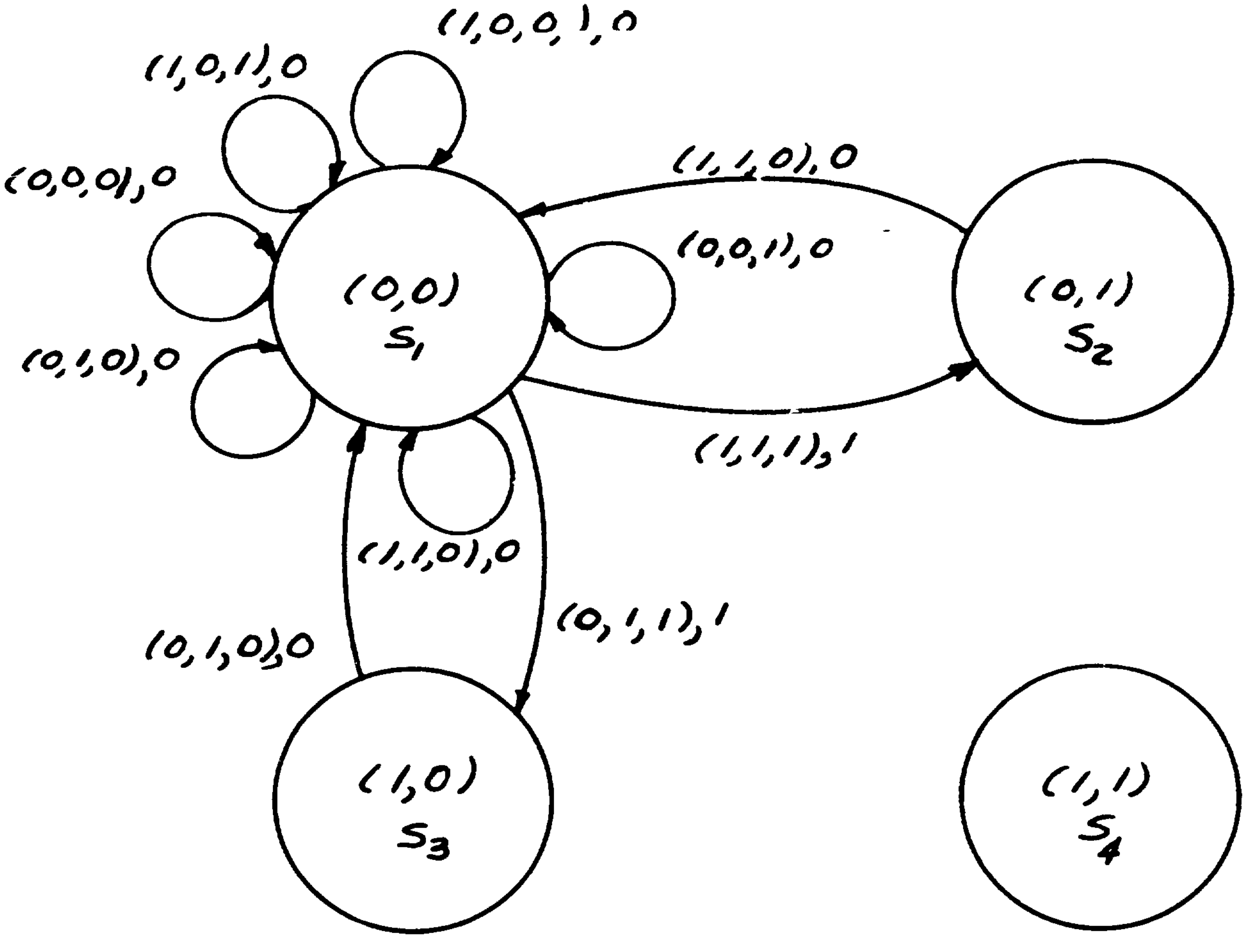
It is impossible to distinguish between equivalent machines on the basis of inputs and outputs alone. We note that the structure of equivalent machines may be quite different. We now define a failed machine.

Definition 2

Machine M' is a failed replicate of machine M if and only if M' is not equivalent to M.

Thus far we have not said anything regarding the structure of the failed machine. To this end we make the following assumptions:

- The set of inputs does not change.
- The set of outputs does not change.
- The number of states does not increase.



STATE DIAGRAM FOR RAM OF 2, 1-BIT WORDS (INCOMPLETE)
FIGURE VII-7

The first two assumptions are relatively weak and imoose minimum constraints on the failure modes. The third assumption is necessary in order to circumscribe the problem. It appears to be reasonable and, in any case, is almost always invoked in the literature. We note, in reference to this assumption, that, given an input sequence of any finite length there is a machine M', beginning in some state, which will yield the same output as M beginning in its initial state.

In summary we assume that a failed replicate of an (n, m, p) machine is, again, a different (n, m, p) machine.

Failures

The following circuit failures can cause a machine to fail:

- stuck-at-0
- stuck-at-1
- opens
- shorts
- bridging (logic)
- intermittent

The first four failures could result in a reduction in states. Bridging could result in either an increase or decrease in the number of states. Intermittent failures may be caused by vibrations or noise and are not necessarily reproducible.

a. Test Philosophies

There are several philosophies regarding testing of digital computers:

- The computer is designod with dedicated additional hardware for the express purpose of detecting failures, usually througŀ redundance and comparison-type monitorinơ.
- Error detection coding of internal computer variables. These variables are coded in such a way that a failure or failures will very likely cause a recognizable change in the code.

- Use of a software program to self test all accessible internal devices. Each device is tested against a stored state table or a portion thereof.

- Generate and output internal variables for comparison with similar variables in an identical computer (comparison-monitoring).

We will discuss only the self test philosophy assuming, as we do, that the computer was not designed with built-in failure detection capability.

b. Computational Requirements of Self Test

In this section we will obtain estimates for the length of the input sequence necessary to completely test an (n, m, p) machine subject to previously stated assumptions regarding the effects of failures. For convenience, we restate these assumptions:

A failed replicate of an (n, m, p) machine is, again, an (n, m, p) machine.

For purposes of obtaining estimates we make the additional assumption that for each (n, m, p) machine, M', there is a set of component failures which will transform the original and non-failed machine, M, into M'.

It is undoubtedly true that the class of machines which can replicate a failed machine is smaller than the class of all (n, m, p) machines. However, since we do not have sufficient information at the present time to significantly limit this class we proceed on our assumption, which, in any case, presents the greater difficulties to the diagnostician.

We now obtain a lower bound on the length of an input sequence required to test an (n, m, p) machine. The example is essentially due to Moore, (Ref. 8). Consider a combination lock with combination a_1, a_2. . . .a_{n-1}, where each digit, a_i, could have assumed one of m values. Such a lock can be represented by an (n, m, 2) machine as shown in Figure VII-9. The combination lock opens when the output equals unity and this can occur, starting in state S_1, if and only if the input sequence is a_1, a_2. . ...a_{n-1}. Now it is obvious that, in order to test the lock, it is necessary to try all of the possible combinations and that the number of combinations is m^{n-1}. An additional example is given in the Supplement to this Appendix.

In order to appreciate the magnitude of this estimate consider a typical 64-bit RAM, which is organized as 16, 4-bit words. The input is a 9-bit binary word, 4 bits of which designate the input word, 4 bits the address, and one bit to read or write. The output is a 4-bit word. Thus, the RAM can be represented by a sequential machine with

$$n = 2^{64} \text{ states}$$

$$m = 2^{9} \text{ inputs}$$

$$p = 2^{4} \text{ outputs}$$

The estimate of m^{n-1} is to be used when no advantage is taken of the unique structure of the device being tested. Thus, if the self test is designed for a particular device it may be possible to do considerably better than m^{n-1}. For the 64-bit RAM we obtain

$$m^{n-1} = (2^{9})^{2^{64}} = 2^{1.12x2^{67}} \quad 10^{3x10^{19}}$$

as the minimum length of the input sequence required to test the device.

From this simple example we may conclude that an efficient and practicable self test must be designed to take advantage of the unique structure of each device being tested.

With regard to an upper bound on the length of input sequence required (in view of the lower bound, the upper bound is of academic interest, only) Moore (Ref. 8) gives the estimate

$$\frac{n^{mn+2}p^{n}}{n!}$$

for an (n,m,p) machine.

We note, in passing, that there is considerable literature available in the area of fault-diagnosis of digital devices. A significant portion of this effort is directed towards developing specialized input sequences designed to detect failures of specific combinatorial and sequential circuits. We have not seen any published data regarding the efficiency of such algorithms when applied to MSI or LSI circuits of a typical minicomputer.

c. Conclusions Regarding Self Test

(1) A Sequential machine provides a good model for representing failed digital devices.

(2) A self test procedure, if it is to be efficient and practicable, must take advantage of the unique structure of the device being tested.

(3) Data must be obtained regarding

(a) the failure modes of typical digital circuits which comprise the flight control computer

(b) the probabilities of occurrence of the failure modes of these circuits.

(4) Validation procedures must be devised to validate a self test algorithm.

In the absence of comprehensive failure data we cannot, at the present time,

- Define the general requirements of a self test procedure in terms of minimum length of input sequence, real time or memory requirements;

- Estimate, with any precision, the efficiency of a self test algorithm when that efficiency approaches 100%;

3. Breadboard Hardware Validation of a Self-Test Program

In the preceding section it was shown that a digital, sequential circuit could be represented by a sequential machine. The sequential machine representation leads to the conclusion that, if no advantage were taken of the unique structure of the device, then the number of inputs required to completely test the device was so large as to render the test impractical. As a consequence, we must settle for something less than a complete test. We cite several factors which give cause for optimism:

- The flight control computer consists of many types of combinatorial and sequential circuits whose inputs and outputs are directly accessible for fault-diagnosis.* Most of these devices are relatively easy to diagnose by self test algorithms.

*A complete tabulation of microcircuits for the Central Processor of the Bendix BDX Digital Computer is presented in Table VII-3.

- Failure rates of hard-to-test failures of a digital device may be acceptably small.

- Failure rates of hard-to-test devices may be acceptably small.

We know, for example, that the most commonly encountered failures are

- Stuck-at input or output bits.

- Stuck-at internal variables which prevent transitions to certain states (e.g., a stuck-at bit of a storage register).

A complete test for these failures can be achieved by forcing each of the variables to a "1" or "0" state. Failures of this kind occur much more frequently than all other failures combined. As a consequence, it may be said that their detection is the primary objective of almost all self test algorithms.

a. Self Test Program

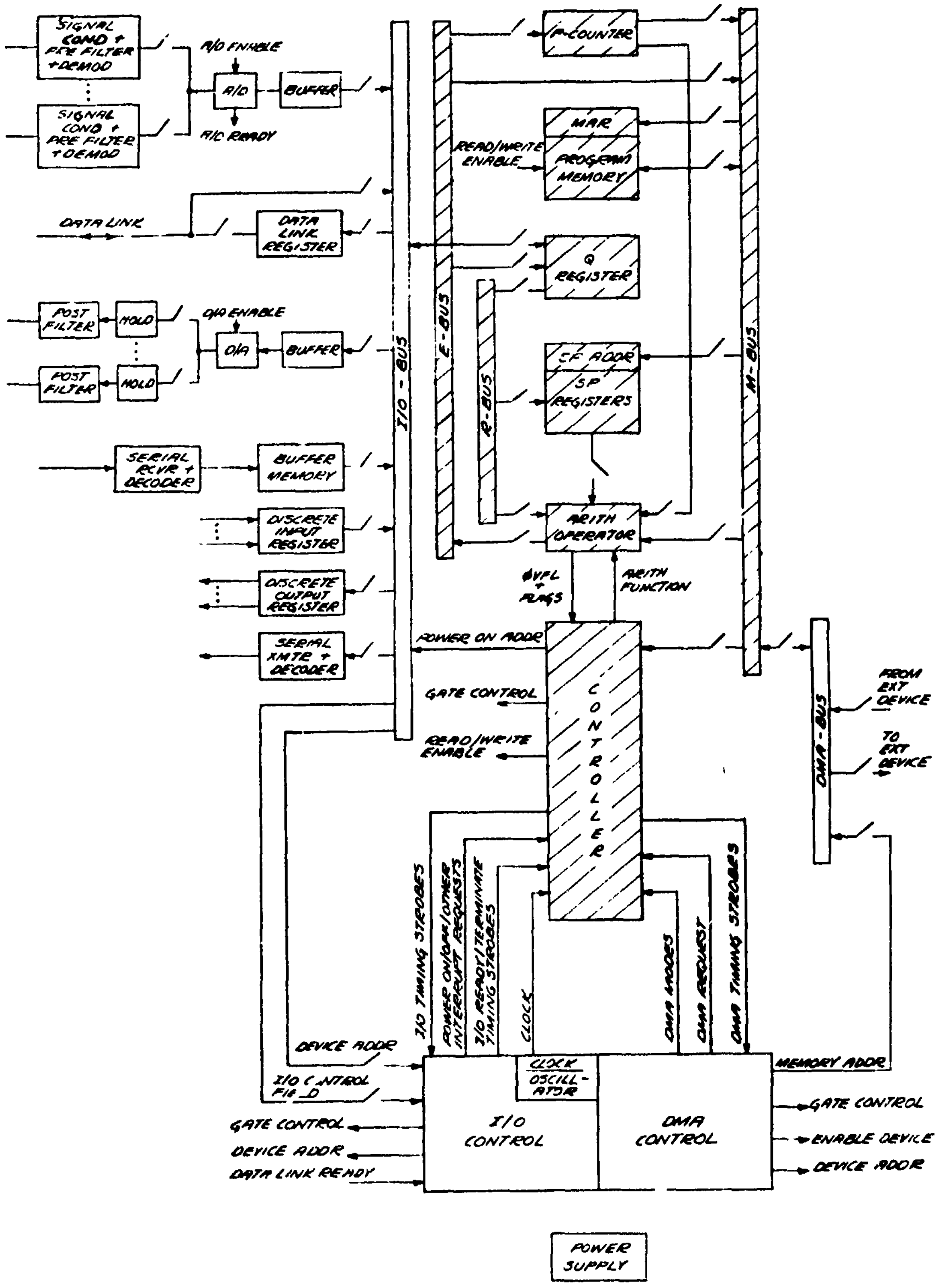
In the next section, we will describe a breadboard set-up which was designed expressly for this study for the purpose of validating a self test program such as might be used in an airborne flight control computer. Because of the similarity of parts and structure of most single address mini-computers, the results of the study are applicable to a wide class of computers.

Because of its availability and also because it has given good service, it was decided to use a software program* which is used to test all of the BDX model computers. A detailed description of the program is contained in Supplement B. Briefly, the program tests all computer busses (except the I/O, DMA busses), instructions, all 16 registers of the scratch pad memory (RAMS), that portion of the main memory containing the self test program, that portion of the program counter which is necessary to address the memory locations containing the self test program, arithmetic operator and the "Q" register. The portion of the computer which is tested is cross-hatched in Figure VII-8. We emphasize that this particular program is not designed to detect failures of the I/O and associated devices such as converters, multiplexers, I/O timing strobes, etc. The self test program requires 1,128 memory words and requires 8,600 memory cycle times to make one complete pass. At the rate of one microsecond

*The program was devised by Mr. T. Weilbacher of Bendix.

Table VII-3
Microcircuits of the Bendix BDX 900 Digital Computer

	MICROCIRCUIT TYPE	DESCRIPTION	CLASSIF-ICATION		QUANTITY	
				ARITH. UNIT	CONTROL UNIT	TOTAL QTY.
1	LM111D	VOLTAGE COMPARATOR	ANALOG		1	
2	936	DTL HEX INVERTER	SSI	1		
3	949	DTL QUAD GATE	SSI	4		
4	CH1032-1D	MOS ROM CLOCK DRIVER	HYBRID		2	
5	3111	ROM (MOS)	LSI		1	
6	3112	↓	↓		1	
7	3113	↓	↓		1	
8	3114	↓	↓		1	
9	3115	↓	↓		1	
10	4009	HEX BUFFER, (MOS)	SSI		9	
11	4609	ADDER/MULTIPLEXER } Spec.	MSI	8		
12	4611	DUAL FF/MULTIPLEXERS }	MSI	8		
13						
14	5400	TTL QUAD GATE	SSI	3	5	
15	5402	TTL QUAD GATE	↓	2	2	
16	5403	TTL QUAD GATE	↓	4		
17	5404	TTL HEX INVERTER	↓	4		
18	5405	TTL HEX INVERTER	↓	2		
19	5410	TTL TRIPLE GATE	↓		1	
20	5437	TTL QUAD GATE/BUFFER	↓	2		
21	5473	TTL DUAL JK FF	↓	2		
22	5474	TTL DUAL D FF	↓	2	3	
23	5475	TTL QUAD LATCH	↓	1	2	
24	5486	TTL QUAD GATE	↓	1	1	
25						
26	9309	TTL DUAL MULTIPLEXER	MSI	4	3	
27	9312	TTL MULTIPLEXER	MSI		2	
28	9316	TTL 4-BIT COUNTER	MSI	4	1	
29						
30	8250	TTL BINARY/OCTAL CONV.	MSI		1	
31	8266	TTL QUAD MULTIPLEXER	MSI		2	
32	8270	TTL 4-BIT SHIFT REG.	MSI	1		
33						
34	31013	64-BIT RAM	LSI	4		
				57	43	



DIGITAL COMPUTER AND ASSOCIATED I/O

FIGURE VII-8

per cycle time, a complete pass requires 8.6 milliseconds. If a fault is not detected the program halts with the program counter equal to $(251)_8$, any other result indicates that a fault was detected. If a fault is detected and if it can exercise sufficient control, the program halts with the program counter equal to $(151)_8$ having first loaded a code into one of the accumulators which identifies the area of the failure.

b. The Breadboard

In order to obtain the maximum information in the shortest time, the validation was confined to a restricted class of failures which included grounded input and output modes and when it was non-destructive and an inverter was accessible, simulated shorts to the supply voltage. The eventual extension of the procedure to include the entire class of stuck-at failures was a paramount consideration, however, and it was understood that the present effort was the first step toward achieving this objective. Altogether, 350 pins, representing the entire complement of accessible nodes, were "failed". After each failure was injected the self test program was initiated and the results tabulated. In the following paragraphs a detailed description of the procedure, hardware, and results is given.

The test was conducted by grounding all input and output nodes, one at a time. However, this did not result in the grounding of each individual input to a device since, frequently, a single node fed two or more inputs via gating circuitry. As a result, the grounding of certain nodes actually resulted in the simultaneous failing of some inputs to a high (if the intervening gate was an inverter) and some inputs to ground. It was not considered advisable to fail all nodes to a high since this could have caused the destruction of an "upstream" gate if a buffer did not intervene.

It appears that, with the proper hardware, this approach can be extended to include the following types of failures:

(1) Input and Output Nodes

(a) always high

(b) always low

(2) Input Nodes

(a) opens (=high in TTL, DTL)

(b) input diode (DTL) short-circuit

(c) emitter-base junction (TTL) short-circuit

(3) Common Package Failures

(a) device ground lead open

(b) device Vcc lead open

There is another class of failures which are extremely difficult to simulate and, at present, no method has been proposed for doing so. These failures are:

(4) Internal Logic Failures of Devices

These failures result in a restructuring of the internal state and transistion branches. As a result, the failure will not be seen at the output until a certain and unknown combination of input and internal state occurs.

Grounding failures are easy to introduce since TTL/DTL outputs may be grounded safely for many seconds. Therefore, it is not necessary to break a conductor path to simulate an open since a ground can be used, instead. Forcing a node to a high can be a problem because of its destructive effect, as noted previously. Forced highs could be introduced with a sequence generator such that the forcing is applied no longer than necessary. Forcing for 30 or 40 milliseconds should not produce undue device degradation while being, at the same time, of adequate duration for the test.

Since both of the above methods do not require physically breaking a wire, no special preparation, other than the generator, is necessary other than providing access to the circuit boards.

Further testing requires opening microcircuit leads and is best accomplished on a specimen machine constructed with sockets for the microcirucits. This approach permits the simulation of failures 2a (input opens), 3a (ground leads open) and 3b (Vcc leads open).

c. The Test

In the fail-to-ground testing routine the two cpu cards of the BDX-900 digital computer were mounted in a test fixture connected to a laboratory core memory and test console. A paper tape reader was used to load the test program. A "DIP-CLIP" test point adaptor was clipped on each DIP and a "normally open" push-button switch was connected to ground each output (or input) to the ground pin of that device.

The test procedure was as follows:

(1) Manually load Bootstrap Loader program into memory using the manual console.

(2) Load self-test program from punched tape into memory.

(3) Set program register to first step of self-test program and initiate computer 'RUN'.

(4) Computer must halt with the program counter equal to 251 indicating self-test was executed correctly.

(5) Connect push-button to pin to be grounded but do not press button. Repeat steps 3 and 4, checking that the push-button has not disturbed the circuits.

(6) While holding the push-button depressed, run the self-test program (steps 3 and 4). If the computer halts with 251 in the program counter, the fault was not detected. Any other result, including no halt or refusal to run, indicates fault detection.

(7) Release push-button.

(8) Record results.

(9) Repeat steps 3 and 4. If a No-Go results, at least a portion of the self-test program in memory has been altered. Repeat steps 2, 3, and 4.

(10) If step 2 will not run, the bootstrap loader program has also been altered. Repeat steps 1, 2, 3, and 4.

(11) Upon receiving a GO, go back to step 5 using next point to be tested.

The self-test program used was designed to test all macro instructions ex:ept the I/O instructions and those skip instructions associated with external signals. All micro words within the micro memory are executed at least once except those associated with "power on", "test set", interrupt and the above mentioned macro instructions.

d. Results

Signals I00* through I15* constitute the data buss to the console. The apparent detection of failures on these points is related to console functions external to the CPU. This group of 16 signals should not be considered tested within the scope of this self-test program.

Signals P11, P12, P13, and P14 represent higher order program counter bits, while TC2 represents the ripple carry from P11 to P12 (since four bit counter/register chips are used, only every fourth carry is available). These five signals represent those signals which are within the scope of the present self-test program and which should be tested but are not. It would be a relatively simple matter to add to or change the program to pick up these points, but for present purposes they illustrate the point.

The score card then reads:

	Control Unit	Arithmetic Unit	Total
Total Nodes Tested	193	185	378
Interrupt Nodes	3	1	4
Manual Halt Nodes	3	0	3
I/O Nodes	10	16	26
Valid Nodes Tested	177	168	345
Nodes not detected	0	5	5
*Efficiency (Node Ground Fault)	100%	97.0%	98.55%

As indicated, the upper program counter bits could be checked by adding to or modifying the self-test program to utilize these upper addresses. This indicates the value of this program testing technique in developing effective self-test.

*If all failures are equi-probable, then this quantity corresponds to test coverage.

e. Summary of Test Validation Procedure

- The hardware validation procedure can be extended to include a large class of frequently encountered stuck-at failures.

- As conducted, the validation did not exercise the full potential of the self-test algorith. For instance, the self-test checks the main memory by a memory sum test and "walks" 1's through 0 fields and 0's through 1 fields in the scratch pad.

- Internal logic failures are difficult to simulate. Work is being done in this area.

- Failure modes of common digital devices should be recorded, as they occur, in order to maintain a continuing record. Design defects should be distinguished from actual failures.

- Probabilities of device failures should be estimated from actual field data.

- From the above data realistic failure modes of digital devices can be estimated. Failure modes with a high probability of occurrence should be recognizable by the self-test algorithm.

SUPPLEMENT A

In order to illustrate the problems connected with the fault diagnosis of a sequential machine we consider the following simple sample:

The original and non-failed machine, M, is shown in Figure VII-9 and the failed copy, M', in Figure VII-10.

The fault diagnostician is presented with machine, M', for diagnosis. He is to determine that M' is or is not equivalent to M by introducing a sequence of inputs into M' and observing the outputs. He does not know what the initial state of M' is. However, he can assume that

a. number of inputs = 2

b. number of outputs = 2

c. number of states = 2.

It would appear that it is sufficient to test each branch of M' as though it were identical to M. The following sequence will accomplish this purpose if M is initially in state, S_1:

x_1, x_2, x_1, x_2

If the sequence is repeated, just for good measure, then we would observe the response

$x_1, x_2, x_1, x_2, x_1, x_2, x_1, x_2$

$y_1, y_2, y_1, y_1, y_1, y_2, y_1, y_1.$

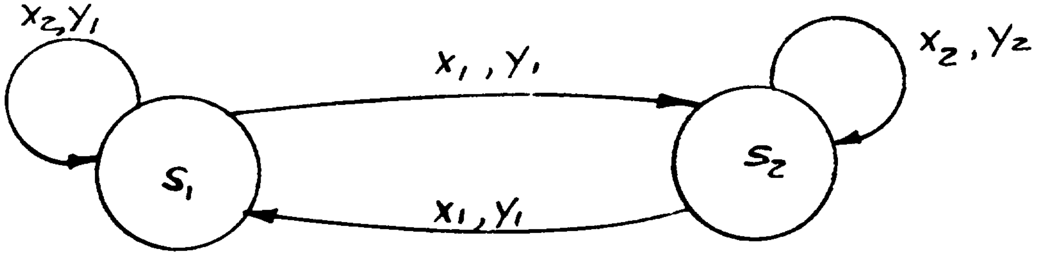
We suppose that the failed copy is initially in state, S_1, when subjected to the above sequence. Then we would observe, as the reader can verify from the state diagram:

$x_1, x_2, x_1, x_2, x_1, x_2, x_1, x_2$

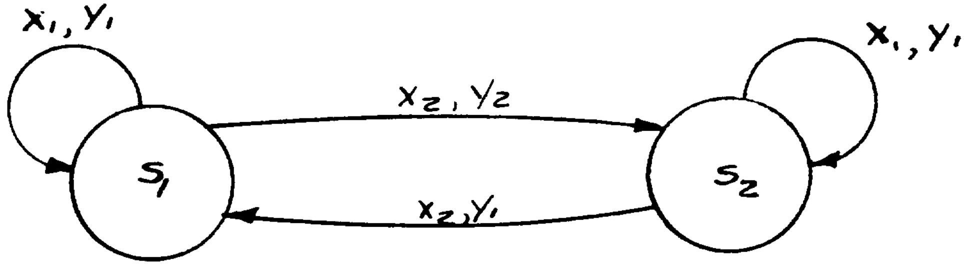
q $y_1, y_2, y_1, y_1, y_1, y_2, y_1, y_1$

The response is the same as would have been obtained from M!

If the tester had been lucky he would have tried the sequence x_2, x_2. The response of M would have been:



NON-FAILED MACHINE = M
FIGURE VII-9



FAILED COPY OF M = M'
FIGURE VII-10

initial state = S_1: x_2, x_2

y_1, y_1

initial state = S_2: x_2, x_2

y_2, y_2

But the response of M' would have been:

initial state = S_1: x_2, x_2

y_2, y_1

initial state = S_2: x_2, x_2

y_1, y_2

Thus, for machine M' an input sequence of length 2 would have been sufficient to distinguish between M' and M.

SUPPLEMENT B

SELF-TEST PROGRAM DESCRIPTION

INSTRUCTION SET

MNEMONIC	INSTRUCTION
ADD	ADD
SUB	SUBTRACT
CMP	COMPARE
LOAD	LOAD
STO	STORE
JU	JUMP
JSA_K	Jump and Mark in A_K
JSM	Jump and Mark in memory
ADDR	Add registers
IAR	Immediate add to register
SUBR	Subtract registers
CMPR	Compare registers
MPY	Multiply registers
DIV	Divide registers
TRA	Transfer
IR	Interchange registers
AND	AND
OR	OR
LCM	Logical complement
ACM	Arithmetic complement
CLA	Clear register
CLAØ	Clear register and overflow
SLSL	Shift left short logical
SRSL	Shift right short logical
SLSA	Shift left short algebraic
SRSA	Shift right short algebraic
RLS	Rotate left short
SLLL	Shift left long logical
SRLL	Shift right long logical
SLIA	Shift left long algebraic
SRLA	Shift right long algebraic
RLL	Rotate left long
DECEQ	Decrement and skip if zero
DECNE	Decrement and skip if not zero
SKGT	Skip if > 0
SKLE	Skip if < 0
SKGE	Skip if ≥ 0
SKLT	Skip if ≤ 0
SKEQ	Skip if $= 0$
SKNE	Skip if $\neq 0$
SSOV	Skip if overflow set
SROV	Skip if overflow reset
SSIE	Skip if interrupt enable set

MNEMONIC	INSTRUCTION
SRIE	Skip if interrupt enable reset
SSF1	Skip if flag 1 set
SRF1	Skip if flag 1 reset
SSF2	Skip if flag 2 set
SRF2	Skip if flag 2 reset
STIR	Skip if interrupt request time
SFIR	Skip if interrupt request false
STE1	Skip if external 1 true
SFE1	Skip if external 1 false
STE2	Skip if external 2 true
SFE2	Skip if external 2 false
STE3	Skip if external 3 true
SFE3	Skip if external 3 false
SET	Set indicators
RESET	Reset indicators
FLIP	Complement indicators
CONT	Control indicators
NOP	No operation
HALT	Halt
OD	Output data
OSR	Output data skip if ready
ID	Input data
ISR	Input data skip if ready
OC	Output control
ISW	Input switch register

1.0 INTRODUCTION

The BDX900 self-test consists of a self-test program, to be loaded into the BDX900 computer memory and then executed.

The self-test program is designed to test all macro instructions except the I/O instructions and those skip instructions associated with external signals. All micro words within the micro memory are executed at least once except those associated with 'power on', 'test set', interrupt and the above mentioned macro instructions.

2.0 DESCRIPTION

The self-test program consists of 24 blocks or sections. Figure VII-11 shows a memory map of the program and the BDX900 assembler print-out shows the actual program. Each of the blocks is described below:

Block 1 - This section consists of temporary storage locations and constants used by the self-test program.

Block 2 - This section contains the sequence control instructions that direct the self-test program through the various test sections and cause the computer to halt when an error occurs.

Block 3 - This section contains the memory test. The test consists of forming a running sum of the contents of all memory locations used in the self-test program. The final sum is compared with a stored constant.

Block 4 - This section contains instructions that interrogate bit 14 of the test set switch register. If bit 14 = 1 the indirect level test is entered. If bit 14 = 0 the indirect level test is by-passed.

The indirect level test attempts to execute all memory reference instructions using sixteen levels of indirect addressing. Correct execution of each instruction causes the computer to come to a halt with the indirect light in the 'on' state.

Address (octal)	Memory Contents	Address (octal)	Memory Contents
000-141	Temporary Storage Contents Memory Block 1	1101-1167	Test #8 Memory Block 13
142-156	Self-Test Program Sequence Control Instructions Memory Block 2	1170-1234	Test #9 Memory Block 14
157-171	Memory Test Memory Block 3	1235-1254	Test #10 Memory Block 15
172-244	Indirect Test (optional) Memory Block 4	1255-1340	Test #11 Memory Block 16
245-377	Halt or Recycle (optional) Memory Block 5	1341-1416	Test #12 Memory Block 17
400-447	Test #1 Memory Block 6	1417-1514	Test #13 Memory Block 18
450-564	Test #2 Memory Block 7	1515-1552	Test #14A Memory Block 19
565-630	Test #3 Memory Block 8	1553-1626	Test #14B Memory Block 20
631-654	Test #4 Memory Block 9	1627-1663	Test #14C Memory Block 21
655-776	Test #5 Memory Block 10	1664-1754	Test #15 Memory Block 22
777-1023	Test #6 Memory Block 11	1755-2074	Test #16 Memory Block 23
1024-1100	Test #7 Memory Block 12	2075-2154	Test #17 Memory Block 24

SELF-TEST MEMORY MAP
FIGURE VII-11

Block 5 - This section contains instructions that interrogate bit 15 of the test set switch register. If bit 15 = 0 the self-test program is executed once and halted. If bit 15 = 1 the self test program is recycled and continuously executed until the switch is thrown to the zero state.

Block 6 (Test 1) - This section is called Test 1 and is the first section entered when the self test program starts execution. Those instructions associated with error detection are partially tested until it is determined that they work sufficiently well for that purpose. The instructions partially tested are:

- CMP (BASE ADDRESS), OVERFLOW NOT TESTED
- LOAD (BASE ADDRESS)
- IAR, OVERFLOW NOT TESTED
- JAS1 (BASE ADDRESS)

a) The "CMP" instruction is partially tested by forming values in accumulators and comparing them against stored constants. All three conditions are tested - greater than, equal to, and less than. The overflow condition is not tested at this time. Only direct, base page addressing is tested.

b) The "LOAD" instruction is tested by loading an accumulator and then comparing against a stored constant. Again only direct, base page addressing is used at this time.

c) The "IAR" instruction is partially tested by incrementing and decrementing an accumulator and comparing the result in each case against a stored constant. Testing of the overflow condition and incrementing and decrementing by larger amounts are deferred to a later section.

d) The "JSA1" instruction is tested in conjunction with the "CMP" instruction by placing the "JSA1" instruction in the proper skip location following the "CMP" instruction. If the compare was executed properly a skip will be made to a "JSA1" instruction which in turn causes a jump to a location where an "IAR" instruction is stored. The "IAR" instruction causes the contents of an accumulator to be incremented thereby building up a check-sum. At the end of test 1, the check-sum is compared against a stored constant to determine if each "JSA1" instruction caused a jump to its proper location.

The address stored in accumulator A1 by the execution of a "JSA1" instruction is also compared against a stored constant.

Only direct, base page addressing is used at this time for the "JSA1" instruction.

Block 7 (Test 2) - This section completely tests the control and skip on indicator instructions. In this section as well as all the other test sections, no instruction is executed within a section unless that instruction has been previously tested in past sections or unless that instruction is presently under test. The instructions tested in this section are:

a) CONT
b) SSOv
c) SROV
d) SSIE
e) SRIE
f) SSF1
g) SRF1
h) SSF2
i) SRF2

The control and skip on indicator instructions are tested by setting the overflow, interrupt enable, flag 1 and flag 2 flip-flops and then attempting to skip on the respective flip-flops being reset as well as being set.

The above is repeated after the flip-flops have been toggled and repeated after the flip-flops have been reset.

Each time a skip is made an accumulator is incremented thereby developing a check-sum which is compared against a stored constant at the end of the test to insure that all skips are made to the correct location.

Block 8 (Test 3) - The complementing instructions LCM and ACM are tested in this section.

The "LCM" and "ACM" instructions are tested by complementing a known value of an accumulator and transferring the result back to the same accumulator and also complementing the value in one accumulator and transferring it to a second accumulator. The results are always compared against stored constants.

In addition, accumulators A0, A1, A2 and A3 are tested in the process because each bit of these accumulators contains a "1" and a "0" during the test.

Block 9 (Test 4)- The IAR instruction is completely tested in this section including arithmetic overflow. An accumulator containing a known value is incremented and decremented using "IAR" instructions such that both the overflow and no overflow condition are generated while incrementing through positive values and also while decrementing through negative values.

The final sum is compared against stored constants.

Block 10 (Test 5) - The TRA, ADDR, and SUBR instructions are tested in this section (including arithmetic overflow). The accumulator registers A4 through A15 are tested by placing a "1" and "0" into every bit of every register. This is accomplished by using the "ADDR", "TRA" and "LCM" instructions. In the process, the "TRA" instruction is tested.

The "ADDR" and "SUBR" instructions are tested including arithmetic overflow by generating predetermined sums and differences that produce overflows and no overflows. Skips are then made on the state of the overflow flip-flop. The sums and differences are compared against stored constants.

Block 11 (Test 6) - The ADD (BASE ADDRESS) and SUB (BASE ADDRESS) are tested (including arithmetic overflow) in this section.

Predetermined sums and differences are generated that produce and don't produce arithmetic overflows. Skips are then made on the state of the overflow flip-flop. The sums and differences are compared against stored constants. Only direct, base addressing mode is used at this time.

Block 12 (Test 7) - The CMPR instruction is completely tested in this section. The CMP (BASE ADDRESS) is further tested for arithmetic overflow.

The "CMPR" instruction is tested by forming predetermined values in accumulators and comparing them against stored constants. All three conditions are tested - greater than, equal to, and less than.

The overflow condition is also tested for both the "CMPR" and "CMP" instructions by generating overflows for both the "CMPR" and "CMP" and skipping on the overflow state.

Only direct, base addressing mode is used at this time for the "CMP" instruction.

Block 13 (Test 8) - The skip on accumulator instructions and the decrement and skip instructions are completely tested here. These instructions are:

a) SKGT
b) SKLE
c) SKGE
d) SKLT
e) SKEQ
f) SKNE
g) DECEQ
h) DECNE

The skip on accumulator instructions are tested for both the skip condition and the non-skip condition by loading an accumulator with a known value and attempting to skip on that accumulator.

If a skip is made on a non-skip condition the self-test program is halted. If a skip is made on a skip condition the first instruction encountered after the skip instruction is an "IAR" instruction which causes an accumulator to be incremented thereby generating a check-sum. This check-sum is compared against a known value at the end of the test to determine if all skips were made to the correct location.

The "DECEO" and "DECNE" instructions are also tested in the same manner as above including the generation of arithmetic overflows.

Block 14 (Test 9) - The logical "AND" and "OR" instructions are tested in this section.

Both instructions are executed with known values in accumulators so that each pair of bits "anded" or "ored" together will successively contain one of the four possible binary combinations.

Block 15 (Test 10) - The register interchange instruction IR is tested here along with its special case CLA.

The "IR" instruction is tested here by loading known values into two accumulators and interchanging the contents of those accumulators. The result is tested and the accumulators are again interchanged back to the initial configuration where result is again tested.

Block 16 (Test 11) - The short shift instructions are tested in this section along with overflow for algebraic left shift. They are:

a) SLLL
b) SRLL
c) SLLA
d) SRLA
e) RLL

In order to execute every micro word associated with the short shifts each shift instruction must be executed at least three times - once with a shift of zero bit positions, second with a shift of one bit position and third with a shift of more than one bit position.

Various bit patterns were used from a "one" in the word to many "one's". After each shift the result was added into a check-sum and the final check sum was compared against a stored constant.

The overflow condition for the (SLSA) instruction was tested.

Block 17 (Test 12) - The long shift instructions are tested in this section along with overflow for the algebraic left shift. These instructions are:

a) SLLL	d) SRLA
b) SRLL	e) RLL
c) SLLA	

Testing the long shifts is accomplished in the same manner as the short shifts.

Block 18 (Test 13) - The multiply instruction MPY is tested in this section.

In order to test the (MPY) instruction eight different multiply instructions were programmed in order to execute every micro word associated with the multiply instruction. After each multiplication the double length product is tested by comparing it against a stored constant.

Blocks 19, 20, 21 (Test 14A, B, C) - The divide instruction is tested in these sections. As the divide test is fairly long it is broken into three sections to provide easier entry for an operator executing the self-test on a "single instruction" basis.

Seventeen divide instructions were programmed in order to execute every micro word associated with the divide instruction. Four of these divisions were needed to test the divide overflow for the possible sign configurations of the operands.

After each divide instruction was executed the resultant quotient and remainder were either used as the operands for another divide or were added into a check-sum which was periodically compared against a stored constant.

Block 22 (Test 15) - This section tests all forms of direct addressing for the memory reference instructions. The instructions tested here are:

a) ADD	d) CMP
b) SUB	e) JSAO
c) LOAD	f) JSAI

Each instruction is tested for the four forms of addressing - base, relative to Program Counter, relative to accumulator A0, and relative to accumulator A1.

A memory constant is loaded in an accumulator via a load instruction. A second memory constant is added to the first via an add instruction followed by the subtraction of a third memory constant via a subtract instruction. The result is compared against a fourth memory constant via a compare instruction. The above is repeated for each form of addressing.

The JSAO, JSA1, and JU instructions are tested by programming jumps in the four forms of addressing.

Block 23 (Test 16) - This section tests all forms of indirect addressing for the instructions listed above in Test 15.

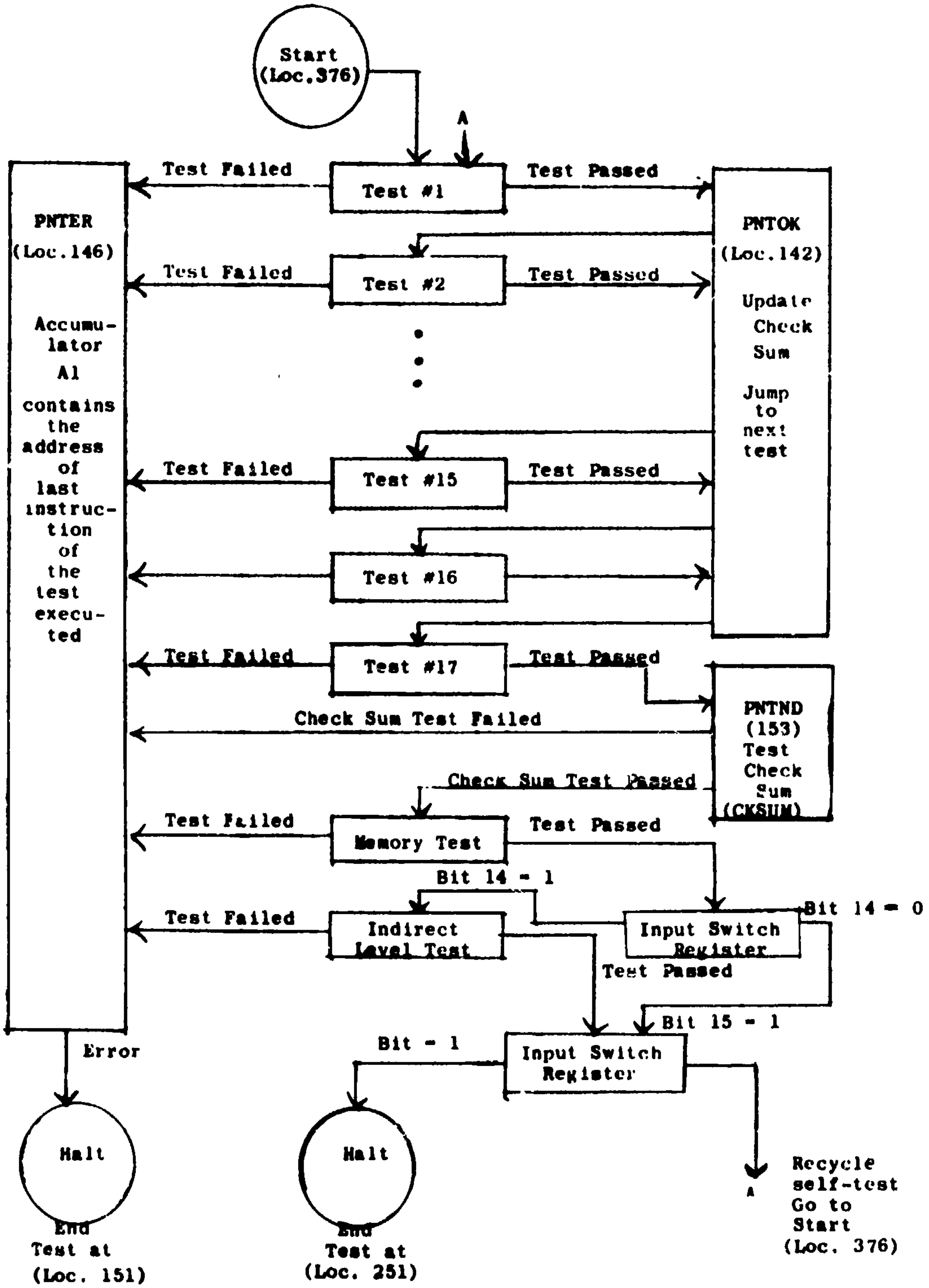
The instructions are tested as in the previous test except that all the instructions are executed with two levels of indirect addressing.

Block 24 (Test 17) - In this section the STO and JSM instructions are completely tested. This is accomplished by successively loading and storing several stored constants into an accumulator. These constants are in reality instructions. A jump is made via a JSM instruction to the first of the successively stored instructions. The short routine is executed and a jump via an indirect JU instruction is made back to the test section. The four forms of addressing are tested in both the direct and indirect mode.

3.0 THEORY OF OPERATION

In this section a description of the general flow of the self-test program will be given. Figure VII-12 shows a flow diagram.

Execution of the self-test program begins with the instruction labeled "START" which clears a check sum location to zero before entering Test 1. If an error is detected during the execution of the early portion of Test 1 before it is



SELF-TEST FLOW DIAGRAM
FIGURE VII-12

determined that the JSA1 instruction is working sufficiently well, the computer will be halted by a HALT instruction imbedded within Test 1. If an error occurs after it has been determined that the JSA1 instruction is working sufficiently well then a jump to the sequence control instruction labeled PNTER is made. At this point the computer is halted and accumulator A1 will contain the address of the instruction from which the jump was made.

If no error is detected when the end of Test 1 is reached a jump is made to the sequence control instruction labeled PNTOK where a check sum is updated followed by a jump to the first instruction of Test 2. The above process is continued from Test 2 through Test 16. If an error is detected a jump is made to PNTER where the computer is halted with the "jump out" address contained in A1. If no error is detected by the end of the test a jump is made to PNTOK where a check sum is updated followed by a jump to the next test.

Eventually, a jump is made to Test 17 via PNTOK. Detected errors are treated in the same manner as in previous tests. However, if no error is detected by the end of Test 17 a jump is made to instruction labeled PNTND where the check sum is tested. The check sum (CKSUM) is simply the cumulative sum of each address of the locations from which the jumps were made to PNTOK, plus +1. If the check sum is not correct a jump is made to PNTER with A1 containing the address of the "jump out" location. If the check sum is correct a jump is made to the memory test.

The memory test forms a running sum of the contents of all memory locations from octal (0 0 1 1) through (2 1 5 4) and compares result against a known value. If the comparison shows the sum is in error a jump is made to PNTER with accumulator A2 containing the "jump out" address. A correct comparison here indicates that the memory test and the self test has passed.

At this point the contents of the test set (console) switch register is interrogated. If bit 14 = 1 the indirect level test is executed. If bit 14 = 0 the indirect level test is passed.

Finally, the contents of the test set switch register bit 15 is interrogated. If bit 15 = 0 the computer is halted. If bit 15 = 1 the self-test program is repeated.

APPENDIX VIII

MULTIPLEX COMMUNICATIONS

Flight control systems and most particularly redundant systems generally require large numbers of communications paths between the flight control computer and the several subsystems which interface with it. Assuming that the flight control computer is a digital computer the primary communications links are:

- Digital computers to and from sensors
- Digital computers to and from digital computers
- Digital computers to and from actuators
- Digital computers to and from control and display panels

The number of links required depends upon the degree of cross-strapping required, the level of redundancy and monitoring strategy.

State-of-the-art development in the area of microminiaturization has made it possible to consider the use of multiplex communications links for data transfer. With this technique a single communications path can be shared by more than one signal, thus reducing the number of paths and the corresponding connectors in the digital computer. These and other potential benefits of multiplexing are summarized as follows:

- Reduction in wires and wire weight.
- Standardization of subsystem interfaces.
- System flexibility - modifications can be imposed on the system in the form of additional sensors or displays without the necessity for extensive rewiring.
- Reduction in connectors and pins particularly in the digital computer I/O.
- Potential improvement in EMI and EMR due to fewer wires and the use of shielded, twisted pair wires for data bussing.

It has been suggested that multiplexing results in improved detection of failures of the bus and improved detection and isolation of failure of interface units. These benefits appear to be specious because failure rates of signal paths are insignificant compared with failure rates of the subsystems. Moreover, since special interface units were introduced to accommodate multiplexing, failure detection and isolation to these units can hardly be considered an advantage. Failures of subsystems will remain as difficult to detect and isolate as formerly.

The advantages of multiplexing are considerable particularly the reduction in wires, standardization of interface and system flexibility. It may reasonably be anticipated that multiplexing will soon become a standard feature of the flight control system. At the present time, however, the cost of multiplexing may tend to offset some of these benefits. Of the enumerated benefits none appears to provide an improvement in system reliability. If anything, a reduction in reliability can be expected due to the proliferation of interface units which are required to accommodate the multiplex system.

In order to achieve standardization, it appears that each subsystem (for instance, a single sensor) will require a dedicated A/D and D/A converter, a serial transmitter and receiver with transformer coupling, encoders, decoders, clock oscillators, etc. Since these units will replace the present multiplexed computer I/O the additional cost in size, weight and dollars could be prohibitive.

In the following sections an attempt will be made to evaluate the impact of multiplexing on redundancy management. Several multiplex and dedicated communications systems will be selected for a triplex configuration and compared with respect to the following parameters:

- Bus loading
- Real time to process data
- Weight of wires and interface units
- Reliability

With the tradeoff as our goal we proceed to define the pertinent characteristics of the multiplex system.

1. Characteristics of the Multiplex System

It will be assumed that the multiplex system is a time division multiplex system as described in the Proposed Military Standard for Aircraft Multiplex Data Bus, revised July, 1973. In this system data is transferred in serial, digital pulse code modulation form. The data code is Manchester Bi-Phase Level as defined in MIL-STD-442. From the standpoint of flight control systems redundancy requirements, the following specifications regarding the multiplex system are pertinent:

a. The communications systems consists of a set of subsystems (e.g., sensors, actuators, digital computers, displays, controls) which may communicate with each other or with the digital computer via a multiplex bus.

b. The interface between each subsystem and the bus consists of:

- Multiplex Terminal Unit (MTU)
- Subsystem Interface Unit (SSIU)
- Additional subsystem electronics to interface with the SSIU

The purpose of the MTU is to interface between the bus and the SSIU. The MTU is a common element in all subsystems and consists of a transmitter, receiver, coupling transformer, clock oscillator and associated electronics. The MTU detects signals on the bus, converts from Manchester to NRZ and performs a parity check. Similarly, the MTU receives NRZ data from the SSIU, encodes the data to Manchester and transmits over the bus.

The SSIU is application dependent and may differ for each subsystem. For purposes of this study, we define a standard SSIU which may interface with a sensor, an actuator on a digital computer. The SSIU receives NRZ data from the MTU and converts it to a parallel word. The appropriate information is extracted and transmitted to the subsystem. In the reverse direction the SSIU receives parallel NRZ data from the subsystem, encodes and converts it to serial form and transfers it to the MTU. When interfacing with a digital computer the SSIU data will be gated onto an internal computer bus to be transferred to an appropriate accumulator or memory location. In the reverse direction the data is gated from the internal computer bus to the SSIU. When the SSIU interfaces with a sensor it will operate primarily (exclusively except for Bit) in the receiving mode to accept data previously converted from analog to digital form in the subsystem. The necessary electronics, including the A/D converter, is contained in the subsystem. When the SSIU interfaces with an actuator it will operate in both the transmit and receive modes. Data will be transmitted to the subsystem for D/A conversion and received from the subsystem having been previously converted from analog to digital form.

Figures VIII-1, VIII-2, VIII-3 and VIII-4 show the functional block diagram of the multiplex system, MTU, SSIU and the subsystem interface electronics, respectively.

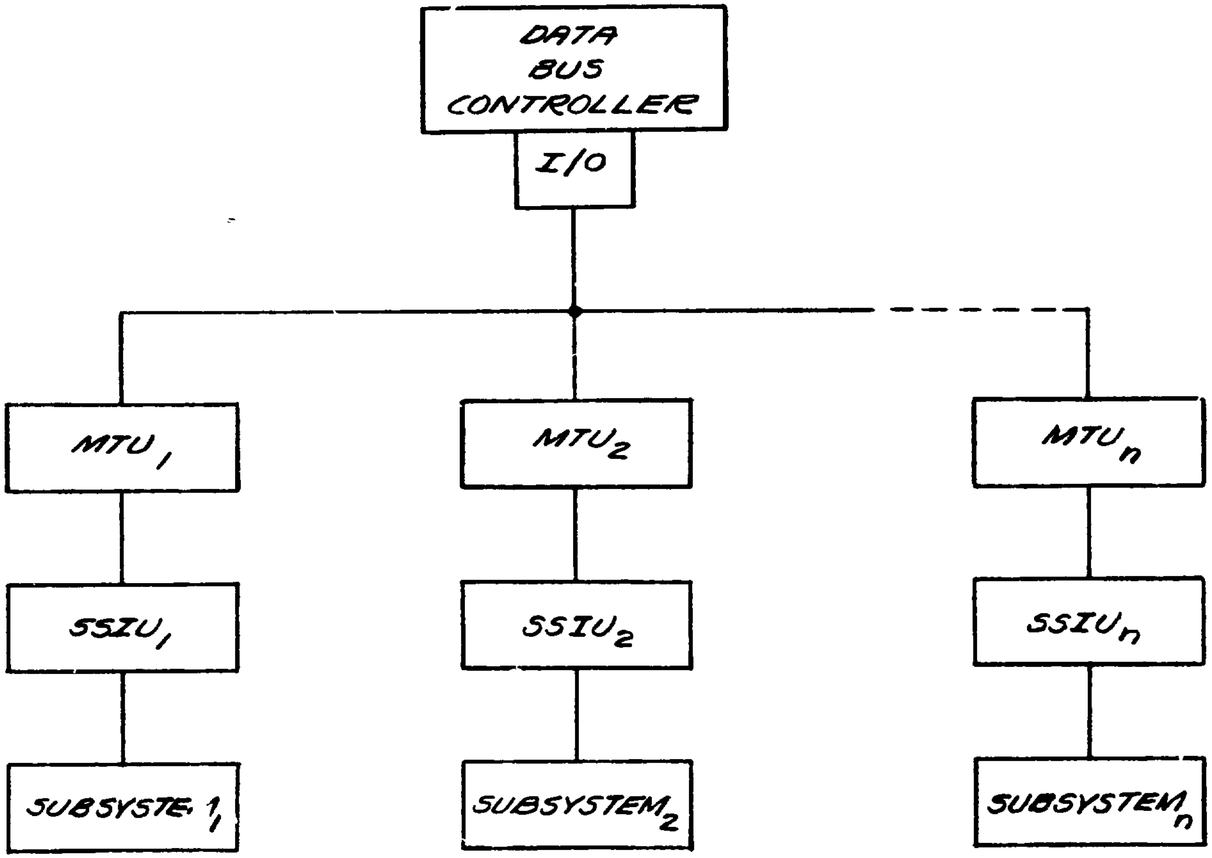
c. The bus traffic is controlled by the command/response rule according to which an MTU will respond only when commanded to by the bus controller.

d. For our purposes the bus controller will be associated with a digital flight control computer and will, if necessary, utilize applicable portions of the I/O or DMA control hardware as well as computer software.

e. The use of transformer coupling reduces the susceptibility to hot shorts and the use of stubbing prevents loss of the main bus due to "opens" at or near the terminal units. The bus, however, is susceptible to extraneous AC signals which may be injected by any transmitter on the line. Accordingly, if a single transmitter interfaces with every redundant bus, then a single failure can result in loss of the entire communications system. While this event may not be very probable (and this must be demonstrated in any case) if detection and disengage capability is provided in each subsystem the possibil.ty, however, remote, of a single failure causing loss of the ent.re system must be avoided whenever possible. As a consequence, the multiplex system will be subject to the same restrictions regarding common failures as all other subsystems. In particular, in no circumstances will a transmitter unit have access to more than one bus.

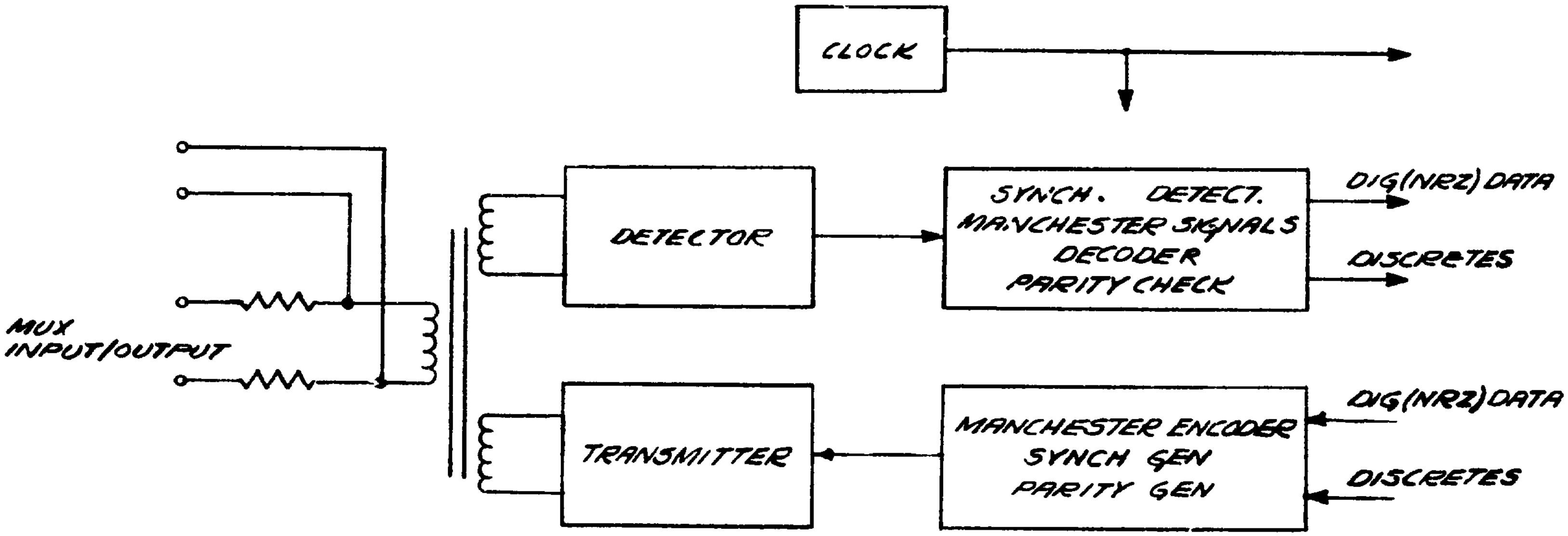
f. Each MTU will perform a self test to detect any signal transmission from itself to the data bus which has not been commanded by the bus controller. Detected failures will cause the MTU to disengage itself from the bus.

g. The data transmission rate of the bus will be one megabit per second (or, equivalently, one bit per microsecond).



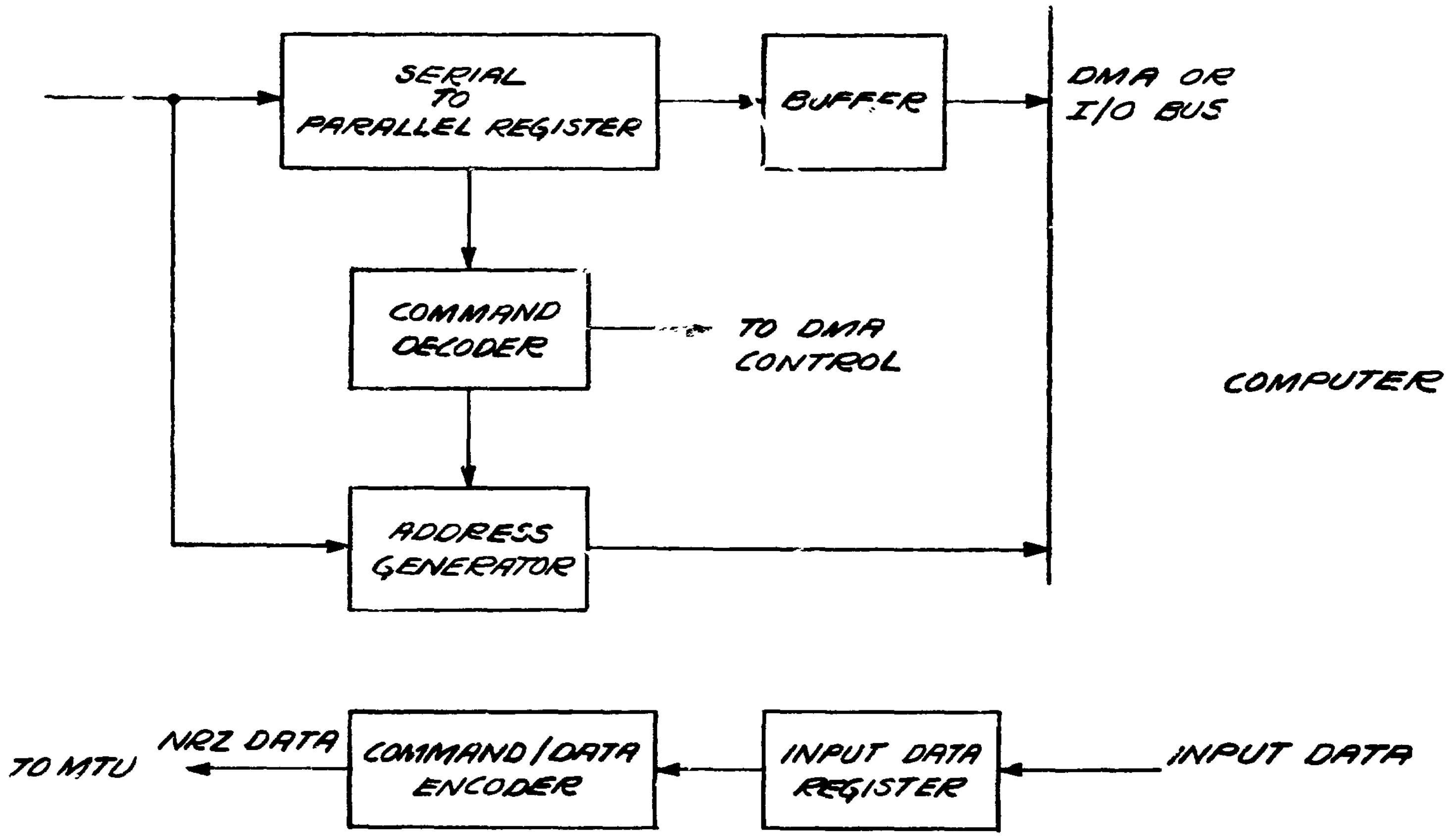
MODULAR MULTIPLEX SYSTEM

FIGURE VIII-1



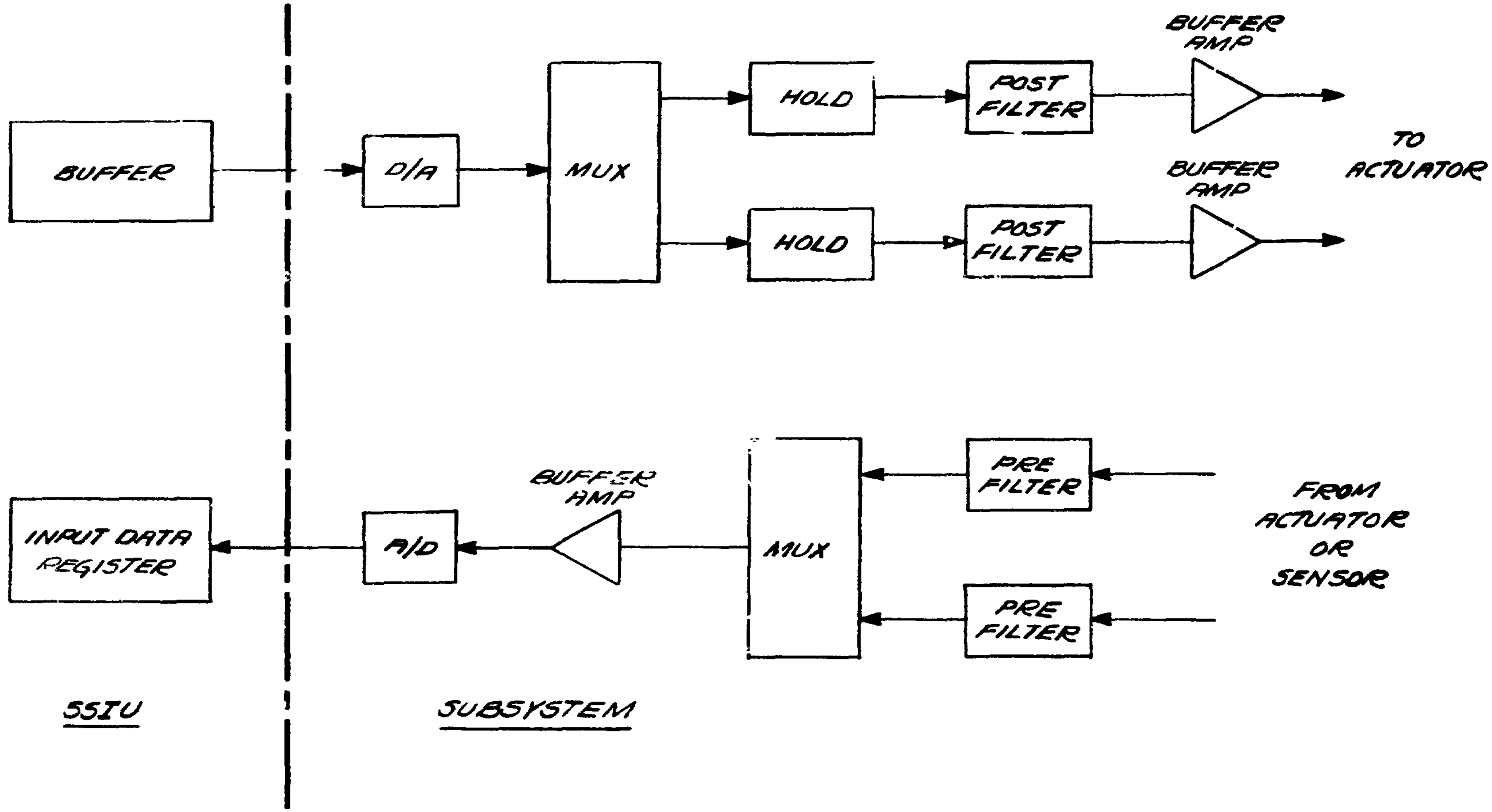
MTU

FIGURE VIII-2



SSIU

FIGURE VIII-3



ADDITIONAL SUBSYSTEM ELECTRONICS

FIGURE VIJI-4

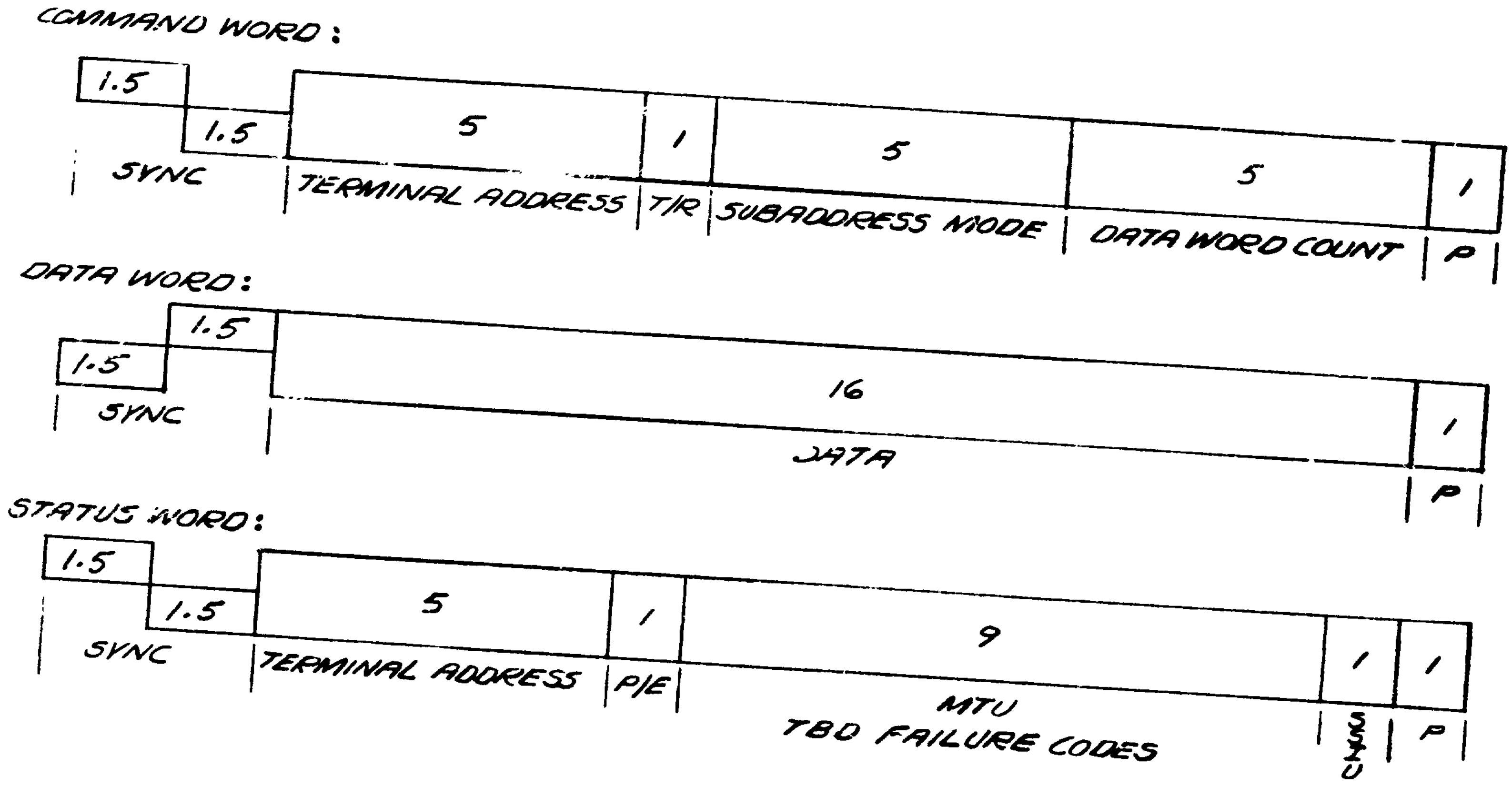
h. Data will be transmitted in words: either a command word, a data word or a status word. Each word will consist of 20 binary bits. The respective word formats are shown in Figure VIII-5.

i. A message from the bus controller to an MTU will consist of a command word to either transmit or receive data. If the command is to receive data the bus controller will then transmit the data words as specified by the word count. Upon reception of the last data word the MTU will transmit a status word back to the controller. If the command is to transmit data then the MTU will transmit a status word to the bus controller followed by the data stream as specified by the word count. In flight control applications the bus controller will request the transmission of a single word at a time, the exception being intercomputer transfers or actuator feedback variables. As a consequence, and according to the MIL Standard, three serial words would be required to transmit a single data word: a command word, the data word and a status word. Since this could result in excessive bus loading we take the liberty of eliminating the status word and reserve the unused bits of the 16 bit data word for error coding. This is justified since only 12 bits can be practicably utilized by an A/D or D/A converter at the present time. In all subsequent estimates we will assume that two, 20 bit serial words are required to transmit one data word.

2. Ground Rules for Trade-Off Estimates

In addition to the aforementioned characteristics of the communications system we postulate the following ground rules which will form the basis for the estimates to follow:

a. DMA - All digital computer input and output variables are accessed via Direct Memory Access (DMA). If the DMA is a Cycle Steal then it requires about one microsecond of real time to access a single data word. This includes both the address (contained in the command) and the data. The selection of DMA for these estimates is not necessarily a recommendation and certainly does not preclude accessing via program control. Under program control data would be requested by the program in the flight control computer. The request would require 2 and possibly 4 microseconds depending upon the location of the address field in the computer. It would then require at least 40 microseconds for the data to be returned in a form ready for access (it requires one microsecond per bit on a one megabit bus). It would then require one or possibly 2 microseconds to transfer the data to a memory location on DMA,



WORD FORMATS

FIGURE VIII-5

or to an accumulator if under program control. While it is possible, in principle, to perform other programmed computations in the interim, it is usually difficult to arrange in practice. Thus, we may assume that it requires at least 43 microseconds to access a single data word under program control.

b. Sampling Rate - The inner loop sampling rate is assumed to be 50 per second. The outer loop variables are sampled at 10 per second.

c. DMA Refresh Rate - The DMA refresh rate is 4 times the sampling rate; i.e., 200 samples per second for inner loop variables and 40 samples per second for outer loop variables.

d. Equalization - We assume that actuator equalization is required and, hence, that all actuators of the same axis require different commands.

e. Configuration - We assume for the purpose of this trade-off a triplex configuration.

f. Display, control panel and bite communications, failure and disengage logic are not included in the trade-off.

g. Sensors - We assume 15 sensors sampled at 50 per second and 15 sensors sampled at 10 per second. Thus, the inner loop sensors require 15 x 50 x 4 x 2 = 6000 serial words per second and the outer loop sensors require 15 x 10 x 4 x 2 = 1200 serial words per second. The total number of serial words required to process sensor information is 7200 serial words.

h. Actuators - We assume 4 actuators, each actuator requiring 3 words of data transmission; i.e., a command, follow-up and equalization data word. All variables are sampled at 50 per second. If a computer supplies an alternate command to another actuator then only the command data word is required, the other words being supplied via the nominal, command computer data bus. Hence, each actuator bus requires 4 x 50 x 4 x 3 x 2 = 4800 serial words per second for direct commands and 4 x 50 x 4 x 2 x 2 = 3200 serial words per second for alternate commands to the other channel actuators.

i. Intercomputer - Intercomputer communications require 15 data words at 50 samples per second to each computer. We assume that the same words are transferred to both computers. Hence, intercomputer transfer requires 15 x 50 x 4 x 2 = 6000 serial words per second. If the transfer is performed under program control then 15 x 50 x 43 = 32,250 microseconds of real time is required.

j. Actuator commands and internal actuator variables are transmitted to all computers. This permits all computers to:

- monitor actuator commands directly,
- supply actuator loop closures (if necessary) for all computers, and
- supply appropriate and possibly different commands to all computers in the event that the nominal command computer fails.

k. The following data regarding topology, wire weights and reliability of the interface units is assumed:

- Distance between digital computers is negligibly small.
- Distance from each sensor to each computer = 100 feet.
- Distance from each computer to each actuator = 100 feet.
- Sensor to computer dedicated wiring = 24 gauge, insulated, twisted pair = 0.5 lbs. per 100 feet.
- Computer to actuator dedicated wiring = 22 gauge, insulated, twisted pair = 0.8 lbs. per 100 feet.
- Multiplexed data bus wiring = shielded, twisted pair = 1.5 lbs./100 feet.
- Each MTU/SSIU weighs 0.5 lbs*x exclusive of the A/D and D/A converters required in each subsystem.

*Based on the use of SSI and MSI devices.

- A combined A/D and D/A converter and associated electronics weighs 0.4 lbs.*

- Stubbing wiring and connector weights not included.

- Failure rate of each MTU/SSIU = 10 x 10^{-6} failures per hour.

- Failure rate of an A/D converter and associated electronics = 10 x 10^{-6} failures per hour.

- Failure rate of a D/A converter and associated electronics = 10 x 10^{-6} failures per hour.

- From i, j and h the total additional failure rate to be added to each subsystem is 30 x 10^{-6} failures per hour.

- Because they are somewhat equivalent and, hence, tend to cancel each other out, weight and reliability of signal selection devices (gates) and analog voters are not included.

- Power supply requirements, including wiring, are not included in the trade-off. It can be expected that multiplex and dedicated systems will require the same number of wires for power supply. Assuming

(1) That the flight control computers will supply power to all subsystems and

(2) power is transmitted on 22 gauge insulated pairs of wire at 0.8 lbs./100 ft.

then the additional weight due to power supply wiring in both systems is 81.6 lbs. (90 sensors and 12 actuators).

*Based on SSI, MIT devices.

3. Trade-offs of Multiplex Configurations

Five multiplex and three dedicated communications systems for a triplex redundant configuration are shown in Figures VIII-6 through VIII-14. The indicated weights only include wiring, interface units and A/D and D/A converters.

CONFIGURATION I

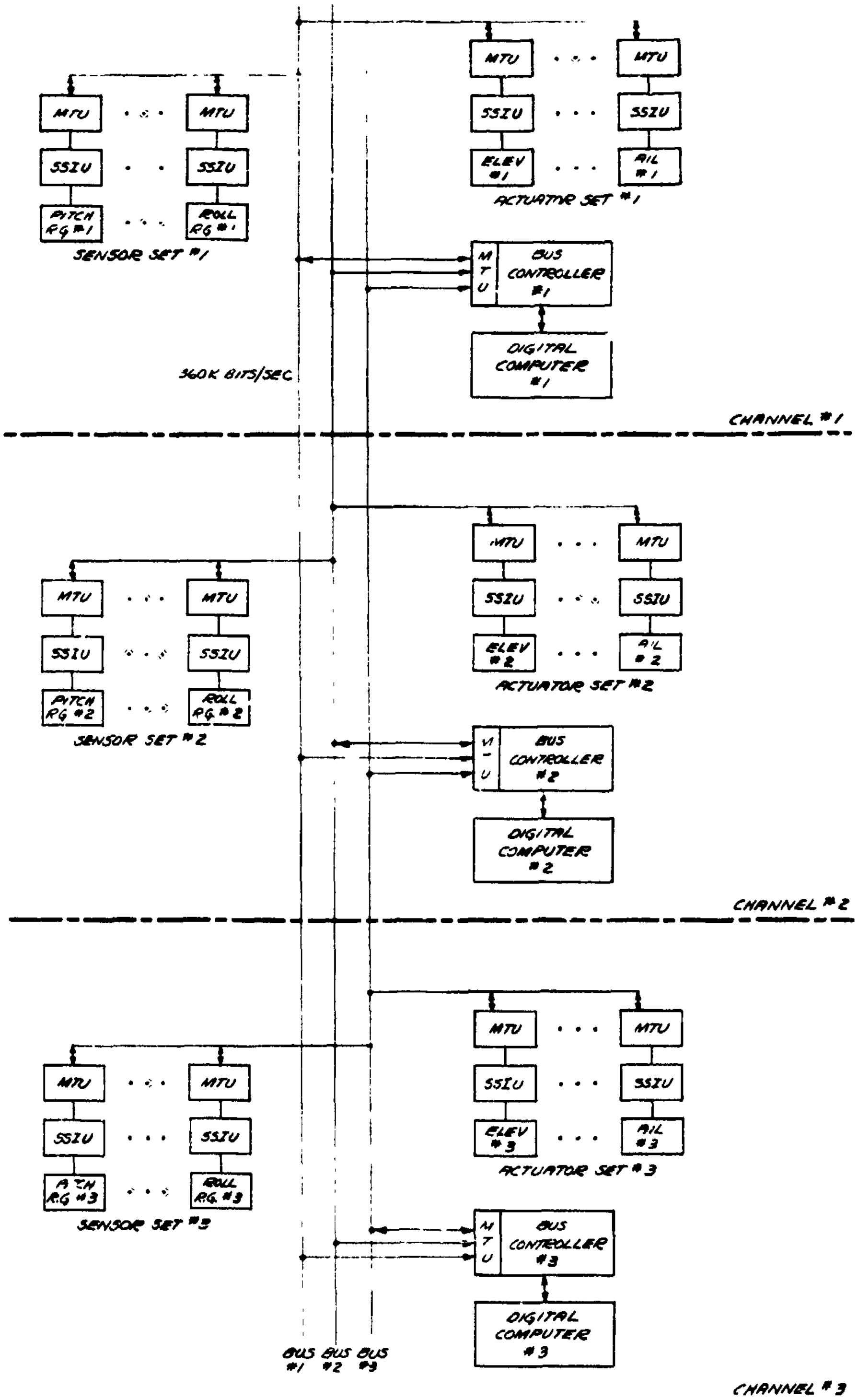
3-BUS SYSTEM
SENSOR/COMPUTER CROSS STRAPPING
NO COMPUTER/ACTUATOR CROSS STRAPPING

7,200 (SENSORS)
4,800 (ACTUATORS)
6,000 (INTECOMPUTER)
18,000 SERIAL WORDS PER SECOND PER BUS
18,000 x 20 = 360,000 BITS PER SECOND PER BUS
18,000 x 3 x.5 = 27,000 μ SEC PROCESSING TIME
= 2.7% REAL TIME
WEIGHT = 106.5 LBS.

CONFIGURATION IA

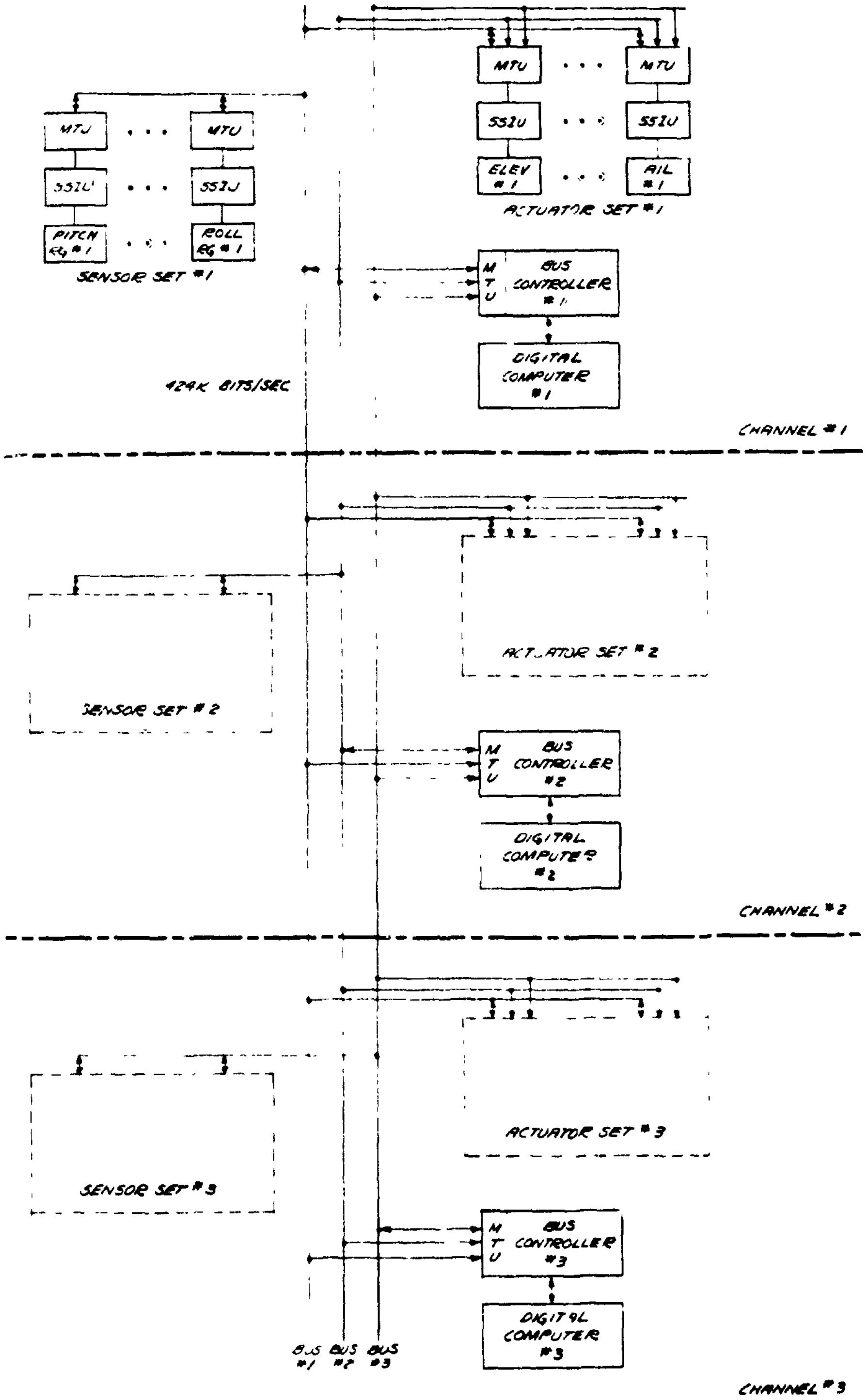
SAME AS I WITH COMPUTER/ACTUATOR CROSS STRAPPING

7,200
8,000
6,000
21,200 SERIAL WORDS PER SECOND PER BUS
21,200 x 20 = 424,000 BITS PER SECOND PER BUS
21,200 x 3 x.5 = 31,500 μ SEC PROCESSING TIME
= 3.15% REAL TIME
WEIGHT = 127.5 LBS.



3-BUS SYSTEM SENSOR/COMPUTER CROSS STRAPPING
NO COMPUTER/ACTUATOR CROSS STRAPPING
CONFIGURATION I

FIGURE VIII-6



3-BUS SYSTEM-SENSOR/COMPUTER/ACTUATOR CROSS STRAPPING
CONFIGURATION IA

FIGURE VIII-7

CONFIGURATION II

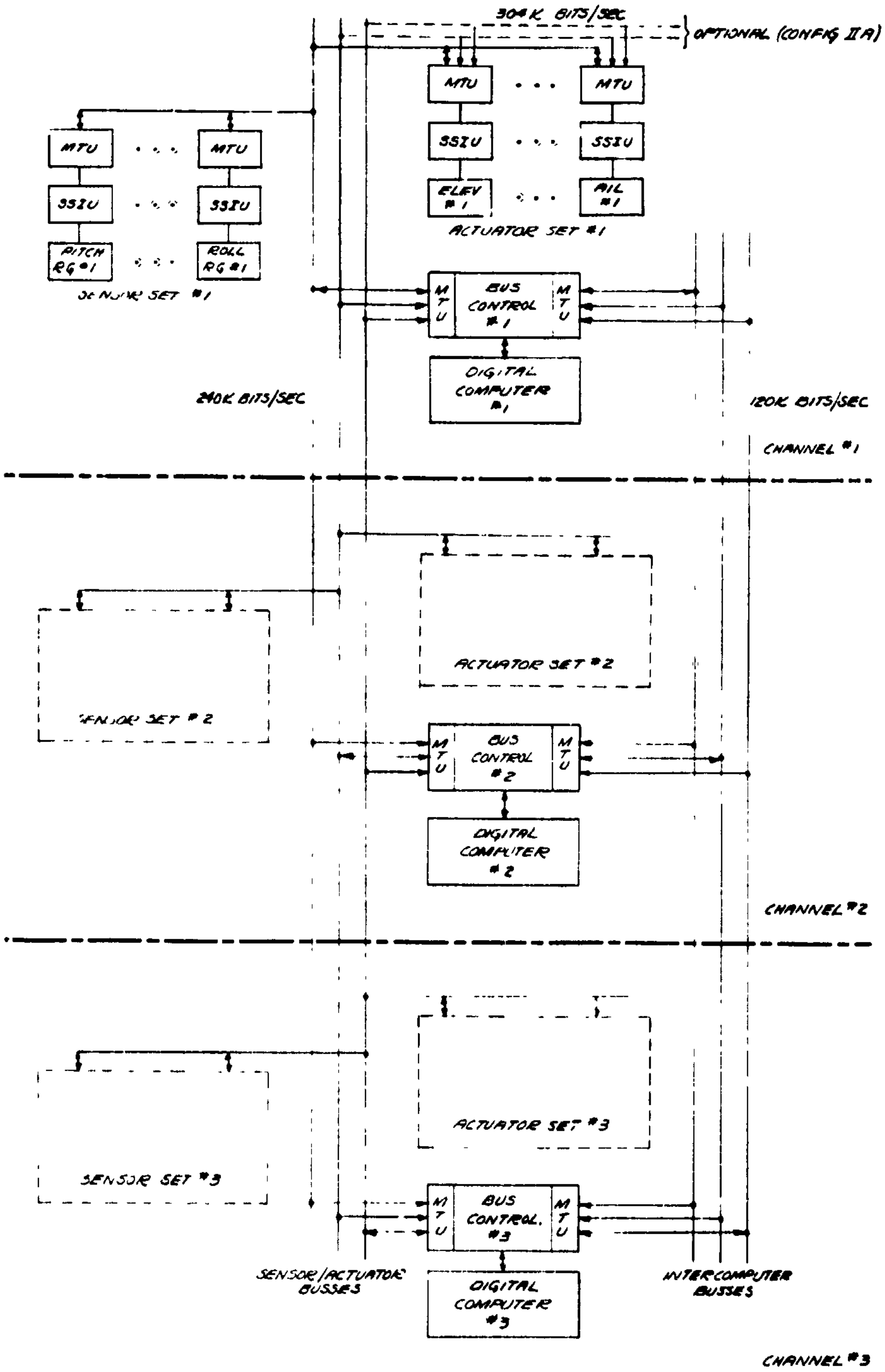
6 - BUS SYSTEM
SENSOR/COMPUTER X STRAPPING
NO COMPUTER/ACTUATOR X STRAPPING
DEDICATED INTERCOMPUTER BUS SYSTEM

7,200	
4,800	
12,000	SERIAL WORDS PER SECOND PER SENSOR/ ACTUATOR BUS
240,000	BITS PER SECOND PER SENSOR/ACTUATOR BUS
6,000	SERIAL WORDS PER SECOND PER INTER-COMPUTER BUS
120,000	BITS PER SECOND PER INTERCOMPUTER BUS
27,000	μSEC PROCESSING TIME = 2.7% REAL TIME WEIGHT = 111.0 LBS.

CONFIGURATION IIA

SAME AS II WITH COMPUTER/ACTUATOR X STRAPPING

15,200	SERIAL WORDS PER SECOND PER SENSOR/ ACTUATOR BUS
304,000	BITS PER SECOND PER SENSOR/ACTUATOR BUS
6,000	SERIAL WORDS PER SECOND PER INTER-COMPUTER BUS
31,500	μ SEC PROCESSING TIME = 3.15% REAL TIME WEIGHT = 132.0 LBS.



6-BUS SYSTEM WITH SEPARATE INTERCOMPUTER BUSSES
AND SENSOR CROSS STRAPPING
CONFIGURATION II

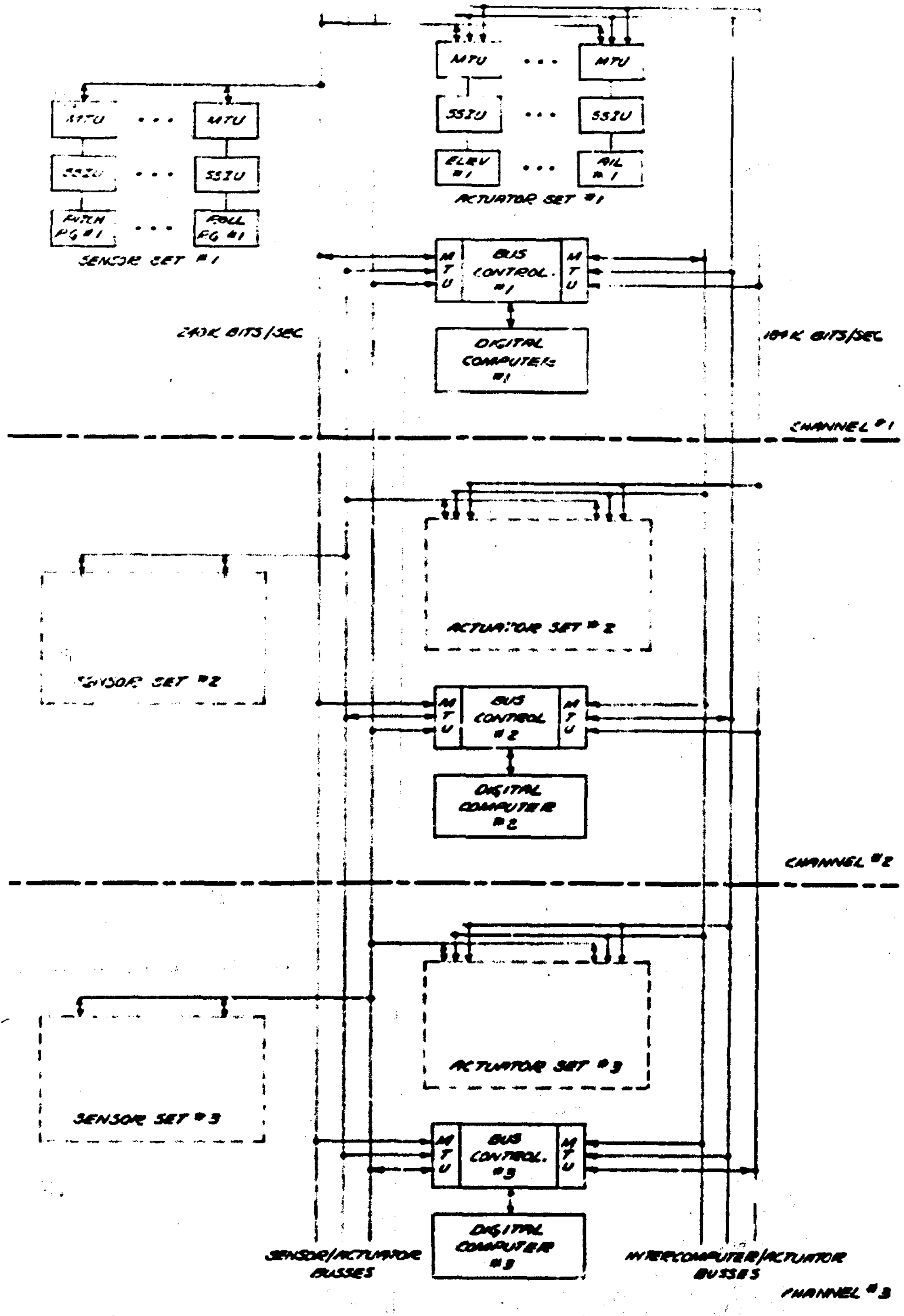
FIGURE VIII-8

CONFIGURATION III

6-BUS SYSTEM
SENSOR/COMPUTER X STRAPPING
COMPUTER/ACTUATOR X STRAPPING

INTERCOMPUTER BUSSES SUPPLY ALTERNATE ACTUATOR COMMANDS

7,200	
4,800	
12,000	SERIAL WORDS PER SECOND PER SENSOR/ ACTUATOR BUS
240,000	BITS PER SECOND PER SENSOR/ACTUATOR BUS
9,200	SERIAL WORDS PER SECOND PER INTER-COMPUTER BUS
184,000	BITS PER SECOND PER INTERCOMPUTER BUS
31,500	μ SEC PROCESSING TIME = 3.15% OF REAL TIME WEIGHT = 132.0 LBS.



6-BUS SYSTEM WITH FULL CROSS STRAPPING
CONFIGURATION III

FIGURE VIII-9

CONFIGURATION IV

6-BUS SYSTEM
SENSOR/COMPUTER X STRAPPING VIA COMPUTER/ACTUATOR BUS
NO COMPUTER/ACTUATOR X STRAPPING

7,200 SERIAL WORDS PER SECOND PER SENSOR BUS
144,000 BITS PER SECOND PER SENSOR BUS

7,200
4,800
6,000

18,000 SERIAL WORDS PER SECOND PER ACTUATOR BUS
360,000 BITS PER SECOND PER ACTUATOR BUS

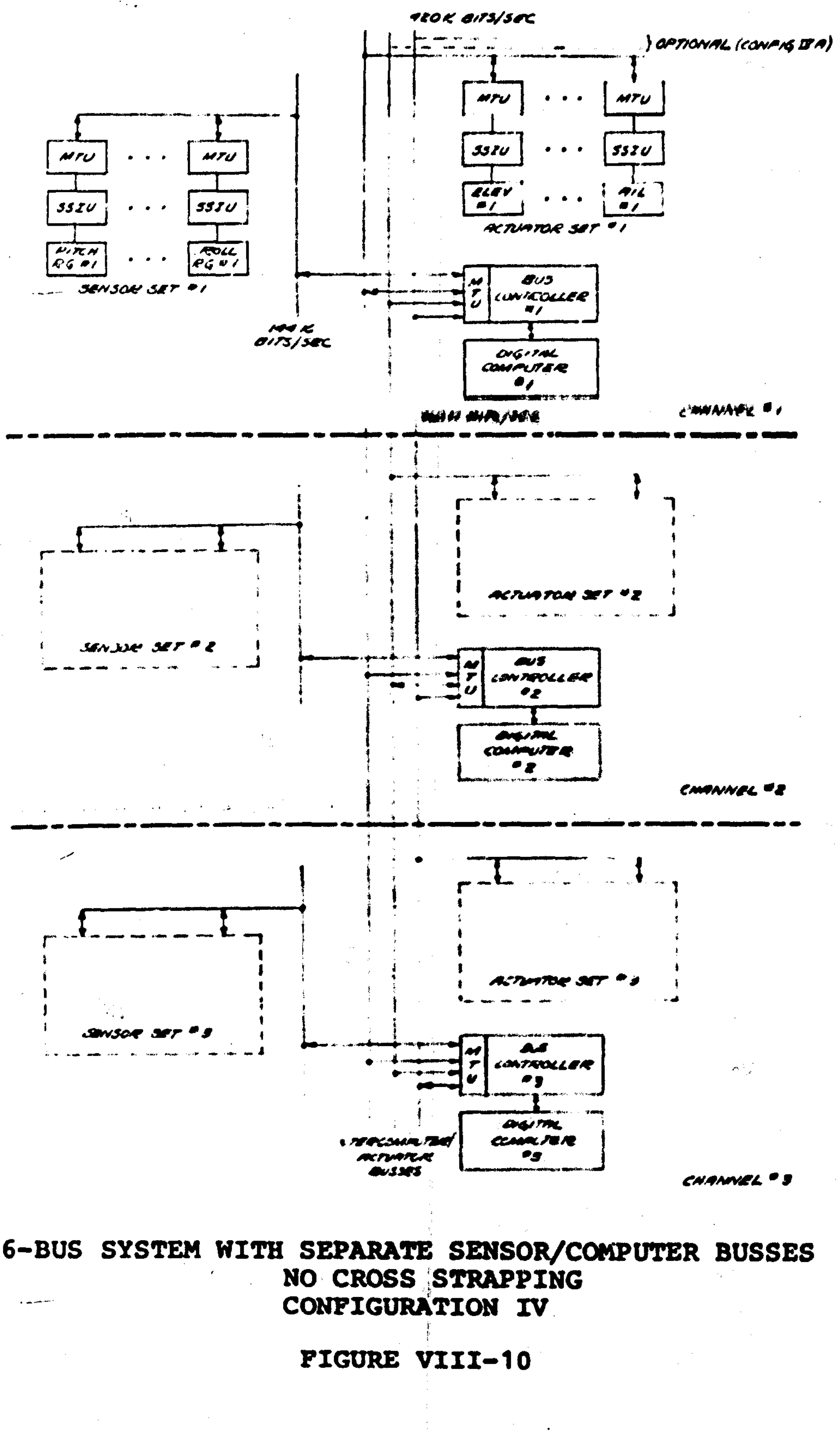
7,200
18,000

25,200 x 3 x.5 = 37,800 μSEC PROCESSING TIME
= 3.78% REAL TIME
WEIGHT = 108.0 LBS.

CONFIGURATION IVA

SAME AS IV WITH COMPUTER/ACTUATOR X STRAPPING

7,200 SERIAL WORDS PER SECOND PER SENSOR BUS
144,000 BITS PER SECOND PER SENSOR BUS
21,000 SERIAL WORDS PER SECOND PER ACTUATOR BUS
424,000 BITS PER SECOND PER ACTUATOR BUS
28,400 x 3 x.5 = 42,000 μSEC PROCESSING TIME
= 4.2% REAL TIME
WEIGHT = 129.0 LBS.



6-BUS SYSTEM WITH SEPARATE SENSOR/COMPUTER BUSSES
NO CROSS STRAPPING
CONFIGURATION IV

FIGURE VIII-10

CONFIGURATION V

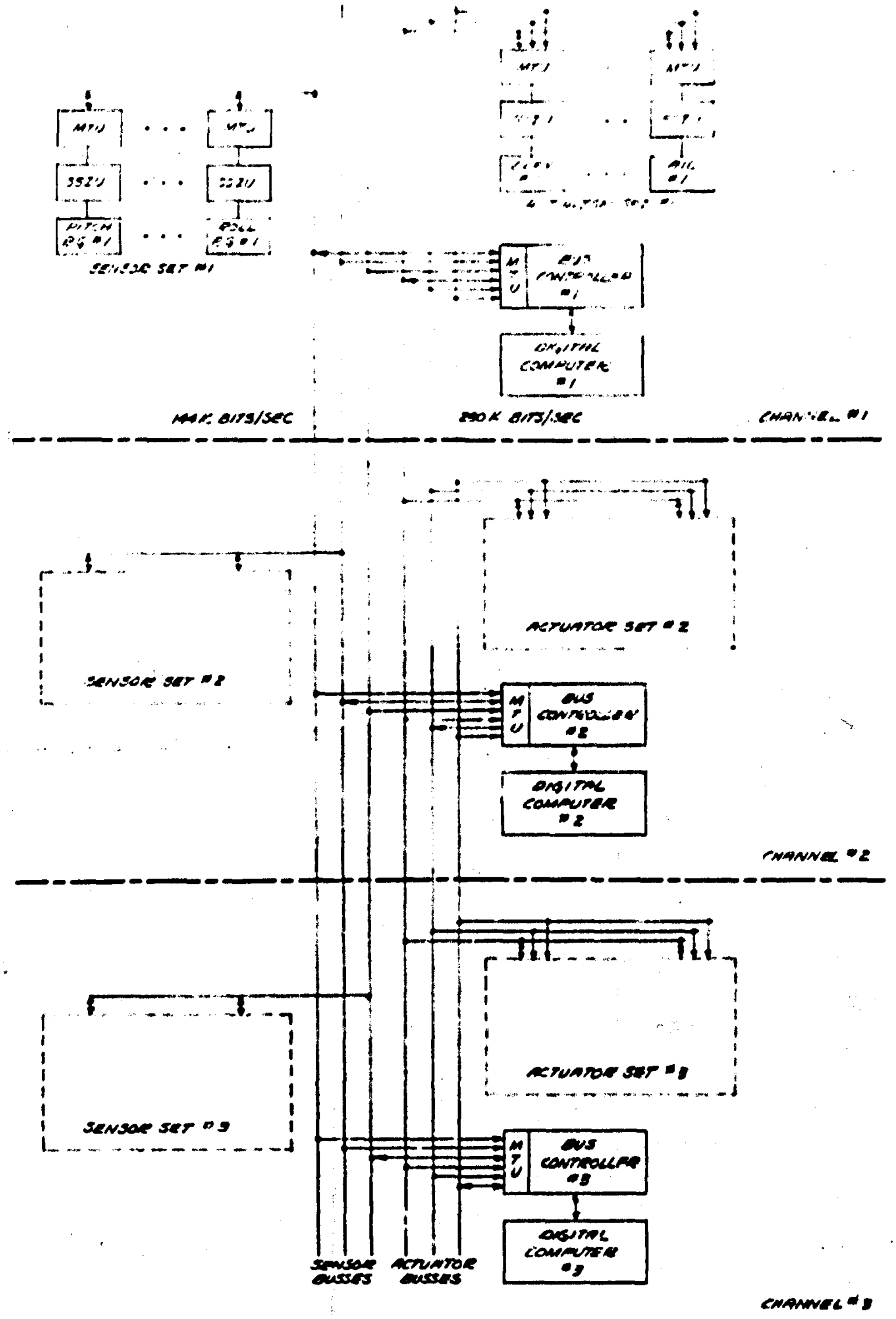
6-BUS SYSTEM
SENSOR/COMPUTER X STRAPPING
COMPUTER/ACTUATOR X STRAPPING

7,200 SERIAL WORDS PER SECOND PER SENSOR BUS
144,000 BITS PER SECOND PER SENSOR BUS

8,000
6,000

14,000 SERIAL WORDS PER SECOND PER ACTUATOR BUS
280,000 BITS PER SECOND PER ACTUATOR BUS

21,200 x 3 x .5 = 31,500 μ SEC PROCESSING TIME
= 3.15% REAL TIME
WEIGHT = 132.0 LBS.

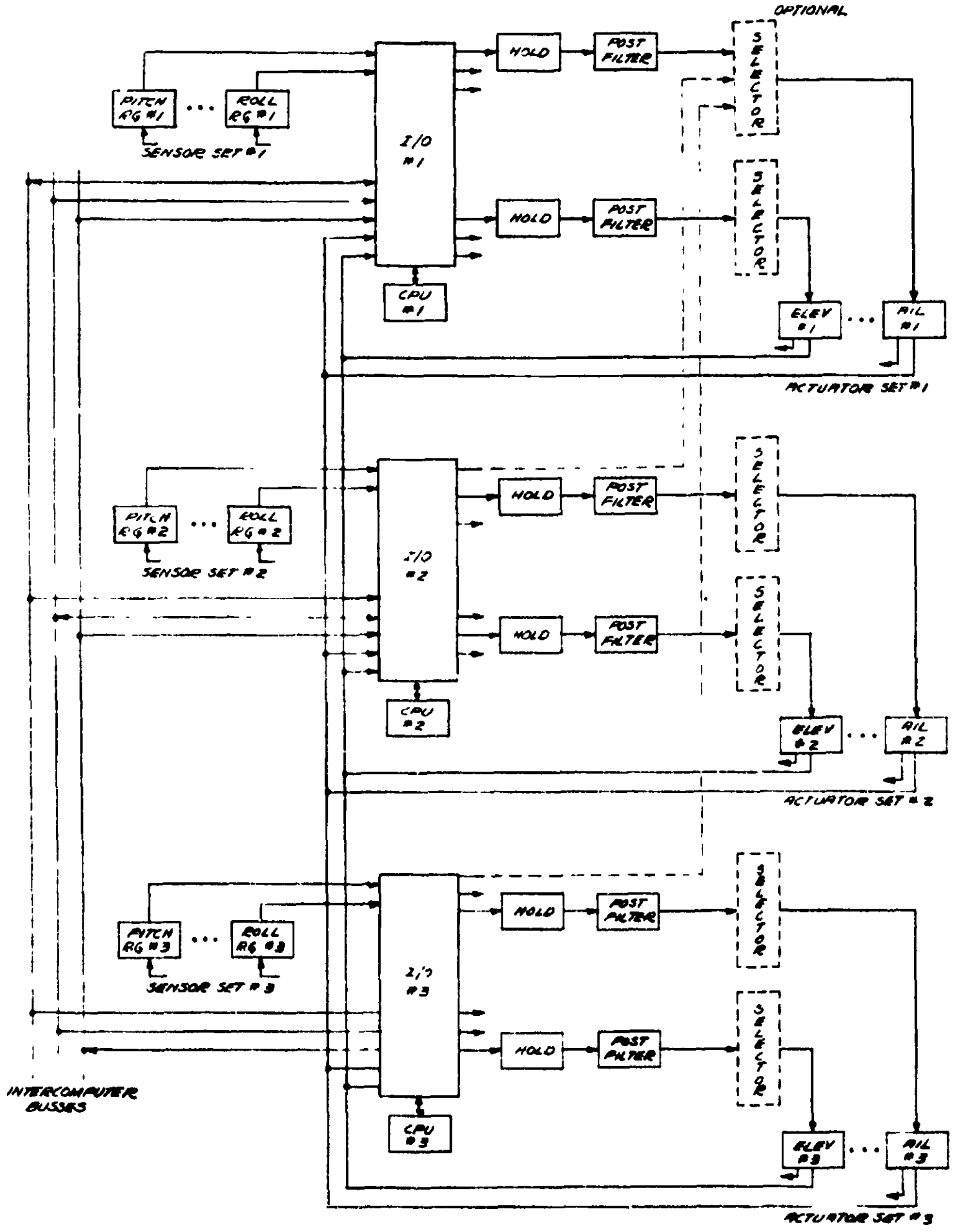


6-BUS SYSTEM - SENSOR/COMPUTER CROSS STRAPPING
COMPUTER/ACTUATOR CROSS STRAPPING
CONFIGURATION V
FIGURE VIII-11

CONFIGURATION VIII

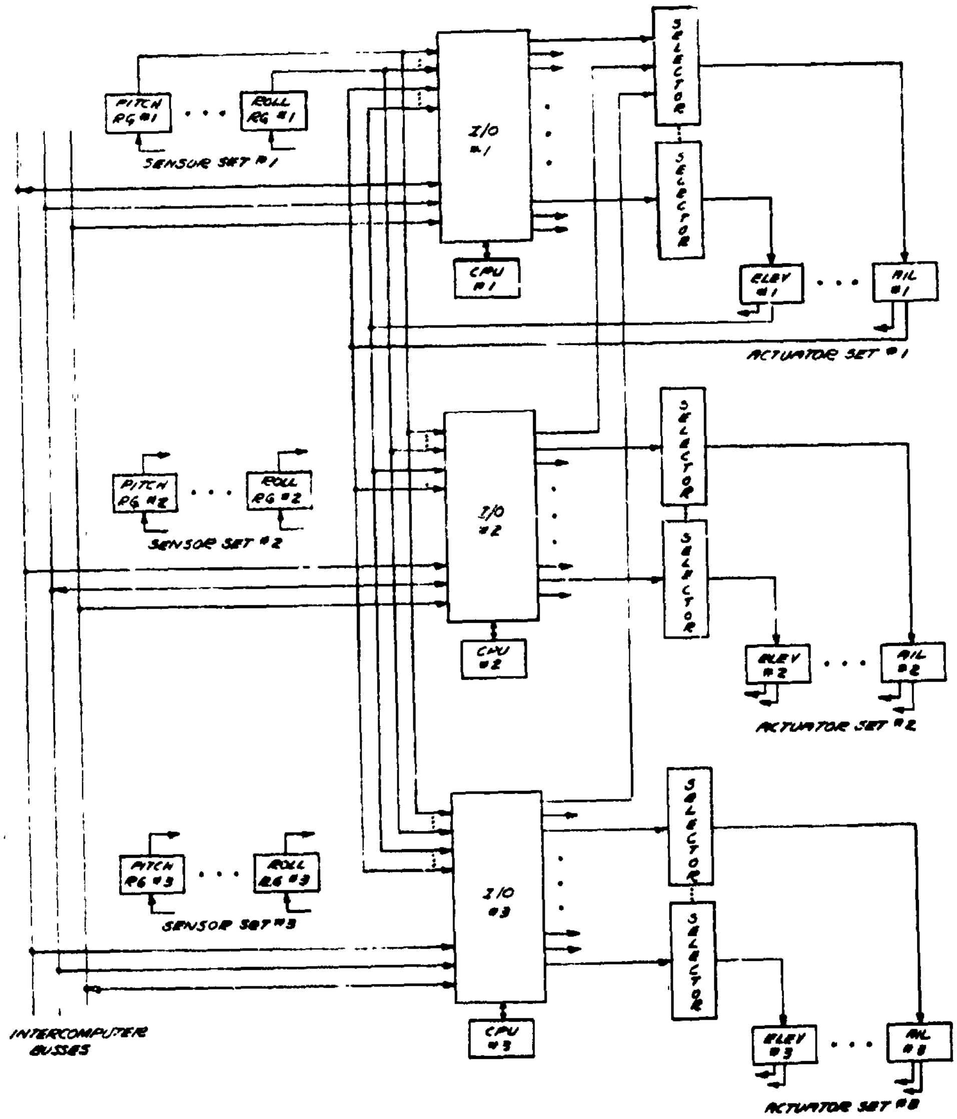
DEDICATED SYSTEM
SENSOR/COMPUTER X STRAPPING VIA ANALOG VOTERS
COMPUTER/ACTUATOR X STRAPPING VIA ANALOG VOTERS

WEIGHT = 124.5 LBS.



DEDICATED SYSTEM
INPUT CROSS STRAPPING VIA INTERCOMPUTER BUSSES
OUTPUT CROSS STRAPPINC (OPTIONAL)
CONFIGURATION VI

FIGURE VIII-12



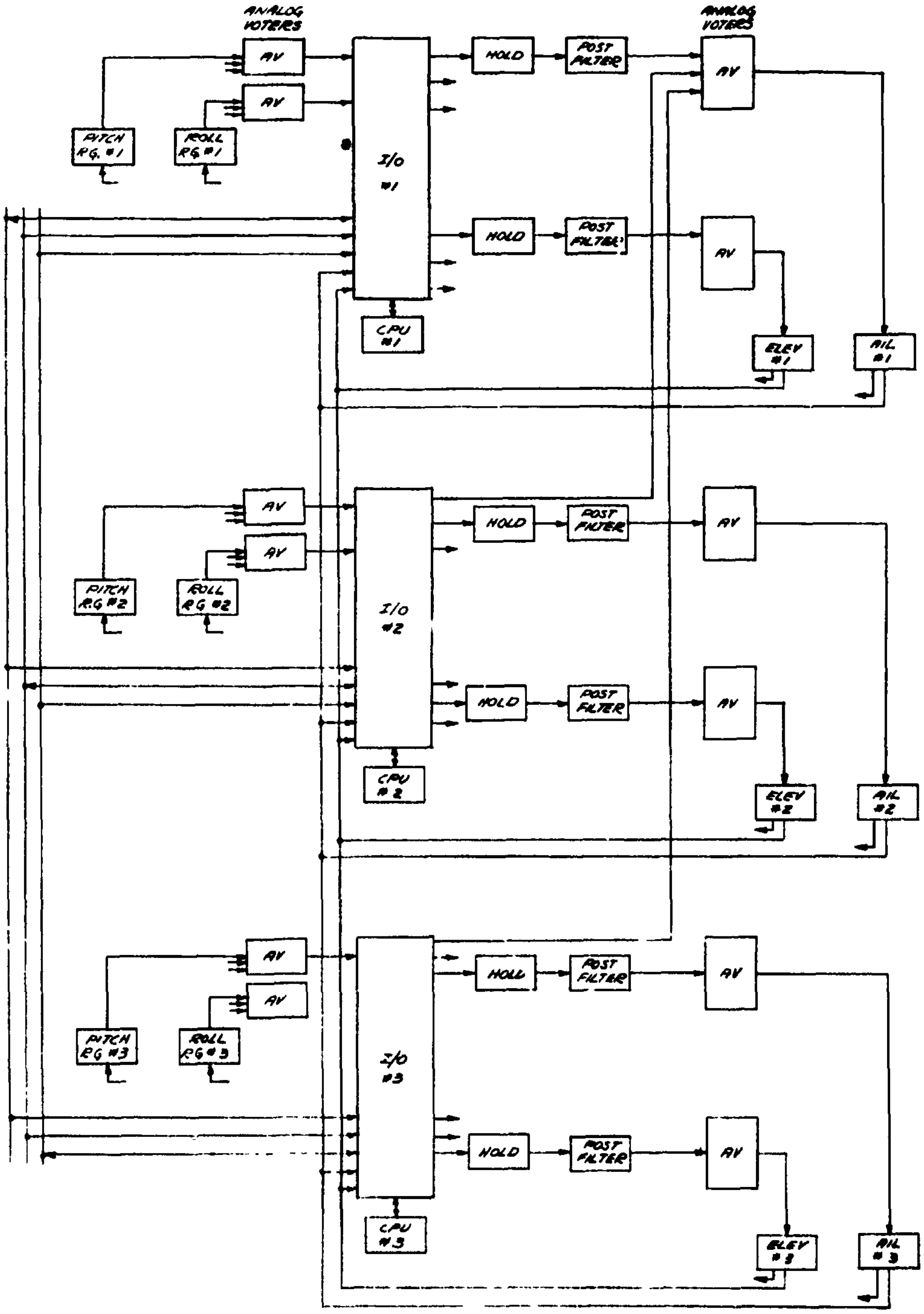
DEDICATED SYSTEM WITH FULL CROSS STRAPPING AND VOTING
CONFIGURATION VII

FIGURE VIII-13

CONFIGURATION VIII

DEDICATED SYSTEM
SENSOR/COMPUTER X STRAPPING VIA ANALOG VOTERS
COMPUTER/ACTUATOR X STRAPPING VIA ANALOG VOTERS

WEIGHT = 124.5 LBS.



DEDICATED SYSTEM
INPUT/OUTPUT CROSS STRAPPING VIA ANALOG VOTERS
CONFIGURATION VIII
FIGURE VIII-14

4. Conclusions

a. Perhaps the most important inference from the results is that the multiplex system can accommodate the required bus loading as indicated by Configuration IA where the bus loading is a maximum.

b. It can be expected that provision for intercomputer communications will be a requirement in any flight control system configuration. The quantity of data to be transferred, however, is very difficult to assess without a knowledge of the details of the specific configuration, the voting and monitoring strategy employed, etc. As a consequence, of the multiplex arrangements, Configuration II is recommended because it permits large quantities of intercomputer transfers without, in any way, affecting the loading of the main busses.

c. The main busses of Configuration II, even with sensor/computer and computer/actuator cross strapping, requires 304 K bits/sec. or less than 1/3 of the bus capacity.

d. The weight trade-off indicates that the dedicated and multiplex configurations are approximately equal. This is due to our assumptions regarding

- number of sensors and actuators
- the use of dedicated MTU/SSIU and A/D, D/A converters for each subsystem
- estimated weights of 0.5 lbs. for each MTU/SSIU and 0.4 lbs. for each A/D, D/A converter combination.

e. Each subsystem of the multiplexing configuration assumes an additional failure rate of 30×10^{-6} failures per hour due to the interface units. This could represent a considerable degradation in mission reliability.

f. The cost of the multiplex system can be reduced considerably if subsystems share a common MTU/SSIU unit. This would result in an increase in wires and wire weight depending upon the proximity of the subsystems to the interface unit. Moreover, when several subsystems share a common interface there is always the problem of common mode failures.

g. The estimates are based on present day technology. It can be expected that the reliability of weight and dollar cost of the interface units will improve over the next several years to the point where multiplexing will indeed become a feasible alternative.

APPENDIX IX

COMMON MODE AND SOFTWARE SINGLE POINT FAILURES

The reliability model used in the tradeoff studies was based on a number of assumptions regarding the effects of failures. Specifically, it was assumed that:

- Two undetected, dissimilar failures in different channels of either a triplex or quadruplex configuration would render the system non-operational.

- Two latent, dissimilar failures in different channels of either a triplex or quadruplex configuration would render the system non-operational.

- A failure in any channel would not significantly reduce test coverage for that channel or any other channel.

- A single, undetected failure in either a triplex or quadruplex configuration will not result in degraded performance.

- Failure in one channel of either a triplex or quadruplex configuration is independent of failures in any other channel.

In practice, of course, one or more of these assumptions may not be valid in a given situation. For example, a failure in one channel frequently reduces the ability of the test to detect subsequent failures, and in the case of comparison monitoring, may even reduce coverage in other channels as well. Nevertheless, the assumptions do not appear to be unreasonable in the context of the present study.

In this section specific attention will be focused on the last assumption above. Failures which affect two or more channels of a redundant system care classified as either *common mode failures* or *single point failures*. The latter type of failure includes failures of primary actuators, control links, power supplies, design defects or single point software failures. Common mode failures are caused by an environment which causes two or more channels to behave as though effected by a single point failure. Typical causes of common mode failures are an excessively noisy environment, EMI, power transients, avalanching in signal selection devices or synchronization lock.

1. Single Point Failures

It is clear that the probability of a single point failure of any kind must be consistent with the reliability goals of the system. In particular, the probability of a single point failure per flight hour must be considerably less than 3.0 x 10^{-6} for a fighter aircraft and 0.23 x 10^{-6} for a commercial transport. As a consequence, the probability of a software single point failure must be a fraction of the total probability of a single point failure. To fix upon a number it is not unreasonable to assume that the probability per flight hour of a software single point failure should be less than 0.3 x 10^{-6} for a fighter and 0.023 x 10^{-6} for a transport, if the system does not provide for dissimilar channels. Unlike conventional single point failure rates, which are determined by equipment failures, software single point failure and their probabilities of occurrence are determined by the environment; i.e., the event of assuming a certain state or exercising a certain transition path. Because it is not practicable to exercise all possible states and transition paths software verification procedures can be extremely costly and time consuming. The large number of possible states makes it unlikely that software verification can be accomplished by a deterministic test algorithm alone. Some form of random selection appears to be required. The development of such procedures for the flight control application is an area for future effort.

2. Examples of Single Point and Common Mode Failures

While the major sources of single point and common mode failures of conventional analog systems are well known, the sources in a digital control system are perhaps less familiar. In any case they are certainly different and, as a consequence, some examples of typical failures will be given. The list of course, is by no means exhaustive and is supplied merely to illustrate the possibilities:

a. An oversight by the programmer which, under certain remote conditions, causes the system to behave in an unpredictable manner.

b. A typical operation of a whole word computer is the negation of a numerical quantity. This is usually accomplished by taking the 2's complement. However, the 2's complement of the most negative number is the most negative number Thus, if the programmer inadvertently takes the 2's complement of -1 the result could be a hardover into all channels.

c. A similar result is obtained if an arithmetic register overflows, and the overflow is not compensated for. An overflow could result in an effective hardover into all channels.

d. Division by zero.

e. Division when the dividend and divisor are equal. In some division algorithms the remainder will assume an erroneous value.

f. Multiplication may require that the multiplicand be located in an even numbered arithmetic register. While violation of this rule is detected in the Assembler, it may happen that, due either to a manual insertion of an instruction or a power transient which causes the program register to assume a random value, the condition is violated. It has been observed that if the multiplicand and multiplier are in odd numbered registers the microgram goes into a "DØ" loop from which there is no recovery except by removing power and then reengaging the system. The "DØ" loop is not interrupted even by the normal external interrupt because the "DØ" is contained entirely within a single micro instruction.

g. When program synchronizing two or more computers via bidirectional intercomputer links it is possible for the computers to continually attempt synchronization without actually being able to do so. This condition could result in the cessation of all computations.

h. A power transient or excessive noise could result in loss of bits in transit on an internal computer bus. If the bits represented an address to the Program Register the result would be unpredictable since the computer would interpret an arbitrary data word as an instruction. The proper instruction sequence is recoverable upon reception of the next external interrupt which causes the computer to execute a predetermined instruction. However, variable storage, such as integrators, would not be recoverable. One solution is to reinitialize the entire set of variable storage. The resultant hiatus in the computations could have serious consequences to the safety of the airplane.

i. Intercomputer links, and particularly bi-directional links, could fail under "hot short" or other line transient conditions. In this event and because the links are eventually gated onto the memory busses, the affected computers could be seriously damaged.

It is emphasized that the above failure conditions may be computer dependent and, in most cases, represent careless programming. In any case, once a failure condition is identified steps can be taken to either eliminate it or minimize its effects. Unfortunately, it is the unidentified conditions that will cause the major problems.

APPENDIX X

TEST VALIDATION CONSIDERATIONS

From the results of the tradeoff studies it may be concluded that test coverage is a critical parameter in determining flight safety reliability of a redundant system. While test coverage requirements will vary considerably, depending upon the configuration, values between 0.99 and 0.999 can reasonably be expected. Compromising between these extremes we will select 0.995 as a tentative goal for purposes of this discussion. Having established the coverage required it remains to determine the coverage actually achieved.

1. Validation Procedure

Assume that a test procedure has been devised for an LRU. The validation procedure will consist of the following steps:

Step 1

Enumerate all component failures of the LRU (Ignore, for the moment, the feasibility of identifying all failures).

Step 2

Enumerate relative failure rates of all component failures.

Step 3

Simulate component failures at random; i.e., according to their relative frequency of occurrence.

Step 4

Tabulate the number of failures detected and compute the ratio SN/N where

S_N = number of failures detected

N = number of failures simulated

Since $\alpha = P(\bar{A}|F)$, we naturally expect that SN/N will approximate the unknown coverage, $1-\alpha$.

Each simulated failure is interpreted as a Bernoulli trial with probability of success equal to $1 - \alpha$. Let it be desired to estimate $1-\alpha$ with an accuracy of ϵ ; i.e., N is chosen so large that

$$\frac{S_N}{N} - \epsilon \le 1-\alpha. \qquad \text{(X-1)}$$

Unfortunately no sample size can give absolute assurance that S_N/N satisfies (X-1). Since absolute certainty is unattainable we settle for an arbitrary confidence level, λ , and only require that N be large enough to insure that

$$P\left(\frac{SN}{N} - \epsilon \le 1-\alpha\right) \ge \lambda. \qquad \text{(X-2)}$$

The number of trials necessary to insure the inequality of (X-2) depends upon the three parameters ϵ, λ and $1-\alpha$.

Accuracy, ϵ

Since it is desirable to be able to distinguish between a coverage of .99 and .999 the accuracy must, *as a minimum*, satisfy the inequality

$$\epsilon \le 0.005.$$

This requirement imposes an additional requirement on the degree of ignorance regarding the known failures of the LRU; i.e., if

M = total number of failures of the device (assumed to be equiprobable)

m = total known failures (also assumed to be equiprobable)

then

$$\frac{M-m}{m} << \epsilon .$$

Thus, for = .005 = 1/200, *at least* 199 of every 200 failures of the device must be known in order to generate a failure model which is consistent with the accuracy requirement of the validation program.

Confidence Level, λ

For repeatability of the validation experiment the confidence level should be approximately unity. However, the cost of high confidence can be considerable in terms of the number of simulated failures which are required. As a consequence, some compromise is desirable. It is proposed to use a confidence level of 90% (i.e., $\lambda = .9$) which yields a reasonable repeatability but, as will be seen does not result in an excessive number of trials.

Test Coverage, $1-\alpha$

Although the purpose of the validation program is to establish the value of $1-\alpha$, it may happen that the test coverage is known, a priori, to exceed a known value. This is not an unreasonable expectation since the test was presumably devised to detect a certain minimal set of failures. Since values of $1-\alpha$ between .8 and .95 are relatively easy to establish we may assume that $1-\alpha \geqslant .95$. This will reduce the number of simulated failures required.

Sample Size

Returning now to the Bernoulli trials, the probability that the number of successes, S_N, lies between K_1 and K_2 is given by

$$P\left(K_1 \le S_N \le K_2\right) = \sum_{j=K_1}^{K_2} \binom{N}{j} (1-\alpha)^j \alpha^{N-j} \qquad \text{(X-3)}$$

where 1- is the probability of success; i.e., of a detected failure.

If K_1 and K_2 arc selected such that

$$K_1 = 0 \qquad \text{(X-4)}$$

and

$$K_2 = N(1-\alpha+\epsilon)$$

then $$P\left(K_1 \le S_N \le K_2\right) = P\left(\frac{S_N}{N} - \epsilon \le 1-\alpha\right)$$

which corresponds to the left side of inequality (X-2). Unfortunately the right side of (X-3) is difficult to evaluate when N is large. For this purpose we use the

<u>DeMoivre-Laplace Theorem:</u>

$$P(K_1 \leq S_N \leq K_2) \sim \phi\left(\frac{K_2 - N(1-\alpha) + .5}{\sqrt{N\alpha(1-\alpha)}}\right) - \phi\left(\frac{K_1 - N(1-\alpha) - .5}{\sqrt{N\alpha(1-\alpha)}}\right) \qquad \text{(X-5)}$$

when

$$\phi(z) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{z} \exp\left(-\frac{x^2}{2}\right) dx$$

and "$\sim$" means that the ratio of the two sides of (X-5) tends to unity as N tends to ∞ .

Substituting K_1 and K_2 of (X-4) into (X-5) yields

$$P\left(\frac{S_N}{N} - \epsilon \leq 1-\alpha\right) \qquad \text{(X-6)}$$

$$\sim \phi\left(\frac{N\epsilon + .5}{\sqrt{N\alpha(1-\alpha)}}\right) - \phi\left(\frac{-N(1-\alpha) - .5}{\sqrt{N\alpha(1-\alpha)}}\right)$$

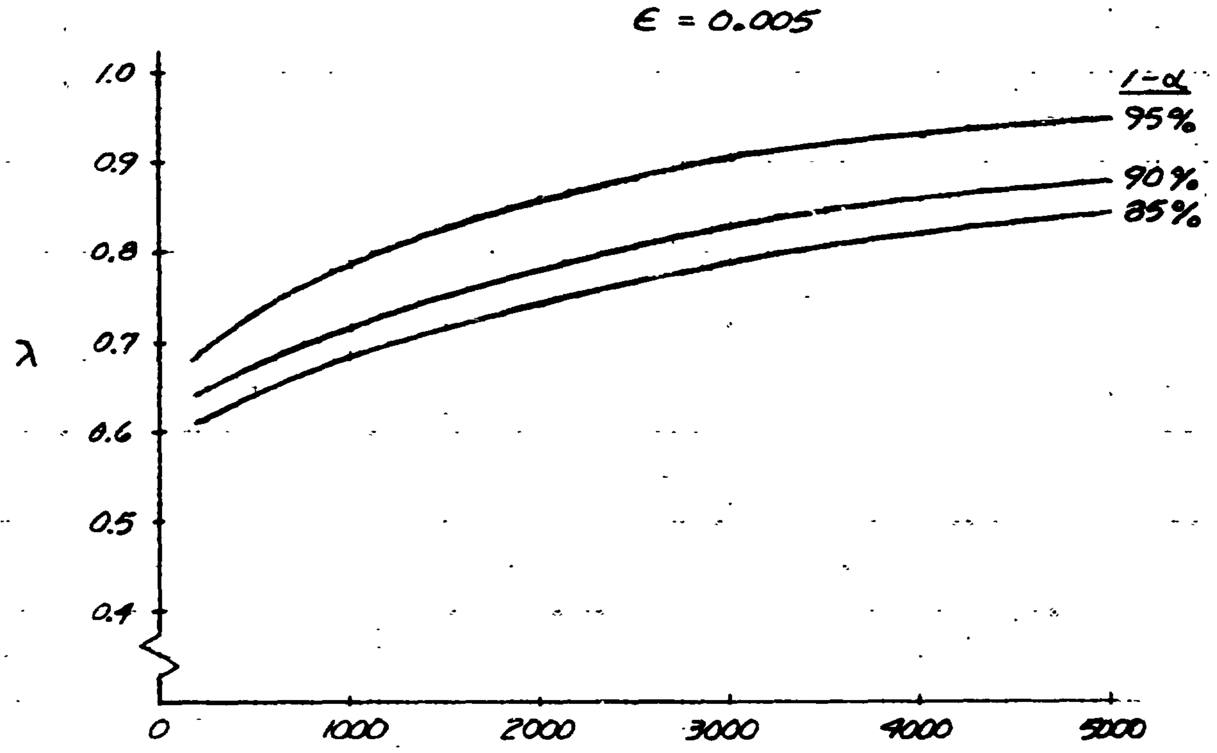
The right side of (X-6) is substituted into (X-2) and the number of samples, N, is evaluated as a function of λ and $1-\alpha$. The result is shown in Figure X-1. where λ is plotted versus N for several values of $1-\alpha$ with an accuracy of .005. It it is known, a priori, that

$$1-\alpha \geq .95$$

then the number of simulated failures required for a 90% confidence in 3000, approximately.

2. Summary

From the preceding discussion and sample computation it can be seen that validating a test coverage goal exceeding 0.995 may require a comprehensive failure model and the capability of simulating large numbers of failures. The failure model must be consistent with the accuracy requirement of the validation program which means that the unknown failures may not exceed 1/200 of the total failures of the device. As pointed out in Appendix VII, simulating non-destructive failures of digital devices can present considerable difficulties particularly when such failures affect internal states or transition paths.



CONFIDENCE LEVEL VERSUS NUMBER OF SAMPLES

FIGURE X-1

APPENDIX XI

SYNCHRONIZATION REQUIREMENTS FOR REDUNDANT DIGITAL FLIGHT CONTROL SYSTEMS

1. General

Rather than presenting particular schemes for synchronizing digital computers, this discussion will attempt to determine what unique factors of digital implementation dictate the degree of synchronization required. Motivation for this review is based on the fact that traditional redundant systems have been designed without a general synchronization scheme.

By way of defining terms, synchronization will be considered to mean (near) simultaneous occurrence of similar events in each of the redundant channels. This can range from sampling a particular input variable; e.g., servo follow-up, at the same time in each redundant channel up to having each micro-program step in each redundant computer occur at the same time.

2. Passive Redundant Configuration

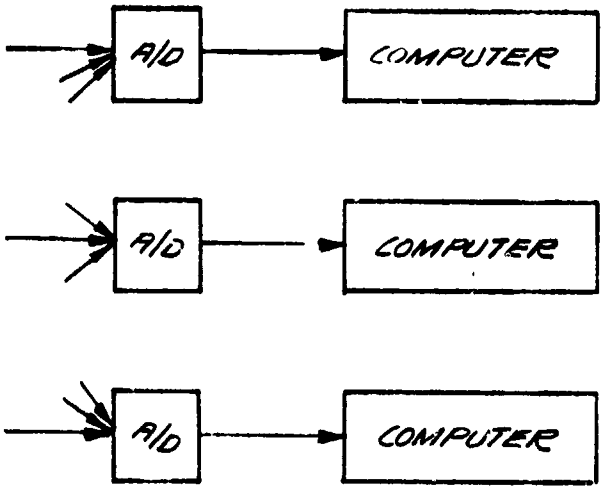
Figure XI-1 shows an elementary triple redundant configuration with no cross strapping of intermediate variables. It is assumed that the output variables command servos whose outputs add on a channel basis to control the aircraft. This type of redundance has been called "passive" since failures are not actively detected nor is the configuration altered as a function of a failure. The effect of a failed channel would be a 33% loss of authority and gain for a passive failure, and in addition a 33% offset for a hardover failure. For this configuration the outputs (non-failed state) would be a control law modified form of the inputs distorted by four factors:

a. transport lag as a function of the iteration rate and algorithm

b. input noise as modified by input filtering and folding effects

c. output ripple due to non-infinite iteration rate and large rate of output change

d. computer errors.



CROSS-STRAPPING ARRANGEMENT
SCHEME 1

FIGURE XI-1

Transport lag is most evident when the input changes immediately after the data is input to the computer so that the input change cannot affect the output until the next iteration. When the inputs are synchronized between the three redundant channels then the net output will suffer an average transport lag. If the three channels are asynchronous, then the average transport lag will be the same. Thus, the net output information will be as fresh on the average for the asynchronous as for the synchronous case.

The effect of input noise folding will depend on the degree of synchronizing the data input events. For example, assume that the three input signals of a given type, three pitch rate signals, contain in-phase noise components (such as power supply frequency) near a multiple of the sampling frequency. This will fold down to give an output noise component at the difference frequency, the amplitude of which will be largest when the data input events are synchronized and, in general, smaller when they are not synchronized. Of course, this noise effect would have to be made small in any case by suitable prefiltering so it is not considered significant in any case. Output ripple will be a function of synchronizing the output event, the worst case (largest ripple) being when synchronized and the best when phased 1/3 sampling interval apart. Obviously, if considered of significance, one could synchronize the output events so that they always occur 1/3 sampling interval apart. In any event, suitable choice of iteration rate and post filtering can reduce this ripple to small enough values so that this effect is not considered of significance.

The effects of computer errors such as truncation and round-off on control system performance has not been extensively studied, to our knowledge. Accordingly, the effect of the degree of synchronization on these errors would be difficult to establish. However, one truncation error problem that has been identified is that associated with integration and lag filter functions that are slow compared to the iteration rate. In such cases a dead space effect can be observed when the increment per iteration required is less than the least significant bit of the data word. In applications where this dead space is of significance, the effect can be made negligible by a simple double precision operation. Therefore, at least in this case, the question of synchronization is not affected. Therefore, the "passive" redundant configuration as outlined here does not have any significant need for synchronization of any type.

3. Analog Output Voting

Although the "passive" redundant configuration is conceptually simple there are several objections to it, the most important being the sensitivity to a hardover failure and the change in gain after failure. These difficulties can be obviated by "voting" the output signals; i.e., by selecting a good signal for transmission downstream. There are many such signal selection schemes available for both analog and digital formats. If analog voters are used for the output signals, the system can be made relatively insensitive to first failures that occur upstream of the voter and the system characteristics are independent of synchronization except for the noise effects mentioned previously.

4. Input Signal Comparison

It is desirable to have the ability to compare redundant input signals for failure detection and for signal selection purposes. Equivalent signals must be compared more or less simultaneously depending on the comparison accuracy required for the particular signal. This is so whether the comparison is accomplished by the digital computer or with dedicated hardware. However, when using differential amplifiers for comparison monitors, their speed of response is so rapid that for control signal frequencies the comparison is essentially simultaneous. However, if the comparison is done in the digital computer(s), nonsimultaneous sampling will cause an error in the comparison which could be significant.

A comparator error due to sampling delay equivalent to 1% of full scale signal would probably be acceptable in most cases. Assuming an input rate of zero to full scale signal in ne second, then the sampling delay between two signals to be compared should be less than 10 milliseconds.

When the comparisons are done in computer software, the redundant signals must be entered into the computers. Two methods of entering such data are shown in Figure 2. In the first method all redundant signal sets are sequentially converted and entered into each computer whereas in the second method intercomputer buses transfer the necessary data on a digital basis.

Considering the first method, the worst case lag beween any two compared signals will be twice the conversion time or about 60 μ s. This will be the only delay of concern if there is assurance that the comparisons are not made between new and stale data; i.e., that the comparison routine is not run in the time period that the signals to be compared are being refreshed.

This assurance is readily provided by proper programming when the input multiplexers and converters are program controlled. However, when the input section is independently controlled, and particularly when DMA is used, this assurance is more difficult to provide. One method that has been suggested is to assign a flag bit in each converted word so that flag bit status indicates common; i.e., adjacent, sampling.

For the second input method indicated in Figure XI-2, the problem is more difficult. If the input multiplexers, A/D converters and computers are all asynchronous, then the relative freshness of data being compared is a function of the basic refresh rates. As an example consider an autopilot having twenty input control signals. At 30 μs per conversion, the maximum staleness at the converter output would be 600 μs. If the inputs are DMA'd into one computer and then DMA'd out to the other computers then only a few additional microseconds of staleness would be added. Thus, in each computers comparisons will be made of data that is at most 600 μs late which is considerably less than the 10 ms. allowed.

However, in method 2 if the conversions and data communications are under program control, then either the programs in each computer must be synchronized to better than 10 ms or else the conversions and data communications must be iterated at least 100 times per second.

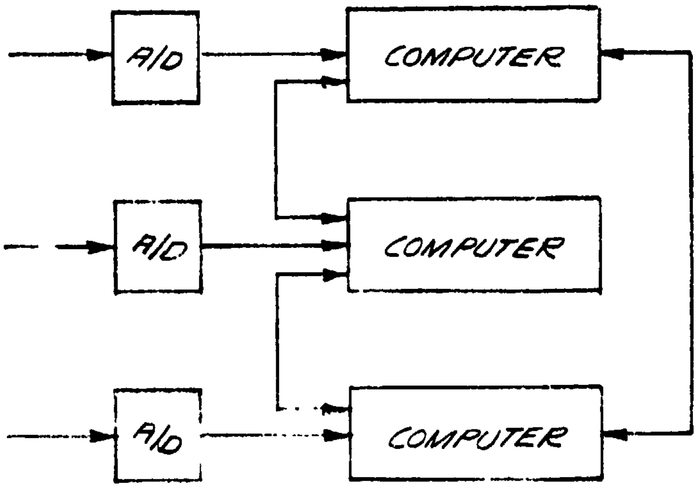
5. Output Signal Comparisons

Comparisons of redundant computers outputs are useful for both failure detection and voting. If the computers have identical inputs and are fully synchronous then with no failures the computer outputs will be identical at every instant. Even if the computers are "clock synchronized" if the input data are not identical then due to conditional branches in the program, the programs may not be synchronous and the outputs may not be identical. If the input data are identical but the computers are asynchronous,then the outputs will be identical but delayed by up to one iteration time.

Therefore, there are two problems to be considered:

a. grossly different outputs due to differences in the operating programs caused by conditional branching when using nonidentical input data

b. timing differences in the outputs caused by asynchronous computer operations resulting in delays of up to one iteration cycle.



CROSS-STRAPPING ARRANGEMENT
SCHEME 2

FIGURE XI-2

The first problem is not unique to digital systems; it also occurs in analog systems. A good example of this is the transition from glide slope track to flare modes in an autoland system. This transition is normally a function of altitude and altitude rate signals. Due to tolerances in the redundant signal sources the channels will not switch to flare at the same time so that one channel is calling for the flare maneuver while the other channels are attempting to track the glide slope beam. As a result the output comparators will alarm. Solutions to this are either to equalize the signals prior to the signal switch or else to design the switching logic so that all redundant signals have to be below the critical value before the mode is initiated. These solutions are applicable to the digital system but their implementation requires that internally generated variables or logic states be communicated between the computers. If this communication link uses DMA or interrupt then the computers can be asynchronous since both computers need not be simultaneously at particular points in their programs. However, if both reception and transmission of data is under program control then synchronization is required so that when one computer transmits, the other receives.

The second problem; i.e., timing differences between redundant outputs, involves the desired speed of comparator response, the desired comparator threshold, the basic iteration rate and the maximum required output rate. Assuming the values used in section 4, comparator threshold of 1% of full scale and maximum rate of zero to full scale in one second, then the maximum increment per iteration will be (FS)/I (where I is the iteration rate). This will also be the comparator error for the worst case computer delay of one iteration. If the input data in each computer is identical then the output values will be identical for no failure so that the delay error is the only error. Therefore, for instantaneous comparison the delay error must be less than the comparator threshold in order to have no nuisance alarm; i.e., (FS)/I ≤ .01 (FS) or 1 ≥ 100 iterations per second. If the input data into each computer is not identical then there will be comparator errors due to tolerances between input data sources. If arbitrarily one half of the allowable error is assigned to these sources and the other one half to output delay then 1 ≥ 200 iterations per second would be needed.

However, it is questionable that instantaneous output comparison is really necessary and that some type of delayed comparison would not be sufficient. As an example, suppose that "B" computer leads the "A" computer. Then the "A" comparison done immediately after the "A" update will be correct. If "B" lags "A" then the "A" comparison done before the "A" update is correct.

Therefore, if no failure exists then the smaller of these two comparisons is correct. If the incremental output per iteration can be larger than the desired failure detection threshold, then failure logic requiring that both comparisons exceed the threshold would be used.

However, for step failures there could be an alarm delay equivalent to one iteration time. It is desirable to make the alarm delay as short as possible but this is at the expense of increased iteration rate. How large the alarm delay may be is a function of the aircraft sensitivity and response characteristics of the servos. If the servo redundancy is such that the servo is insensitive to command failures, then relatively large alarm delays would be acceptable since the main purpose would be to alert the pilot. If the servos respond to the failure and the alarm is to be used to disconnect the failed computer, then the alarm delay should be shorter to reduce the amount of servo motion and resultant aircraft transient due to the failure. In that case, an alarm delay of 50 to 100 ms would be appropriate. As an example, for an aircraft having 1 deg/g sensitivity and 3 rad/sec. second order response, and a servo with slew rate of 40 deg/sec., a disengage delay of 100 ms would meet the normal .1 g transient requirement.

6. Conclusions

The question of synchronization arises where there is communication between computers. Communication which is program controlled requires synchronization at least on a program basis. Communications which are independently controlled do not require synchronization since the refresh rates for typical flight control applications can be made high enough to make relative errors negligible.

The major area where synchronization might be desirable is where variables computed in each redundant computer are to be compared. If these comparisons must be made and action taken very rapidly (on the order of 10 ms), then some type of synchronization would be preferable to an increase in the basic iteration rate to values higher than would otherwise be necessary.

APPENDIX XII

ANA_OG INNER LOOPS/DIGITAL OUTER LOOPS

General Observations

1. In the context of supplying an autopilot command, the digital computer can be treated as any other sensor with a relatively high failure rate. Because its failure rate will be approximately equal to the failure rate of the inner loop, cross strapping between the digital computer and the inner loop is desirable.

2. The signal interface between the digital outer loop and the analog inner loop presents no unusual problems.

3. Because of its computational flexibility the digital computers can compensate for undesirable feedback in the inner loop. For example, the autopilot command, as supplied by the digital computer, may cancel the stick command and acceleration feedback or effectively increase rate feedback in a particular mode of flight. Obviously, stick force, rate and acceleration sensors must be accessible to the digital computer in order to achieve this compensation.

4. Variable authority limits can be computed in the digital computer and the autopilot command computed accordingly. However, in the event of a failure of the digital computer, the computed limit must be superseded by a slightly higher limit contained in the inner loop.

5. Easy on, easy off and synchronization functions of the autopilot commands are computed in the digital computer--thus eliminating the need for dedicated circuitry in the inner loop.

6. Care must be taken to prevent an inner loop channel from disengaging in the event of a digital computer failure. For example, in a triplex system with three digital computer outer loops, if each computer supplies a dedicated autopilot command to an inner loop, a failure of a digital computer could result in disengagement of the inner loop channel. This situation could occur if the inner loop monitoring detects a difference between the servo commands before the autopilot failure is detected. This is an extremely undesirable situation because it significantly reduces the

reliability of the inner loops. Assuming that the inner loop monitoring must be rapid in order to reduce undesirable failure transients it appears that the solution is (1) to supply the same (or effectively the same) autopilot command to all inner loops or (2) monitor, isolate and disengage autopilot command failures before the failure is detected by inner loop monitoring. In this latter strategy, however, an alternate autopilot command must be available; otherwise an imbalance will develop between inner loops causing either a disengagement or a reduction in servo authority. In summary,

(a) each inner loop must receive effectively the same autopilot command and

(b) alternate autopilot commands must be provided if high reliability of the autopilot is required without performance degradation.

7. All autopilot commands should be authority limited.* The authority limit may be varied as a function of g's, dynamic pressure, airspeed, trim, etc. If the authority limit (for safety) is compatible with autopilot performance, then outer loop monitoring may be performed by the digital computers which may either disengage one autopilot command or annunciate the failure at the control panel for manual disengagement by the pilot. In any case, the outer loop authority limit which should preferably be located in the inner loop is sufficient protection against failure. In the critical case, however, the authority limit is not compatible with autopilot performance. In this event the inner loops should be provided with the means of detecting autopilot failures. While it is possible, in principle, to monitor the outer loops external to the inner loops, it is desirable for Bit to concentrate all monitoring in the inner loops.

8. In order to reduce the effect of autopilot disengagement transients due to failures of the outer loops it is desirable to include a fade-out circuit in the inner loop. This circuit could be a simple lag filter. The undesirable lag effects can be compensated for in the digital computer for normal operation.

* By authority limit we mean rate and amplitude of signal.

9. Digital computers can be used for Bite signal generation and testing. When used for this purpose, at least one digital computer must be operational before each flight.

Configurations

Several outer loop/inrer loop configurations are presented for consideration. It is emphasized that our objective is to prevent alternatives rather than recommendations. For simplicity it is assumed that the inner loop (FBW) is a triplex arrangement.

Figure XII-1

In this arrangement a single digital computer supplies the autopilot commands to all axes. Implicit in this configuration is the assumption that the authority limit is compatible with autopilot performance. The failure rate of the autopilot is at least as great as that of the digital computer; e.g., 120 x 10^{-6} failure per hour. If autopilot sensors are not cross-strapped, the failure rate could be considerably worse. With a single digital computer, a single computer failure could result in a hardover to all axes.

Figure XII-2

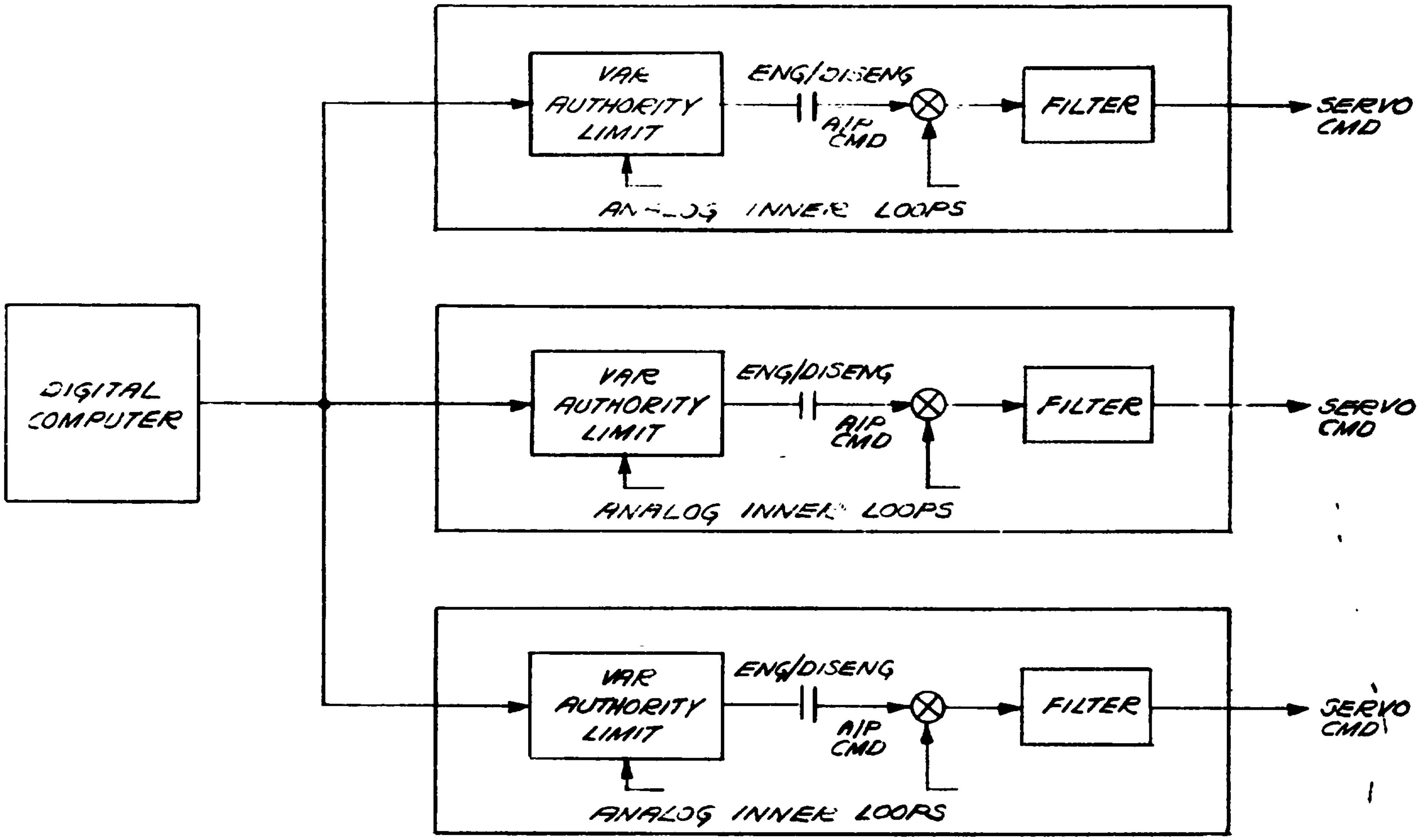
In this configuration two digital computers supply a single autopilot command to all axes and all channels. The authority limit is presumed to be compatible with performance. Selection of one of the two available autopilot commands is performed by the pilot, assisted by comparison monitoring between computers followed by computer self test in the event of a comparison difference. This arrangement results in a considerable improvement in outer loop availability.

Figure XII-3

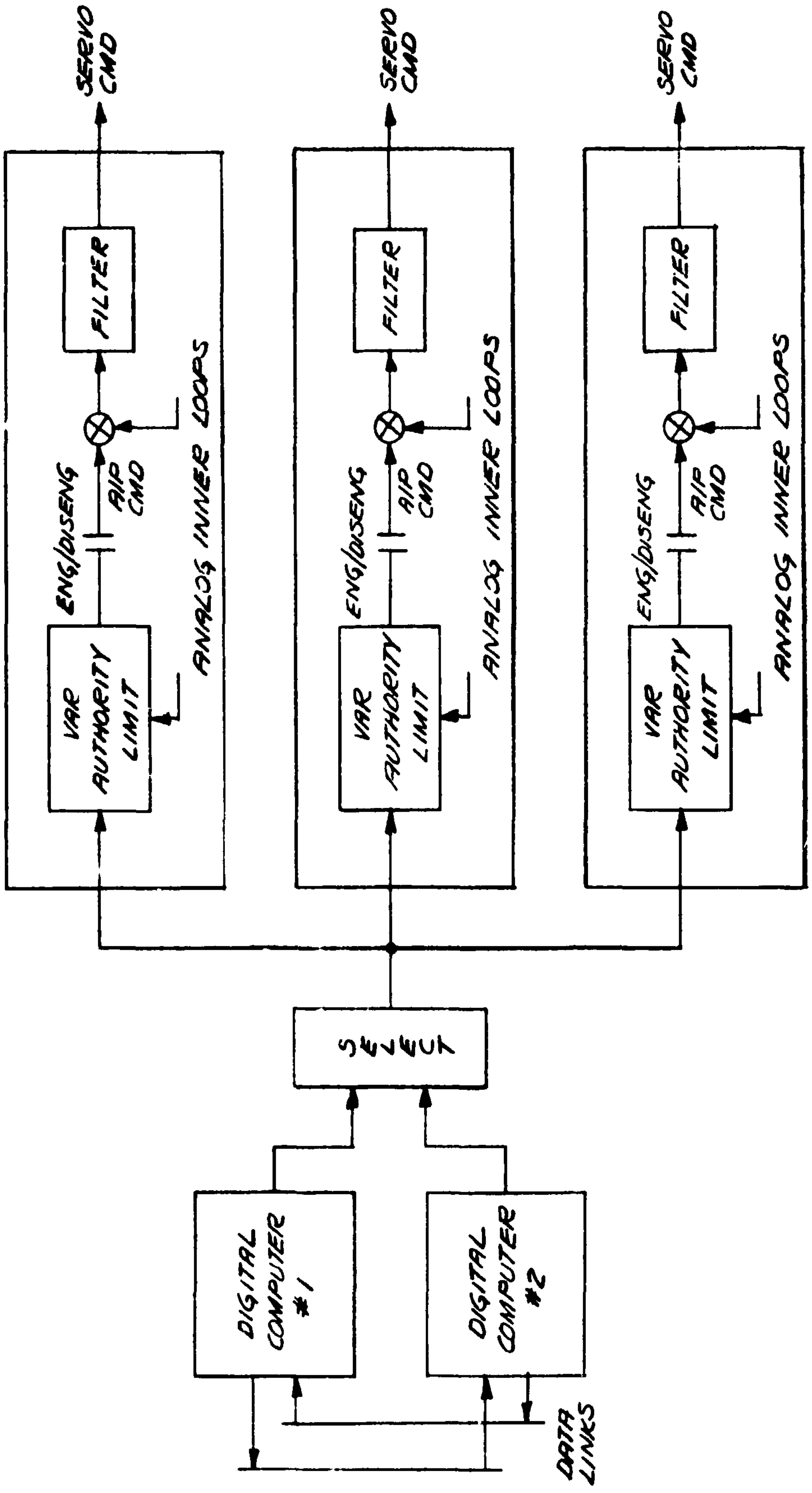
In this arrangement two digital computers supply two separate autopilot commands for all axes and all channels. The two autopilot commands are compared in each inner loop channel. Detected failures result in rapid autopilot disengagement. While some authority limit is provided, it is assumed that the safety limit is not compatible with autopilot performance. Reliability and availability of the autopilot is considerably worse than that of a single computer as in Figure 1, since loss of one of two computers will result in loss of the outer loop.

Figure XII-4

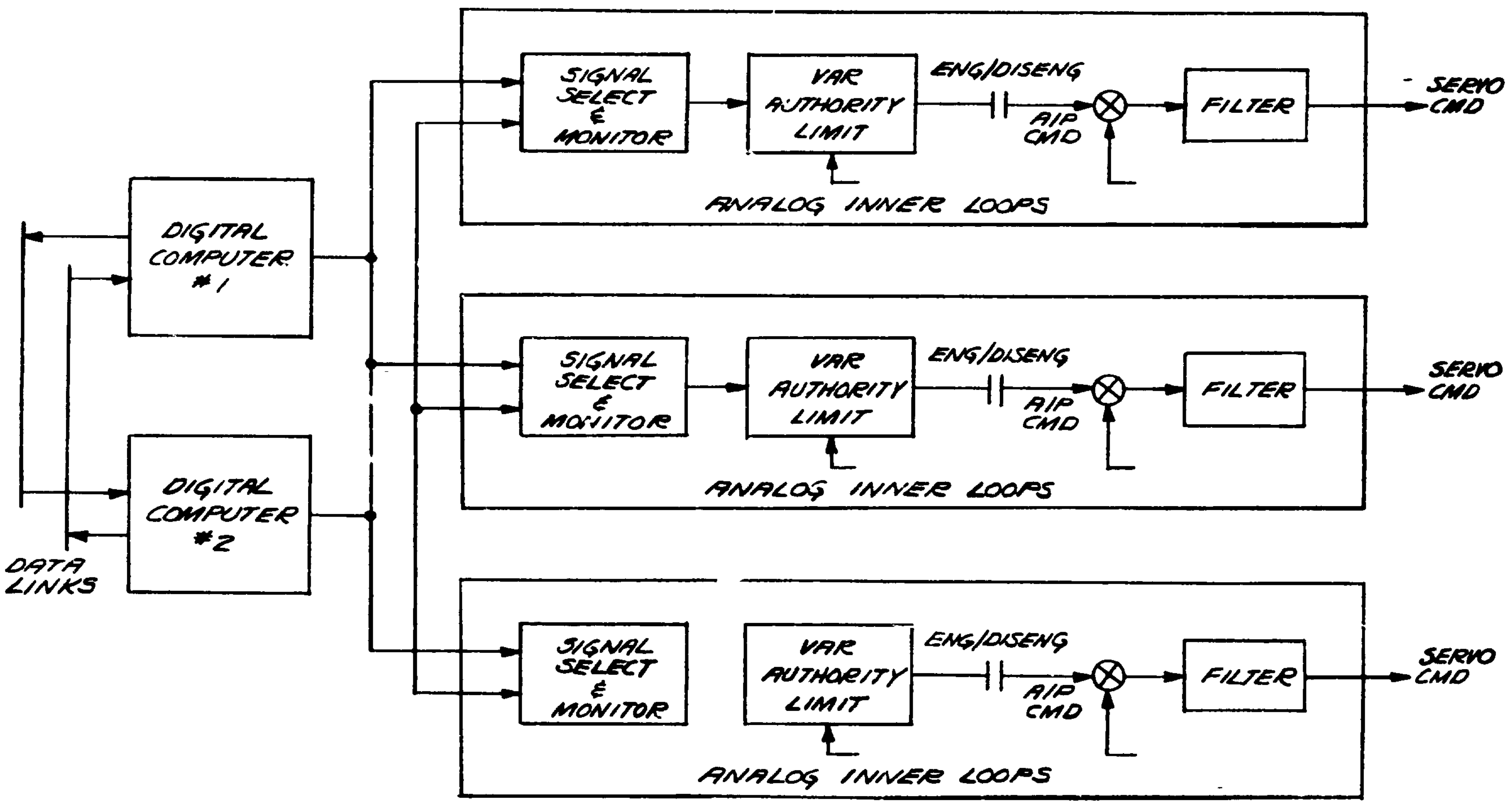
In this configuration there are three digital computers, each supplying the autopilot command for all axes and all channels. Command selection and monitoring is performed in the inner loops. It is assumed that selection and failure monitoring of autopilot sensors, if performed by the digital computers, is compatible with safety requirements since it must be presumed that the authority limit is not compatible with autopilot performance.



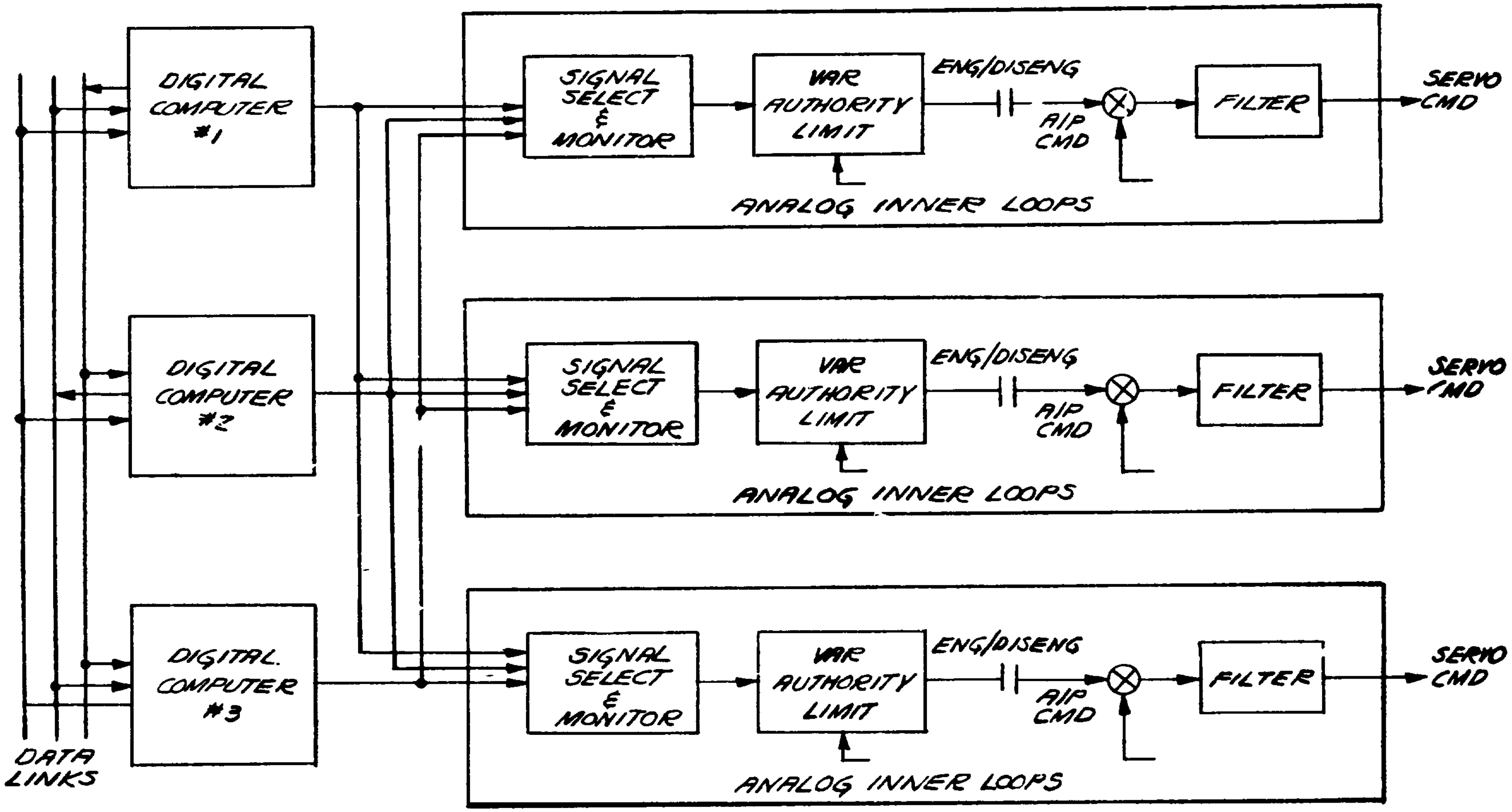
SINGLE DIGITAL OUTER LOOP
FIGURE XII-1



DUAL/STANDBY DIGITAL OUTER LOOP
FIGURE XII-2



DUAL/FAIL PASSIVE DIGITAL OUTER LOOP
FIGURE XII-3



FAIL OPERATIONAL TRIPLEX DIGITAL OUTER LOOPS
FIGURE XII-4

☆U.S.Government Printing Office: 1975 — 657-018/392