NASA CR-
147493

R-932

SPACE SHUTTLE ORBITAL MANEUVERING SYSTEM FAILURE DETECTION AND IDENTIFICATION SOFTWARE REQUIREMENTS (UNCONTROLLED)

by

Louis A. D'Amario, John P. Vullo

January 1976

Recommended for Inclusion in
Space Shuttle Orbiter
Orbital Flight Test
Level C
Functional Subsystem
Software Requirements Document
Guidance, Navigation, and Control
Part D
Redundancy Management

(NASA-CR-147493) SPACE SHUTTLE ORBITAL MANEUVERING SYSTEM FAILURE DETECTION AND IDENTIFICATION SOFTWARE REQUIREMENTS (UNCONTROLLED) (Draper (Charles Stark) Lab., Inc.) 65 p HC $4.50 CSCL 22B G3/17
N76-19209
Unclas
22064

The Charles Stark Draper Laboratory, Inc.
Cambridge, Massachusetts 02139

R-932

SPACE SHUTTLE
ORBITAL MANEUVERING SYSTEM
FAILURE DETECTION AND IDENTIFICATION
SOFTWARE REQUIREMENTS
(UNCONTROLLED)

by

Louis A. D'Amario

John P. Vullo

Recommended for Inclusion in
Space Shuttle Orbiter
Orbital Flight Test
Level C
Functional Subsystem
Software Requirements Document
Guidance, Navigation, and Control
Part D
Redundancy Management

January 1976

Approved: ______________________
for N. E. Sears

The Charles Stark Draper Laboratory, Inc.
Cambridge, Massachusetts 02139

ACKNOWLEDGMENT

This report was prepared by The Charles Stark Draper Laboratory, Inc. under contract NAS9-13809 with the Johnson Space Center, of the National Aeronautics and Space Administration.

Technical coordination for the report was provided by Kim Kirchwey and George Silver.

The publication of this report does not constitute approval by the National Aeronautics and Space Administration of the findings contained herein. It is published for the exchange and stimulation of ideas.

ABSTRACT

This report presents candidate designs and their software implementation for the Orbital Maneuvering System (OMS) Failure Detection and Identification (FDI) algorithms in the Redundancy Management (RM) module of the Space Shuttle Guidance, Navigation, and Control (GN&C) software. The OMS engine FDI algorithm monitors OMS engine thrust performance, and the OMS actuator FDI algorithm monitors OMS gimbal actuator performance.

Section 1 describes the software functional requirements of the algorithms. It contains a statement of the objective of each algorithm, a list of the assumptions which have governed its design, input-output requirements, a functional description of the algorithm (including a functional block diagram), and input interface requirements. Section 2 is concerned with the HAL* software formulation of the algorithms. This section contains structured flowcharts of the procedures, estimates of flight computer core storage and CPU time, and processing requirements. Section 3 contains a glossary of the symbols used to define the software requirements and formulation, and the Appendices contain material which is supportive in nature to the preceding sections.

---

* HAL is the language of the Space Shuttle flight computer.

FOREWORD

This document details a preliminary baseline design for the Orbital Maneuvering System (OMS) Failure Detection and Identification (FDI) algorithms. This report is intended to be a comprehensive presentation of the material introduced by The Charles Stark Draper Laboratory, Inc. SSV Memo 75-10C-43, "Preliminary OMS FDI Algorithm Description Report."

This publication describes the design of the OMS FDI software in its present state of development. The primary intent of this document is to provide a reference for the OMS FDI algorithms incorporated into the Redundancy Management (RM) module. It does not specify either the structure or the design of the entire RM module, but presents the software functional requirements and software formulation of that portion of the RM module which is concerned with OMS FDI. The design of the OMS FDI software is continually undergoing revision, and the integration of OMS FDI software with the remainder of the RM software is a concern to be addressed in the future. Consequently, the material contained herein should be considered a snapshot of an evolving process.

The OMS FDI system presented in this report consists of an OMS FDI executive, an OMS engine FDI procedure, an OMS actuator FDI procedure, and two input interface routines. The OMS FDI executive interrogates the status of mode or event flags set by higher-level software, performs various initialization actions, and calls the OMS engine and actuator FDI procedures and their respective input interface routines. The OMS engine and actuator FDI procedures monitor OMS engine thrust performance and OMS gimbal actuator performance, respectively, and set failure flags which signify the fault status of these OMS components. The function of an input interface routine is to restructure RM input data into a form acceptable to its corresponding OMS FDI procedure.

The OMS FDI algorithms have been coded in HAL and will be implemented and tested on the Statement Level Simulator (SLS) which is being developed at the Draper Laboratory for the testing of Space Shuttle flight programs written in HAL.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

SECTION 1

SOFTWARE FUNCTIONAL REQUIREMENTS

The functional requirements of the Orbital Maneuvering System (OMS) Failure Detection and Identification (FDI) algorithms in the Redundancy Management (RM) module are described in this section. The description begins with an overview which relates the RM module to the other Guidance, Navigation, and Control (GN&C) major functions, and defines its internal structure and interfaces only to the extent necessary for a clear understanding of the OMS FDI system. An illustration of the relationship between the OMS FDI procedures in the RM module and the Flight Control (FC) functions and vehicle systems with which it interacts completes the overview. The section continues with a discussion of the functional requirements of the OMS FDI executive. The OMS engine and actuator FDI algorithms are then discussed individually. For each algorithm, its objectives, assumptions, input-output requirements, and a functional description illustrated by a functional block diagram are presented. Also included is a discussion of the functional requirements of each algorithm's input interface routine. The functional block diagrams in this section relate on a one-for-one basis with the actual HAL procedure flowcharts in Section 2. A glossary of the symbols used in this section to represent computer variables appears in Section 3.

1.1 Overview

The FC module controls the attitude and translation of the Space Shuttle Orbiter (SSO) during the on-orbit flight phase by utilizing as effectors the Reaction Control System (RCS) and/or Thrust Vector Control (TVC) of the OMS. The RCS consists of 38 primary fixed jets (900-pound thrust) and 6 vernier fixed jets (25-pound thrust). The OMS is a pair of 6000-pound-thrust rocket engines which can be gimballed independently of one another in pitch and yaw by electromechanical gimbal actuators. The OMS provides the propulsive thrust for orbit insertion, orbit circularization, orbit transfer, rendezvous, and deorbit. The OMS FDI function, however, is contained within the RM module. Figure 1-1 shows the relationship of the RM and FC modules to one another, to the other GN&C major functions, and to the Space Shuttle sensors, effectors, controls, and displays. The RM module is scheduled by the Moding, Sequencing, and Control (MSC) software via the Flight Computer Operating System (FCOS).

The primary OMS FDI processing is accomplished by two procedures: the OMS engine FDI procedure, which monitors OMS engine thrust performance, and the OMS actuator FDI procedure, which monitors OMS gimbal actuator performance. Other OMS FDI-related functions in the RM module are the OMS FDI executive and the input interface routines. The internal structure and interfaces of the RM module (only insofar as they relate to OMS FDI) are shown in Figure 1-2. The OMS FDI executive is responsible for scheduling the OMS engine and actuator FDI procedures and the input interface routines, whose general function is to translate data supplied externally to the RM module into a form acceptable to the FDI procedure. The OMS engine and actuator FDI procedures may run concurrently.

The interactions between the OMS FDI procedures, the relevant functions of the FC module, and other related vehicle components and systems are shown in Figure 1-3. OMS TVC steering commands (from the Guidance module) drive the OMS TVC Digital Autopilot (DAP) in the FC module. The OMS TVC DAP supplies OMS gimbal deflection commands to the OMS gimbal actuator servomechanism.* The actuator output extension causes the engine to deflect and the resulting vehicle rotational and translational dynamics are sensed by the IMU. The IMU attitude signals are returned to the FC module for use in the state estimator and various other submodules (not shown). The roll disturbance acceleration estimate of the FC state estimator, the FC-generated OMS engine ON/OFF commands, and IMU velocity signals are inputs to the OMS engine FDI procedure in the RM module. The FC-generated OMS gimbal deflection commands and the OMS actuator output extension (sensed by a position transducer) are inputs to the OMS actuator FDI procedure. Note that in addition to their function as crew display failure indicators, the OMS engine failure flags are used to automatically reconfigure the FC module. Similarly, the OMS actuator failure flags are used by RM to activate redundant actuator components.

---

* Only one OMS gimbal actuator channel is shown here for simplicity.

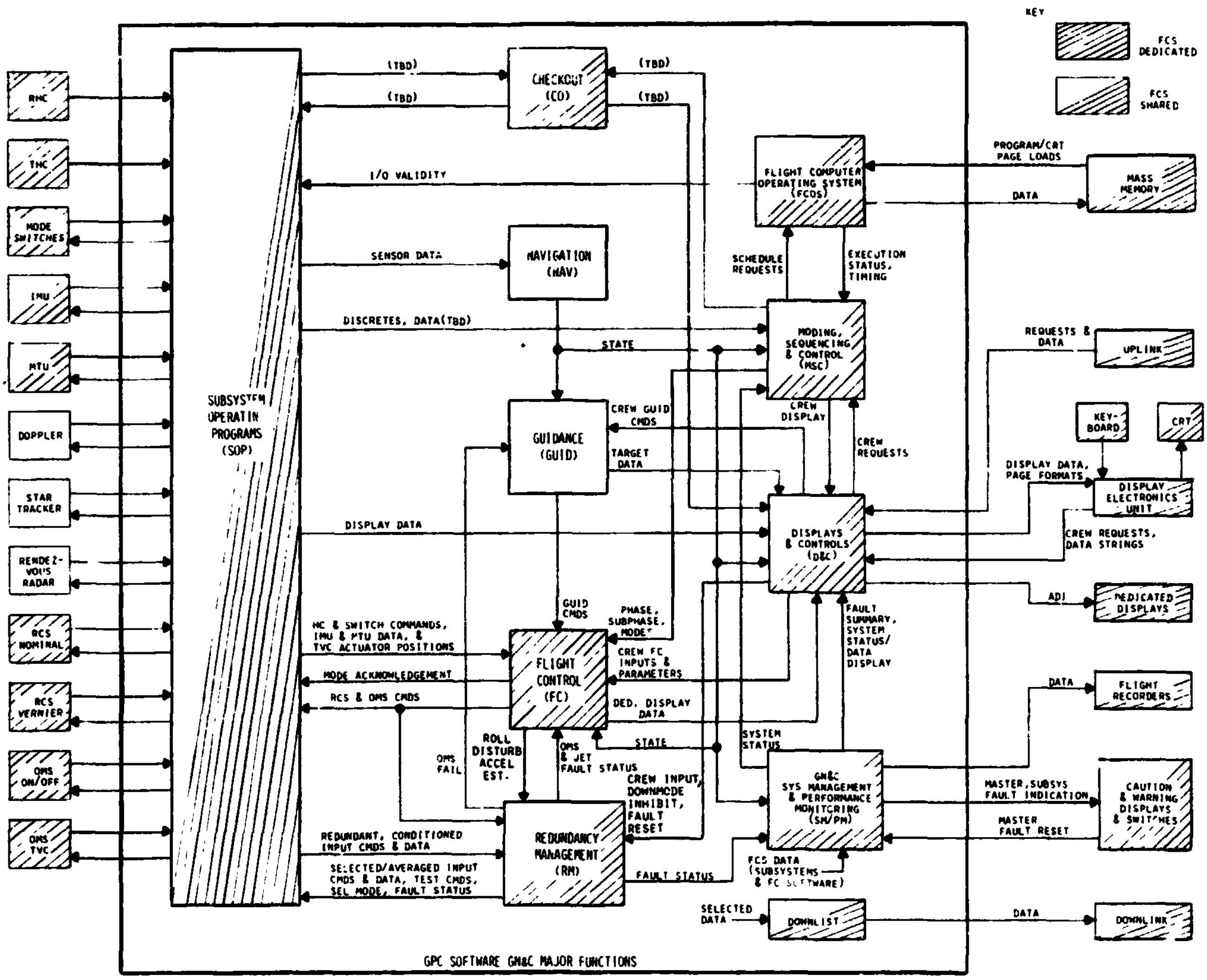

Figure 1-1. Relationship between the RM module and other GN&C major functions.

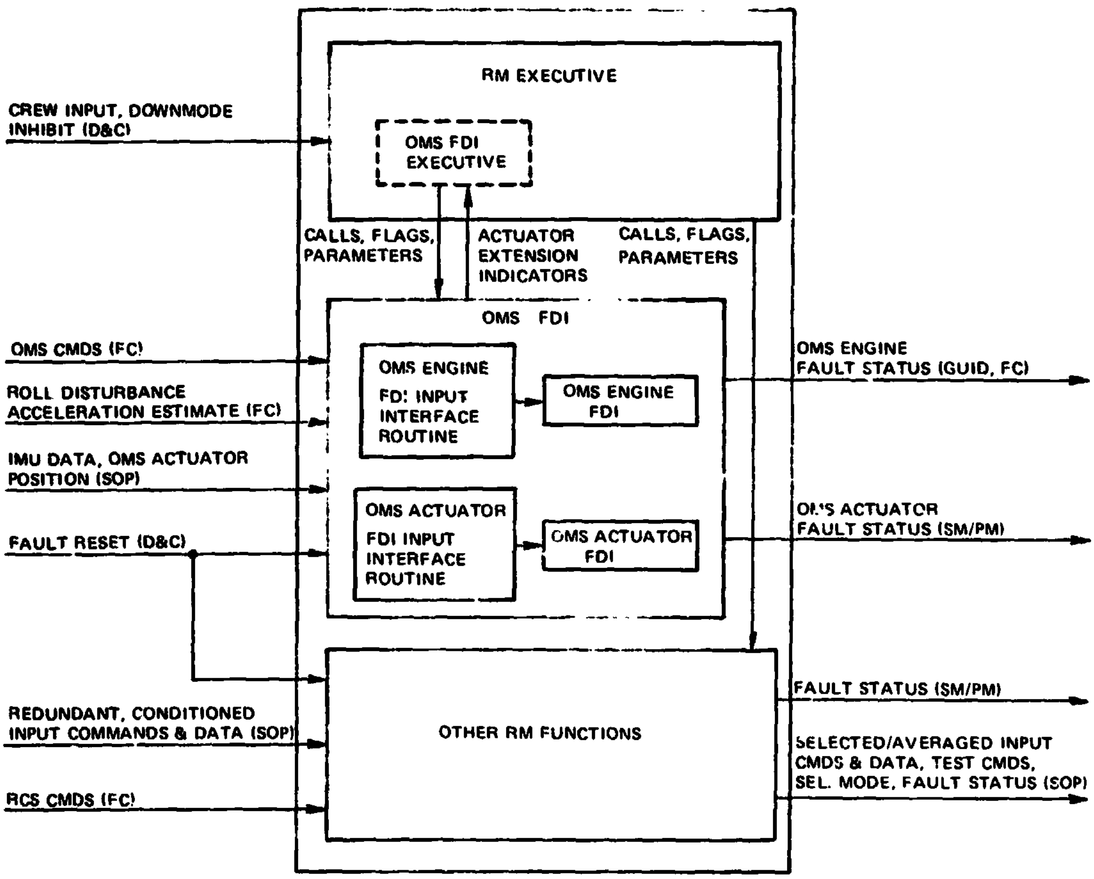

Figure 1-2. Relationship between the OMS FDI executive and the OMS FDI procedures within the RM module.

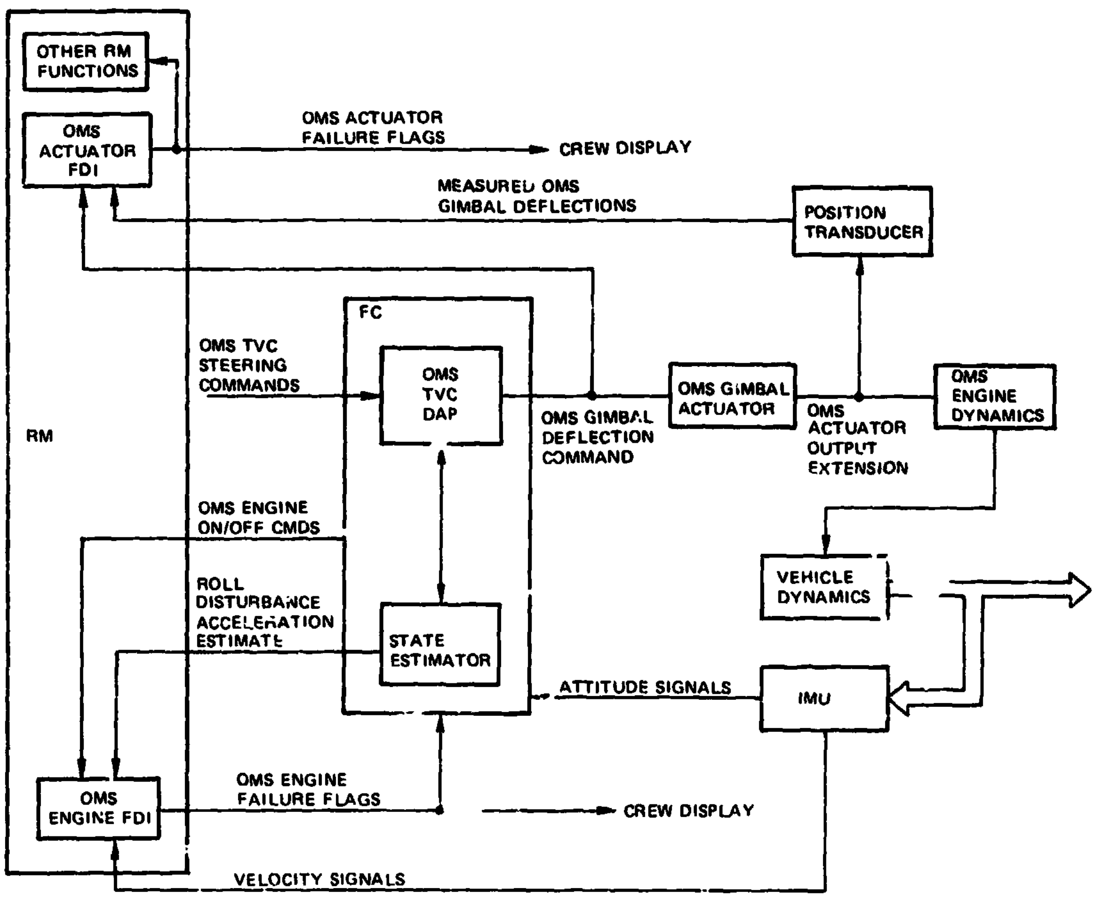

Figure 1-3. Block diagram relating the OMS FDI procedure to the FC module, vehicle dynamics, and the IMU.

1.2 OMS FDI Executive

The OMS FDI executive is that subset of the RM executive which performs the following functions:

(1) OMS engine and actuator FDI procedure calls.

(2) Input interface routine calls.

(3) Initialization.

It should be noted here that at the time of publication of this report it is not clear whether the OMS FDI executive will be a separate procedure or part of the RM executive.

The primary task of the OMS FDI executive software is to interrogate the status of mode or event flags in order to schedule the OMS FDI procedures and the input interface routines during those flight phases for which OMS FDI processing is needed. In general, these mode or event flags are set or cleared by higher-level software associated with crew interface and mission sequencing. For example, during normal OMS operation, a mode or event flag would be set indicating that an OMS burn is in progress, and the OMS FDI executive would begin invoking both the OMS engine and actuator FDI procedures and the input interface routines. To carry the example further, if an OMS burn and OMS FDI processing are in progress and a mode or event flag is set indicating the initiative of RCS-assist for attitude control, then the OMS FDI executive would terminate OMS engine FDI processing (for reasons explained in Section 2.3.3). As one final example, if a test of the OMS gimbal actuators were to be performed before each OMS burn, a mode or event flag indicating that situation would be set, and the OMS FDI executive would begin only OMS actuator FDI processing. To summarize, in response to the status of various mode or event flags, the OMS FDI executive performs calls to the FDI procedures and the input interface routines. These calls are performed in the proper sequence and with the proper frequency such that OMS engine and actuator performance are monitored correctly.

The OMS FDI executive is also responsible for certain initialization actions:

(1) Before the first entry into the OMS engine and actuator FDI procedures (i.e., before the first instance of OMS operation), the OMS FDI executive must ensure that the OMS engine and actuator failure flags are initialized to OFF.

(2) The OMS FDI executive will also have to reinitialize OMS engine or actuator failure flags in response to crew input. For example, if an OMS engine or actuator failure flag had been turned ON in the past, the crew might decide to override the flag either to force the use of a previously failed component or because it was thought that a false alarm had occurred.

(3) The OMS FDI executive is responsible for setting individual initialization flags used as inputs to the OMS engine and actuator FDI procedures upon first entry into each of the procedures, or upon entry after a period during which a procedure was not being scheduled. (The initialization flag is set for only one call to the procedure each time and is cleared immediately thereafter.)

The actual software formulation of the OMS FDI executive will depend upon the general requirements discussed in this section and the more specific processing requirements discussed in Sections 2.3.3 and 2.4.3.

1.3 OMS Engine FDI

1.3.1 Objective

The objective of the OMS engine FDI algorithm is to detect and identify off-nominal thrust performance of the two OMS engines.

Off-failures of one or both engines are detected by comparing the actual increment in the added velocity due to OMS thrust over a specified time interval to one-engine and two-engine threshold values. This velocity increment is derived from IMU-mounted accelerometer data. Ignition of either engine (if not commanded) is considered to be an on-failure, and is detected in the same manner.

Identification, or pinpointing a failure to a specific engine, is accomplished by testing the roll disturbance acceleration estimate as generated by the on-orbit FCS state estimator. The roll disturbances acceleration estimate is essentially the difference between the measured acceleration and the predicted (modelled) acceleration.

If a failure is detected but not yet identified, a single failure-detection flag will be activated. Identified failures, on the other hand, cause individual failure flags, one for each engine, to be activated.

1.3.2 Assumptions

The following assumptions apply to the OMS engine FDI algorithm:

(1) The OMS engine FDI algorithm is capable of detecting and identifying hard failures only; i.e., full-off or full-on failures.

  (a) A full-off failure is defined as the case in which an engine is providing essentially zero thrust when commanded ON.

  (b) A full-on failure is defined as the case in which an engine is providing essentially full thrust when commanded OFF.

(2) The OMS engine FDI algorithm is capable of detecting and identifying single engine failures and failures of both engines whether simultaneous or sequential.

(3) The nominal operational alignment of the OMS engines is such that the OMS thrust vectors are parallel and point in a direction which results in zero net torque on the vehicle. (For this alignment, the thrust vectors will be in or near the vehicle XZ plane.)

(4) OMS engine FDI processing is inhibited during RCS jet firings.

1.3.3 Input-Output Requirements

The inputs to the OMS engine FDI procedure are:

(1) The IMU measured added velocity since the beginning of the OMS burn.

(2) The roll disturbance acceleration estimate from the on-orbit FC state estimator.

(3) The OMS engine ON/OFF commands.

(4) An initialization flag.

The outputs of the OMS engine FDI procedure are:

(1) An OMS detect flag which indicates that an OMS engine failure has been detected.

(2) Two OMS failure flags which indicate that an OMS engine failure has been identified.

Table 1-1 lists these parameters. Included for each variable or constant are its computer and mathematical notation, description, units, value or range, and sampling rate.

Table 1-1. Input-output requirements for OMS_ENGINE_FDI.

Category	Name of Variable or Constant: Computer Notation	Name of Variable or Constant: Mathematical Notation	Description of Variable or Constant	Units	Value of Range	Sample Rate (words/s)
Inputs	ACCUM_ DELTA_V	$\underline{v}_a$	Added velocity due to OMS thrust	ft/s	$0 \leq x < 3.0$	25
	ROLL_DISTURB_ ACCEL	$\hat{a}_{d_r}$	Roll disturbance acceleration estimate	deg/s^2	$-2.0 \leq x \leq 2.0$	25
	OMS1_ON_CMD	OMS1_ON_CMD	OMS Engine 1 ON/OFF command	None	0,1	25
	OMS2_ON_CMD	OMS2_ON_CMD	OMS Engine 2 O /OFF command	None	0,1	25
	OMSE_INIT_ FLAG	OMSE_INIT_ FLAG	Initialization flag	None	0,1	1
Outputs	OMS1_FAIL	OMS1_FAIL	OMS Engine 1 failure flag	None	0 1	25
	OMS2_FAIL	OMS2_FAIL	OMS Engine 2 failure flag	None	0,1	25
	OMS_FAIL_ DETECT	OMS_FAIL_ DETECT	OMS engine failure detection flag	None	0,1	25

1.3.4 Functional Description

The logic for activating the three failure flags which can be set for off-nominal performance of the OMS engines is shown in Figure 1-4.

Note that the OMS engine FDI procedure makes two basic tests (explained in the following) during its cycle. The procedure is called once per second, and the variables involved in the tests are:

(1) The computed increment in the added velocity due to OMS thrust which has occurred (as sensed by IMU-mounted accelerometers) over the 1-second interval since the last call.

(2) The roll disturbance acceleration estimate from the FC state estimator.

By comparing the actual sensed increment in added velocity with threshold values for one and two engines, the number of engines that are truly firing is determined. By inspecting the roll disturbance acceleration estimate a failure of either engine, if one has occurred, can be identified. In the case of a failure, the failure flag for Engine 1, OMS1_FAIL, or for Engine 2, OMS2_FAIL, will be activated if the roll disturbance acceleration estimate is outside a deadzone. The sign of the roll disturbance acceleration estimate determines which engine has failed. If Engine 1 is firing, the roll disturbance acceleration is always negative, and if Engine 2 is firing, the roll disturbance acceleration is always positive, provided that the disturbance is caused by a failure of a single engine during a two-engine burn or when neither engine is commanded ON. (Assumption 3 in Section 1.3.2 ensures that there will be a disturbance acceleration.)

If the roll disturbance acceleration estimate is inside the deadzone, either the failed engine cannot be identified, or there has been no failure (i.e., single-engine burn). However, the third flag, OMS_FAIL_DETECT will be activated if both engines have the same ON/OFF command (i.e., not a single-engine burn).

Latching features are incorporated in the logic so that a failure flag is not cleared if it were set during any previous cycle. That is, a failure flag which had been previously set to ON will not be reset to OFF when the engine command for that failed engine is reset to OFF.

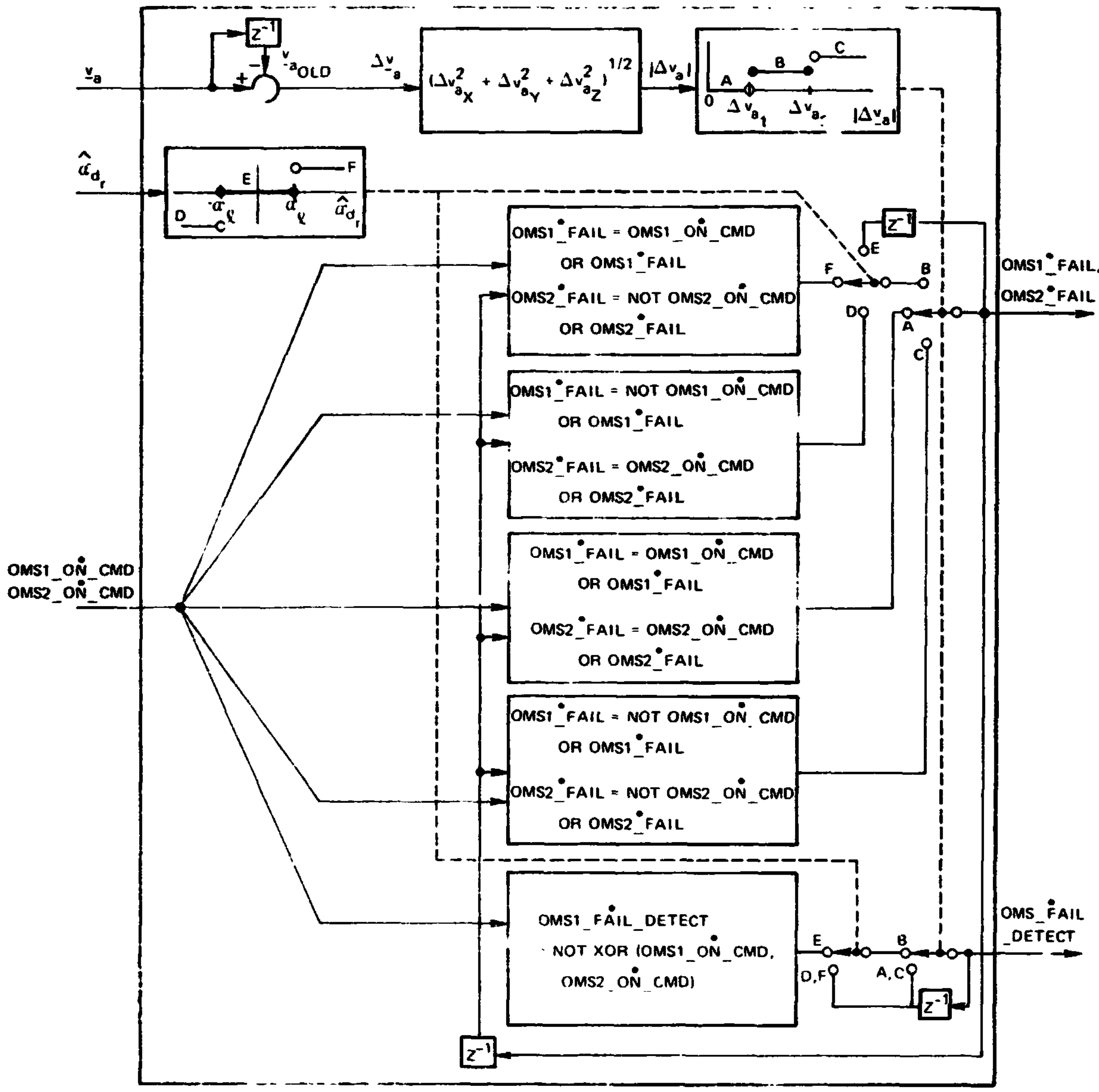

Figure 1-4. Functional block diagram of OMS_ENGINE_FDI.

When two engines are firing, the failure flags will be set to OFF as long as both engines have been commanded to fire. However, if it is determined that two engines are firing, but only one engine has been commanded to fire, then the failure flag of the engine not commanded to fire will be set to ON.

Similarly, when one engine is firing, a failure flag will not be activated as long as the commanded engine is firing. It it is determined that one engine is firing and the roll disturbance test indicates that the wrong engine is firing, the appropriate failure flag will be set to ON.

Note that since the OMS_FAIL_DETECT flag is activated by an exclusive-or function in the single-engine-firing case (with no appreciable roll disturbance), it will be set to ON only if both engines have the same firing commands. That is, the flag will come ON if either both engine commands are ON or both engine commands are OFF. Those two possible combinations of commands in conjunction with thrust from only a single engine imply a failure.

Possible combinations of failure flags generated by the OMS_ENGINE_FDI procedure for the cases of two, one, or zero engines firing are discussed further and illustrated in Section A.3 of the Appendix.

1.3.5 OMS Engine FDI Input Interface Routine

The OMS engine FDI input interface routine accepts IMU accelerometer data from the Subsystem Operating Programs (SOP) and derives from that data the velocity added to the vehicle due to OMS thrust since the beginning of the current OMS burn.

Essentially, this input interface routine reformats SOP velocity information into the quantity, $\underline{v}_a$, specified in the input-output requirements of the OMS engine FDI procedure given in Section 1.3.3.

1.4 OMS Actuator FDI

1.4.1 Objective

The objective of the OMS actuator FDI algorithm is to detect and identify off-nominal performance of the pitch and yaw gimbal actuators of the two OMS engines. Since inputs for performance monitoring of the four actuators are made available and processed separately for each actuator, the failure detection and identification problem reduces to one of detection only. Failures are detected by testing the increment in the measured gimbal deflection over a specified interval of time if, during that time, the actuator is commanded to deflect the engine gimbal continuously in the same direction, and the gimbal is not driven to a stop. Two successive failure indications are necessary before an OMS actuator failure flag is set.

The four OMS actuator failure flags are monitored for activation of redundant actuator channels and for display to the crew.

1.4.2 Assumptions

The following assumptions apply to the OMS actuator FDI algorithm:

(1) The OMS actuator FDI algorithm is capable of detecting and identifying full-off failures only.

(2) An unfailed OMS actuator achieves a steady-state nominal extension rate within two minor cycles (80 ms) in response to an applied voltage.

(3) The accuracy of the OMS actuator output position transducer is sufficient to permit the use of gimbal deflection increments over six minor cycles (240 ms) to detect full-off failures with zero probability of a false alarm.

(4) The OMS actuator FDI algorithm is not capable of detecting failures downstream of the actuator output, i.e., in the gimbal mounting structure.

1.4.3 Input-Output Requirements

The inputs to the OMS actuator FDI procedure are:

(1) The OMS gimbal deflection commanded for each actuator from the Thrust Vector Control (TVC) DAP in the FC module.

(2) The OMS gimbal deflection for each actuator as measured by the actuator output position transducer, a linear voltage differential transformer.*

(3) A procedure call counter.

(4) An initialization flag.

---

* The actuator extension length (measured from null), rather than the OMS gimbal angle itself, is the sensed quantity.

The outputs of the OMS actuator procedure are:

(1) Four OMS actuator failure flags.

(2) Four OMS actuator extension indicators, which signify whether each actuator is being commanded to extend, retract, or remain stationary.

Table 1-2 lists these parameters. Included for each variable or constant are its computer and mathematical notation, description, units, value or range, and sampling rate.

Table 1-2. Input-output requirements for OMS_ACTUATOR_FDI.

Category	Name of Variable or Constant: Computer Notation	Name of Variable or Constant: Mathematical Notation	Description of Variable or Constant	Units	Value of Range	Sample Rate (words/s)
Inputs	[OMS_GIMBAL]	$[\delta_m]$	Array of measured OMS gimbal deflections	deg	$-8.0 \le x \le 8.0$	25
	[OMS_GIMBAL_CMD]	$[\delta_c]$	Array of commanded OMS gimbal deflections	deg	$-8.0 \le x \le 8.0$	25
	OMSA_CALL COUNTER	OMSA_CALL COUNTER	Procedure call counter	None	1,2,3,4,5,6,7,8,9	1
	OMSA_INIT_FLAG	OMSA_INIT_FLAG	Initialization flag	None	0,1	1
Outputs	[OMS_ACTUATOR_FAIL]	[F]	Array of OMS actuator failure flags	None	0,1	25
	[OMS_ACTUATOR_EXTEND]	[E]	Array of OMS actuator extension indicators	None	-1,0,1	1

1.4.4 Functional Description

Each of the OMS engines is equipped with two gimbal actuators which are used to control nozzle deflections in pitch and yaw. The pitch and yaw actuators are identical in design except for the stroke length, and contain redundant channels which couple to a common drive assembly.

The OMS gimbal actuator servo loop is described in more detail in References 2, 3, 4, and 5. However, familiarity with its operation will aid in understanding the OMS actuator FDI procedure. Figure 1-5 is a simplified block diagram of an OMS gimbal actuator servo loop. The gimbal deflection command is differenced with the measured gimbal deflection to obtain the gimbal deflection error. The deflection error is then supplied to a servo amplifier which is a bang-bang amplifier with deadzone and hysteresis. The output of the servo amplifier is the actuator input voltage which drives the actuator motor. The output of the actuator motor is the actuator extension measured from null. A Linear Voltage Differential Transformer (LVDT) position transducer senses the actuator output extension and converts it to the measured gimbal deflection. The <u>actual</u> gimbal deflection is the output of the OMS gimbal mounting structure dynamics, while the <u>measured</u> gimbal deflection is inferred from the actuator output extension.

If the gimbal deflection error is outside the servo amplifier deadzone, the actuator is being commanded to extend or retract. If the gimbal deflection error is within the servo amplifier deadzone, the actuator may or may not be commanded to extend or retract depending upon the past history of the gimbal deflection error. For purposes of OMS actuator failure detection, the actuator is assumed to be continuously extending or retracting only if the gimbal deflection error remains outside the servo amplifier deadzone.

The OMS actuator FDI algorithm determines whether an actuator has failed by testing the increment in the measured gimbal deflection which occurs over six minor cycles, if the actuator is being commanded to extend or retract continuously for that period (according to the definition given immediately above), and if the engine gimbal is not driven to a stop. The OMS actuator FDI procedure is called at least once every second. It may be called on eight successive minor cycles after the first pass, depending upon the results of the first and subsequent passes. The "procedure call counter" input indicates to the procedure which of the nine possible passes is occurring. The "actuator extension

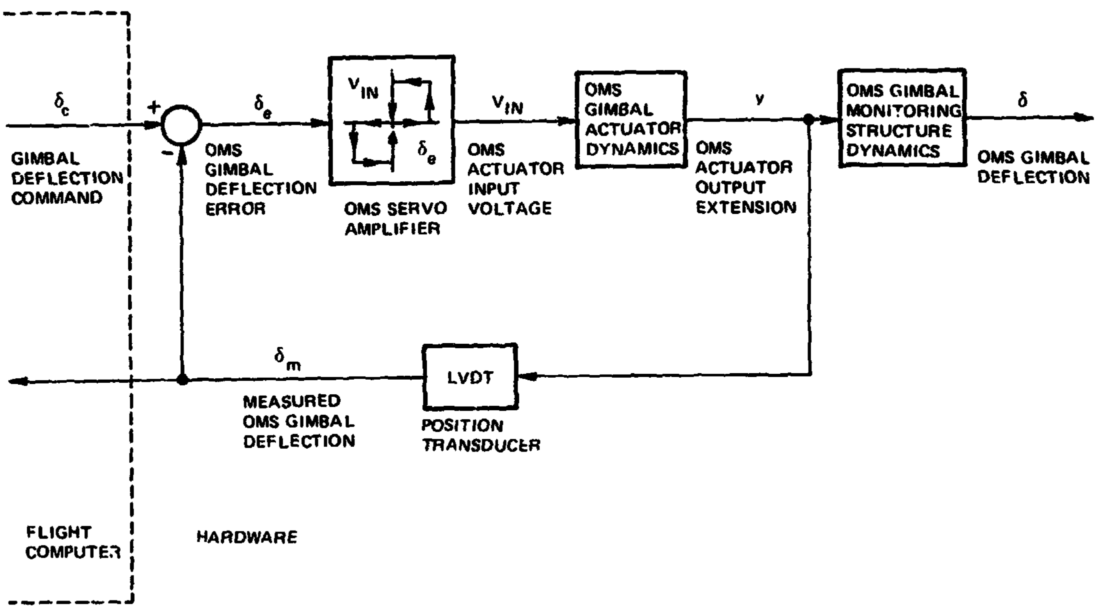

Figure 1-5. Simplified block diagram of an OMS gimbal actuator servo loop.

indicator" outputs are used by the OMS FDI executive to determine whether the next call should be executed. An explanation of what occurs on each pass is given in Sections 1.4.4.1 through 1.4.4.5.

1.4.4.1 First Pass

Once per seccnd a first call is made to the procedure. Figure 1-6 is a functional block diagram of this first pass, and illustrates actuator FDI actions at this point of the procedure. Note that the actuator extension indication, E_{OLD}, is established on this pass. Note also that the actuator failure flag, F, does not change status.

On this pass, and on all subsequent passes, the gimbal deflection error of each actuator is computed and checked as shown in Figure 1-7. The extension indicator for each actuator is set to -1, 0, or +1, according to whether that actuator is being commanded to retract, remain stationary, or extend, respectively. Also, the extension indicator is set to zero to inhibit gimbal deflection increment threshold testing if the measured gimbal deflection indicates that the engine gimbal is at a stop. The failure counter is set to zero only if the gimbal deflection error is within the servo amplifier deadzone.

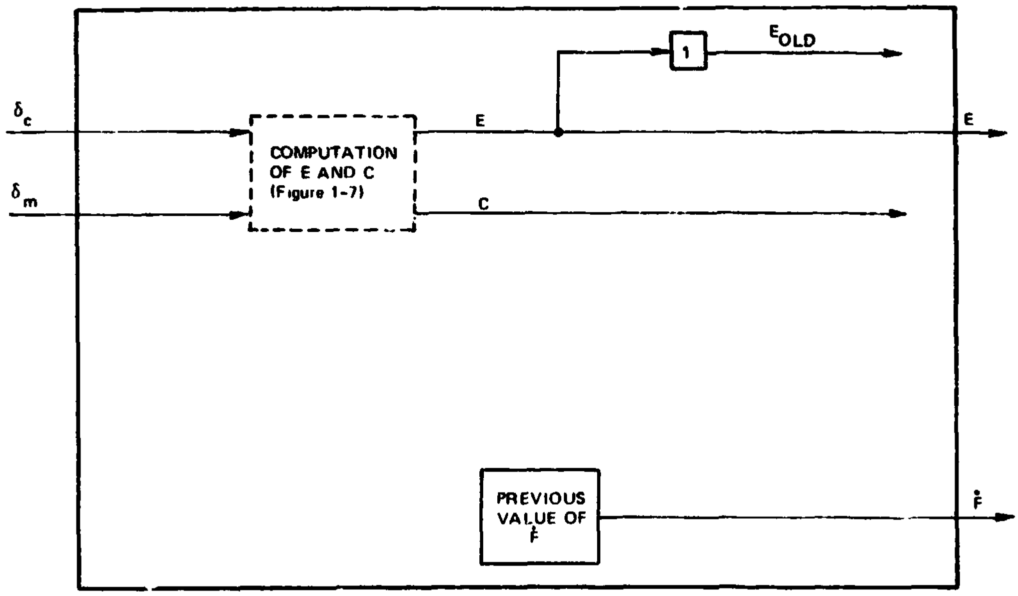

Figure 1-6. Functional block diagram of the first pass of OMS_ACTUATOR_FI .

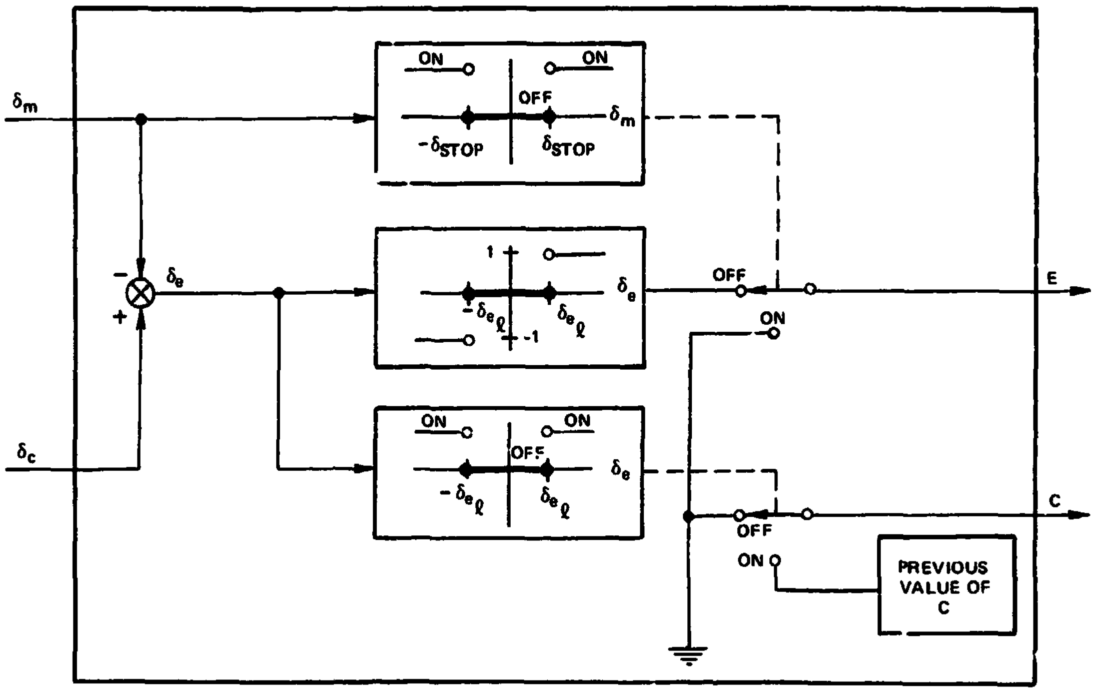

Figure 1-7. Functional block diagram of the computation of E and C within OMS_ACTUATOR_FDI.

1.4.4.2 Second Pass

One minor cycle (40 ms) after the first call, the second call is made by the OMS FDI executive if any actuator extension indicator output from the previous pass is nonzero. On this pass, the gimbal deflection error and extension indicator are recomputed for those actuators which had a nonzero extension indicator output from the first pass. If the second-pass value differs from the first-pass value, the extension indicator and the failure counter are set to zero. The actuator failure flag does not change status.

These actions are illustrated in Figure 1-8, and are identical to those occurring during the fourth through eighth passes.

1.4.4.3 Third Pass

One minor cycle after the second call, the third call is made by the executive procedure if any actuator extension indicator output from the previous pass is still nonzero. On this pass, the gimbal deflection error and extension indicator are computed for those actuators which had a nonzero extension indicator output from the second pass.

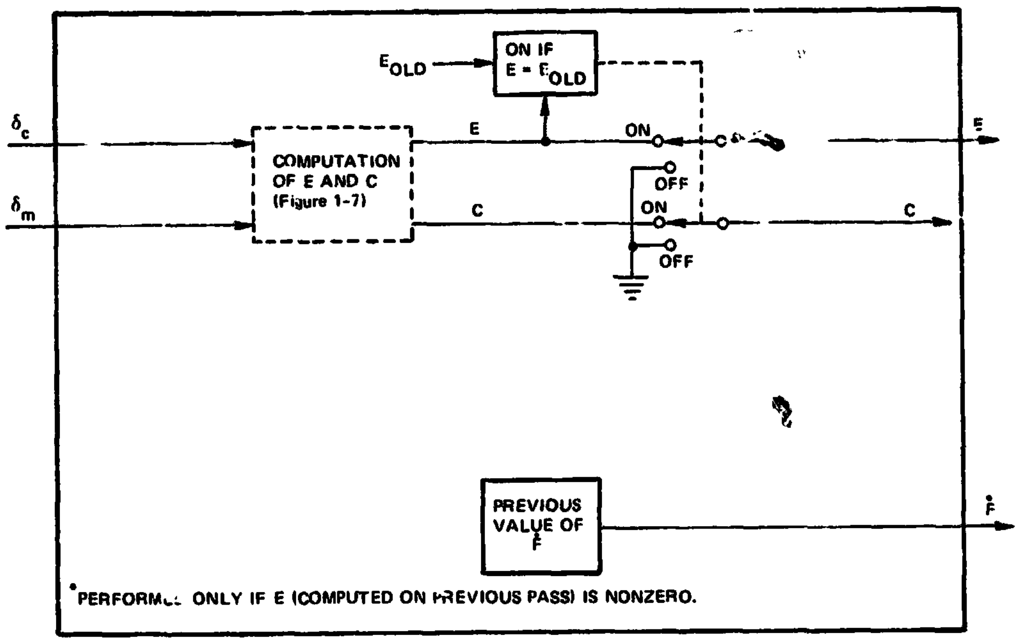

Figure 1-8. Functional block diagram of the second and fourth through eighth passes of OMS_ACTUATOR_FDI.*

Note from Figure 1-9 that if the third-pass value of the extension indicator is equal to the first-pass value, the present measured gimbal deflection is saved as $\delta_{m_{OLD}}$. Otherwise, the extension indicator and failure counter are set to zero.

The actuator failure flag does not change status.

1.4.4.4 Fourth through Eighth Passes

These passes perform the same operations as the second pass.

1.4.4.5 Ninth Pass

One minor cycle after the eighth call, the ninth call is made by the OMS FDI executive if any actuator extension indicator output from the previous pass is still nonzero. On this pass, the gimbal deflection error and extension indicator are recomputed for those actuators which had a nonzero extension indicator output from the eighth pass.

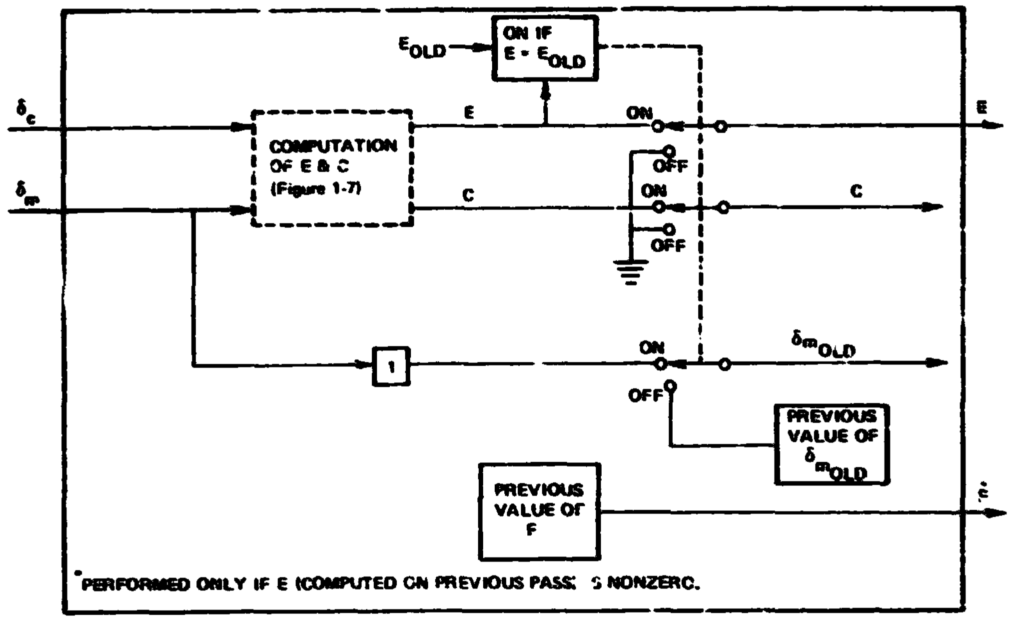

Figure 1-9. Functional block diagram of the third pass of OMS_ACTUATOR_FDI.*

Refer to Figure 1-10. If the ninth-pass value of the extension indicator differs from the first-pass value, the extension indicator and the failure counter are set to zero. If the ninth-pass value is the same as the first-pass value and the product of the extension indicator and gimbal deflection increment is less than a threshold value, the failure counter will be incremented by one; otherwise the failure counter is set to zero. The actuator failure flag, F, is set to ON if the failure counter has reached two.

The primary functions of each pass can be summarized on a time scale (see Figure 1-11).

The OMS actuator FDI algorithm waits two minor cycles (80 ms) after the first call before recording the first measured gimbal deflection because an unfailed OMS actuator requires a finite time to achieve a steady-state nominal extension rate in response to an applied voltage. Results published in References 2 and 5 and also unpublished results (by L. A. D'Amario) of simulated OMS actuator behavior indicate that it requires less than 40 ms (one minor cycle) for an OMS actuator to achieve a steady-state nominal deflection rate starting from a rest initial

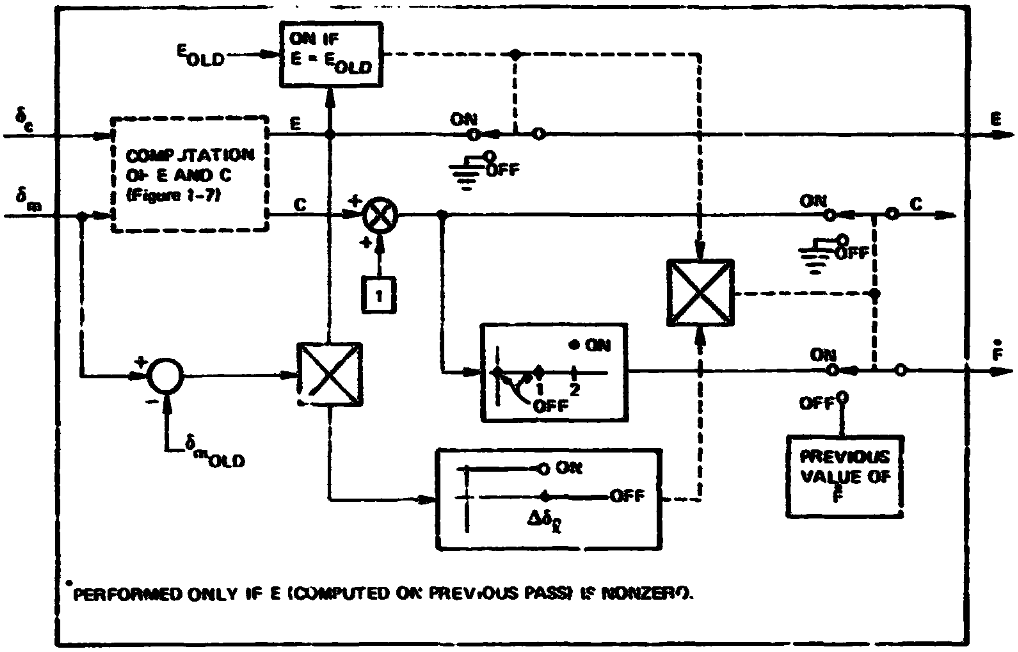

Figure 1-10. Functional block diagram of the ninth pass of OMS_ACTUATOR_FDI.*

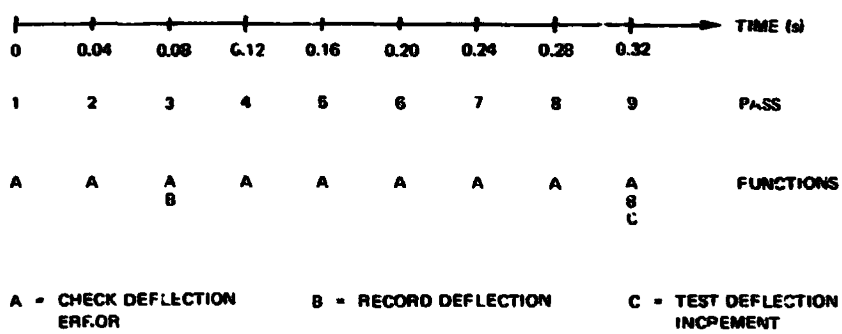

Figure 1-11. Primary functions of each pass of OMS_ACTUATOR_FDI illustrated on a time scale.

condition. Further, it requires between 40 ms (one minor cycle) and 80 ms (two minor cycles) if, as an initial condition, the actuator is extending at a steady-state rate in a direction opposite to the sense of the applied voltage

The OMS actuator FDI algorithm waits six minor cycles (240 ms) after recording the first measured gimbal deflection before recording the second measurement and testing the computed increment, so that the error in the measured increment relative to the nominal size of the increment is low enough that:

(1) An actuator which has experienced a full-off failure and is not moving has zero probability of passing the threshold test.

(2) An actuator which is performing nominally has zero probability of failing the threshold test.

The accuracy of the computed gimbal deflection increment is directly dependent on the accuracy of the OMS actuator position transducer. The effects of OMS actuator position transducer accuracy on FDI CPU time and the justification for waiting six minor cycles is discussed in Section 2.4.2.

In summary, if any OMS actuator is being commanded to extend or retract continuously (as previously defined in this section) for eight minor cycles, and if the engine gimbal is not driven to a stop, then the computed increment in the measured gimbal deflection over the last six cycles is compared to a threshold value to test actuator performance. Two successive failure indications (one second apart) are necessary before an actuator failure flag is set to ON. If the gimbal deflection error of any actuator is inside the deadzone on the first pass or moves into or through the deadzone after the first pass, or if the engine gimbal is indicated to be at a stop on any pass, then no gimbal deflection increment threshold test is performed during that second.

One of the assumptions in Section 1.4.2 was that the OMS actuator FDI algorithm is capable of detecting full-off failures only. Full-on failures will also be detected, but not directly. That is, the test performed on the measured gimbal deflection increment determines only whether the gimbal has deflected far enough, not whether it has deflected too far. For instance, if a full-on extension failure occurs, the gimbal actuator servo loop will eventually sense that the gimbal has extended too far and will command it to retract. At this point, a test under commanded retraction will be performed and failed. In other words, a full-on failure in gimbal extension becomes a full-off failure in gimbal retraction through the actions of the gimbal actuator servo loop.

1.4.5 <u>OMS Actuator FDI Input Interface Routine</u>

The OMS actuator FDI input interface routine accepts commanded OMS gimbal deflections from the FC TVC DAP and measured OMS gimbal deflections from the SOP. The measured OMS gimbal deflections are obtained from the OMS actuator output position transducer.

The commanded OMS gimbal deflections are provided by the OMS TVC DAP as two 2-dimensional arrays containing the pitch and yaw deflection commands for each engine. The actuator FDI input interface routine reformats the gimbal deflection commands into one 4-dimensional array, as specified in the input-output requirements in Section 1.4.3.

In addition, this input interface routine formats the measured OMS gimbal deflections into one 4-dimensional array, as specified also in Section 1.4.3.

SECTION 2

SOFTWARE FORMULATION

The software formulation of the OMS FDI algorithms is described in this section. A structured flowchart, estimates of flight computer core storage and CPU time, and processing requirements are presented for the OMS engine and actuator FDI procedures. A glossary of the symbols used in this section appears in Section 3.

2.1 Overview

2.2 OMS FDI Executive

The software of the OMS FDI executive has not yet been formulated. The OMS FDI executive may be a separate procedure, or it may be simply part of the larger RM executive. The actual software formulations, must satisfy the general functional requirements given in Section 1.2 and the more specific processing requirements discussed in Sections 2.3.3 and 2.4.3.

2.3 OMS Engine FDI

2.3.1 OMS_ENGINE_FDI: PROCEDURE

A structural flowchart for HAL procedure OMS_ENGINE_FDI is shown in Figure 2-1. A functional description of the operations illustrated in Figure 2-1 is given in Section 1.3.4.

Definitions of the computer variables and constants are given in the glossary in Section 3. Estimates of flight computer core storage and maximum possible CPU time derived from HAL-FC compilations are given in Section 2.3.2.

```
"""""""""""""""""""
" OMS_ENGINE_FDI:  "
"                  "--------" B L O C K   S U M M A R Y           "
" PROCEDURE;       "        "                                     "
"""""""""""""""""""         " COMPOOL VARIABLES USED:             "
  |                         "    DECLARED IN COMPOOL FDIPOOL:     "
  |                         "        ACCUM_DELTA_V                "
  |                         "        OMS_FAIL_DETECT*             "
  |                         "        OMSF_INIT_FLAG               "
  |                         "        OMS1_FAIL                    "
  |                         "        OMS1_FAIL*                   "
  |                         "        OMS1_ON_CMD                  "
  |                         "        OMS2_FAIL                    "
  |                         "        OMS2_FAIL*                   "
  |                         "        OMS2_ON_CMD                  "
  |                         "        ROLL_DISTURB_ACCEL           "
  |                         """""""""""""""""""""""""""""""""""""""
  |
"""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""
" DECLARE ACCUM_DELTA_V_INCR SCALAR SINGLE AUTOMATIC,                 "
"                                                                     "
"         ONE_ENGINE_THRESHOLD SCALAR SINGLE CONSTANT(1),             "
"                                                                     "
"         TWO_ENGINE_THRESHOLD SCALAR SINGLE CONSTANT(2),             "
"                                                                     "
"         ROLL_ACCEL_THRESHOLD SCALAR SINGLE CONSTANT(3),             "
"                                                                     "
"         OLD_ACCUM_DELTA_V VECTOR(3) SINGLE STATIC;                  "
"""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""
  |
  |
**************************
*                        *             THEN ************************************
* IF OMSF_INIT_FLAG = ON *-----------+-----*                                  *
*                        *           |     * OLD_ACCUM_DELTA_V = ACCUM_DELTA_V; *
**************************           |     ************************************
                                     |
                                     |ELSE ********************************************
                                     +-----*                                          *
                                           * ACCUM_DELTA_V_INCR = ABVAL(ACCUM_DELTA_V - *
                                           *                                          *
                                           * OLD_ACCUM_DELTA_V);                      *
                                           *                                          *
                                           * OLD_ACCUM_DELTA_V = ACCUM_DELTA_V;       *
                                           ********************************************
                                             |
                                             |
```

Figure 2-1. Structured automatic flowchart of OMS_ENGINE_FDI (sheet 1 of 5).

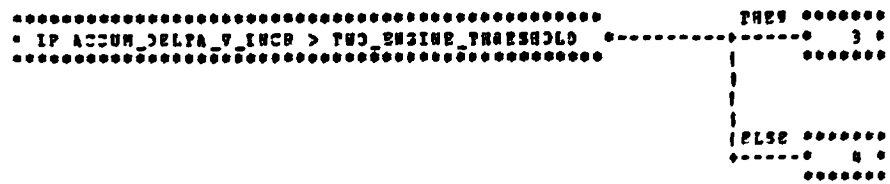

Figure 2-1. Structured automatic flowchart of OMS_ENGINE_FDI (sheet 2 of 5).

```
*******
*   3 *
*******
  |
  |
  |
*****************************************
*                                       *
* OMS1_FAIL = NOT OMS1_ON_CMD OR OMS1_FAIL; *
*                                       *
*                                       *
* OMS2_FAIL = NOT OMS2_ON_CMD OR OMS2_FAIL; *
*****************************************
```


Figure 2-1. Structured automatic flowchart of OMS_ENGINE_FDI (sheet 3 of 5).

```
*******
*   4 *
*******
   |
   |
   |
***********************************************************                THEN ****************************************
* IF ACCUM_DELTA_V_INCR < OMF_ENGINE_THRESHOLD          *----------+------*                                      *
***********************************************************           |      * OMS1_FAIL = OMS1_ON_CMD OR OMS1_FAIL; *
                                                                      |      *                                      *
                                                                      |      *                                      *
                                                                      |      * OMS2_FAIL = OMS2_ON_CMD OR OMS2_FAIL; *
                                                                      |      ****************************************
                                                                      |
                                                                      |
                                                                      |ELSE ***********************************************         *******
                                                                      +-----* IF ROLL_DISTURB_ACCEL > ROLL_ACCEL_THRESHOLD *---------*   5 *
                                                                            ***********************************************         *******
```


Figure 2-1. Structured automatic flowchart of OMS_ENGINE_FDI (sheet 4 of 5).

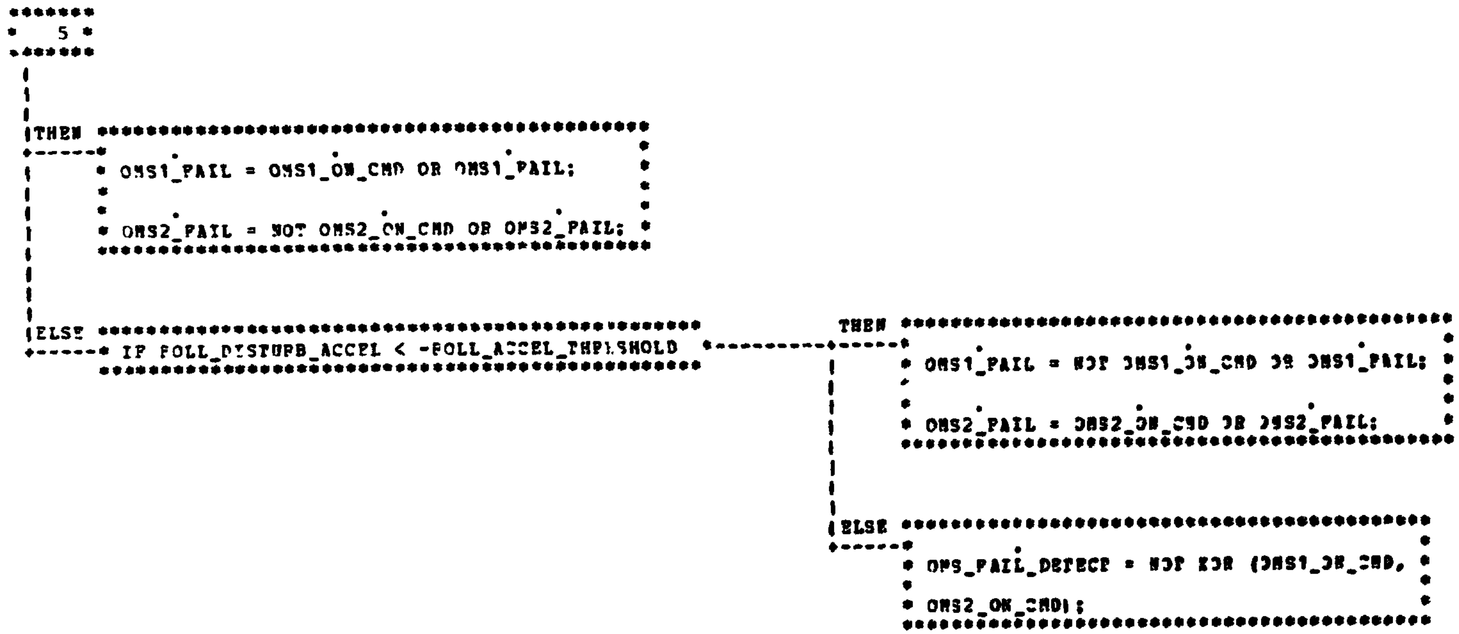

Figure 2-1. Structured automatic flowchart of OMS_ENGINE_FDI (sheet 5 of 5).

2.3.2 Flight Computer Core Storage and CPU Time

The following OMS engine FDI algorithm data is derived from HAL-FC compilations:

Flight Computer Core Storage:

Data	=	11 words
Procedure	=	70 words
Total	=	81 words

Flight Computer CPU Time: 0.341 ms/s (0.034%)

Core storage requirements are computed by the HAL compiler and printed out as part of the compilation, while the CPU time is an estimate of the maximum possible time required to execute all the computations of the procedure.

The maximum time estimate is based on combining the CPU times for each statement in the longest possible path (timewise) through the procedure's logic. The CPU times for each statement are also computed as part of the compilation.

2.3.3 Processing Requirements

The procedure OMS_ENGINE_FDI is normally called once per second during an OMS burn. It may also be called prior to and after an OMS burn for on-failure detection. There are three situations, however, in which OMS engine FDI processing should be inhibited:

(1) During OMS thrust buildup immediately after commanded engine ignition.

(2) During OMS thrust tailoff immediately after commanded engine shutdown.

(3) During RCS DAP operation.

With respect to the first two situations, the problem is that thrust buildup and tailoff are not modeled in the computation of the increment in the added velocity due to OMS thrust. Because the procedure is designed to detect and identify full-off or full-on failures only, it implicitly assumes that the thrust from an OMS engine is either at the nominal design level or at zero. Thus, for example, during thrust build-up in a two-engine burn, it might appear to the OMS engine FDI procedure that the velocity increment over the last second indicates that only one engine is firing. This incorrect indication would cause a false alarm. Therefore, the procedure OMS_ENGINE_FDI should not be called during a 1-second interval immediately after commanded engine ignition or shutdown. The 1-second interval, which allows for thrust buildup or tailoff, is based on OMS data in Reference 3.

The third situation occurs whenever the RCS DAP is operating, thus causing RCS jets to fire. In the event of RCS jet firings, the increment in the added velocity due to OMS thrust, which is computed directly from IMU-mounted accelerometer information, would mistakenly contain contributions from RCS jet thrust also. The incorrect velocity increment could cause an incorrect determination of the number of engines which are firing, thus leading to misalarms or false alarms. This third situation, in which there is a conflict between RCS DAP operation and desired OMS engine FDI processing, can be identified for several flight control modes:

(1) RCS attitude-hold prior to OMS ignition (prevents pre-burn OMS on-failure detection).

(2) RCS attitude-hold after an OMS burn (prevents post-burn OMS on-failure detection).

(3) General RCS-assist for attitude control immediately after a single OMS failure during a two-engine burn (prevents FDI on the remaining good engine).*

(4) Roll-only RCS-assist during single-engine OMS operation:

   (a) For the remainder of a two-engine burn after a single failure.

   (b) For a single-engine burn (prevents FDI during single-engine operation).

The OMS engine FDI procedure should not be called during RCS DAP operation in any of these four situations.

Two qualifications of these statements should be noted here. The first involves the specific restrictions on OMS engine FDI processing given in items (1), (2), and (4). It may be possible to adjust the velocity increment thresholds (i.e., provide enough margin) to allow OMS engine FDI processing during these flight control modes. The

---

* The RCS DAP is not activated until a failure has been identified.

reasons are that during RCS attitude-hold and roll-only RCS-assist, the frequency and duration of RCS jet firings, relative to the 1-second interval over which the velocity increment is calculated, are such that the ΔV contribution due to RCS jet firings should be small relative to that of an OMS engine. Also, the candidate jets for use in roll-only RCS operation (item (4)) are limited. The direction and magnitude of their thrust is known, and if their ΔV contribution is small enough, it may be ignored. Further analysis and simulation results will resolve the question of the need for the restriction listed in items (1), (2), and (4).

The second qualification involves the restriction of items (1) through (4) in general. If the RCS DAP would compute on a periodic basis the expected ΔV due to RCS jet firings, then OMS engine FDI processing could continue in any situation of RCS DAP operation. The reason is that the ΔV contribution of RCS jet firings could then be subtracted from the IMU-derived velocity increment in order to obtain the contribution from OMS thrust only. This computation could be performed in either the FC or RM module.

2.4 OMS Actuator FDI

2.4.1 OMS_ACTUATOR_FDI: PROCEDURE

A structured flowchart for HAL procedure OMS_ACTUATOR_FDI is shown in Figure 2-2. A functional description of the operations illustrated in the Figure 2-2 is given in Section 1.4.4.

Definitions of the computer variables and constants are given in the glossary in Section 3. Estimates of flight computer core storage and maximum possible CPU time derived from HAL-FC compilations are given in Section 2.4.2.

2.4.2 Flight Computer Core Storage and CPU Time

The following OMS actuator FDI algorithm data are derived from HAL-FC compilations.

Flight Computer Core Storage:

Data	=	21 words
Procedure	=	83 words
Total	=	104 words

Flight Computer CPU Time = 7.665 ms/s (0.767%)

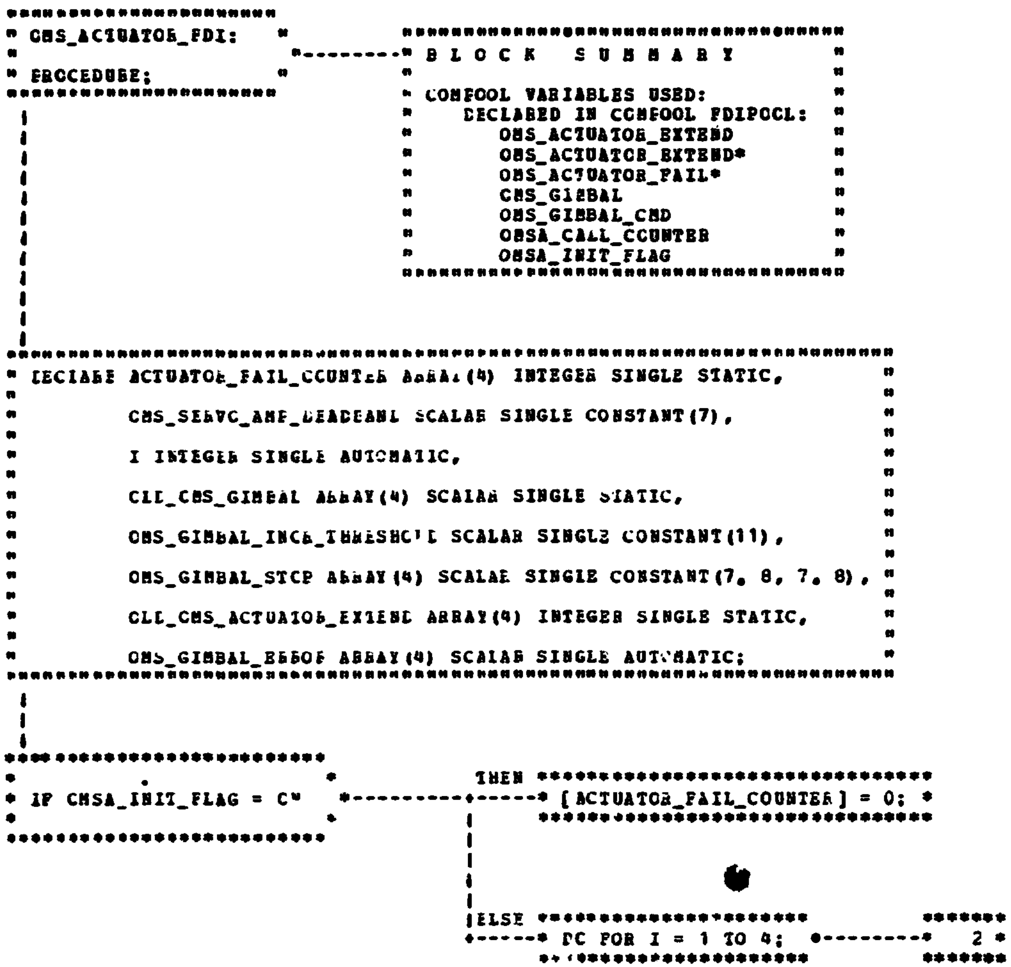

Figure 2-2. Structured automatic flowchart of OMS_ACTUATOR_FDI (sheet 1 of 7).

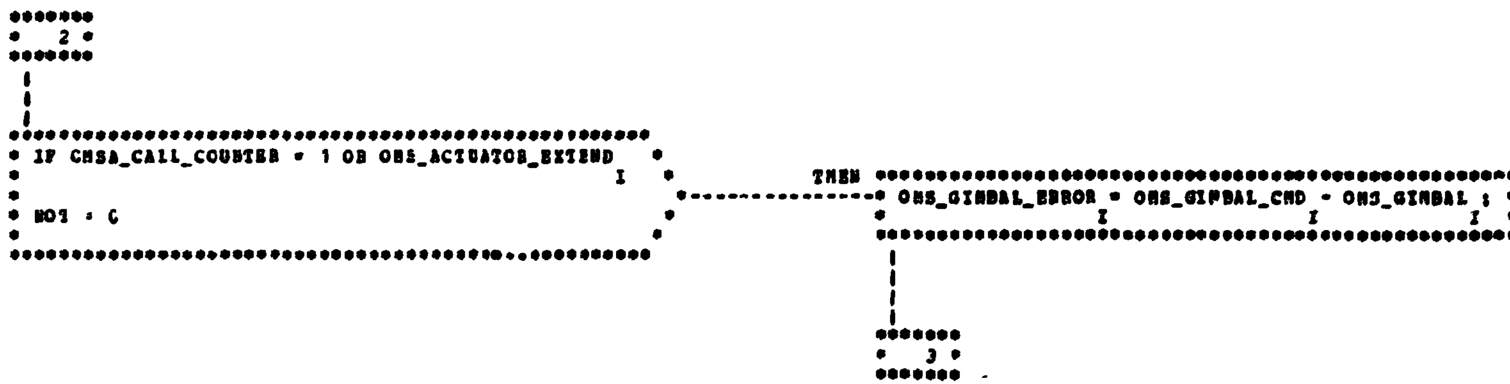

Figure 2-2. Structured automatic flowchart of OMS_ACTUATOR_FDI (sheet 2 of 7).

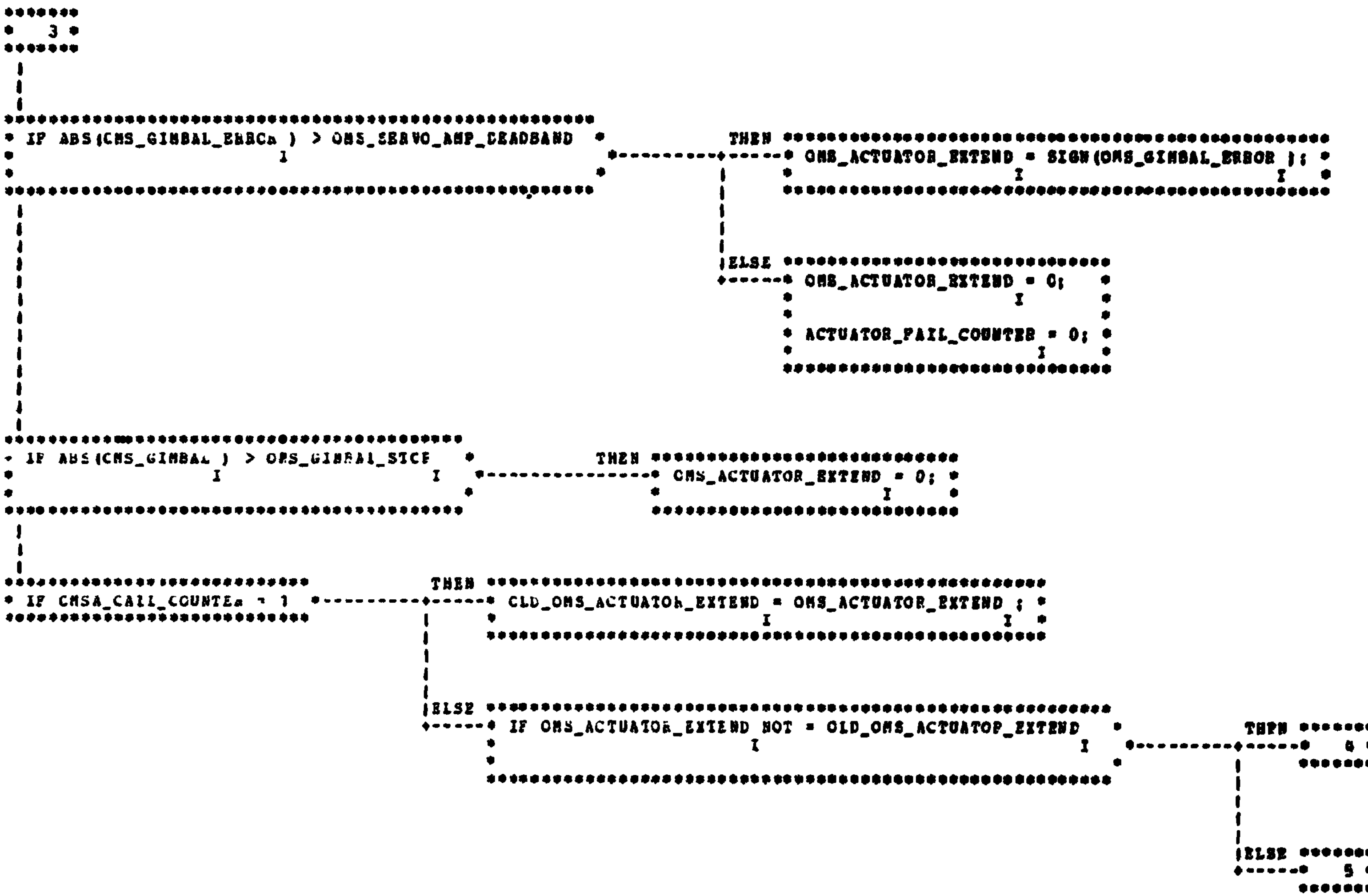

Figure 2-2. Structured automatic flowchart of OMS_ACTUATOR_FDI (sheet 3 of 7).

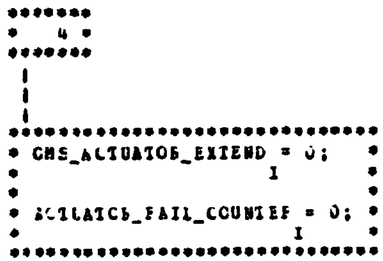

Figure 2-2. Structured automatic flowchart of OMS_ACTUATOR_FDI (sheet 4 of 7).

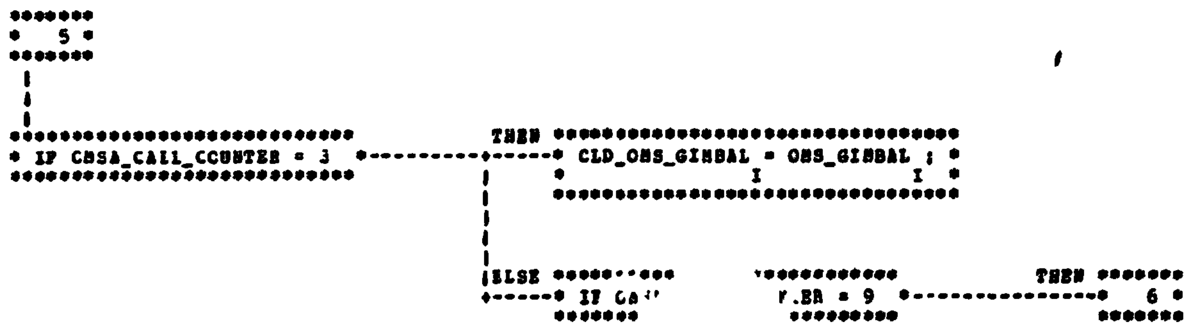

Figure 2-2. Structured automatic flowchart of OMS_ACTUATOR_FDI (sheet 5 of 7).

```
*******
*   6 *
*******
  |
  |
  |
*************************************************************
* IF (OMS_GIMBAL - CLC_OMS_GIMBAL ) OMS_ACTUATOR_EXTEND <   *
*                  I                  I                  I  *                THEN *****************************************************
*                                                           *---------+------* ACTUATOR_FAIL_COUNTER = ACTUATOR_FAIL_COUNTER + *
* OMS_GIMBAL_INCR_THRESHOLD                                 *         |      *                                  I              I  *
*                                                           *         |      *                                                    *
*************************************************************         |      * 1;                                                 *
                                                                      |      ******************************************************
                                                                      |        |
                                                                      |        |
                                                                      |        |
                                                                      |      ********************************
                                                                      |      * IF ACTUATOR_FAIL_COUNTER = 2 *          *******
                                                                      |      *                            I *----------*   7 *
                                                                      |      *                              *          *******
                                                                      |      ********************************
                                                                      |
                                                                      |ELSE ******************************
                                                                      +-----* ACTUATOR_FAIL_COUNTER = 0; *
                                                                            *                      I     *
                                                                            ******************************
```


Figure 2-2. Structured automatic flowchart of OMS_ACTUATOR_FDI (sheet 6 of 7).

```
*******
*   7 *
*******
 |
 |
 |
 |THEN ****************************
 *------*          .              *
        * OMS_ACTUATOR_FAIL  = ON; *
        *                 I;      *
        ****************************
```


Figure 2-2. Structured automatic flowchart of OMS_ACTUATOR_FDI (sheet 7 of 7).

Core storage requirements are computed by the HAL compiler and printed out as part of the compilation, while the CPU time is an estimate of the maximum possible time required to execute all the computations of the procedure.

The maximum time estimate is based on combining the CPU times for each statement in the longest possible path (timewise) through the procedure's logic. The CPU times for each statement are also computed as part of the compilation.

The maximum possible actuator FDI CPU time as summarized above can be reduced if the OMS actuator position transducer accuracy specification is upgraded. The method of determining the effect of position transducer accuracy on OMS actuator FDI CPU time is described in detail in Section 4.4 and in Reference 6. Only the results will be presented here.

According to the accuracy specification stated in Reference 7, the accuracy is ±0.1 volt for a transducer scale factor of 2.0 volts per inch, or ±0.05 inch of actuator extension, which translates to ±0.157° of gimbal deflection. The OMS actuator FDI algorithm described herein has been designed with such an accuracy constraint, and it requires waiting six minor cycles to compute a measured gimbal deflection. If the position transducer accuracy can be upgraded to ±0.019 volt with the same scale factor, thus yielding ±0.0095 inch or ±0.03°, the actuator FDI algorithm can be modified to wait only one minor cycle, and the maximum possible CPU time can be reduced to 3.476 ms/s or 0.348%. This is a decrease in CPU time of a factor of about 2.2 for an increase in position transducer accuracy of about 5. For an earlier version of the OMS actuator FDI algorithm (evaluated in Reference 6), the decrease in CPU time was about a factor of 2.5.

2.4.3 Processing Requirements

The first call to the procedure OMS_ACTUATOR_FDI is made once per second during OMS gimbal actuator operation. Each subsequent call to the procedure, at intervals of one minor cycle (40 ms), is performed by the OMS FDI executive only if at least one of the four actuator extension indicator outputs from the previous call is nonzero. Nine calls would be necessary during any 1-second interval for a complete actuator performance test.

SECTION 3

GLOSSARY

This section is a glossary of the constants and variables used in this document. (See Table 3-1 for the OMS engine FDI constants and variables and Table 3-2 for the OMS actuator FDI constants and variables.) Included for each variable and constant are its name, type/attribute, description, units, and value or range. The modules where the constant or variable is declared, assigned, and referenced are also given. Codes and definitions of the symbols used in the type/attribute designation appear in Table 3-3.

Table 3-1. OMS engine FDI glossary.

Name of Variable or Constant		Type/ Attribute	Description	Units	Value or Range	Declared in	Assigned in	Referenced in
Computer Notation	Mathematical Notation							
ROLL_DISTURB_ ACCEL	$\hat{\alpha}_{d_r}$	Sc/SP	Roll disturbance acceleration estimate	deg/s^2	$-2.0 \le x \le 2.0$	COMPOOL	PART1_FILTER	OMS_ENGINE_FDI
ROLL_ACCEL_ THRESHOLD	α_t	Sc/SP,C	Roll acceleration threshold	deg/s^2	TBD	OMS_ENGINE_FDI	OMS_ENGINE_FDI	OMS_ENGINE_FDI
OMS1_ON_CMD	OMS1_ON_CMD	B/-	OMS Engine 1 ON/OFF command	none	0,1	COMPOOL	OMS_ENG_CMD	OMS_ENGINE_FDI
OMS2_ON_CMD	OMS2_ON_CMD	B/-	OMS Engine 2 ON/OFF command	none	0,1	COMPOOL	OMS_ENG_CMD	OMS_ENGINE_FDI
ACCUM_DELTA_ V_INCR	$\lvert \Delta \underline{v}_a \rvert$	Sc/SP,A	Increment in the added velocity due to OMS thrust	ft/s	$0 \le x \le 3.0$	OMS_ENGINE_FDI	OMS_ENGINE_FDI	OMS_ENGINE_FDI
ONE_ENGINE_ THRESHOLD	Δv_{a_1}	Sc/SP,C	One-engine velocity increment threshold	ft/s	TBD	OMS_ENGINE_FDI	OMS_ENGINE_FDI	OMS_ENGINE_FDI
TWO_ENGINE_ THRESHOLD	Δv_{a_2}	Sc/SP,C	Two-engine velocity increment threshold	ft/s	TBD	OMS_ENGINE_FDI	OMS_ENGINE_FDI	OMS_ENGINE_FDI
OMS1_FAIL	OMS1_FAIL	B/-	OMS Engine 1 failure flag	none	0,1	COMPOOL	OMS_ENGINE_FDI	OMS FDI Executive
OMS2 FAIL	OMS2_FAIL	B/-	OMS Engine 2 failure flag	none	0,1	COMPOOL	OMS_ENGINE_FDI	OMS FDI Executive
OMSE_INIT_ FLAG	OMSE_INIT_ FLAG	B/-	Initialization flag	none	0,1	COMPOOL	OMS FDI Executive	OMS_ENGINE_FDI
ACCUM_ DELTA_V	$\underline{v}_a$	V(3)/SP	Added velocity due to OMS thrust	ft/s	$-1000 \le x \le 1000$	COMPOOL	Hardware/SOP/ OMS Engine FDI IIR	OMS_ENGINE_FDI
OLD_ACCUM_ DELTA_V	$\underline{v}_{a_{OLD}}$	V(3)/SP, St	Last value of added velocity due to OMS thrust	ft/s	$-1000 \le x \le 1000$	OMS_ENGINE_FDI	OMS_ENGINE_FDI	OMS_ENGINE_FDI

Table 3-2. OMS actuator FDI glossary.

Name of Variable or Constant		Type/ Attribute	Description	Units	Value or Range	Declared in	Assigned in	Referenced in
Computer Notation	Mathematical Notation							
[OMS_GIMBAL]	[δ_m]	A(4),Sc/SP	Array of OMS gimbal deflections	deg	$-8.0 \le x \le 8.0$	COMPOOL	Hardware/SOP/ OMS Actuator FDI IIR	OMS_ACTUATOR_FDI
[OLD_OMS_ GIMBAL]	[$\delta_{m_{OLD}}$]	A(4),Sc/ SP,St	Array of values of measured OMS gimbal deflections	deg	$-8.0 \le x \le 8.0$	OMS_ACTUATOR_ FDI	OMS_ACTUATOR_ FDI	OMS_ACTUATOR_FDI
[OMS_GIMBAL_ CMD]	[δ_c]	A(4),Sc/ SP	Array of commanded OMS gimbal deflections	deg	$-8.0 \le x \le 8.0$	COMPOOL	TVC_DAP/OMS Actuator FDI IIR	OMS_ACTUATOR_FDI
[OMS ACTU- ATOR_EXTEND]	[E]	A(4),I/SP	Array of OMS actuator extension indicators	none	-1,0,1	COMPOOL	OMS_ACTUATOR_ FDI	OMS_ACTUATOR_FDI, OMS FDI Executive
[OLD_OMS_ ACTUATOR_ EXTEND]	[E_{OLD}]	A(4),I/SP, St	Array of first-pass values of OMS actuator extension indicators	none	-1,0,1	OMS_ACTUATOR_ FDI	OMS_ACTUATOR_ FDI	OMS_ACTUATOR_FDI
OMS_SERVO_ AMP_ DEADBAND	δ_{e_ℓ}	Sc/SP,C	OMS servo amplifier deadzone	deg	0.43	OMS_ACTUATOR_ FDI	OMS_ACTUATOR_ FDI	OMS_ACTUATOR_FDI
[OMS_ACTUATOR_ FAIL]	[F]	A(4),B/	Array of OMS actuator failure flags	none	0,1	COMPOOL	OMS_ACTUATOR_ FDI	OMS FDI Executive
I	I	I/SP,A	Index variable and counter	none	1,2,3,4	OMS_ACTUATOR_ FDI	OMS_ACTUATOR_ FDI	OMS_ACTUATOR_FDI
OMSA_INIT_ FLAG	OMSA_INIT_FLAG	B/-	Initialization flag	none	0,1	COMPOOL	OMS FDI Executive	OMS_ACTUATOR_FDI
[ACTUATOR_ FAIL_COUNTER]	[C]	A(4),I/SP	Array of OMS actuator failure counters	none	0,1,2	OMS_ACTUATOR_ FDI	OMS_ACTUATOR_ FDI	OMS_ACTUATOR_FDI
OMSA_CALL_ COUNTER	OMSA_CALL_ COUNTER	I/SP	Procedure call counter	none	1,2,3,4,5,6, 7,8,9	COMPOOL	OMS FDI Executive	OMS_ACTUATOR_FDI, OMS FDI Executive
OMS_GIMBAL_ INCR_ THRESHOLD	$\Delta\delta_\ell$	Sc/SP,C	OMS gimbal deflection increment threshold	deg	0.360	OMS_ACTUATOR_ FDI	OMS_ACTUATOR_ FDI	OMS_ACTUATOR_FDI
[OMS_GIMBAL_ STOP]	[δ_{STOP}]	A(4),Sc/ SP,C	Array of OMS gimbal stop values	deg	[5.98,6.98, 5.98,6.98]	OMS_ACTUATOR_ FDI	OMS_ACTUATOR_ FDI	OMS_ACTUATOR_FDI
[OMS_GIMBAL_ ERROR]	[δ_e]	A(4),Sc/ SP	Array of OMS gimbal deflection errors	deg	$-16.0 \le x \le 16.0$	OMS_ACTUATOR_ FDI	OMS_ACTUATOR_ FDI	OMS_ACTUATOR_FDI

Table 3-3. Codes for type/attribute column.

Key for Type	Key for Attribute
A(i) - Array (i)	SP - Single Precision
V(i) - Vector (i)	A - Automatic
Sc - Scalar	C - Constant
I - Integer	St - Static
B - Boolean	

APPENDIX A

SUPPLEMENTARY MATERIAL TO OMS FAILURE DETECTION AND IDENTIFICATION ALGORITHM DESIGNS

This appendix contains material supportive to the OMS Failure Detection and Identification Algorithm designs presented in Sections 1 and 2.

A.1 Overview

A.2 Appendix to OMS FDI Executive

A.3 Appendix to OMS Engine FDI
Failure Flag Summary Charts

The three charts in Figures A-1 and A-2 illustrate the failure flag outputs produced by OMS_ENGINE_FDI as a result of all possible combinations of procedure inputs (engine firing commands and roll disturbance acceleration estimate) and the actual engine firing conditions (two, one, or zero engines firing). Each chart corresponds to one of the engine firing conditions.

The OMS1 and OMS2 failure flag outputs are given in the row entitled "present failure fla values" for the columns headed by each possible combination of engine commands. The OMS_FAIL_DETECT flag output is listed in Figure A-2 for the case of one engine firing. For all of the charts, logic "0" is defined herein as a flag or command set to OFF, and logic "1" is a flag or command set to ON. Note that the logic symbols in each case are located in a box. Each box is divided by a diagonal where the upper-left value corresponds to OMS Engine 1, and the lower right value corresponds to OMS Engine 2.

Included with the One-Engine-Firing chart on Figure A-2 is a diagram which defines the three regions A, B, and C of the roll disturbance acceleration estimate which are used in that chart. The diagram also indicates that a positive roll disturbance acceleration estimate occurs when OMS1 is OFF and OMS2 is ON, and a negative roll disturbance acceleration estimate occurs when OMS1 is ON and OMS2 is OFF. This relationship between the sign of the roll disturbance acceleration estimate and the ON/OFF status of the OMS may be explained as follows.

ENGINE CONDITIONS		TWO ENGINES FIRING															
PRESENT ENGINE COMMANDS	OMS1_CMD	0				1				0				1			
	OMS2_CMD	0				0				1				1			
PREVIOUS FAILURE FLAG VALUES	OMS1_FAIL	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
	OMS2_FAIL	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
PRESENT FAILURE FLAG VALUES	OMS1_FAIL	1	1	1	1	0	1	0	1	1	1	1	1	0	1	0	1
	OMS2_FAIL	1	1	1	1	1	1	1	1	0	0	1	1	0	0	1	1

ENGINE CONDITIONS		ZERO ENGINES FIRING															
PRESENT ENGINE COMMANDS	OMS1_CMD	0				1				0				1			
	OMS2_CMD	0				0				1				1			
PREVIOUS FAILURE FLAG VALUES	OMS1_FAIL	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
	OMS2_FAIL	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
PRESENT FAILURE FLAG VALUES	OMS1_FAIL	0	1	0	1	1	1	1	1	0	1	0	1	1	1	1	1
	OMS2_FAIL	0	0	1	1	0	0	1	1	1	1	1	1	1	1	1	1

Figure A-1. Failure flag summary charts: two engines firing and zero engines firing.

In the OMS engine FDI procedure, the roll disturbance acceleration estimate is used to identify which engine has failed in the situations where only one engine is firing and both engines are commanded ON or OFF. Under the assumption that in those situations the alignment of the engines is at or near trim,* then the yaw deflection of each engine would be such that neither engine could point behind the center of gravity in the yaw plane. Therefore the sign of the roll acceleration produced by each engine is the same for all possible pitch deflections: negative for OMS1 and positive for OMS2.

* That is, near the vehicle XZ plane.

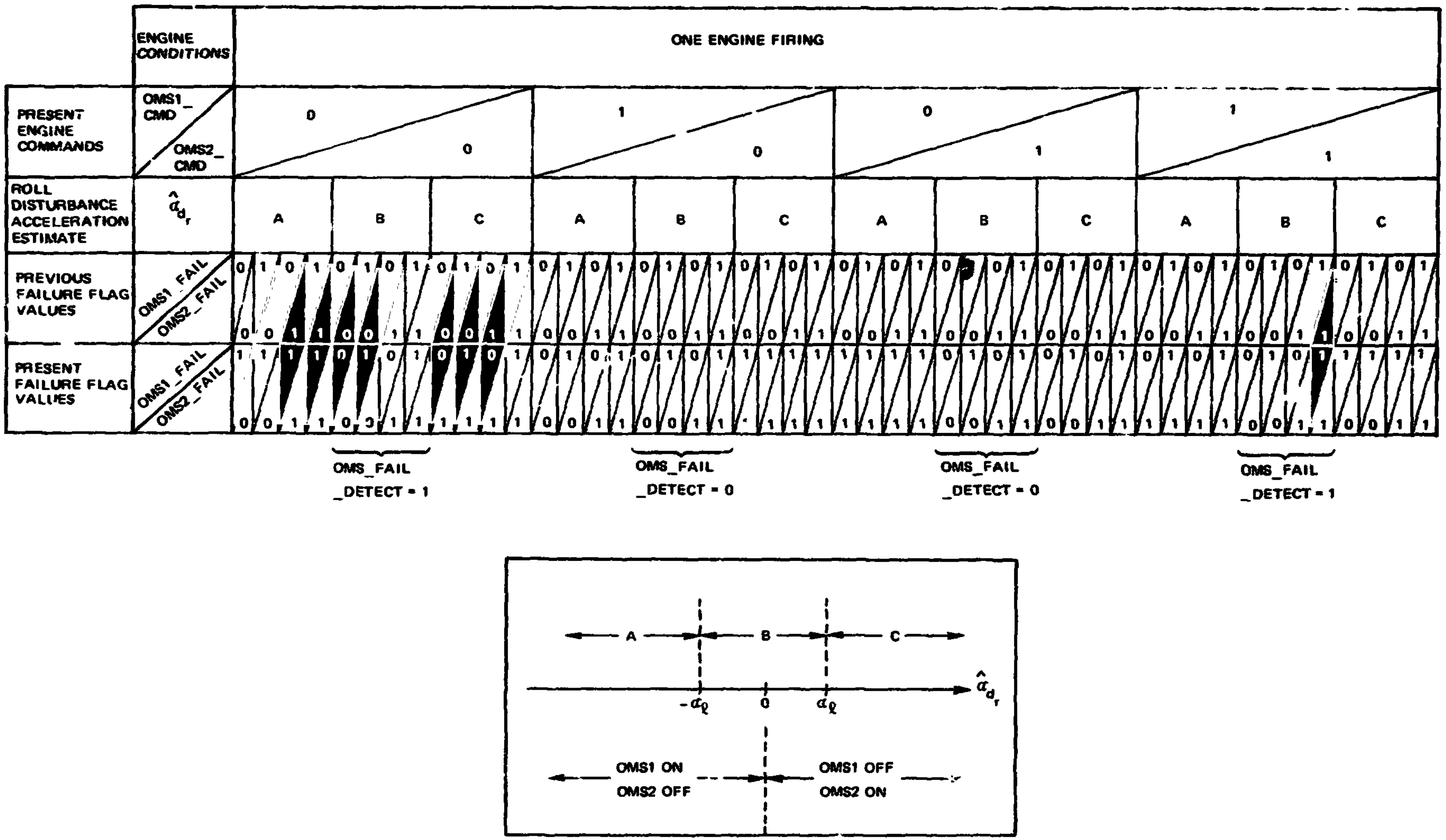

Figure A-2. Failure flag summary chart: one engine firing.

The roll disturbance acceleration estimate is essentially defined as

$$\hat{\alpha}_{d_r} = \tilde{\alpha}_r - \alpha_r' \qquad \text{(A-1)}$$

where

$\tilde{\alpha}_r$ = measured roll acceleration

α_r' = extrapolate roll acceleration*

Hence, if both engines are commanded ON or OFF, the roll disturbance acceleration will be approximately equal to

(1) The negative of the extrapolated roll acceleration of a failed-off engine.**

or

(2) The measured roll acceleration of a failed-on engine.

Combining these two definitions with the fact that OMS1 produces negative roll acceleration and OMS2 positive roll acceleration yields the relationship between the sign of the roll disturbance acceleration estimate and the ON/OFF status of the engines as

OMS1 commanded OFF OMS2 commanded OFF OMS1 fails ON	$\hat{\alpha}_{d_r} \approx \tilde{\alpha}_1 = (-)$
OMS1 commanded OFF OMS2 commanded OFF OMS2 fails ON	$\hat{\alpha}_{d_r} \approx \tilde{\alpha}_2 = (+)$
OMS1 commanded ON OMS2 commanded ON OMS1 fails OFF	$\hat{\alpha}_{d_r} \approx -\alpha_1' = (+)$

---

* An extrapolated quantity is defined as the extrapolation of the quantity's previous estimate, and it may be thought of as a prediction of the quantity.

** Here it is assumed that the measured and extrapolated roll acceleration of an unfailed engine are nearly equal.

OMS1 commanded ON
OMS2 commanded ON
OMS2 fails OFF

$$\left.\begin{array}{l}\text{OMS1 commanded ON}\\ \text{OMS2 commanded ON}\\ \text{OMS2 fails OFF}\end{array}\right\} \quad \hat{\alpha}_{d_r} \approx -\alpha_2' = (-)$$

To illustrate the way the charts may be used and to verify the correctness of the outputs, consider two cases for which the OMS1_FAIL flag should be set to ON:

(1) OMS1 fails OFF when both engines are commanded ON.

(2) OMS1 fails ON when both engines are commanded OFF.

In both cases, since one engine is firing, the chart in Figure A-2 is applicable. Also, it is assumed that $\hat{\alpha}_{d_r}$ is outside the deadzone.

<u>Case 1</u>:

OMS1_CMD = 1

OMS2_CMD = 1

$$\left.\begin{array}{l}\text{OMS1 is OFF}\\ \text{OMS2 is ON}\end{array}\right\} \quad \hat{\alpha}_{d_r} > \alpha_\ell \rightarrow C$$

Previous OMS1_FAIL = 0 (assumed)

Previous OMS2_FAIL = 0 (assumed)

∴ Present OMS1_FAIL = 1

Present OMS2_FAIL = 0

<u>Case 2</u>:

OMS1_CMD = 0

OMS2_CMD = 0

$$\left.\begin{array}{l}\text{OMS1 is ON}\\ \text{OMS2 is OFF}\end{array}\right\} \quad \hat{\alpha}_{d_r} < -\alpha_\ell \rightarrow A$$

Previous OMS1_FAIL = 0 (assumed)

Previous OMS2_FAIL = 0 (assumed)

∴ Present OMS1_FAIL = 1

Present OMS2_FAIL = 0

In both cases, when OMS1 failed (assuming both OMS1 and OMS2 had not failed in the past), OMS1_FAIL was set to ON and OMS2_FAIL was left OFF.

A.4 Appendix to OMS Actuator FDI

The OMS actuator FDI procedure compares the increment in the measured gimbal deflection over six minor cycles (240 ms) to a threshold value to determine if an actuator has failed. Two successive failed threshold tests result in activation of a failure flag. The threshold value is set such that the measured gimbal deflection of an actuator which has experienced a full-off failure is less than the threshold, and the measured gimbal deflection of an actuator which is performing nominally is greater than the threshold. The desired result of no full-off misalarms or full-on false alarms is thus achieved. The accuracy of the OMS actuator output extension position transducer* directly affects not only the number of minor cycles over which the deflection increment is computed, but also the value at which the threshold is set.

The measured increment in the OMS gimbal deflection is given by

$$\Delta\delta_m = \delta_m(t + n\,\Delta T) - \delta_m(t) \qquad \text{(A-2)}$$

where ΔT is the minor cycle time (40 ms), and n is the number of minor cycles over which the increment is computed. The error in the deflection increment is therefore

$$\varepsilon_{\Delta\delta} = \varepsilon_\delta(t + n\,\Delta T) - \varepsilon_\delta(t) \qquad \text{(A-3)}$$

where $\varepsilon_\delta(t)$ is the error in the measured gimbal deflection at time t introduced by the OMS LVDT position transducer. (The additional error introduced by the A/D conversion of the position transducer output signal is an order of magnitude less and is neglected).

A pessimistic upper bound on the error in the deflection increment is found by assuming that the errors in the gimbal deflection measurements at the two times are negatively correlated and of maximum magnitude

$$|\varepsilon_{\Delta\delta}|_{max} = 2|\varepsilon_\delta|_{max} \qquad \text{(A-4)}$$

---

* The measured gimbal deflection is equal to the measured actuator output position scaled to angular measure.

In reality, if the actual position transducer output is a sufficiently "smooth" function of the actuator extension, then the error in the gimbal deflection measurement at time $t + n\Delta T$ would be positively correlated with the error at time t, provided that n were not too large. However, in the absence of a realistic model of the position transducer error sources, the more conservative approach of assuming negative correlation is taken.

The OMS actuator position transducer accuracy specification in Reference 7 lists the accuracy of the LVDT as ±0.1 volt.* The scale factor of the LVDT is 2.0 volt per inch, and there are 3.14 degrees of gimbal deflection per inch of actuator extension.[3,7]** Combining this data and referring to Eq. (A-4), one obtains

$$|\varepsilon_{\Delta\delta}|_{max} = 0.314° \qquad \text{(A-5)}$$

The lower limit of acceptable performance in the OMS actuator procurement specification[7] is 3.0°/s. In order that there be no misalarms in the case of a full-off failure and no false alarms in the case of lower-limit nominal performance, the error in the deflection increment must be less than one half the increment itself. Otherwise, a deflection rate of 0.0°/s could not be distinguished from 3.0°/s by means of a measured gimbal deflection increment.

For a constant deflection rate of 3.0°/s, the actual gimbal deflection (in degrees) is

$$\Delta\delta = 3\ n\ \Delta T = 0.12\ n \qquad \text{(A-6)}$$

In order that $|\varepsilon_{\Delta\delta}|_{max}$ be less than $0.5\Delta\delta$, n must be at least 6. Thus, the accuracy of the OMS actuator position transducer is such that it requires waiting six minor cycles (240 ms) to compute a gimbal

---

* After the algorithm designs presented in this report were completed, but before publication, a new OMS actuator math model, a new OMS position transducer accuracy specification, and new OMS actuator performance requirements were received. The analysis of this section and the design of the OMS actuator FDI procedure will be reworked in light of the new data. However, only minor changes of parameter values are anticipated. The basic algorithm structure will be unchanged.

** Superscript numerals refer to similarly numbered references in the List of References.

deflection increment which can only distinguish between a full-off failure (0.0°/s) and a lower-limit nominal performance (3.0°/s).

Ideally, one would want to wait only one minor cycle between the two samples of the measured gimbal deflection for the increment computation. The reason is that the CPU time consumed by the OMS actuator FDI procedure in any 1-second interval is directly proportional to the number of minor cycles between the two samples, because at each of those minor cycles the procedure must be called to check the gimbal deflection error to determine if the actuator input voltage is being continuously applied over the interval. Otherwise the actuator is not extending continuously, and the threshold test should not be performed. To wait only one cycle ($n = 1$), the accuracy of the measured gimbal deflection would have to be ±0.03°, implying an OMS position transducer accuracy of ±0.019 volt for the same scale factor. This represents an increase in accuracy of about a factor of 5. The CPU time savings which can be obtained by this increased position transducer accuracy are described in Section 2.4.2.

Returning to the situation based on the accuracy specifications of Reference 7 for which $n = 6$, the actual gimbal deflection increment over six minor cycles for a constant rate of 3.0°/s is 0.72° (see Eq. (A-6)). Given that the error in the deflection increment can be as large as 0.314° (see Eq. (A-5)), the permissible range for the threshold $\Delta\delta_\ell$ is

$$0.314° < \Delta\delta_\ell < 0.406° \qquad \text{(A-7)}$$

That is, if $\Delta\delta_\ell$ were less than 0.314°, a full-off failure might pass the test, and if $\Delta\delta_\ell$ were greater than 0.406°, an actuator performing nominally at 3.0°/s might fail the test. The threshold is arbitrarily set at the midpoint of the permissible range

$$\Delta\delta_\ell = 0.360° \qquad \text{(A-8)}$$

To summarize the possibilities:

(1) An OMS actuator performing below 0.192°/s will always cause a failure indication.

(2) There will never be a false alarm for an OMS actuator which is performing above 2.808°/s.

(3) OMS actuators performing between 0.192°/s and 2.808°/s may or may not cause failure indications.

Of course, if either the accuracy of the OMS position transducer is upgraded or more than six minor cycles are used for the gimbal deflection increment computation, the area of uncertainty will shrink and more specific information about actuator performance can be deduced from the threshold test.

LIST OF ABBREVIATIONS

Symbol	Description
cg	center of gravity
CO	checkout
CPU	central processing unit
CRT	cathode ray tube
DAP	digital autopilot
D&C	displays & controls
FC	flight control
FCS	flight control system
FCOS	flight computer operating system
FDI	failure detection and identification
GN&C	guidance, navigation, & control
GPC	general-purpose computer
GUID	guidance
HC	hand controller
IIR	input interface routine
IMU	inertial measuring unit
LVDT	linear voltage differential transformer
MSC	moding, sequencing, & control
MTU	master timing unit
NAV	navigation
OFC	on-orbit flight control
OMS	orbital maneuvering system

LIST OF ABBREVIATIONS (Cont.)

Symbol	Description
PM	performance monitoring
RCS	reaction control system
RHC	rotational hand controller
RM	redundancy management
SM	system management
SOP	subsystem operating programs
TBD	to be determined
THC	translational hand controller
TVC	thrust vector control

NOTATION CONVENTIONS

^	estimated quantity
'	extrapolated quantity
_ or ¯	vector
~	measured quantity
•	matrix
[ ]	array
·	Boolean

LIST OF REFERENCES

1. On-Orbit Flight Control System Algorithm Description (Second printing), Charles Stark Draper Laboratory Report R-881, May 1975.
2. Vander Voort, R.J., "OMS Gimbal Actuator Performance Study," Technical Coordination Letter No. Rockwell-074-082, Aerospace Division, Honeywell, Inc., St. Petersburg, Florida, 11 November 1974.
3. "Space Shuttle Flight Control System Data Book, Volume II: Orbiter," Rockwell Document SD 73-SH-0097-2D, Space Division, Rockwell International, Downey, California, May 1975.
4. D'Amario, L.A., "New OMS Gimbal Actuator Servo Loop Models for Implementation in the SLS," Charles Stark Draper Laboratory SSV Memo 75-10C-39, 16 September 1975.
5. D'Amario, L.A., "Numerical Integration Step Size and Position Transducer Error Representation for Inclusion in the SLS OMS Gimbal Actuator Servo Loop Models," Charles Stark Draper Laboratory SSV Memo 75-10C-41, 4 November 1975.
6. D'Amario, L.A., "Impact of Actuator FDI CPU Time of the OMS Gimbal Actuator Position Transducer Accuracy," Charles Stark Draper Laboratory SSV Memo 75-10C-15, 20 May 1975.
7. Vander Voort, R.J., "OMS Actuator Procurement Specification," Technical Coordination Letter No. Rockwell-074-086, Aerospace Division, Honeywell, Inc., St. Petersburg, Florida, 2 December 1974.
8. D'Amario, L.A. "Preliminary OMS FDI Algorithm Description Report," Charles Stark Draper Laboratory SSV Memo 75-10C-43, 11 November 1975.
9. "Shuttle Operational Data Book, Volume II, Mission Mass Properties," JSC Document JSC-08934 (Vol. II), Lyndon B. Johnson Space Center, NASA, Houston, Texas, February 1975.