Systems Cost/Performance Analysis (Study 2.3) Final Report


Prepared by
ADVANCED MISSION ANALYSIS DIRECTORATE
Advanced Orbital Systems Division

31 March 1975

Prepared for
OFFICE OF MANNED SPACE FLIGHT
NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
Washington, D.C. 20546

Contract No. NASW-2727

THE AEROSPACE CORPORATION
SYSTEMS COST/PERFORMANCE ANALYSIS (STUDY 2.3)  
FINAL REPORT  

Prepared by  
Advanced Mission Analysis Directorate  
Advanced Orbital Systems Division  

31 March 1975  
(Supersedes and replaces ATR-74(7343)-1, Vol. III, dated 27 September 1974)  

Systems Engineering Operations  
THE AEROSPACE CORPORATION  
El Segundo, California  

Prepared for  
OFFICE OF MANNED SPACE FLIGHT  
NATIONAL AERONAUTICS AND SPACE ADMINISTRATION  
Washington, D.C.  

Contract No. NASW-2727
SYSTEMS COST/PERFORMANCE ANALYSIS
(STUDY 2.3) FINAL REPORT

Prepared

R. F. Janz, Manager
Data Systems Analysis Section
Data Processing Subdivision

Approved

L. Sashkin, Director
Data Processing Subdivision
Information Processing Division
Engineering Science Operations

R. H. Herndon, Group Director
Advanced Mission Analysis
Directorate
Advanced Orbital Systems Division
FOREWORD

This report documents The Aerospace Corporation effort on Study 2.3, Systems Cost/Performance Analysis, performed under NASA Contracts NASW-2575 and NASW-2727 during Fiscal Years 1974 and 1975. The effort was directed by Mr. B. H. Campbell. Mr. R. D. Kramer, Marshall Space Flight Center and Mr. R. R. Carley, NASA Headquarters were the NASA Study Directors for this study. Their efforts in providing technical direction throughout the duration of the study are greatly appreciated.

This volume is one of three volumes of the final report for Study 2.3. The three volumes are:

Volume I Executive Summary
Volume II Systems Cost/Performance Model
Appendix Data Base

Volume I summarizes the overall report. It includes the relationship of this study to other NASA efforts, significant results, study limitations, and suggested additional effort.

Volume II provides a detailed description of the Systems Cost/Performance Model. It also includes the model checkout and the results for three payload test cases. The Data Base is provided in the Appendix to Volume II.

Volume III provides a detailed description of how the Systems Cost/Performance Computer Program is organized and operates. The program listing, detailed flow charts and user restrictions are included.
ACKNOWLEDGMENTS

The Aerospace Corporation effort on Study 2.3 was supported by various Members of the Technical Staff (MTS). The contributions of the following MTS to the Systems Cost/Performance Computer Program are gratefully acknowledged:

R. M. Harris
R. F. Janz
H. S. Porjes
R. E. Rice
D. E. Sakaguchi
J. C. Thacker
## CONTENTS

**FOREWORD** .................................................. i

**ACKNOWLEDGMENTS** ........................................ vii

1. **INTRODUCTION** ........................................ 1-1

2. **OPERATING ENVIRONMENT** ............................. 2-1
   2.1 Hardware .................................................. 2-1
      2.1.1 Computer ........................................... 2-1
      2.1.2 Main Memory Utilization ......................... 2-1
      2.1.3 Magnetic Tapes ..................................... 2-1
      2.1.4 Card Punch ......................................... 2-1
      2.1.5 Plotter .............................................. 2-1
      2.1.6 Disk ................................................ 2-1
   2.2 Software ............................................... 2-2
      2.2.1 Operating System .................................. 2-2
      2.2.2 Programming Language .............................. 2-2
      2.2.3 Type of Run ........................................ 2-2
      2.2.4 Library Subroutines ............................... 2-2

3. **PROGRAM SPECIFICATIONS** ............................. 3-1
   3.1 Computational Flow Chart .............................. 3-1
      3.1.1 MACRO-MICRO ....................................... 3-1
   3.2 COMMON Regions ......................................... 3-8
      3.2.1 USER1, USER3, USER4, USER6, USER8, USER9, USERR, USERC, USERI, USERP .... 3-8
      3.2.2 DBCOM ............................................... 3-12
      3.2.3 CHOSE ............................................... 3-13
      3.2.4 PRTCOM ............................................. 3-15
      3.2.5 BTWN ............................................... 3-18
   3.3 Hardware Selection Procedure ........................ 3-21
## CONTENTS (Continued)

4. SUBROUTINES WHICH SELECT HARDWARE .................................. 4-1

<table>
<thead>
<tr>
<th>SUBROUTINE</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1 Subroutine SANDC</td>
<td>4-1</td>
</tr>
<tr>
<td>4.1.1 Purpose of Subroutine</td>
<td>4-1</td>
</tr>
<tr>
<td>4.1.2 Communication with Main Program</td>
<td>4-3</td>
</tr>
<tr>
<td>4.1.3 Variables Specified in DATA Statements</td>
<td>4-3</td>
</tr>
<tr>
<td>4.1.4 Other Subroutines Called</td>
<td>4-3</td>
</tr>
</tbody>
</table>

| 4.2 Subroutine AUXPRO | 4-3 |
| 4.2.1 Purpose of Subroutine | 4-3 |
| 4.2.2 Communication with Main Program | 4-6 |
| 4.2.3 Variables Specified in DATA Statements | 4-6 |
| 4.2.4 Other Subroutines Called | 4-6 |

| 4.3 Subroutine DPI | 4-6 |
| 4.3.1 Purpose of Subroutine | 4-6 |
| 4.3.2 Communication with Main Program | 4-8 |
| 4.3.3 Variables Specified in DATA Statements | 4-8 |
| 4.3.4 Other Subroutines Called | 4-8 |
| 4.3.4.1 Subroutine MIS | 4-8 |
| 4.3.4.2 Subroutine ORDER | 4-9 |

| 4.4 Subroutine COMM | 4-9 |
| 4.4.1 Purpose of Subroutine | 4-9 |
| 4.4.2 Communication with Main Program | 4-11 |
| 4.4.3 Variables Specified in DATA Statements | 4-11 |
| 4.4.4 Other Subroutines Called | 4-12 |
| 4.4.4.1 Subroutine BESS | 4-12 |
| 4.4.4.2 Function RESET | 4-12 |

| 4.5 Subroutine EP | 4-13 |
| 4.5.1 Purpose of Subroutine | 4-13 |
| 4.5.2 Communication with Main Program | 4-14 |
| 4.5.3 Variables Specified in DATA Statements | 4-14 |
| 4.5.4 Other Subroutines Called | 4-15 |
CONTENTS (Continued)

5. SUBROUTINES WHICH DO NOT SELECT HARDWARE .. 5-1

5.1 Subroutine PRESET .................. 5-1
   5.1.1 Purpose of Subroutine ............ 5-1
   5.1.2 Communication with Main Program .... 5-1
   5.1.3 Variables Specified in DATA Statements .. 5-1
   5.1.4 Other Subroutines Called .......... 5-1

5.2 Subroutine FILTER .................. 5-1
   5.2.1 Purpose of Subroutine ............ 5-1
   5.2.2 Communication with Main Program .... 5-2
   5.2.3 Variables Specified in DATA Statements .. 5-2
   5.2.4 Other Subroutines Called .......... 5-2

5.3 Subroutine INITIL .................. 5-2
   5.3.1 Purpose of Subroutine ............ 5-2
   5.3.2 Communication with Main Program .... 5-2
   5.3.3 Variables Specified in DATA Statements .. 5-2
   5.3.4 Other Subroutines Called .......... 5-2

5.4 Subroutine READDB .................. 5-3
   5.4.1 Purpose of Subroutine ............ 5-3
   5.4.2 Communication with Main Program .... 5-3
   5.4.3 Variables Specified in DATA Statements .. 5-3
   5.4.4 Other Subroutines Called .......... 5-3

5.5 Subroutine SAVE ................... 5-3
   5.5.1 Purpose of Subroutine ............ 5-3
   5.5.2 Communication with Main Program .... 5-4
   5.5.3 Variables Specified in DATA Statements .. 5-4
   5.5.4 Other Subroutines Called .......... 5-4

5.6 Subroutine VESIZE .................. 5-4
   5.6.1 Purpose of Subroutine ............ 5-4
   5.6.2 Communication with Main Program .... 5-5
   5.6.3 Variables Specified in DATA Statements .. 5-6
   5.6.4 Other Subroutines Called .......... 5-6
CONTENTS (Continued)

5.7 Subroutine STRUCT ........................................ 5-6
  5.7.1 Purpose of Subroutine ................................ 5-6
  5.7.2 Communication with Main Program .................... 5-7
  5.7.3 Variables Specified in DATA Statements ............ 5-7
  5.7.4 Other Subroutines Called ............................. 5-7

5.8 Subroutine RELY ........................................... 5-7
  5.8.1 Purpose of Subroutine ................................. 5-7
  5.8.2 Communication with Main Program .................... 5-8
  5.8.3 Variables Specified in DATA Statements ............ 5-9
  5.8.4 Other Subroutines Called ............................. 5-9
    5.8.4.1 Subroutine RIMOD ................................ 5-9
    5.8.4.2 Subroutine QSF .................................. 5-9
    5.8.4.3 Subroutine GAM .................................. 5-10
    5.8.4.4 Subroutine CERF ................................ 5-10

5.9 Subroutine THRML .......................................... 5-10
  5.9.1 Purpose of Subroutine ................................ 5-10
  5.9.2 Communication with Main Program .................... 5-11
  5.9.3 Variables Specified in DATA Statements ............ 5-11
  5.9.4 Other Subroutines Called ............................. 5-11

5.10 Subroutine COSTS ......................................... 5-12
  5.10.1 Purpose of Subroutine ................................ 5-12
  5.10.2 Communication with Main Program .................... 5-13
  5.10.3 Variables Specified in DATA Statements ............ 5-13
  5.10.4 Other Subroutines Called ............................. 5-14

5.11 Subroutine SKED ......................................... 5-14
  5.11.1 Purpose of Subroutine ................................ 5-14
  5.11.2 Communication with Main Program .................... 5-14
  5.11.3 Variables Specified in DATA Statements ............ 5-14
  5.11.4 Other Subroutines Called ............................. 5-14
CONTENTS (Continued)

5.12 Subroutine PRNT ........................................ 5-15
  5.12.1 Purpose of Subroutine ................................ 5-15
  5.12.2 Communication with Main Program .................... 5-15
  5.12.3 Variables Specified in DATA Statements .............. 5-15
  5.12.4 Other Subroutines Called .......................... 5-15

6. DATA BASE ............................................. 6-1
  6.1 Format .................................................. 6-1
  6.2 Presort .................................................. 6-3

7. RESTRICTIONS AND/OR LIMITATIONS .......................... 7-1

8. SAMPLE TEST CASE ........................................ 8-1
  8.1 User Input Variable List ................................ 8-1
  8.2 Input Variables for Test Case ......................... 8-1
  8.3 Sample Test Case Results ................................ 8-16

9. SOURCE CODE LISTING ..................................... 9-1

10. DETAILED FLOW CHARTS .................................... 10-1
FIGURES

3-1 Main Program Logic
3-2 Explanation of Arrays: IDB and ICHOSE
3-3 Hardware Selection Flow Chart
6-1 Data Base Format
8-1 Input Variables for Test Case
8-2 Sample Test Case Results

TABLES

3-1 Hardware Selection Procedure in Kth Subsystem
7-1 Stabilization and Control Configuration Selection
7-2 Stabilization and Control Configuration Compatibility
7-3 Auxiliary Propulsion Configuration Selection
7-4 Data Processing Configuration Compatibility
7-5 Communication Configuration Selection
7-6 Electrical Power Configuration Compatibility
7-7 Vehicle Shape Compatibility
7-8 Structural Configuration Compatibility
8-1 User Input List
1. INTRODUCTION

The objective of the programming task within Study 2.3 was to implement the entire Systems Cost/Performance Model as a digital computer program. This document contains a discussion of the operating environment in which the program was written and checked; the program specifications such as discussions of logic and computational flow; the different subsystem models involved in the design of the spacecraft; and routines involved in the nondesign area such as costing and scheduling of the design. Preliminary results for the DSCS-II design are also included.

Section 2 of this volume covers the Operating Environment. This includes both hardware and software considerations for the UNIVAC 1108 and the CDC 7600.

Section 3 contains the Program Specifications. These include the computational flow, a discussion of the MACRO-MICRO concept, a detailed discussion of the COMMON structures used for communication in the model, and the Hardware Selection procedure.

Section 4 covers the subroutines that select hardware from the data base. These include Stabilization and Control (Subroutine SANDC), Auxiliary Propulsion (Subroutine AP), Data Processing and Instrumentation (Subroutine DPI), Communications (Subroutine COMM), and Electrical Power (Subroutine EP). A discussion of the communication with the main program is included along with the default parameters set in the DATA statements.

Section 5 covers the subroutines that do not select equipment, but do size or calculate information that is pertinent to the design. Subroutines included are: FILTER, which filters out incompatible designs; PRESET, which computes constants as a function of the inputs; INITIL, which initializes certain default numbers that are needed early in the model, but are not computed until later in the model; READDB, which reads the data base for any one subsystem at a time; SAVE, which saves
certain matrices to be used by later subroutines; VESIZE, the vehicle sizing routine that computes weights, lengths, and inertias for the design; STRUCT, that computes other mechanical design data needed to size the structure; RELY, which computes the reliability for the spacecraft; THRML, which computes the thermal requirements for the spacecraft; COSTS, which calculates the various costs involved in building and integrating the entire spacecraft system; SKED, which computes the schedule for the spacecraft from initial design phase to the launch phase; and PRNT, which outputs the final design attributes.

Section 6 contains a discussion of the data base format and tape requirements. Also discussed is the PRESORT routine which allows one to presort the data base into a different order based on cost, weight, or reliability.

Section 7 summarizes the restrictions and limitations established within the program.

Section 8 contains a discussion of the actual sample case used to check the program including all input default values and changes pertaining to the sample case. The results of the test case are discussed here also.

Sections 9 and 10 contain the source code listing and the detailed flow charts, respectively.
2. OPERATING ENVIRONMENT

Section 2 contains a description of the operating environment within which the program was coded and checked. Paragraph 2.1 summarizes the hardware involved and Paragraph 2.2 summarizes the software involved.

2.1 HARDWARE

2.1.1 Computer

UNIVAC 1108 and CDC 7600

2.1.2 Main Memory Utilization

a. UNIVAC 1108 (Octal Words)
   135K (to compile)
   135K (to link edit)
   135K (to execute)

b. CDC 7600 (Octal Words)
   120K (to compile)
   20K (to link edit)
   74K (to execute)

2.1.3 Magnetic Tapes

Optional for input or presort (see Paragraph 6.2).

2.1.4 Card Punch

Not required

2.1.5 Plotter

Not required

2.1.6 Disk

Optional for input or presort (see Paragraph 6.2).
2.2 SOFTWARE

2.2.1 Operating System
   a. UNIVAC 1108
      EXEC 8
   b. CDC 7600
      SCOPE 3.4.1 Vers. 373

2.2.2 Programming Language
   FORTRAN

2.2.3 Type of Run
   BATCH

2.2.4 Library Subroutines
   SQRT
   SIN
   COS
   TAN
   ATAN
   ARSIN (ASIN on the CDC 7600)
   EXP
   FLOAT
   INT
   ALOG
3. PROGRAM SPECIFICATIONS

Paragraph 3.1 contains a description of the overall program flow and a discussion of the MACRO-MICRO concept. Paragraph 3.2 contains a discussion of the common structures. Paragraph 3.3 contains a discussion of the hardware selection procedure. Detailed discussions of all subroutines can be found in Sections 4 and 5.

3.1 COMPUTATIONAL FLOW CHART

In general, it can be said that the program has an outer loop on configurations and an inner loop on iterations. The inner loop on iterations includes the calling of all subsystem subroutines and for ITER = 0 the calling of the reliability subroutine. For ITER = 1 (second pass) reliability is bypassed. The structures, thermal, cost, and print subroutines are called once per outer loop on configurations. A general flow chart is shown in Figure 3-1.

3.1.1 MACRO-MICRO

A prerequisite to the understanding of the MACRO-MICRO concept is an understanding of "configuration." A set of rules for selecting equipments is associated with each subsystem. Which set of rules is to be used at any moment in time is determined by NCONF (configuration number) for that subsystem. For example, if NCONF (1) = 5, a star sensor will be selected by reference to the appropriate equations. However, if NCONF (1) = 1, a star sensor will never be selected. Thus, the configuration numbers determine a subset of the sets of equipments, and only this subset is considered for the configuration design.

A MACRO search is a method for testing all possible combinations of configuration numbers (one per subsystem) and determining within this subset of equipments and within the subset of selection procedures the first acceptable equipments for each. Some combinations of config-
Figure 3-1. Main Program Logic
Figure 3-1. Main Program Logic (Continued)
Figure 3-1. Main Program Logic (Continued)

3-4
Figure 3-1. Main Program Logic (Continued)
Figure 3-1. Main Program Logic (Continued)
Figure 3-1. Main Program Logic (Continued)
urations are never acceptable, and some are ruled out by mission re-
requirements; but a MACRO search will, in general, produce many acceptable
designs.

In a MICRO mode all configuration numbers except the one
being "MICROed" are fixed. Care must be taken that these numbers are
compatible. For example, VESIZE should not be set to configuration 2
when SANDC is set at 2; that is, a box shape for the equipment bay is in-
compatible with dual spin. Within the subsystem being "MICROed", all
configurations are checked and all possible combinations of equipments
within the subset determined by configuration are checked. Within a con-
figuration then, it is essentially the stopping point which determines the
difference between a MACRO and a MICRO. A MACRO selects the first
acceptable equipment in each category. The MICRO selects all combinations
of acceptable equipments. The reader should note that in the current ver-
sion of the program, equipments for which selection criteria are not available
are not included in the MICRO "search."

3.2 COMMON REGIONS

The main COMMON regions consist of the USER series, DBCOM,
CHOSE, PRTCOM and BTWN.

3.2.1 USER1, USER3, USER4, USER6, USER8, USER9,
USERR, USERC, USERI, USERP

The various "user" COMMONS are for storage of user inputs.
Variables included in each of these nine common areas are used by only one
of nine major subsystems: Stabilization and Control, Data Processing and
Instrumentation, Communications, Vehicle Sizing, Schedules, Structures,
Reliability, Costs and Print. All user-specified values are preset to either
default values or flags by the BLOCK DATA routine, but can be overwritten
by values specified in the NAMELIST input from the user. The default flags
inform subroutine PRESET to calculate (from other inputs) those variables
which are not overwritten by user-supplied inputs. A complete list of all
variables in each of the user common areas as well as the applicable subsystem and namelist block names are given below. Refer to Paragraph 8.1 for a definition of each of these variables.

<table>
<thead>
<tr>
<th>COMMON Block Name</th>
<th>Subsystem</th>
<th>Variable</th>
<th>NAMELIST Block Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>USERI</td>
<td>Stabilization and Control</td>
<td>ALPHA</td>
<td>OPTION</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AX</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>AY</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>AZ</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>DPHI</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>EA</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>EANT</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>EP1</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>K</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>MANV</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>OMEGR</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PDOTAV</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PDOTRX</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PDOTRY</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PDOTRZ</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PDOTST</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PDOTX</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PDOTY</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PDOTZ</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PDOT0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PHIFOV</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PHIRX</td>
<td>DESIRE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PHIRY</td>
<td>OPTION</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PHIRZ</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>TAGCEL</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>THETMX</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>THOLD</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>TL</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>TPMIN</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>TSMALL</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>XN</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>XNN</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>XN\n</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>XNNN</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>XNU</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>YN</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>ZN</td>
<td></td>
</tr>
</tbody>
</table>

3-9
<table>
<thead>
<tr>
<th>COMMON Block Name</th>
<th>Subsystem</th>
<th>Variable</th>
<th>NAMELIST Block Name</th>
</tr>
</thead>
</table>
| USER3            | Data Processing and Instrumentation | ARRAYN (11, 3)  
                   |                | BTRMX                | DESIRE             |
|                  |                            | NMSEQ                    | OPTION              |
|                  |                            | OPSM                     | DESIRE              |
|                  |                            | SCSFL                    | OPTION              |
|                  |                            | TPRFL                    |                      |
| USER4            | Communications             | BWIDTH(2)                | OPTION              |
|                  |                            | FREQ(2)                  |                      |
|                  |                            | FREQR                    |                      |
|                  |                            | IOPTCM (3)               |                      |
|                  |                            | LINK                     |                      |
|                  |                            | NADIR                    |                      |
|                  |                            | NET                      |                      |
| USER6            | Vehicle Sizing             | CGEXX (9)                | DESIRE              |
|                  |                            | EELOG (9)                |                      |
|                  |                            | EEOVL (9)                |                      |
|                  |                            | EMIXCG                   |                      |
|                  |                            | EMIZCG                   |                      |
|                  |                            | EM2XCG                   |                      |
|                  |                            | EM2ZCG                   |                      |
|                  |                            | EQPF                     | OPTION              |
|                  |                            | ISBQFG                   |                      |
|                  |                            | NUMEEQ                   | DESIRE              |
|                  |                            | XCGSA3                   | OPTION              |
| USER8            | Schedules                  | SKDME (7, 3)             | DESIRE              |
| USER9            | Structures                 | CA                       | OPTION              |
|                  |                            | CE                       | OPTION              |
| USERR            | Reliability                | ISPT                     | OPTION              |
|                  |                            | ISUB                     |                      |
|                  |                            | KEOPT                    |                      |
|                  |                            | RFIXED                   |                      |
|                  |                            | SLBMX                    |                      |
| USERC            | Costs                      | FEEPCT                   | OPTION              |
|                  |                            | IMETYP                   | DESIRE              |
|                  |                            | NFV                      | REQUIR              |
|                  |                            | NQV                      |                      |
|                  |                            | PI                       | DESIRE              |
In the above-mentioned COMMON regions the variables are either input from the user, defaulted by the BLOCK DATA routine or calculated by subroutine PRESET and used only by the subsystem mentioned. In the following COMMON region the variables are again either input from the user, defaulted by the BLOCK DATA routine or calculated by subroutine PRESET, but are used by more than one subsystem.

<table>
<thead>
<tr>
<th>COMMON Block Name</th>
<th>Subsystem</th>
<th>Variable</th>
<th>NAMELIST Block Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>USERP</td>
<td>Print</td>
<td>IPRINT</td>
<td>REQUIRE *</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ITITLE</td>
<td></td>
</tr>
</tbody>
</table>

* ITITLE is a special card input. It is the first card of any run prior to the NAMELISTs and describes the run (all 80 columns).
<table>
<thead>
<tr>
<th>COMMON Block Name</th>
<th>Subsystem</th>
<th>Variable</th>
<th>NAMELIST Block Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>USERI PRESET, SANDC, VESIZE</td>
<td>EQM2YL</td>
<td>DESIRE</td>
<td></td>
</tr>
<tr>
<td>PRESET, SANDC, VESIZE</td>
<td>EQM2ZL</td>
<td>DESIRE</td>
<td></td>
</tr>
<tr>
<td>SANDC, AUXPRO PRESET, THERMAL, COMM REL, COST</td>
<td>FE (TTHST in AP) IAGNCY</td>
<td>OPTION DESIRE</td>
<td></td>
</tr>
<tr>
<td>PRESET, SICE ALL</td>
<td>MB12SH</td>
<td>DESIRE</td>
<td></td>
</tr>
<tr>
<td>SPANDC, PRESET EP, REL</td>
<td>MICRO</td>
<td>REQUIR</td>
<td></td>
</tr>
<tr>
<td>EP, REL</td>
<td>OPTEMP</td>
<td>OPTION</td>
<td></td>
</tr>
<tr>
<td>SANDC, PRESET ORBINC</td>
<td>OPTIM</td>
<td>OPTION</td>
<td></td>
</tr>
<tr>
<td>THRML</td>
<td>PERIGE</td>
<td>REQUIR</td>
<td></td>
</tr>
<tr>
<td>PRESET, SANDC, EP REL</td>
<td>RELME</td>
<td>DESIRE</td>
<td></td>
</tr>
<tr>
<td>PRESET, REL</td>
<td>SPEC (6)</td>
<td>REQUIR</td>
<td></td>
</tr>
<tr>
<td>PRESET, REL</td>
<td>SPEC1</td>
<td>REQUIR</td>
<td></td>
</tr>
<tr>
<td>PRESET, INTIL, SANDC, REL</td>
<td>T</td>
<td>REQUIR</td>
<td></td>
</tr>
<tr>
<td>SANDC, VESIZE</td>
<td>XCGSA1</td>
<td>OPTION</td>
<td></td>
</tr>
<tr>
<td>COST, PNRT</td>
<td>XMER</td>
<td>DESIRE</td>
<td></td>
</tr>
<tr>
<td>COST, PRNT</td>
<td>XMEU</td>
<td>DESIRE</td>
<td></td>
</tr>
</tbody>
</table>

### 3.2.2 DBCOM

DBCOM acts as storage for blocks of the data base. All data base values for one of the hardware selection subroutines (i.e., all 55 attributes associated with all equipments relevant to that subsystem) are read at one time. These values are stored in matrix DATAB (55, 100). In addition, the COMMON contains IDB(30), (see Paragraph 3.3) which is filled by the read routine. IDB(I) contains the last column number for the Ith equipment of the active subsystem.

3-12
3. 2. 3. **CHOSE**

The named COMMON block CHOSE contains values pertaining to equipment already chosen. ICHOSE(60) and NCHOSE(60) are concatenations of the separate ICHOSE(I) and NCHOSE(I) of each subsystem which selects hardware as discussed in Paragraph 3.3. COST(5, 60) is a matrix formed by selecting the following rows from the data base for each equipment selected:

<table>
<thead>
<tr>
<th>Matrix</th>
<th>Row of DATAB</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>COST (1, I)</td>
<td>46</td>
<td>Design engineering cost</td>
</tr>
<tr>
<td>COST (2, I)</td>
<td>47</td>
<td>Test and evaluation cost</td>
</tr>
<tr>
<td>COST (3, I)</td>
<td>48</td>
<td>Unit production cost</td>
</tr>
<tr>
<td>COST (4, I)</td>
<td>49</td>
<td>Reference quantity</td>
</tr>
<tr>
<td>COST (5, I)</td>
<td>50</td>
<td>Factor</td>
</tr>
</tbody>
</table>

SKD(7, 60) is a matrix formed by selecting the following rows from the data base for each equipment selected.

<table>
<thead>
<tr>
<th>Matrix</th>
<th>Row of DATAB</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SKD (1, I)</td>
<td>46</td>
<td>Design engineering cost</td>
</tr>
<tr>
<td>SKD (2, I)</td>
<td>47</td>
<td>Test and evaluation cost</td>
</tr>
<tr>
<td>SKD (3, I)</td>
<td>51</td>
<td>Development constant</td>
</tr>
<tr>
<td>SKD (4, I)</td>
<td>52</td>
<td>Development variable</td>
</tr>
<tr>
<td>SKD (5, I)</td>
<td>53</td>
<td>Qualification constant</td>
</tr>
<tr>
<td>SKD (6, I)</td>
<td>54</td>
<td>Qualification variable</td>
</tr>
<tr>
<td>SKD (7, I)</td>
<td>55</td>
<td>State-of-the-art factor</td>
</tr>
</tbody>
</table>

REL(6, 60) is a matrix formed by selecting the following rows from the data base for each equipment selected.
<table>
<thead>
<tr>
<th>Matrix</th>
<th>Row of DATAB</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>REL (1, I)</td>
<td>23</td>
<td>Weight</td>
</tr>
<tr>
<td>REL (2, I)</td>
<td>41</td>
<td>Failure model</td>
</tr>
<tr>
<td>REL (3, I)</td>
<td>42</td>
<td>$\lambda$ or $\mu$</td>
</tr>
<tr>
<td>REL (4, I)</td>
<td>43</td>
<td>$\sigma$</td>
</tr>
<tr>
<td>REL (5, I)</td>
<td>44</td>
<td>$q$</td>
</tr>
<tr>
<td>REL (6, I)</td>
<td>45</td>
<td>Maximum redundancy</td>
</tr>
</tbody>
</table>

THM(4, 60) is a matrix formed by selecting the following rows from the data base for each equipment selected:

<table>
<thead>
<tr>
<th>Matrix</th>
<th>Row of DATAB</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>THM (1, I)</td>
<td>17</td>
<td>Maximum power</td>
</tr>
<tr>
<td>THM (2, I)</td>
<td>18</td>
<td>Minimum power</td>
</tr>
<tr>
<td>THM (3, I)</td>
<td>27</td>
<td>Maximum temperature</td>
</tr>
<tr>
<td>THM (4, I)</td>
<td>28</td>
<td>Minimum temperature</td>
</tr>
</tbody>
</table>

DPIA(11, 60) is a matrix formed by selecting the following rows from the data base for each equipment selected:

<table>
<thead>
<tr>
<th>Matrix</th>
<th>Row of DATAB</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DPIA (1, I)</td>
<td>30</td>
<td>Number power commands</td>
</tr>
<tr>
<td>DPIA (2, I)</td>
<td>31</td>
<td>Number other commands</td>
</tr>
<tr>
<td>DPIA (3, I)</td>
<td>32</td>
<td>Number time tags</td>
</tr>
<tr>
<td>DPIA (4, I)</td>
<td>33</td>
<td>Number high rate analog points</td>
</tr>
<tr>
<td>DPIA (5, I)</td>
<td>34</td>
<td>Number high rate digital points</td>
</tr>
<tr>
<td>DPIA (6, I)</td>
<td>35</td>
<td>High sample rate</td>
</tr>
<tr>
<td>DPIA (7, I)</td>
<td>36</td>
<td>Word length</td>
</tr>
<tr>
<td>DPIA (8, I)</td>
<td>37</td>
<td>Number low rate analog points</td>
</tr>
<tr>
<td>Matrix</td>
<td>Row of DATABASE</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------------</td>
<td>-------------------------------------------------</td>
</tr>
<tr>
<td>DPIA (9, I)</td>
<td>38</td>
<td>Number low rate digital points</td>
</tr>
<tr>
<td>DPIA (10, I)</td>
<td>39</td>
<td>Low sample rate</td>
</tr>
<tr>
<td>DPIA (11, I)</td>
<td>40</td>
<td>Word length</td>
</tr>
</tbody>
</table>

These matrices are needed by the subroutines that have similar names. For example, COST is used by COSTS, REL is used by RELY, SKD is used by SKED, THM is used by THRML, and DPIA is used by DPI.

3.2.4 **PRTCOM**

PRTCOM is used to pass values to the print subroutine which are not needed (except for output) outside of a given routine. A description of the variables in this COMMON block is given below:

<table>
<thead>
<tr>
<th>Name</th>
<th>From</th>
<th>Units</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACCRCY</td>
<td>SANDC</td>
<td>deg</td>
<td>S&amp;C accuracy</td>
</tr>
<tr>
<td>AM</td>
<td>STRUCT</td>
<td>---</td>
<td>Number of frames</td>
</tr>
<tr>
<td>AN</td>
<td>STRUCT</td>
<td>---</td>
<td>Number of stringers</td>
</tr>
<tr>
<td>BF</td>
<td>STRUCT</td>
<td>in.</td>
<td>Frame height</td>
</tr>
<tr>
<td>BS</td>
<td>STRUCT</td>
<td>in.</td>
<td>Stringer height</td>
</tr>
<tr>
<td>CDPI (7, 2)</td>
<td>DPI, MIS</td>
<td>---</td>
<td>Engineering &amp; mission equipment data for CDPI</td>
</tr>
<tr>
<td>CISTAR</td>
<td>EP</td>
<td>amp-hr</td>
<td>Battery capacity</td>
</tr>
<tr>
<td>CTOT</td>
<td>COST</td>
<td>$</td>
<td>Flight operations cost</td>
</tr>
<tr>
<td>DDTE</td>
<td>COST</td>
<td>$</td>
<td>DDT&amp;E program total</td>
</tr>
<tr>
<td>DE</td>
<td>COST</td>
<td>$</td>
<td>Design engineering cost</td>
</tr>
<tr>
<td>DRIWT</td>
<td>AUXPRO</td>
<td>lb</td>
<td>Weight of AP less expendables</td>
</tr>
<tr>
<td>EQBSTR</td>
<td>VESIZE</td>
<td>lb</td>
<td>Equipment bay structure weight</td>
</tr>
<tr>
<td>FEEINV</td>
<td>COST</td>
<td>$</td>
<td>Investment contractor fee</td>
</tr>
<tr>
<td>FEEOPS</td>
<td>COST</td>
<td>$</td>
<td>Operations contractor fee</td>
</tr>
</tbody>
</table>

3-15
<table>
<thead>
<tr>
<th>Name</th>
<th>From</th>
<th>Units</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FEER</td>
<td>COST</td>
<td>$</td>
<td>DDT&amp;E contractor fee</td>
</tr>
<tr>
<td>GSE</td>
<td>COST</td>
<td>$</td>
<td>DDT&amp;E GSE</td>
</tr>
<tr>
<td>IREL</td>
<td>RELY</td>
<td>---</td>
<td>0 means single system; 1 means dual system</td>
</tr>
<tr>
<td>ITRUNC</td>
<td>RELY</td>
<td>---</td>
<td>Index for reliability</td>
</tr>
<tr>
<td>MMDOLD</td>
<td>RELY</td>
<td>mo</td>
<td>Mean mission duration</td>
</tr>
<tr>
<td>NAME (3, 60)</td>
<td>SAVE</td>
<td>---</td>
<td>Name of equipment type</td>
</tr>
<tr>
<td>OPS</td>
<td>COST</td>
<td>$</td>
<td>Operations program total</td>
</tr>
<tr>
<td>PAYINV</td>
<td>COST</td>
<td>$</td>
<td>Total payload investment cost</td>
</tr>
<tr>
<td>PAYQUL</td>
<td>COST</td>
<td>$</td>
<td>DDT&amp;E qual. units cost</td>
</tr>
<tr>
<td>PAYR</td>
<td>COST</td>
<td>$</td>
<td>DDT&amp;E total payload cost</td>
</tr>
<tr>
<td>PE</td>
<td>COST</td>
<td>$</td>
<td>Unit engineering cost</td>
</tr>
<tr>
<td>PMP</td>
<td>COST</td>
<td>$</td>
<td>Investment program management cost</td>
</tr>
<tr>
<td>PMR</td>
<td>COST</td>
<td>$</td>
<td>DDT&amp;E program management cost</td>
</tr>
<tr>
<td>POWER (6)</td>
<td>MAIN</td>
<td>watts</td>
<td>Power requirement of each subsystem</td>
</tr>
<tr>
<td>PU</td>
<td>COST</td>
<td>$</td>
<td>Unit production cost</td>
</tr>
<tr>
<td>PWR (60)</td>
<td>SAVE</td>
<td>watts</td>
<td>Power requirement of each component</td>
</tr>
<tr>
<td>QCP</td>
<td>COST</td>
<td>$</td>
<td>Investment quality control cost</td>
</tr>
<tr>
<td>QCR</td>
<td>COST</td>
<td>$</td>
<td>DDT&amp;E quality control cost</td>
</tr>
<tr>
<td>ROLD(60)</td>
<td>RELY</td>
<td>---</td>
<td>Reliability of each module</td>
</tr>
<tr>
<td>SABMWT</td>
<td>VESIZE</td>
<td>lb</td>
<td>Solar array boom weight</td>
</tr>
<tr>
<td>SATADP</td>
<td>VESIZE</td>
<td>lb</td>
<td>Adapter weight</td>
</tr>
<tr>
<td>SATINV</td>
<td>COST</td>
<td>$</td>
<td>Spacecraft investment cost</td>
</tr>
<tr>
<td>SATR</td>
<td>COST</td>
<td>$</td>
<td>DDT&amp;E spacecraft cost</td>
</tr>
<tr>
<td>SEIP</td>
<td>COST</td>
<td>$</td>
<td>Investment systems engineering &amp; integration cost</td>
</tr>
<tr>
<td>Name</td>
<td>From</td>
<td>Units</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-------</td>
<td>-------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>SEIR</td>
<td>COST</td>
<td>$</td>
<td>DDT&amp;E systems engineering and integration cost</td>
</tr>
<tr>
<td>SKTAU(1)</td>
<td>SKED</td>
<td>mo</td>
<td>Design and component development time (critical subsystem)</td>
</tr>
<tr>
<td>SKTAU(2)</td>
<td>SKED</td>
<td>mo</td>
<td>Component qualification time (critical subsystem)</td>
</tr>
<tr>
<td>SKTAU(3)</td>
<td>SKED</td>
<td>mo</td>
<td>Subsystem development time (critical subsystem)</td>
</tr>
<tr>
<td>SKTAU(4)</td>
<td>SKED</td>
<td>mo</td>
<td>Subsystem qualification time (critical subsystem)</td>
</tr>
<tr>
<td>SKTAU(5)</td>
<td>SKED</td>
<td>mo</td>
<td>Subsystem development and flight readiness time (critical subsystem)</td>
</tr>
<tr>
<td>SKTAU(6)</td>
<td>SKED</td>
<td>mo</td>
<td>Total subsystem critical time</td>
</tr>
<tr>
<td>SSREL(6)</td>
<td>RELY</td>
<td>---</td>
<td>Subsystem reliabilities</td>
</tr>
<tr>
<td>SUBE(7)</td>
<td>COST</td>
<td>$</td>
<td>Subsystem design eng. cost</td>
</tr>
<tr>
<td>SUBT(7)</td>
<td>COST</td>
<td>$</td>
<td>Subsystem test &amp; eval. cost</td>
</tr>
<tr>
<td>SUBUE(7)</td>
<td>COST</td>
<td>$</td>
<td>Subsystem unit eng. cost</td>
</tr>
<tr>
<td>SUBUP(7)</td>
<td>COST</td>
<td>$</td>
<td>Subsystem unit prod. cost</td>
</tr>
<tr>
<td>TA</td>
<td>STRUCT</td>
<td>in.</td>
<td>End cover thickness center</td>
</tr>
<tr>
<td>TAU(6, 6)</td>
<td>SKED</td>
<td>mo</td>
<td>Critical path for each subsys</td>
</tr>
<tr>
<td>TB</td>
<td>STRUCT</td>
<td>in.</td>
<td>End cover thickness aft</td>
</tr>
<tr>
<td>TC</td>
<td>STRUCT</td>
<td>in.</td>
<td>End cover thickness forward</td>
</tr>
<tr>
<td>TE</td>
<td>COST</td>
<td>$</td>
<td>Test and evaluation cost</td>
</tr>
<tr>
<td>TF</td>
<td>STRUCT</td>
<td>in.</td>
<td>Frame thickness</td>
</tr>
<tr>
<td>TOOLR</td>
<td>COST</td>
<td>$</td>
<td>DDT&amp;E tooling and test equipment cost</td>
</tr>
<tr>
<td>TOOLO</td>
<td>COST</td>
<td>$</td>
<td>Investment tooling and test equipment cost</td>
</tr>
<tr>
<td>TOTOPS</td>
<td>DPI</td>
<td>ips</td>
<td>Computer operations rate</td>
</tr>
<tr>
<td>TRUNC</td>
<td>RELY</td>
<td>mo</td>
<td>Reliability truncation time</td>
</tr>
<tr>
<td>TS</td>
<td>STRUCT</td>
<td>in.</td>
<td>Stringer thickness</td>
</tr>
<tr>
<td>TTT</td>
<td>STRUCT</td>
<td>in.</td>
<td>Skin thickness</td>
</tr>
</tbody>
</table>

3-17
### Communication of all design variables between subsystems

Communication of all design variables between subsystems is accomplished via COMMON block BTWN. A description of all variables contained in BTWN is given below:

<table>
<thead>
<tr>
<th>Name</th>
<th>From</th>
<th>To</th>
<th>Units</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACSSN</td>
<td>SANDC</td>
<td>REL</td>
<td>---</td>
<td>Number of sensors</td>
</tr>
<tr>
<td>ACSWP</td>
<td>AUXPRO</td>
<td>VS</td>
<td>lb</td>
<td>Propellant weight</td>
</tr>
<tr>
<td>ALT</td>
<td>PRESET</td>
<td>ALL</td>
<td>nmi</td>
<td>Average altitude</td>
</tr>
<tr>
<td>AREA</td>
<td>EP</td>
<td>VS</td>
<td>ft²</td>
<td>Solar array area</td>
</tr>
<tr>
<td>BATCAP</td>
<td>EP</td>
<td>REL,</td>
<td>amp-hr</td>
<td>Battery capacity</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PRNT</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BITRAT(2)</td>
<td>DPI</td>
<td>COMM</td>
<td>bps</td>
<td>Bit rate (mission equipment and housekeeping)</td>
</tr>
<tr>
<td>CLIFE</td>
<td>SANDC</td>
<td>AUXPRO</td>
<td>---</td>
<td>Cycle life of thrusters</td>
</tr>
<tr>
<td>CONVWT</td>
<td>SANDC</td>
<td>COSTS</td>
<td>lb</td>
<td>Converters weight</td>
</tr>
<tr>
<td></td>
<td>&amp; COMM</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>INITIL, VS</td>
<td>SANDC</td>
<td>ft</td>
<td>Vehicle diameter</td>
</tr>
<tr>
<td>DT</td>
<td>INITIL, VS</td>
<td>SANDC</td>
<td>ft</td>
<td>Dist. from c.g. to engine</td>
</tr>
</tbody>
</table>

3.2.5 BTWN

Communication of all design variables between subsystems is accomplished via COMMON block BTWN. A description of all variables contained in BTWN is given below:
<table>
<thead>
<tr>
<th>Name</th>
<th>From</th>
<th>To</th>
<th>Units</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DX</td>
<td>INITIL, VS</td>
<td>SANDC</td>
<td>ft</td>
<td>Gas jet lever arm (roll, pitch, and yaw)</td>
</tr>
<tr>
<td>DY</td>
<td>INITIL, VS</td>
<td>SANDC</td>
<td>ft</td>
<td></td>
</tr>
<tr>
<td>DZ</td>
<td>INITIL, VS</td>
<td>SANDC</td>
<td>ft</td>
<td></td>
</tr>
<tr>
<td>EQB LG</td>
<td>PRESET, VS</td>
<td>SANDC</td>
<td>in.</td>
<td>Equipment bay length</td>
</tr>
<tr>
<td>EQBSID</td>
<td>PRESET, VS</td>
<td>SANDC</td>
<td>in.</td>
<td>Equipment bay side</td>
</tr>
<tr>
<td>FC</td>
<td>SANDC</td>
<td>REL</td>
<td>hr^-1</td>
<td>APS thruster cycle rate</td>
</tr>
<tr>
<td>FF</td>
<td>SANDC</td>
<td>AUXPRO</td>
<td>lb</td>
<td>Attitude &amp; control thrust</td>
</tr>
<tr>
<td>HARNWT</td>
<td>VS</td>
<td>COSTS</td>
<td>lb</td>
<td>Harness weight (wiring)</td>
</tr>
<tr>
<td>HPT</td>
<td>THERMAL</td>
<td>PRNT</td>
<td>Btu/hr</td>
<td>Total heater power</td>
</tr>
<tr>
<td>HTPipe</td>
<td>THERMAL</td>
<td>PRNT</td>
<td>Btu/hr</td>
<td>Heat pipe</td>
</tr>
<tr>
<td>HTPT</td>
<td>THERMAL</td>
<td>PRNT</td>
<td>Btu/hr</td>
<td>Total heat pipe</td>
</tr>
<tr>
<td>HTRPRB</td>
<td>THERMAL</td>
<td>PRNT</td>
<td>Btu/hr</td>
<td>Battery heater power</td>
</tr>
<tr>
<td>HTRPWR</td>
<td>THERMAL</td>
<td>PRNT</td>
<td>Btu/hr</td>
<td>Heater power</td>
</tr>
<tr>
<td>IBTLOC</td>
<td>EP</td>
<td>THERMAL</td>
<td>---</td>
<td>Battery location (column no.)</td>
</tr>
<tr>
<td>LMBDD</td>
<td>EP</td>
<td>RELY</td>
<td>---</td>
<td>Depth of discharge of battery capacity</td>
</tr>
<tr>
<td>NC</td>
<td>EP</td>
<td>RELY</td>
<td></td>
<td>Number of cells</td>
</tr>
<tr>
<td>OMEGS</td>
<td>PRESET</td>
<td>SANDC</td>
<td>rad/ sec</td>
<td>Spin rate about yaw axis</td>
</tr>
<tr>
<td>PASSTR</td>
<td>VS</td>
<td>COSTS</td>
<td>lb</td>
<td>Equivalent structures weight</td>
</tr>
<tr>
<td>PJ</td>
<td>INITIL, VS</td>
<td>SANDC</td>
<td>slug-ft^2</td>
<td>Platform spin axis inertia</td>
</tr>
<tr>
<td>PL</td>
<td>ALL</td>
<td>EP</td>
<td>watts</td>
<td>Average power</td>
</tr>
<tr>
<td>PLMIN</td>
<td>ALL</td>
<td>EP</td>
<td>watts</td>
<td>Minimum power</td>
</tr>
<tr>
<td>PCOCNWT</td>
<td>SANDC &amp; EP</td>
<td>PRNT, COST</td>
<td>lb</td>
<td>Power control weight</td>
</tr>
<tr>
<td>RADA</td>
<td>THERMAL</td>
<td>PRNT</td>
<td>ft^2</td>
<td>Radiator area</td>
</tr>
<tr>
<td>RADAB</td>
<td>THERMAL</td>
<td>PRNT</td>
<td>ft^2</td>
<td>Battery radiator area</td>
</tr>
<tr>
<td>RAT</td>
<td>THERMAL</td>
<td>PRNT</td>
<td>ft^2</td>
<td>Total radiator area</td>
</tr>
<tr>
<td>RJ</td>
<td>INITIL, VS</td>
<td>SANDC</td>
<td>slug-ft^2</td>
<td>Rotor spin axis inertia</td>
</tr>
<tr>
<td>SABOLG</td>
<td>VS</td>
<td>STRUCT</td>
<td>in.</td>
<td>Solar array boom length</td>
</tr>
<tr>
<td>SATLG</td>
<td>VS</td>
<td>THERMAL</td>
<td>ft</td>
<td>Vehicle length</td>
</tr>
</tbody>
</table>

3-19
<table>
<thead>
<tr>
<th>Name</th>
<th>From</th>
<th>To</th>
<th>Units</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SATTWT</td>
<td>VS</td>
<td>PRNT</td>
<td>lb</td>
<td>Launch weight</td>
</tr>
<tr>
<td>SATWT</td>
<td>VS</td>
<td>PRNT</td>
<td>lb</td>
<td>Vehicle weight</td>
</tr>
<tr>
<td>SATXCG</td>
<td>INITIL, VS</td>
<td>SANDC</td>
<td>in.</td>
<td>X Axis center of gravity</td>
</tr>
<tr>
<td>SATYCG</td>
<td>INITIL, VS</td>
<td>SANDC</td>
<td>in.</td>
<td>Y Axis center of gravity</td>
</tr>
<tr>
<td>SATZCG</td>
<td>INITIL, VS</td>
<td>SANDC</td>
<td>in.</td>
<td>Z Axis center of gravity</td>
</tr>
<tr>
<td>SAIXL</td>
<td>INITIL, VS</td>
<td>SANDC</td>
<td>in.</td>
<td>X location of solar array</td>
</tr>
<tr>
<td>SAIYL</td>
<td>INITIL, VS</td>
<td>SANDC</td>
<td>in.</td>
<td>Y location of solar array</td>
</tr>
<tr>
<td>SAIZL</td>
<td>INITIL, VS</td>
<td>SANDC</td>
<td>in.</td>
<td>Z location of solar array</td>
</tr>
<tr>
<td>SIDE</td>
<td>INITIL, VS</td>
<td>SANDC</td>
<td>in.</td>
<td>Side of box shape vehicle</td>
</tr>
<tr>
<td>SYSLB</td>
<td>VS</td>
<td>REL</td>
<td>lb</td>
<td>System weight</td>
</tr>
<tr>
<td>THCMWT</td>
<td>VS</td>
<td>COSTS</td>
<td>lb</td>
<td>Thermal control weight</td>
</tr>
<tr>
<td>THRUST(2)</td>
<td>SANDC</td>
<td>AUXPRO</td>
<td>lb</td>
<td>Attitude and translational thrusts</td>
</tr>
<tr>
<td>TI</td>
<td>SANDC</td>
<td>AUXPRO</td>
<td>lb-sec</td>
<td>Total impulse</td>
</tr>
<tr>
<td>TNKWT</td>
<td>AUXPRO</td>
<td>COSTS</td>
<td>lb</td>
<td>Propellant feed systems weight</td>
</tr>
<tr>
<td>TPRIM</td>
<td>REL</td>
<td>SANDC</td>
<td>mo</td>
<td>Mission length</td>
</tr>
<tr>
<td>VB</td>
<td>EP</td>
<td>VESIZE, PRNT</td>
<td>ft³</td>
<td>Volume of battery</td>
</tr>
<tr>
<td>VCHP</td>
<td>THERMAL</td>
<td>PRNT</td>
<td>Btu/hr</td>
<td>Variable conductance heat pipe</td>
</tr>
<tr>
<td>VOL</td>
<td>ALL</td>
<td>VS</td>
<td>ft³</td>
<td>Accumulated volume</td>
</tr>
<tr>
<td>WATE</td>
<td>EP</td>
<td>VS, RELY1b</td>
<td></td>
<td>Solar array weight</td>
</tr>
<tr>
<td>WB</td>
<td>EP</td>
<td>VESIZE, PRNT</td>
<td>lb</td>
<td>Volume of battery</td>
</tr>
<tr>
<td>WBT</td>
<td>EP</td>
<td>VESIZE</td>
<td>lb</td>
<td>Weight of batteries</td>
</tr>
<tr>
<td>WT</td>
<td>ALL</td>
<td>VS</td>
<td>lb</td>
<td>Accumulated equipment wt.</td>
</tr>
<tr>
<td>XNZERO</td>
<td>EP</td>
<td>RELY</td>
<td>rad/sec</td>
<td>Orbital mean motion</td>
</tr>
<tr>
<td>XJ</td>
<td>INITIL, VS</td>
<td>SANDC</td>
<td>slug-ft²</td>
<td>Vehicle inertia (roll, pitch, yaw)</td>
</tr>
<tr>
<td>YJ</td>
<td>INITIL, VS</td>
<td>SANDC</td>
<td>slug-ft²</td>
<td></td>
</tr>
<tr>
<td>ZJ</td>
<td>INITIL, VS</td>
<td>SANDC</td>
<td>slug-ft²</td>
<td></td>
</tr>
</tbody>
</table>

3-20
This section describes the hardware selection procedure, the method of communication between the MAIN program and the hardware selection subroutines, and the general procedure used in systematically checking all hardware parameters until a component is found that meets the specifications. Discussions as to which hardware is selected can be found in the appropriate subsystem subroutine sections (see Section 4). There are five subroutines in which hardware is selected: SANDC (Stabilization and Control), AUXPRO (Auxiliary Propulsion), DPI (Data Processing and Instrumentation), COMM (Communications), and EP (Electrical Power). The procedures described in Table 3-1, Figures 3-2 and 3-3 are applicable to all of these subroutines.
Table 3-1. Hardware Selection Procedure in Kth Subsystem

Calling Sequence

SUBROUTINE SSK (IPIC, IERR, ITER, NCONF, ICHOOSE, NCHOOSE)

Definition of Variable Names

1. IPIC(NSIZE) = hardware index indicating data base column
   NSIZE = maximum number of equipments sized for any configuration

2. IERR = message flag
   0 means no message
   1 means first message only
   10 means second message only
   ...
   111 means first, second and third messages are applicable

3. ITER = iteration flag (0 means first time through)

4. NCONF(6) = system configuration indices

5. ICHOOSE(NEQUIP) = I.D. of hardware chosen
   NEQUIP = maximum equipments (in general, more than one manufacturer
            per equipment) in any configuration

6. NCHOOSE(NEQUIP) = number of identical pieces of hardware required

Additional Variables Used in Selection Procedure

7. DATAB(NR, NC) = data base for subsystem
   NR = total equipment attributes
   NC = total number of individual pieces of hardware

8. IDB(NTOTL)* = last hardware column index for all equipment
   NTOTL = total number of equipments in data base for this subsystem (all configurations)

* See Figure 3-2 for further explanation
FIRST ROW CONTAINS HARDWARE I. D.

<table>
<thead>
<tr>
<th>EO1</th>
<th>EO2</th>
<th>EO3</th>
<th>EO4</th>
<th>EO5</th>
<th>EO6</th>
</tr>
</thead>
</table>

NOT SIZED

SIZED

IF THIS PIECE OF HARDWARE IS ACCEPTABLE,

\[ \text{CHOSE}(3) = \text{DATAB}(1, J1) = 4 \text{ digit I. D.} \]

3RD EQUIPMENT, 2ND CONFIGURATION

WHERE,

\[ J1 = \text{IDB}(5) + 3 \]

---

**Figure 3-2.** Explanation of Arrays: IDB and CHOOSE
SECOND CONFIGURATION

INITIALIZE VARIABLES
ICHOOSE(I) = 0
NCHOOSE(I) = 0
I = 1, NEQUIP
IERR = 0

SELECT HARDWARE NOT SIZED
i.e., ICHOOSE(N1) = DATAB(1,N2)
FOR APPROPRIATE N1 AND N2
SET NONZERO VALUES OF NCHOOSE

Figure 3-3. Hardware Selection Flow Chart
Figure 3-3. Hardware Selection Flow Chart (Continued)
4. SUBROUTINES WHICH SELECT HARDWARE

4.1 SUBROUTINE SANDC (IPIC, IERR, ITER, NCONF, ICHOSE, NCHOSE)

4.1.1 Purpose of Subroutine

The Stabilization and Control Subsystem stabilizes a spacecraft to a desired accuracy about a tracking line from a reference on the vehicle to an external reference. The external reference may be the local vertical of a planet, the sun, or a more distant star; an inertial reference; or the line of sight to a natural phenomenon like a gravity gradient or the lines of the earth's magnetic field. In many cases, a platform free to rotate with respect to the main structure of the vehicle must also be aligned with an external reference. The necessary accuracy of attitude stabilization depends, of course, on the mission of the vehicle.

In the beginning of SANDC, the subroutine computes the disturbance torques (XMD, YMD, and ZMD). These disturbance torques are the combination of gravity gradient torques, aerodynamic torques and solar torques. The solar or aerodynamic torques are a function of the altitude. The disturbance torques are then used in the selection criteria equations for all configurations.

The principal calculations, other than those necessary to select stabilization and control equipment, are contained in equations for thrust, cycle life, and total impulse. (These are necessary for the correct selection or sizing of equipment in auxiliary propulsion). Sensor selection is based on factors such as deadband and pointing errors (with respect to various axes). The equations for sensor selection tend to be quite complicated and involve user input, numbers from other subsystems, and values from the data base for many of the selected equipments. As an example, star sensors are selected on the basis of type (mappers or trackers), rate error, pointing error, sensitivity, and compatibility with the selected gyro and control moment.
gyros (CMGs). CMGs are selected on the basis of momentum, gimbal rate, and torque. Reaction (or momentum) wheels are selected on the basis of the angular momentum required.

Those equipments which are not chosen on the basis of selection criteria in the model are simply "called up" from the database.

The five configurations and their equipments are as follows:

<table>
<thead>
<tr>
<th>Configuration</th>
<th>Equipoements</th>
</tr>
</thead>
</table>
| **a. Dual Spin** \( [NCONF(1) = 1] \) | 1. Despin mechanical and electrical assembly  
2. Valve driver assembly  
3. Sun sensor  
4. Nutation damper \( \text{called up} \)  
5. Gimbal electronics  
6. Control timing assembly  
7. Gimbal drive assembly  
8. Nonscanning earth sensor \( \text{selected} \)  
9. Power converter \( \text{called up} \) |
| **b. Yaw Spin** \( [NCONF(1) = 2] \) | 1. Sun sensor  
2. ACS electronics \( \text{called up} \)  
3. Rate gyro  
4. Horizon sensor \( \text{selected} \)  
5. Reaction wheel  
6. Power converter  
7. Valve driver \( \text{called up} \) |
| **c. Three-Axis Mass Expulsion** \( [NCONF(1) = 3] \) | 1. Attitude reference electronics  
2. Valve driver \( \text{called up} \)  
3. Power converter  
4. Rate integrating gyro \( \text{selected} \)  
5. Scanning earth sensor |
| **d. Momentum Exchange** \( [NCONF(1) = 4] \) | 1. Electronics processor \( \text{called up} \)  
2. Valve driver  
3. Horizon sensor or sun sensor \( \text{called up} \)  
4. Control moment gyros  
5. Rate integrating gyros \( \text{selected} \)  
6. Star sensor |
e. **Pitch Momentum Bias**

1. Valve driver 
2. Electronic error processor 
3. Horizon sensor 
4. Momentum wheel 

4.1.2 Communication with Main Program

The variables in the calling sequence are discussed in Paragraph 3.3

User inputs are communicated by the COMMON areas USER1 and USER1. These are discussed in Paragraph 8.1 (see NAMELISTs REQUIR, DESIRE and OPTION). Variables are passed to and received from other subroutines through the COMMON area BTWN, which is discussed in Paragraph 3.2. The fourth COMMON area in this subroutine is DBCOM, which contains all necessary data base values and an indexing scheme to reference the values. DBCOM is also discussed in Paragraph 3.2.

4.1.3 Variables Specified in DATA Statements

Four variables appear in DATA statements. Three of these (XMD2, YMD2, and ZMD2) are approximations for external torques (ft-lb). The other, DI, is a minimum gas jet on-time in seconds.

4.1.4 Other Subroutines Called

None

4.2 **SUBROUTINE AUXPRO (IPIC, IERR, ITER, NCONF, ICHOOSE, NCHOOSE)**

4.2.1 Purpose of Subroutine

The auxiliary propulsion subroutine selects hardware which is required to provide attitude control forces and stationkeeping or maneuvering forces. Three configurations are considered in the subroutine. These
configurations are characterized by the nature of the propellant under investigation: cold gas, monopropellant, bipropellant.

All thrusters, isolation valves, filters, regulators, and tanks are selected by comparing appropriate attributes listed in the data base with satellite performance requirements determined by the model. Thrusters are selected on the basis of thrust level, isolation valves on the basis of effective flow area, filters on the basis of flow impedance, regulators on the basis of effective flow area and pressure operating range, and tanks on the basis of volume and pressure.

The model does not include selection criteria for the fill and vent valves, fill and drain valves, or the relief valves. The first valves in the appropriate equipment slots in the data base are simply called up.

The sequence in which equipments are selected in each configuration are given below:

a. Cold Gas \([NCONF(2) = 1]\)

1. Attitude and control thrusters *
2. Translational thrusters *
3. Pneumatic isolation valves (selected)
4. Pneumatic filters
5. Pneumatic regulator
6. Pneumatic tank
7. Fill and vent valve
8. Relief valve (called up)

b. Monopropellant \([NCONF(2) = 2]\)

1. Attitude and control thrusters *
2. Translational thrusters *
3. Fuel circuit isolation valves (selected)
4. Fuel circuit filters
5. Pneumatic regulator
6. Pneumatic isolation valve

* Those thrusters which come closest to satisfying the thrust requirements are always chosen, whether the program is in a MICRO or MACRO mode of calculation.
7. Fuel tank **
8. Pneumatic tank
9. Fill and drain valve
10. Fill and vent valve
11. Relief valve

**

Pneumatic tank

\{ called up \}

Relief valve

\{ called up \}

\{ selected \}

\{ selected \}

\{ called up \}

\{ called up \}

\} (called up)

\} (called up)

\} (called up)

\} (called up)

\} (called up)
Plumbing and connector weight in each configuration is estimated from the combined tank weight.

4.2.2 Communication with Main Program

The variables listed in the calling sequence are common to all subroutines which select hardware and are discussed in Paragraph 3.3. In addition to the calling sequence, subroutine AUXPRO communicates with the main program via three COMMON blocks: USERI, BTWN, and DBCOM. Variables coming through USERI are user inputs discussed in Paragraph 8.1 (see NAMELISTs REQUIR, DESIRE and OPTION). The variables in BTWN and DBCOM are discussed in Paragraph 3.2.

4.2.3 Variables Specified in DATA Statements

DATA XMR/1.5/
XMR = mixing ratio for bipropellant configuration

4.2.4 Other Subroutines Called

None

4.3 SUBROUTINE DPI (IPIC, IERR, ITER, NCONF, ICHOSE, NCHOSE, NOWAT)

4.3.1 Purpose of Subroutine

The data processing and instrumentation subroutine selects hardware which is required for mission equipment data processing, command decoding, and monitoring purposes. Two configurations are considered in the subroutine: general purpose processing and special purpose processing. In the general purpose mode, a computer on board the satellite performs all data processing tasks unless there is a requirement for separate processing of telemetry data. In this case, a separate digital telemetry unit (DTU) is used to process the housekeeping data. In the special purpose mode, all processing is performed by DTUs. If the communications configuration involves uplink plus downlink, unified link-common antenna, or unified
link-separate antennas, a single DTU performs all mission equipment and housekeeping data processing. If the communications configuration involves unified link-common antenna plus downlink, or unified link-separate antennas plus downlink, one DTU is used for mission equipment data processing and one DTU is used for housekeeping data processing.

The general purpose computer is selected on the basis of total required instructions (or operations) per second. The DTUs are not sized. The first DTUs in the appropriate equipment slot in the data base are simply called-up.

The following quantities are computed in the sequence indicated:

a. Requirement for a digital multiplexer
b. Number of mainframe words
c. Word length
d. Bit rate
e. Number of words per subframe
f. Number of subframes

The above quantities are computed regardless of the configuration in subroutine DPI. Depending on the configuration, the following operations are performed in the sequence indicated:

a. **General Purpose Processing** [NCONF(3) = 1]
   1. If telemetry data is processed separately, select one DTU. Otherwise, compute telemetry operations per second.
   2. Compute attitude control, command, and total operations per second.
   3. Select general purpose computer.

b. **Special Purpose Processing** [NCONF(3) = 2]
   1. Depending on the communications configuration (as discussed previously), a DTU may or may not be selected for mission equipment data processing.
   2. Select DTU for housekeeping data processing.
4.3.2 Communication with Main Program

The variables listed in the calling sequence with the exception of NOWAT, are common to all subroutines which select hardware and are discussed in Paragraph 3.3. NOWAT is one greater than the number of entries in the ARRAY table (i.e., DPIA matrix in Paragraph 3.2).

In addition to the calling sequence, subroutine DPI communicates with the main program via six COMMON blocks: CHOSE, BTWN, DBCOM, USERI, USER3, and PRTCOM. Variables coming through USER3 are user inputs described in Paragraph 8.1 (see NAMELIST REQUIR, DESIRE and OPTION).

4.3.3 Variables Specified in DATA Statements

DATA ACSRT, ACSOP, COMOP, OPREQ/10., 50., 6., 4./

where:

ACSRT = ACS rate (sec\(^{-1}\))
ACSOP = ACS operations
COMOP = Command operations
OPREQ = TLM operations required

4.3.4 Other Subroutines Called

4.3.4.1 Subroutine MIS (IPIC, IERR, ITER, NCONF, ICHOSE, NCHOSE)

The purpose of this subroutine is to select a DTU for mission equipment data processing. It is called from subroutine DPI in the special purpose processing configuration for the specific communications configurations discussed in Paragraph 4.3.1. The same six quantities (i.e., requirement for digital multiplexer, number of mainframe words, word length, bit rate, number of words per subframe, and number of subframes) which are computed in subroutine DPI for all equipment on board the satellite are computed for the mission equipment in subroutine MIS.

The variables listed in the calling sequence are discussed in Paragraph 3.3.
4.3.4.2 Subroutine ORDER (N, A, B, C, XM2, MEDIAN)

The purpose of this subroutine is to order array A from the highest to the lowest entry and determine the median entry in this array. The high rate telemetry points are ordered with respect to both sample rate and word length while the low rate telemetry points are ordered only with respect to sample rate. This information is used to determine mainframe sample rate and maximum word length.

This subroutine is called by both subroutines DPI and MIS. The variables in the calling sequence are defined as follows:

N       = Number of entries in telemetry points table
A       = One-dimensional array consisting of sample rates or word lengths
B       = One-dimensional array consisting of number of analog and digital points
C       = One-dimensional array consisting of sample rates or word lengths
XM2     = Twice the median value of array A after it has been ordered
MEDIAN  = Median entry in array A

4.4 SUBROUTINE COMM (IPIC, IERR, ITER, NCONF, ICHOSE, NCHOSE)

4.4.1 Purpose of Subroutine

The communication subroutine selects hardware for the satellite command and telemetry system. Five configurations are provided for in the subroutine. These are determined by the complexity of the data processor being used and the amount of data to be transmitted. The pieces of equipment which may be selected are: baseband assembly unit, transmitter antenna(s), transmitter(s), receiver antenna, receiver, diplexer, and signal conditioner. The pieces chosen and the number chosen are configuration dependent. Each piece of equipment to be chosen is selected by comparing the attributes as computed from the user input, configuration number, and default parameter values, with the attributes for that piece of equipment in the data base.
There are pieces of equipment which have constraints placed on
them for the selection process. For example, a given baseband assembly unit
may be constrained for use with a given transmitter and no other. These
constraints are built into the data base.

The sequences in which equipment are selected in each configura-
tion are given below:

a. **Uplink Plus Downlink [NCONF(4) = 1]**

1. Transmitter antenna
2. Transmitter
3. Receiver antenna
4. Receiver
5. Signal conditioner

b. **Unified Link, Common Antenna [NCONF(4) = 2]**

1. Baseband assembly unit
2. Antenna
3. Transmitter
4. Receiver
5. Signal conditioner
6. Diplexer

b. **Unified Link, Separate Antennas [NCONF(4) = 3]**

1. Baseband assembly unit
2. Transmitter antenna
3. Transmitter
4. Receiver antenna
5. Receiver
6. Signal conditioner

b. **Unified Link, Common Antenna plus Downlink [NCONF(4) = 4]**

1. Baseband assembly unit
2. Transmitter antenna (unified)
3. Transmitter antenna (nonunified)
4. Transmitter (unified)
5. Transmitter (nonunified)
6. Receiver
7. Signal conditioner
8. Diplexer
4.4.2 Communication with Main Program

The variables listed in the calling sequence are common to all subroutines which select hardware and are discussed in Paragraph 3.3.

In addition to the calling sequence, Subroutine COMM communicates with the main program via four COMMON blocks: USER4, USERI, BTWN; and DBCOM. Variables coming through USER4 and USERI are user inputs discussed in Paragraph 8.1 (see NAMELIST REQUIRE, DESIRE and OPTION). The variables in BTWN and DBCOM are discussed in Paragraph 3.2.

4.4.3 Variables Specified in DATA Statements

\[
\text{DATA SIGNOI/10.,10./, LMARG/6.,6./, SLANT/-1.E+10/, GTOT/-1.E+10/, GR/-1.E+10/, T/-1.E+10/, NF/-1.E+10/;}
\]

\[
\text{TCLOSS/0.,0./, POLLOSS/0./, GAMMA/.1/, BETA/1.8/, GT/-1.E+10, -1.E+10/, MODX/0., 0./, ANTLOS/0./,}
\]

\[
\text{COVER/0./}
\]

\[
\text{where:}
\]

\[
\text{SIGNOI(2) = Signal-to-noise ratios for transmitter(s) (dB)}
\]

\[
\text{LMARG(2) = Link margin(s) (dB)}
\]

\[
\text{SLANT = Slant range (nmi)}
\]

\[
\text{GTOT = Gain-to-temperature ratio}
\]

\[
\text{GR = Receiving antenna (downlink) gain (dB)}
\]

\[
\text{T = System noise temperature (°K)}
\]

\[
\text{NF = Noise figure (dB)}
\]
TCLOSS(2) = Transmitter(s) circuit loss
POLOSS  = Polarization loss
ANTLCS  = Satellite antenna off-axis loss
GAMMA   = PRN modulation index
BETA    = Subcarrier modulation index
GT(2)   = Antenna(s) gain (dB)
MODX(2) = Transmitter(s) modulation type
  MODX = 0 no equipment dependence
  MODX = 1 phase modulation
  MODX = 2 frequency modulation
  MODX = 3 amplitude modulation
COVER   = Transmitter antenna coverage (in percent)

4.4.4 Other Subroutines Called

4.4.4.1 Subroutine BESS (X, BESJ, NMAX)

This subroutine uses a recursive procedure for evaluating
tables of the Bessel function, \( J_n(x) \).

The variables in the calling sequence are defined as follows:

\( X \) = floating point single precision argument

\( BESJ \) = one-dimensional array of values of \( J_n(x) \)

\( NMAX \) = one less than the number of values in \( BESJ \) array:
  \( i.e., \ BESJ \ (n+1) = J_n(x), \ n = 0, \ldots, /NMAX/ \)

4.4.4.2 Function RESET (K)

This subroutine, as the name implies, resets or initializes
equipment indices in the data base.
4.5 SUBROUTINE EP (IPIC, IERR, ITER, NCONF, ICHOSE, NCHOSE)

4.5.1 Purpose of Subroutine

The electrical power subroutine selects hardware which is required to regulate the electrical power for the spacecraft and batteries to store the electrical power. Six configurations are considered in the subroutine. These configurations are characterized by the nature of the regulation and the configuration of the solar arrays.

All regulators, batteries and battery chargers are selected by comparing appropriate attributes listed in the data base with satellite performance determined by the model. Regulators are selected on the basis of their ability to regulate the power load, batteries on the basis of the capacity needed during the eclipse portion of orbit, and battery chargers on the basis of being able to use the excess power to store energy back into the battery.

The model does not include selection criteria for power control units, central control units, solar power distributor, and power distributors. The first equipments available in the data base are simply called up. The solar array area and weight are sized primarily on the average power load required for the spacecraft.

The sequences in which equipments are selected in each configuration are given below:

a. Shunt Regulation - Paddle or Body Mounted Arrays

\[
NCONF(5) = 1 \text{ or } 2
\]

1. Shunt regulator
2. Battery
3. Battery charger
4. Power control unit

\{(selected)\}

\{(called up)\}

b. Shunt and Discharge Regulation - Paddle or Body Mounted Arrays

\[
NCONF(5) = 3 \text{ or } 4
\]

1. Discharge regulator
2. Shunt regulator
3. Battery
4. Battery charger
5. Central control unit

\{(selected)\}

\{(called up)\}
c. Series Load Regulation - Paddle or Body Mounted Arrays

[CONF(5) = 5 or 6]

1. Series load regulator
2. Battery (selected)
3. Battery charger
4. Power distributors
5. Solar power distributors (called up)

4.5.2 Communication with Main Program

The variables listed in the calling sequence are common to all subroutines which select hardware and are discussed in Paragraph 3.3.

In addition to the calling sequence, subroutine EP communicates with the main program via four named COMMON blocks: PRTCOM, USERI BTWN, and DBCOM. Variables coming through USERI are user inputs discussed in Paragraph 8.1 (see NAMELISTs REQUIR, DESIRE and OPTION). The variables in BTWN, DBCOM and PRTCOM are discussed in Paragraph 3.2.

4.5.3 Variables Specified in DATA Statements

\[
\begin{align*}
\text{DATA} & \ D ELF/0.03/, D E L I/0.02/, D E L M/0.01/, E T A I/0.105/, \\
& \ E T A R/1.0/, K 1/1.02/, K 2/1.4/, L M B D P/0.9/, S O L/1353/, \\
& \ V C/1.1/, P I E/3.1416/, C H M I N T/2.0/ \\
\end{align*}
\]

where:

- **DELF** = Coverglass and coverglass adhesive transmissivity loss factor (dimensionless)
- **DELI** = Array fabrication loss factor (dimensionless)
- **DELM** = Miscellaneous loss factor (dimensionless)
- **ETAI** = Solar cell efficiency at 28°C, AMO illumination (dimensionless)
- **ETAR** = Power distribution loss factor (array to loads)
- **K1** = Battery packing factor (dimensionless)
- **K2** = Battery structure weight factor (dimensionless)
- **LMBDP** = Solar array factor (dimensionless) (active surface area/actual surface area)
- **SOL** = Average solar intensity (watts/meter²)

4-14
VC = Minimum allowable cell voltage (V dc)
CHMINT = Minimum allowable charge time (hr)

4.5.4 Other Subroutines Called
None
5. SUBROUTINES WHICH DO NOT SELECT HARDWARE

5.1 SUBROUTINE PRESET (IERR)

5.1.1 Purpose of Subroutine

The purpose of the subroutine PRESET is to calculate values for those input variables for which flags have been specified, provided these flags have not been overwritten by user supplied inputs.

5.1.2 Communication with Main Program

IERR is a flag that informs the main program that subsystem reliabilities cannot be preset with the given information. Subroutine PRESET communicates with the main program via four named COMMON blocks: USER1, USERR, USERI and BTWN. Variables coming through USER1, USERR, and USERI are discussed in Paragraph 8.1.

5.1.3 Variables Specified in DATA Statements

None.

5.1.4 Other Subroutines Called

None.

5.2 SUBROUTINE FILTER (NCONF, ICODE)

5.2.1 Purpose of Subroutine

Some combinations of configurations are known to be unacceptable. These are filtered out without the necessity of calling any subsystems. As an example, configuration 1 in SANDC and configuration 1 in EP are incompatible because 1 in SANDC is a spinning vehicle and 1 in EP requires solar array paddles which cannot be used on a spinning vehicle. A complete description of these restrictions is presented in Section 7.
5.2.2 Communication with Main Program

NCONF is an array containing the number of each subsystem's configuration. ICODE is a return code of 0 for compatible configurations or -1 for unacceptable combinations of configurations.

FILTER also uses values from COMMONs USER1, USER3, USER4, and USER5, all of which are discussed in Paragraph 8.1.

5.2.3 Variables Specified in DATA Statements

None.

5.2.4 Other Subroutines Called

None.

5.3 SUBROUTINE INITIL (NCONF, IERRI)

5.3.1 Purpose of Subroutine

Some values are needed before they are calculated. For example, subroutine SANDC needs moments and lengths which are calculated "downstream" in vehicle sizing. Approximations for such values are calculated here.

5.3.2 Communication with Main Program

NCONF is discussed in Paragraph 3.1.1. IERRI is a flag which is set when the estimated satellite diameter exceeds the maximum allowable size. Subroutine INITIL communicates with the main program via four named COMMON blocks: USER1, USER3, BTWN and PRTCOM.

5.3.3 Variables Specified in DATA Statements

None.

5.3.4 Other Subroutines Called

None.
5.4  **SUBROUTINE READDB (IENDDB)**

5.4.1  **Purpose of Subroutine**

This subroutine reads all data base values for one subsystem at a time and determines the IDB array. Of major importance are the equipment numbers which exist as the first two digits of the four digit equipment identification numbers. These are counted by groups (all 1's, all 2's, all 3's, ....) and these counts exist as IDB(1), IDB(2), and so on. The routine returns when equipment for the next subsystem is encountered, i.e., when the equipment numbers begin to decrease.

5.4.2  **Communication with Main Program**

IENDDB is the last column in the data base for the active subsystem. This is needed for the SAVE routine. DBCOM is the common area in which the data base values for each subsystem are stored (see Paragraph 3.2).

5.4.3  **Variables Specified in DATA Statements**

DATA STORE/55*0. /
STORE = variable used for temporary storage

5.4.4  **Other Subroutines Called**

None

5.5  **SUBROUTINE SAVE (IIN, NIN, NOWAT, ITEST, IENDDB)**

5.5.1  **Purpose of Subroutine**

The purpose of this subroutine is to build matrices needed by other subsystems. Specifically this routine concatenates separate ICHOSE and NCHOSE arrays (with zeros taken out) which contain the hardware I.D. 's of the equipment selected for the five satellite subsystems and the number of each equipment type. It also saves the data required to fill the COST, REL, THM, DPIA, and SKD arrays for their subroutines and the component volume, power and name for the PRNT routine.
5.5.2 Communication with Main Program

IIN and NIN are ICHOSE and NCHOSE of the active subsystem (described in Paragraph 3.3). NOWAT is described in Paragraph 4.3.) ITEST is the largest possible number of types of equipment chosen by a subsystem. The three COMMON areas (DBCOM, CHOSE, PRTCOM) which are also used for communication with the main program are discussed in Paragraph 3.2.

5.5.3 Variables Specified in DATA Statements

None.

5.5.4 Other Subroutines Called

None.

5.6 SUBROUTINE VESIZE (IERR, NCONF, ICHOSE)

5.6.1 Purpose of Subroutine

The vehicle sizing subroutine determines the satellite structural weight, the total weight, the satellite volume, dimensions, center of gravity locations and the satellite inertial characteristics. Three configurations are considered in the subroutine. These configurations are characterized by the shape of the equipment bay: cylinder, box, sphere. This corresponds to NCONF(6) = 1, 2, 3, respectively.

The following quantities are computed in the sequence indicated:

a. Equipment bay equipment weight and volume
b. Equipment bay length
c. Satellite length
d. Solar array dimensions
e. Equipment bay structural weight
f. Mission equipment bay structural weight
g. Mission equipment support weight
h. Total volume of mission equipment bay
i. Solar array boom and mechanism weight (paddles)
j. Total mission equipment and external equipment weight and volume
k. Harness weight
l. Structural thermal protection system weight
m. Satellite dry weight
n. Satellite gross weight
o. Satellite launch weight
p. Mission equipment and mission equipment bay structure CGs
q. Equipment bay structure CGs
r. External equipment CGs
s. Solar array CGs
t. Satellite CGs
u. Equipment bay structure and equipment bay equipment incremental inertia
v. External equipment incremental inertia
w. Solar array incremental inertia
x. Mission equipment bay incremental inertia
y. Total satellite inertia
z. Distance from satellite CG to main engine
_a. Gas jet lever arms on roll, pitch, and yaw axes

5.6.2 Communication with Main Program

All three variables listed in the calling sequence are discussed in Paragraph 3.3. (In this subroutine ICHOSE is a scalar which is set to -1 when the current design is unacceptable.)

In addition to the calling sequence, subroutine VESIZE communicates with the main program via four COMMON blocks: USER1, USER6, BTWN, and PRTCOM. Variables coming through USER1 and USER6 are user inputs discussed in Paragraph 8.1 (see NAMELISTs REQUIR, DESIRE, and OPTION). The variables in BTWN and PRTCOM are discussed in Paragraph 3.2.
5.6.3 Variables Specified in DATA Statements

None.

5.6.4 Other Subroutines Called

None.

5.7 SUBROUTINE STRUCT (NCONF)

5.7.1 Purpose of Subroutine

The structures subroutine specifies the satellite loads environment and sizes the solar array extension supports, the equipment bay structure, the end covers and the midsection bulkhead if appropriate.

One configuration is considered in the subroutine. This configuration is characterized by the type of equipment bay structure: semi-monocoque.

The following quantities are computed in the sequence indicated:

a. Solar array paddle applied load
b. Nominal radius and wall thickness of solar array extension supports
c. Loads applied to equipment bay structure
d. Equivalent axial load on semi-monocoque structure
e. Equivalent thickness of stiffened cylinder
f. Skin thickness of skin-stringer assembly
g. Stringer thickness, height, spacing, and efficiency
h. Number of stringers
i. Cylinder frame, radius of gyration, area, height, thickness and spacing
j. Number of frames
k. Forward and aft end cover thickness
l. Applied uniform load on midsection bulkhead
m. Midsection bulkhead thickness

If the equipment bay shape is a box instead of a cylinder, quantities comparable to those listed above in steps d – k are computed for the box shape.
5.7.2 Communication with Main Program

The variable in the calling sequence is discussed in Paragraph 3.3. In addition to the calling sequence, subroutine STRUCT communicates with the main program via four COMMON blocks: USER9, USERI, BTWN, and PRTCOM. Variables coming through USER9 and USERI are user inputs discussed in Paragraph 3.1 (see NAMELISTs REQUIR, DESIRE, and OPTION). The variables in BTWN and PRTCOM are discussed in Paragraph 3.2.

5.7.3 Variables Specified in DATA Statements

DATA E, XNU, RHO, SIGY, PI/1.1E7, .33, .1, 3.E4, 3.1416/

where:

E = Young's modulus (psi)
XNU = Poisson's ratio
RHO = Weight density (lb/in^3)
SIGY = Yield stress (psi)

5.7.4 Other Subroutines Called

None.

5.8 SUBROUTINE RELY (IRTN, IDS, NEQUIP)

5.8.1 Purpose of Subroutine

The reliability subroutine incrementally increases the level of redundancy in the spacecraft system until the system reliability, R(TRUNC), and the mean mission duration, MMD, specifications are met. The procedure is constrained by a maximum total satellite weight or cost and available equipment reserves. The subroutine operates to meet the system reliability specification prior to meeting the mean mission duration requirement.

Two configurations are considered in the subroutine. These configurations are single system redundancy and dual system redundancy. This corresponds to NCONF(7) = 0, 1 respectively.
The principle of operation is to add a redundancy to a single module, then calculate the new system reliability and the payoff, as defined by

\[ \text{RHO} = \frac{\Delta R(\text{TRUNC})}{\Delta \text{weight}} \]

This is repeated for each module where equipment reserves are available. The module offering the greatest payoff is selected, and the following three tests are applied:

a. Is RHO large enough? (The threshold is preselected.)
b. Is spacecraft weight or cost below the maximum allowed?
c. Is the R(\text{TRUNC}) still short of the requirement?

If these tests are passed, the subroutine begins the selection process again. This loop is retraced until one or more of the tests is failed. Failure of tests a or b results in termination of the design procedure. If a configuration is found which meets the system reliability requirement, then the above is repeated replacing R(\text{TRUNC}) with MMD. A final design is recognized as optimum subject to the imposed R(\text{TRUNC}), MMD, weight, and cost constraints.

The subroutine contains the additional feature in that subsystem reliabilities may be specified. The task of meeting subsystem requirements is performed prior to any total system considerations. The same logic as presented above is used for determining the appropriate subsystem redundancies.

5.8.2 Communication with Main Program

The variables listed in the calling sequence are: a return indicator, a double string design indicator, and a vector of the number of equipment types per subsystem, respectively.

Subroutine RELY additionally communicates with the main program through the COMMON blocks: USERR, USERI, BTWN, DBCOM,
CHOSE, and PRTCOM. Variables in USERR and USERI are user inputs and are discussed in Paragraph 8.1 (see NAMELISTs REQUIRE, DESIRE, and OPTION). The variables in BTWN, CHOSE, DBCOM, and PRTCOM are discussed in Paragraph 3.2.

5.8.3 Variables Specified in DATA Statements

None

5.8.4 Other Subroutines Called

5.8.4.1 Subroutine RIMOD (J, DELH, ITRUNC, NT, IADD, IOPT)

Subroutine RIMOD is called by subroutine RELY. Subroutine RIMOD computes the reliability function for a specified module with or without a redundancy added. Five different models are used, depending on the failure mode of an individual module. The calling parameters are:

\[ J = \text{Current module number} \]
\[ \text{DELH} = \text{Time increment} \]
\[ \text{ITRUNC} = \text{Number of time points} \]
\[ \text{NT} = \text{Input option} \]
\[ \text{IADD} = \text{Input option} \]
\[ \text{IOPT} = \text{Input option} \]

Parameters passed through COMMON block CHOSE are:

\[ \text{NCHOSE} = \text{Initial number of elements by module} \]
\[ \text{SYSPAR} = \text{Matrix of model parameters (called DATAB in subroutine RELY)} \]

Parameters passed through COMMON block DBCOM are:

\[ \text{R} = \text{Resultant reliability function} \]
\[ \text{NR} = \text{Number of redundancies by module} \]

5.8.4.2 Subroutine QSF (H, Y, Z, NDIM)

Subroutine QSF is called by subroutine RELY. Subroutine QSF computes a vector of integral values for a given equidistant table of function values. QSF is a member of the System/360 Scientific Subroutine
Package. The calling parameters are:

- \( H \) = Increment of argument values
- \( Y \) = Input vector of function values
- \( Z \) = Resulting vector of integral values
- \( NDIM \) = Dimension of vectors \( Y \) and \( Z \)

No parameters are passed in common.

5.8.4.3 Subroutine GAM (X)

The function GAM is called by RIMOD. Function GAM computes the gamma function of its argument, \( X \). GAM uses a polynomial approximation on the interval \((1.0, 2.0)\).

5.8.4.4 Subroutine CERF (X)

The function CERF is called by RIMOD. Function CERF computes the error function for \( X \) in \((0.0, 4.0)\) and the compliment of the error function for \( X \) in \((4.0, \infty)\). A Chebyshev approximation is used in both cases.

5.9 SUBROUTINE THRML (IERR, NCONF)

5.9.1 Purpose of Subroutine

The thermal sizing subroutine determines the phase change material weight, insulation area, heater power, radiator area, and types of heat pipes to be used. Various configurations are considered in the subroutine dependent upon variables such as orbit, shape of vehicle, type of stabilization, power requirements, temperature limits, and battery temperatures. These variables are determined elsewhere in the model and passed to THRML via the common blocks.

The output quantities are computed in the following sequence:

- a. Radiator area (RADA)
- b. Heater power (HTRPWR)
- c. Heat pipe (HTPIPE)
d. Battery radiator area (RADAB)
e. Battery heater power (HTRPRB)
f. Battery variable conductance heat pipe (VCHP)
g. Total radiator area (RAT)
h. Total heater power (HPT)
i. Total heat pipes (HTPT)

5.9.2 Communication with Main Program

Both variables listed in the calling sequence are discussed in Paragraph 3.3.

In addition to the calling sequence, subroutine THRML communicates with the main program via three COMMON blocks: USERI, CHOSE, and BTWN. Variables coming through USERI are user inputs discussed in Paragraph 8.1 (see NAMELISTs REQUIRE, DESIRE, and OPTION). The variables in BTWN and CHOSE are discussed in Paragraph 3.2.

5.9.3 Variables Specified in DATA Statements

\[
\begin{align*}
\text{DATA} & \quad \text{SIGMA}/0.1714 \ E-08/, \ QS/442./, \ EMISS/60./, \ ALBDO/155./, \ CONST/1.5/, \ PIE/3.1416/ \\
\end{align*}
\]

where:

- SIGMA = Boltzmann constant in Btu/(hr*ft-deg $R^4$)
- QS = Solar constant in Btu/(hr*ft)$^2$
- EMISS = Earth emission in Btu/(hr*ft)$^2$
- ALBDO = The Albedo in Btu/(hr*ft)$^2$
- CONST = The K constant (dimensionless)

5.9.4 Other Subroutines Called

None
5.10  **SUBROUTINE COSTS (NCONF, NEQUIP)**

5.10.1  **Purpose of Subroutine**

The cost subroutine determines the cost of building and integrating a payload from the design engineering phase to the launch phase. Costs are broken down into the following categories (variable names are in parenthesis):

<table>
<thead>
<tr>
<th>DDT&amp;E (Nonrecurring)</th>
<th>Investment (Recurring)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Design engineering (DE)</td>
<td>Unit engineering (PE)</td>
</tr>
<tr>
<td>Test and evaluation (TE)</td>
<td>Unit production (PU)</td>
</tr>
<tr>
<td>Tooling and equipment (TOOLR)</td>
<td>Tooling and equipment (TOOLU)</td>
</tr>
<tr>
<td>Quality control (QCR)</td>
<td>Quality control (QCU)</td>
</tr>
<tr>
<td>Systems engineering and integration (SEIR)</td>
<td>Systems engineering and integration (SEIP)</td>
</tr>
<tr>
<td>Program management (PMR)</td>
<td>Program management (PMP)</td>
</tr>
</tbody>
</table>

Other costs which are computed are listed in the table below (variable names are written in where computed):

<table>
<thead>
<tr>
<th>Cost Category</th>
<th>DDT&amp;E</th>
<th>Investment</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spacecraft</td>
<td>SATR</td>
<td>SATINV</td>
<td></td>
</tr>
<tr>
<td>Mission equipment</td>
<td>XMER</td>
<td>XMEINV</td>
<td></td>
</tr>
<tr>
<td>Total payload</td>
<td>PAYR</td>
<td>PAYINV</td>
<td></td>
</tr>
<tr>
<td>Quality Units</td>
<td>PAYQUL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GSE</td>
<td>GSE</td>
<td></td>
<td>XLTOT</td>
</tr>
<tr>
<td>Launch support</td>
<td></td>
<td></td>
<td>CTOT</td>
</tr>
<tr>
<td>Flight operations</td>
<td></td>
<td></td>
<td>FEEINV</td>
</tr>
<tr>
<td>Contractor fee</td>
<td>FEER</td>
<td>FEEINV</td>
<td>FEEOPS</td>
</tr>
<tr>
<td>Program total</td>
<td>DDTE</td>
<td>XVEST</td>
<td>OPS</td>
</tr>
</tbody>
</table>

5-12
5.10.2 Communication with Main Program

Both variables listed in the calling sequences are discussed in Paragraph 3.3.

In addition to the calling sequence, subroutine COSTS communicates with the main program via five COMMON blocks: USERC, USERI, BTWN, CHOSE, and PRTCOM. Variables coming through USERC are user inputs discussed in Paragraph 8.1 (see NAMELISTS REQUIR, DESIRE, and OPTION). The variables in BTWN, CHOSE, and PRTCOM are discussed in Paragraph 3.2.

5.10.3 Variables Specified in DATA Statements

DATA FR, FP, FT, FE, RE, RT, RP, BE, BT, BP, PI, SF

where:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FR(6)</td>
<td>Subsystem design engineering cost factor</td>
</tr>
<tr>
<td>FP(6)</td>
<td>Subsystem unit production cost factor</td>
</tr>
<tr>
<td>FE(6)</td>
<td>Subsystem unit engineering cost factor</td>
</tr>
<tr>
<td>FT(6)</td>
<td>Subsystem test evaluation cost factor</td>
</tr>
<tr>
<td>RE(6)</td>
<td>Design engineering CER constant</td>
</tr>
<tr>
<td>RT(6)</td>
<td>Test evaluation CER constant</td>
</tr>
<tr>
<td>RP(6)</td>
<td>Production CER constant</td>
</tr>
<tr>
<td>BE(6)</td>
<td>Design engineering CER exponent</td>
</tr>
<tr>
<td>BT(6)</td>
<td>Test evaluation CER exponent</td>
</tr>
<tr>
<td>BP(6)</td>
<td>Production CER exponent</td>
</tr>
</tbody>
</table>

The six values in each of the above arrays are associated with the following equipment or systems in the order indicated:

a. Solar array
b. Wiring harness
c. Thermal
d. Converters
e. Propellant feed systems
f. Structures
In addition:

\[ PI = \text{Price index (i.e., change of the value of the dollars)} \]

\[ SF = \text{Optional factor (e.g., standardization factor)} \]

5.10.4 Other Subroutines Called

None

5.11. SUBROUTINE SKED (NEQUIP, NCONF)

5.11.1 Purpose of Subroutine

The purpose of this subroutine is to calculate component development lead time, subsystem development lead time, component qualification time, subsystem qualification lead time, test lead time, and a total time for each subsystem and for the mission equipment. The critical path is determined and the associated times are passed to the PRNT routine.

5.11.2 Communication with Main Program

Both variables in the calling sequence are discussed in Paragraph 3.3. Subroutine SKED also communicates with the main program via three COMMON areas: CHOSE, USER8, and PRTCOM. Variables coming through USER8 are user inputs discussed in Paragraph 8.1 (See NAMELISTs REQUIR, DESIRE, and OPTION). The variables in BTWN and PRTCOM are discussed in Paragraph 3.2.

5.11.3 Variables Specified in DATA Statements

DATA CONF, ICI

where:

\[ \text{CONF}(22, 5) = \text{Configuration dependent weighting factors} \]

\[ \text{ICI}(5) = \text{Index with which the CONF array is addressed} \]

5.11.4 Other Subroutines Called

None
5.12  SUBROUTINE PRNT (IERR, NEQUIP, NACCEP, NCONF)

5.12.1  Purpose of Subroutine

This subroutine prints all output determined by the model. A sample of the output may be found in Paragraph 8.3. This sample includes all three possible levels (system, subsystem, assembly) of output which are available as well as a glossary containing descriptive information. Depending on the value of the parameter, IPRINT, system, system plus subsystem or system, subsystem and assembly design information will be printed out for each design.

5.12.2  Communication with Main Program

The variables IERR and NCONF listed in the calling sequence are discussed in Paragraph 3.3. NEQUIP is discussed in Paragraph 3.1. NACCEP is a counter maintained by MAIN and used only by PRNT. It is the acceptable design number identifying the particular run.

In addition to the calling sequence, subroutine PRNT communicates with the main program via five COMMON blocks: BTWN, PRTCOM, CHOSE, USERP, and USERI.

5.12.3  Variables Specified in DATA Statements

None.

5.12.4  Other Subroutines Called

None.
6. DATA BASE

Paragraph 6.1 contains the discussion of the data base, the position of the attributes contained therein, and a description of the data base tape. Paragraph 6.2 discusses the PRESORT program which may reorder the data base prior to exercising the model.

6.1 FORMAT

The data base tape is a seven track, BCD tape, 800 bpi and blocked 84 characters per record. The format is illustrated in Figure 6-1.

Equipments in the data base are ordered by: (1) subsystems, (2) configuration within each subsystem, and (3) equipment types within each configuration [sized equipment(s) first, selected equipment(s) second]. Within equipment types, the equipment is ordered according to the prime technical performance parameter. (This ordering may be changed by the PRESORT routine discussed in Paragraph 6.2.) A list of the data base equipment in the order determined by these considerations is given below:

a. Stabilization and Control

1. Despin mechanical and electronics assembly
2. Valve driver assembly
3. Sun sensor with electronics
4. Nutation damper
5. Gimbal electronics assembly
6. Control timing assembly
7. Biaxial drive assembly
8. Nonscanning earth sensor
9. Sun sensor with electronics
10. Control electronics assembly
11. Rate gyro assembly
12. Horizon sensor
13. Reaction wheel
14. Power converter
15. Attitude reference electronics
16. Valve driver assembly
17. Rate integrating gyros
18. Horizon sensor (with electronics)
19. Electronics processing assembly
20. Single gimbal control moment gyro
21. Star sensor with electronics
22. Electronic error processor

b. **Auxiliary Propulsion**
1. Cold gas thruster
2. Cold gas isolation valve
3. Cold gas filter
4. Cold gas pressure regulator
5. Cold gas pneumatic tank
6. Cold gas fill and vent valve
7. Cold gas relief valve
8. Monopropellant thruster
9. Monopropellant isolation valve
10. Monopropellant filter
11. Monopropellant spherical tank
12. Monopropellant fill valve
13. Bipropellant thruster
14. Bipropellant isolation valve
15. Bipropellant filter
16. Bipropellant tank
17. Bipropellant fill valve

c. **Data Processing and Instrumentation**
1. General purpose processor
2. Special purpose processor (digital telemetry unit)

d. **Communications**
1. Base band assembly unit
2. Antenna
3. Transmitters
4. Receiver
5. Signal conditioner
6. Diplexer
7. Converters (transmitter and receiver)

e. **Electrical Power**
1. Shunt regulator
2. Battery cells
3. Battery charger
4. Discharge regulator
5. Shunt regulator
6. Battery charger
7. Central control unit
8. Series load regulator
9. Battery charger
10. Solar power distributor
11. Power distributor
12. Power control unit

6.2 PRESORT

A small program exists to sort the data base prior to submitting a run for obtaining preliminary spacecraft designs. It will sort the data base according to weight, cost, or reliability. A single digit in Column 1 of a card (to be read on unit 5) determines the sort-variable: 1 = weight (row 23), 2 or 3 = cost (row 46 and 47 or 48), 4 = reliability (row 42). Input tape is expected on unit 8. Output tape is unit 9. Either disk or tape is acceptable for both input and output. Output should be input to the main run. If this presort capability is not used, the order of the data base is determined by technical performance as discussed in Paragraph 6.1.
| Identification | | Technical Characteristics | | Performance | | CDPI Inputs |
|----------------|-----------------|-------------------------|-------------------|-------------------|-------------------|
| 1              | ID              | E5.0                    | T. P. 1           | Ave Pow           | No. Pow Cmd       |
| 2              | CO              | A2                      | 2                 | Max Pow           | No. Other Cmd     |
| 3              | Type            | 3A6                     | 3                 | Min Pow           | Time Tags         |
| 4              |                 |                         | 4                 |                   |                   |
| 5              |                 |                         | 5                 |                   |                   |
| 6              |                 |                         |                   |                   |                   |
| 7              |                 |                         |                   |                   |                   |
| 8              |                 |                         |                   |                   |                   |
| 9              |                 |                         |                   |                   |                   |
| 10             |                 |                         |                   |                   |                   |
| 11             |                 |                         |                   |                   |                   |
| 12             |                 |                         |                   |                   |                   |
| 13             |                 |                         |                   |                   |                   |
| 14             |                 |                         |                   |                   |                   |
| 15             |                 |                         |                   |                   |                   |
| 16             |                 |                         |                   |                   |                   |
| 17             |                 |                         |                   |                   |                   |
| 18             |                 |                         |                   |                   |                   |
| 19             |                 |                         |                   |                   |                   |
| 20             |                 |                         |                   |                   |                   |
| 21             |                 |                         |                   |                   |                   |
| 22             |                 |                         |                   |                   |                   |
| 23             |                 |                         |                   |                   |                   |
| 24             |                 |                         |                   |                   |                   |
| 25             |                 |                         |                   |                   |                   |
| 26             |                 |                         |                   |                   |                   |
| 27             |                 |                         |                   |                   |                   |
| 28             |                 |                         |                   |                   |                   |
| 29             |                 |                         |                   |                   |                   |
| 30             |                 |                         |                   |                   |                   |
| 31             |                 |                         |                   |                   |                   |
| 32             |                 |                         |                   |                   |                   |
| 33             |                 |                         |                   |                   |                   |
| 34             |                 |                         |                   |                   |                   |

Figure 6-1. Data Base Format (7 Cards/Equipment)
<table>
<thead>
<tr>
<th></th>
<th>Samp Rate</th>
<th>Granularity</th>
<th>Safety</th>
<th>Fail Mod</th>
<th>λ or μ</th>
<th>σ</th>
<th>Cost</th>
<th>D.E. Cst</th>
<th>T.E. Cst</th>
<th>Unit Prod</th>
<th>Ref Quant</th>
<th>Factor</th>
<th>Schedule</th>
<th>Devel Const</th>
<th>Devel Var</th>
<th>Qual Const</th>
<th>Qual Var</th>
<th>State-Art</th>
</tr>
</thead>
<tbody>
<tr>
<td>35</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>8E10.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>36</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Card 5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>37</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>38</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>39</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>40</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>41</td>
<td></td>
<td></td>
<td></td>
<td>Fail Mod</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>42</td>
<td></td>
<td></td>
<td></td>
<td>λ or μ</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>43</td>
<td></td>
<td></td>
<td></td>
<td>σ</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>44</td>
<td></td>
<td></td>
<td></td>
<td>q</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>45</td>
<td></td>
<td></td>
<td></td>
<td>Max Redund</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>46</td>
<td></td>
<td></td>
<td></td>
<td>D.E. Cst</td>
<td></td>
<td></td>
<td></td>
<td>8E10.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>47</td>
<td></td>
<td></td>
<td></td>
<td>T.E. Cst</td>
<td></td>
<td></td>
<td></td>
<td>Card 6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>48</td>
<td></td>
<td></td>
<td></td>
<td>Unit Prod</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>49</td>
<td></td>
<td></td>
<td></td>
<td>Ref Quant</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>50</td>
<td></td>
<td></td>
<td></td>
<td>Factor</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>51</td>
<td></td>
<td></td>
<td></td>
<td>Devel Const</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>52</td>
<td></td>
<td></td>
<td></td>
<td>Devel Var</td>
<td></td>
<td></td>
<td></td>
<td>5E10.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>53</td>
<td></td>
<td></td>
<td></td>
<td>Qual Const</td>
<td></td>
<td></td>
<td></td>
<td>Card 7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>54</td>
<td></td>
<td></td>
<td></td>
<td>Qual Var</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>55</td>
<td></td>
<td></td>
<td></td>
<td>State-Art</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 6-1. Data Base Format (7 Cards/Equipment) (Continued)
7. RESTRICTIONS AND/OR LIMITATIONS

The following tables detail both restrictions and limitations of the model. The first type of restriction is that of incompatibility between subsystem configurations and user requirements (Table 7-1). The second type of restriction is that of incompatibility between subsystem configurations (Tables 7-2 through 7-8).
Table 7-1. Stabilization and Control Configuration Selection

<table>
<thead>
<tr>
<th>Requirements</th>
<th>Dual Spin</th>
<th>Yaw Spin</th>
<th>Three-Axis Mass Expulsion</th>
<th>ME with CMGs</th>
<th>ME and Momentum Wheel</th>
</tr>
</thead>
<tbody>
<tr>
<td>Orientation</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Inertial</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Earth pointing</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Sun pointing</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Maneuverability requirements</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Vehicle slewing</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pointing accuracy</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>35-170 mrad (2-10 deg)</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>3.5-35 mrad (0.2-2 deg)</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>0.17-3.5 mrad (0.01-0.2 deg)</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>&lt; 0.17 mrad (&gt; 0.01 deg)</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Rate accuracy</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.7-17 mrad/sec (0.1-1.0 deg/sec)</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>0.17-1.7 mrad/sec (0.01-0.1 deg/sec)</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>&lt; 0.17 mrad/sec (0.01 deg/sec)</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
</tbody>
</table>

Legend: Yes - Configuration can be used  
No - Configuration cannot be used
<table>
<thead>
<tr>
<th>Stabilization and Control Subsystem Configurations</th>
<th>Data Processing Subsystem</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>General Purpose Processors</td>
<td>Special Purpose Processors</td>
</tr>
<tr>
<td>Dual Spin</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Yaw Spin</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Three-Axis Mass Expulsion</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Mass Expulsion with Control Moment Gyros.</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Mass Expulsion with Pitch Momentum Wheel</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Legend:
- Yes - Compatible
- No - Incompatible
Table 7-3. Auxiliary Propulsion Configuration Selection

<table>
<thead>
<tr>
<th>Input Requirements</th>
<th>Cold Gas</th>
<th>Monopropellant</th>
<th>Bipropellant</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Thrust</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt; 224 newtons (&lt; 50 lb)</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>224-4450 newtons (50-1000 lb)</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>&gt; 4450 newtons (&gt; 1000 lb)</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>Total Impulse</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt; 4.4 x 10^4 newton-sec (&lt; 10^4 lb-sec)</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>4.4 x 10^4-2.2 x 10^5 newton-sec (10^4-5 x 10^4 lb-sec)</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>2.2 x 10^5-8.9 x 10^5 newton-sec (5 x 10^4-2 x 10^5 lb-sec)</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>&gt; 8.9 x 10^5 newton-sec (2 x 10^5 lb-sec)</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Legend:
- Yes - Acceptable
- No - Unacceptable
<table>
<thead>
<tr>
<th>Communication Configuration</th>
<th>General Purpose Processor</th>
<th>Special Purpose Processors</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uplink, plus downlink</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Unified link, common antenna</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Unified link, separate antennas</td>
<td>Yes (1 Data Rate Computed)</td>
<td>Yes</td>
</tr>
<tr>
<td>Unified link, common antenna plus downlink</td>
<td>Yes (2 Data Rates Computed)</td>
<td>No</td>
</tr>
<tr>
<td>Unified link, separate antennas plus downlink</td>
<td>Yes (2 Data Rates Computed)</td>
<td>No</td>
</tr>
</tbody>
</table>

Legend:
- **Yes** - Compatible
- **No** - Incompatible
Table 7-5. Communication Configuration Selection

<table>
<thead>
<tr>
<th>Configurations</th>
<th>Ranging Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uplink plus downlink</td>
<td>No</td>
</tr>
<tr>
<td>Unified link, common antenna</td>
<td>Yes</td>
</tr>
<tr>
<td>Unified link, separate antennas</td>
<td>Yes</td>
</tr>
<tr>
<td>Unified link, common antenna plus downlink</td>
<td>Yes</td>
</tr>
<tr>
<td>Unified link, separate antennas plus downlink</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Legend:
Yes - Acceptable
No - Unacceptable
Table 7-6. Electrical Power Configuration Compatibility

<table>
<thead>
<tr>
<th>Configuration</th>
<th>Vehicle Orientation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Spinning</td>
</tr>
<tr>
<td>Solar Arrays</td>
<td></td>
</tr>
<tr>
<td>Body Mounted</td>
<td>Yes</td>
</tr>
<tr>
<td>Oriented Paddles</td>
<td>No</td>
</tr>
</tbody>
</table>

Legend:
- Yes - Compatible
- No - Incompatible

Table 7-7. Vehicle Shape Compatibility

<table>
<thead>
<tr>
<th>SANDC Configuration</th>
<th>Cylinder</th>
<th>Sphere</th>
<th>Box</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spinning</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>3-Axis</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Legend:
- Yes - Compatible
- No - Incompatible
Table 7-8. Structural Configuration Compatibility

<table>
<thead>
<tr>
<th>Structural Configuration</th>
<th>Vehicle Shape</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Cylinder</td>
</tr>
<tr>
<td>Monocoque</td>
<td>Yes</td>
</tr>
<tr>
<td>Semi-Monocóque</td>
<td>Yes</td>
</tr>
<tr>
<td>Truss</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Legend:
- Yes - Compatible
- No - Incompatible
8. SAMPLE TEST CASE

Paragraph 8.1 discusses the input variables to the model. Paragraph 8.2 discusses values that were used in the sample test case. Paragraph 8.3 contains the results of the sample test case.

8.1 USER INPUT VARIABLE LIST

Inputs to the model are listed in Table 8-1. NAMELIST names are shown in parenthesis. All NAMELIST blocks must be in the order given. If the user wishes to use the default parameters, the variables need not be entered. However, NAMELIST control input must exist for each NAMELIST section. For example:

Title Card (80 columns)
$ REQUIR
  
  
$ END
$ DESIRE
  
  
$ END
$ OPTION
  
  
$ END

8.2 INPUT VARIABLES FOR TEST CASE

Figure 8-1 lists the variables which were used for the sample test case. Only those variables that are changed from the default values need to be entered.
### Table 8-1: User Input List

#### Required Input Data

<table>
<thead>
<tr>
<th>FORTRAN Name</th>
<th>Default Value</th>
<th>Units</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MICRO</td>
<td>0</td>
<td></td>
<td>Set to 0 for macro; set to 1, 2, 3, 4 or 5 for micro. If 0, program operates in macro mode. If 1, 2, 3, 4, or 5, program performs micro search for SANDC, AUXPRO, DPI, COMM, or EP subsystems, respectively. For micro search on a specific subsystem, user must restrict all other subsystems to one configuration each.</td>
</tr>
<tr>
<td>IPRINT</td>
<td>1</td>
<td></td>
<td>Set to 1 for system level printout. Set to 2 for system and subsystem level printouts. Set to 3 for system, subsystem, and assembly level printouts. Zero allows no printout.</td>
</tr>
<tr>
<td>T</td>
<td>24.0</td>
<td>mo</td>
<td>Mission lifetime</td>
</tr>
<tr>
<td>APOGEE</td>
<td>500.0</td>
<td>nmi</td>
<td>Orbit apogee</td>
</tr>
<tr>
<td>PERIGE</td>
<td>500.0</td>
<td>nmi</td>
<td>Orbit perigee</td>
</tr>
<tr>
<td>SPEC1 *</td>
<td>18.0</td>
<td>mo</td>
<td>System mean mission duration requirement</td>
</tr>
<tr>
<td>SPEC6 *</td>
<td>0.6</td>
<td></td>
<td>System reliability requirement at end of mission life</td>
</tr>
<tr>
<td>NQV</td>
<td>1</td>
<td></td>
<td>Number of qualification vehicles</td>
</tr>
<tr>
<td>NFV</td>
<td>4</td>
<td></td>
<td>Number of flight vehicles</td>
</tr>
<tr>
<td>EQM1WT</td>
<td>435.0</td>
<td>lb</td>
<td>Mission equipment weight (must be zeroed out if there is no mission equipment one)</td>
</tr>
<tr>
<td>EQM2WT</td>
<td>435.0</td>
<td>lb</td>
<td>Mission equipment weight (must be zeroed out if there is no mission equipment two)</td>
</tr>
<tr>
<td>EPME</td>
<td>0.200</td>
<td>watts</td>
<td>Mission equipment power requirement</td>
</tr>
</tbody>
</table>

*Either SPEC 1 or SPEC 6 can be omitted if the other is given. If SPEC 1 ≤ 0.1, the requirement is ignored, thereby reducing the execution time of the program. If SPEC 6 ≤ 0.00001, this requirement is ignored; however, the program execution time is not reduced.*
### Table 8-1. User Input List (Continued)

#### Desirable Input Data

<table>
<thead>
<tr>
<th>FORTRAN Name</th>
<th>Default Value</th>
<th>Units</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IAGNCY</td>
<td>1</td>
<td></td>
<td>1 = USAF, 2 - NASA</td>
</tr>
<tr>
<td>IMETYP</td>
<td>2</td>
<td></td>
<td>Mission equipment type (1 means Communications, 2 means Earth Observation, 3 means Lunar, 4 means Planetary)</td>
</tr>
<tr>
<td>ISATOR</td>
<td>1</td>
<td></td>
<td>1 earth oriented, 2 sun oriented, 3 inertially oriented</td>
</tr>
<tr>
<td>PHIRX</td>
<td>0.75 deg</td>
<td></td>
<td>Required attitude accuracy about roll, pitch, and yaw axes</td>
</tr>
<tr>
<td>PHIRY</td>
<td>0.75 deg</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PHIRZ</td>
<td>0.75 deg</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NMSEQ</td>
<td>1</td>
<td></td>
<td>Number of mission equipment command and telemetry data arrays in ARRAYN (maximum of 3)</td>
</tr>
<tr>
<td>ARRAYN (1,-)</td>
<td>(0., 0., 0.)</td>
<td></td>
<td>Mission data for up to three (3) equipments *</td>
</tr>
<tr>
<td>ARRAYN (2,-)</td>
<td>(0., 0., 0.)</td>
<td></td>
<td>Power switching command</td>
</tr>
<tr>
<td>ARRAYN (3,-)</td>
<td>(0., 0., 0.)</td>
<td></td>
<td>Other commands</td>
</tr>
<tr>
<td>ARRAYN (4,-)</td>
<td>(100., 0., 0.)</td>
<td></td>
<td>Time tagged commands</td>
</tr>
<tr>
<td>ARRAYN (5,-)</td>
<td>(100., 0., 0.)</td>
<td></td>
<td>High rate telemetry **</td>
</tr>
<tr>
<td>ARRAYN (6,-)</td>
<td>(5000., 0., 0.)</td>
<td>sec^{-1}</td>
<td>Number of analog points</td>
</tr>
<tr>
<td>ARRAYN (7,-)</td>
<td>(8., 0., 0.)</td>
<td>bits</td>
<td>Sample rate</td>
</tr>
<tr>
<td>ARRAYN (8,-)</td>
<td>(280., 0., 0.)</td>
<td></td>
<td>Word length</td>
</tr>
<tr>
<td>ARRAYN (9,-)</td>
<td>(280., 0., 0.)</td>
<td></td>
<td>Low rate telemetry **</td>
</tr>
<tr>
<td>ARRAYN (10,-)</td>
<td>(1., 0., 0.)</td>
<td>sec^{-1}</td>
<td>Number of digital points</td>
</tr>
<tr>
<td>ARRAYN (11,-)</td>
<td>(8., 0., 0.)</td>
<td>bits</td>
<td>Sample rate</td>
</tr>
<tr>
<td>OPSMS</td>
<td>0. ops/sec</td>
<td></td>
<td>Word length</td>
</tr>
<tr>
<td>MB12SH</td>
<td>1</td>
<td></td>
<td>Number of mission operations</td>
</tr>
<tr>
<td>ERM1XL</td>
<td>(Calculated)</td>
<td>in.</td>
<td>No. 1 mission equipment bay length ***</td>
</tr>
<tr>
<td>ERM1YL</td>
<td>(Calculated)</td>
<td>in.</td>
<td></td>
</tr>
<tr>
<td>ERM1ZL</td>
<td>(Calculated)</td>
<td>in.</td>
<td></td>
</tr>
<tr>
<td>ERM2XL</td>
<td>(Calculated)</td>
<td>in.</td>
<td>No. 2 mission equipment bay width ****</td>
</tr>
<tr>
<td>ERM2YL</td>
<td>(Calculated)</td>
<td>in.</td>
<td></td>
</tr>
<tr>
<td>ERM2ZL</td>
<td>(Calculated)</td>
<td>in.</td>
<td>No. 2 mission equipment bay height ****</td>
</tr>
</tbody>
</table>

* Representative values shown for ARRAYN apply to the separate downlink configurations. Designs not using a separate downlink for the mission equipment should specify substantially smaller input values than those values suggested since the mission equipment data is combined with the housekeeping data for transmission purposes.

** For separate downlink designs, nonzero high rate and low rate telemetry data must be specified for at least one mission equipment. Designs not using a separate downlink for the mission equipment can have ARRAYN zeroed out.

*** Must be zeroed out if not used.
<table>
<thead>
<tr>
<th>FORTRAN Name</th>
<th>Default Value</th>
<th>Units</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EM1YCG</td>
<td>0.0</td>
<td>in.</td>
<td>Mission equipment CGs relative to equipment bay interface</td>
</tr>
<tr>
<td>EM1ZCG</td>
<td>0.0</td>
<td>in.</td>
<td></td>
</tr>
<tr>
<td>EM2YCG</td>
<td>0.0</td>
<td>in.</td>
<td></td>
</tr>
<tr>
<td>EM2ZCG</td>
<td>0.0</td>
<td>in.</td>
<td></td>
</tr>
<tr>
<td>NUMEEQ</td>
<td>0</td>
<td></td>
<td>Number of external equipments (Maximum of 9)</td>
</tr>
<tr>
<td>EEQWT(1)</td>
<td>0.0</td>
<td>lb</td>
<td>External equipment weights</td>
</tr>
<tr>
<td>EEQWT(2)</td>
<td>0.0</td>
<td>lb</td>
<td></td>
</tr>
<tr>
<td>EEQWT(3)</td>
<td>0.0</td>
<td>lb</td>
<td></td>
</tr>
<tr>
<td>EEQWT(4)</td>
<td>0.0</td>
<td>lb</td>
<td></td>
</tr>
<tr>
<td>EEQWT(5)</td>
<td>0.0</td>
<td>lb</td>
<td></td>
</tr>
<tr>
<td>EEQWT(6)</td>
<td>0.0</td>
<td>lb</td>
<td></td>
</tr>
<tr>
<td>EEQWT(7)</td>
<td>0.0</td>
<td>lb</td>
<td></td>
</tr>
<tr>
<td>EEQWT(8)</td>
<td>0.0</td>
<td>lb</td>
<td></td>
</tr>
<tr>
<td>EEQWT(9)</td>
<td>0.0</td>
<td>lb</td>
<td></td>
</tr>
<tr>
<td>EEQVL(1)</td>
<td>0.0</td>
<td>ft³</td>
<td>External equipment volumes</td>
</tr>
<tr>
<td>EEQVL(2)</td>
<td>0.0</td>
<td>ft³</td>
<td></td>
</tr>
<tr>
<td>EEQVL(3)</td>
<td>0.0</td>
<td>ft³</td>
<td></td>
</tr>
<tr>
<td>EEQVL(4)</td>
<td>0.0</td>
<td>ft³</td>
<td></td>
</tr>
<tr>
<td>EEQVL(5)</td>
<td>0.0</td>
<td>ft³</td>
<td></td>
</tr>
<tr>
<td>EEQVL(6)</td>
<td>0.0</td>
<td>ft³</td>
<td></td>
</tr>
<tr>
<td>EEQVL(7)</td>
<td>0.0</td>
<td>ft³</td>
<td></td>
</tr>
<tr>
<td>EEQVL(8)</td>
<td>0.0</td>
<td>ft³</td>
<td></td>
</tr>
<tr>
<td>EEQVL(9)</td>
<td>0.0</td>
<td>ft³</td>
<td></td>
</tr>
<tr>
<td>CGEEX(1)</td>
<td>2.0</td>
<td></td>
<td>Location of external equipment (1 means front, 2 means center, 3 means aft end along axis of symmetry)</td>
</tr>
<tr>
<td>CGEEX(2)</td>
<td>2.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CGEEX(3)</td>
<td>2.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CGEEX(4)</td>
<td>2.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CGEEX(5)</td>
<td>2.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CGEEX(6)</td>
<td>2.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CGEEX(7)</td>
<td>2.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CGEEX(8)</td>
<td>2.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CGEEX(9)</td>
<td>2.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EELOC(1)</td>
<td>3.0</td>
<td></td>
<td>Location of external equipment (1 means right, 2 means left, 3 means top, 4 means bottom-looking along the axis of symmetry from the aft end)</td>
</tr>
<tr>
<td>EELOC(2)</td>
<td>3.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EELOC(3)</td>
<td>3.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EELOC(4)</td>
<td>3.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EELOC(5)</td>
<td>3.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EELOC(6)</td>
<td>3.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EELOC(7)</td>
<td>3.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EELOC(8)</td>
<td>3.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EELOC(9)</td>
<td>3.0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Table 8-1. User Input List (Continued)

#### Desirable Input Data

<table>
<thead>
<tr>
<th>FORTRAN Name</th>
<th>Default Value</th>
<th>Units</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RELME</td>
<td>1.0</td>
<td></td>
<td>Mission equipment reliability at end of mission life</td>
</tr>
<tr>
<td>XMER</td>
<td>0.</td>
<td>$</td>
<td>Mission equipment DDT&amp;E cost</td>
</tr>
<tr>
<td>XMEU</td>
<td>0.</td>
<td>$</td>
<td>Mission equipment average unit cost</td>
</tr>
<tr>
<td>PI</td>
<td>1.0</td>
<td></td>
<td>Price index factor</td>
</tr>
<tr>
<td>SKDME(1,-)</td>
<td>(0., 0., 0.)</td>
<td>$1000</td>
<td>Schedule data for up to three mission equipment: Design engineering cost</td>
</tr>
<tr>
<td>SKDME(2,-)</td>
<td>(0., 0., 0.)</td>
<td>$1000</td>
<td>Test and evaluation cost</td>
</tr>
<tr>
<td>SKDME(3,-)</td>
<td>(0., 0., 0.)</td>
<td>mo</td>
<td>Development lead time constant</td>
</tr>
<tr>
<td>SKDME(4,-)</td>
<td>(0., 0., 0.)</td>
<td>mo</td>
<td>Development lead time variable</td>
</tr>
<tr>
<td>SKDME(5,-)</td>
<td>(0., 0., 0.)</td>
<td>mo</td>
<td>Qualification lead time constant</td>
</tr>
<tr>
<td>SKDME(6,-)</td>
<td>(0., 0., 0.)</td>
<td>mo</td>
<td>Qualification lead time variable</td>
</tr>
<tr>
<td>SKDME(7,-)</td>
<td>(0., 0., 0.)</td>
<td></td>
<td>State-of-art factor</td>
</tr>
</tbody>
</table>
Table 8-1. User Input List (Continued)

Optional Input Data

<table>
<thead>
<tr>
<th>FORTRAN Name</th>
<th>Default Value</th>
<th>Units</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IDEBUG</td>
<td>0</td>
<td></td>
<td>Input value of 1 causes cost and reliability debugging information to be printed out.</td>
</tr>
<tr>
<td>ISTRT1</td>
<td>1</td>
<td></td>
<td>First of all allowable five configurations to be designed for the Stabilization and Control subsystem. ISTRTI and IENDINGI effectively limit the number of configurations whose designs will be attempted. (Must be equal for micro search on another subsystem).</td>
</tr>
<tr>
<td>IENDING1</td>
<td>5</td>
<td></td>
<td>Last of the allowable five configurations to be designed for the Stabilization and Control subsystem.</td>
</tr>
<tr>
<td>ISTRT2</td>
<td>1</td>
<td></td>
<td>As above for Auxiliary Propulsion</td>
</tr>
<tr>
<td>IEND2</td>
<td>3</td>
<td></td>
<td>As above for Data Processing and Instrumentation</td>
</tr>
<tr>
<td>ISTRT3</td>
<td>1</td>
<td></td>
<td>As above for Communications</td>
</tr>
<tr>
<td>IEND3</td>
<td>2</td>
<td></td>
<td>As above for Electrical Power</td>
</tr>
<tr>
<td>ISTRT4</td>
<td>1</td>
<td></td>
<td>As above for Vehicle Sizing</td>
</tr>
<tr>
<td>IEND4</td>
<td>5</td>
<td></td>
<td>As above for Reliability</td>
</tr>
<tr>
<td>ISTRT5</td>
<td>1</td>
<td></td>
<td>As above for Reliability</td>
</tr>
<tr>
<td>IEND5</td>
<td>6</td>
<td></td>
<td>As above for Reliability</td>
</tr>
<tr>
<td>ISTRT6</td>
<td>1</td>
<td></td>
<td>As above for Electrical Power</td>
</tr>
<tr>
<td>IEND6</td>
<td>3</td>
<td></td>
<td>As above for Vehicle Sizing</td>
</tr>
<tr>
<td>ISTRTR</td>
<td>0</td>
<td></td>
<td>As above for Reliability</td>
</tr>
<tr>
<td>IENDR</td>
<td>1</td>
<td></td>
<td>As above for Reliability</td>
</tr>
<tr>
<td>ORBINC</td>
<td>(Calculated)</td>
<td>deg</td>
<td>Orbit inclination</td>
</tr>
<tr>
<td>DPHI</td>
<td>.25</td>
<td>deg</td>
<td>Main engine alignment to thrust axis</td>
</tr>
<tr>
<td>FE</td>
<td>4.1</td>
<td>lb</td>
<td>Translational thrust (must be non-zero)</td>
</tr>
<tr>
<td>TSMALL</td>
<td>100</td>
<td>sec</td>
<td>Main engine burn time (AV and stationkeeping)</td>
</tr>
<tr>
<td>XNU</td>
<td>3</td>
<td></td>
<td>Control system efficiency</td>
</tr>
<tr>
<td>PDOT0</td>
<td>1</td>
<td>deg/sec</td>
<td>Maximum initial rate</td>
</tr>
<tr>
<td>PDOTX</td>
<td>1</td>
<td>deg/sec</td>
<td>Maximum maneuver rates</td>
</tr>
<tr>
<td>PDOTY</td>
<td>1</td>
<td>deg/sec</td>
<td>Maximum maneuver rates</td>
</tr>
<tr>
<td>PDOTZ</td>
<td>1</td>
<td></td>
<td>Maximum maneuver rates</td>
</tr>
<tr>
<td>XN</td>
<td>1</td>
<td></td>
<td>Number of maneuvers about roll, pitch, and yaw axes</td>
</tr>
<tr>
<td>YN</td>
<td>1</td>
<td></td>
<td>Number of maneuvers about roll, pitch, and yaw axes</td>
</tr>
<tr>
<td>ZN</td>
<td>1</td>
<td></td>
<td>Number of maneuvers about roll, pitch, and yaw axes</td>
</tr>
<tr>
<td>FORTRAN Name</td>
<td>Default Value</td>
<td>Units</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>---------------</td>
<td>-------</td>
<td>-------------</td>
</tr>
<tr>
<td>PDOTRX</td>
<td>.012</td>
<td>deg/sec</td>
<td>Required system rate accuracy</td>
</tr>
<tr>
<td>PDOTRY</td>
<td>.012</td>
<td>deg/sec</td>
<td></td>
</tr>
<tr>
<td>PDOTRZ</td>
<td>.012</td>
<td>deg/sec</td>
<td></td>
</tr>
<tr>
<td>TMIN</td>
<td>10.0</td>
<td>sec</td>
<td>Minimum payload scan period (applies only to yaw spin configuration)</td>
</tr>
<tr>
<td>OMEGR</td>
<td>(Calculated)</td>
<td>rpm</td>
<td>Spin rate of rotor (applies only to dual spin configuration)</td>
</tr>
<tr>
<td>XNN</td>
<td>(Calculated)</td>
<td>days</td>
<td>Time between spin axis corrections (applies only to dual spin configuration)</td>
</tr>
<tr>
<td>K</td>
<td>1</td>
<td></td>
<td>0 if errors for spin axis relative to nadir; 1 if errors for payload relative to nadir (applies only to dual spin configuration)</td>
</tr>
<tr>
<td>MANV</td>
<td>1</td>
<td></td>
<td>4 means vehicle skewing and prevents design of the dual spin configuration; otherwise, no effect</td>
</tr>
<tr>
<td>EPI</td>
<td>.0001</td>
<td>deg/sec</td>
<td>Maximum programmed pitch over rate (applies only to three-axis mass expulsion configuration)</td>
</tr>
<tr>
<td>AX</td>
<td>.05</td>
<td></td>
<td>Misalignment errors in mounting inertia measurement units (applies only to three-axis mass expulsion configuration)</td>
</tr>
<tr>
<td>AY</td>
<td>.05</td>
<td>deg</td>
<td></td>
</tr>
<tr>
<td>AZ</td>
<td>.05</td>
<td>deg</td>
<td></td>
</tr>
<tr>
<td>EA</td>
<td>0.10</td>
<td>deg</td>
<td>Antenna misalignment (applies only to pitch momentum bias configuration)</td>
</tr>
<tr>
<td>EAANT</td>
<td>.1</td>
<td>rad</td>
<td>Antenna elevation (applies only to pitch momentum bias configuration and should be set to less than one radian)</td>
</tr>
<tr>
<td>ALPHA</td>
<td>12.0</td>
<td>deg</td>
<td>Thruster offset in roll-yaw plane (applies only to pitch momentum bias configuration)</td>
</tr>
<tr>
<td>TL</td>
<td>1.0</td>
<td>day</td>
<td>Time between unloading wheel momentum (applies only to CMG configuration)</td>
</tr>
<tr>
<td>TACCEL</td>
<td>(Calculated)</td>
<td>sec</td>
<td>Acceleration time for maneuvering (applies only to CMG configuration)</td>
</tr>
<tr>
<td>XNNN</td>
<td>4.0</td>
<td></td>
<td>Number of single gimbaled gyros (applies only to CMG configuration)</td>
</tr>
<tr>
<td>THOLD</td>
<td>100000.</td>
<td>min</td>
<td>Time vehicle in inertial hold (applies only to CMG configuration)</td>
</tr>
<tr>
<td>PDOTAV</td>
<td>0.01</td>
<td>deg/sec</td>
<td>Average body rate for low orbit when high accuracy is required (applies only to CMG configuration)</td>
</tr>
</tbody>
</table>
Table 8-1. User Input List (Continued)

**Optional Input Data**

<table>
<thead>
<tr>
<th>FORTRAN Name</th>
<th>Default Value</th>
<th>Units</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PDOTST</td>
<td>0.0667</td>
<td>deg/sec</td>
<td>Maximum rate at which star information is obtained (applies only to CMG configuration)</td>
</tr>
<tr>
<td>THETMX</td>
<td>180.</td>
<td>deg</td>
<td>Maximum maneuver angle (applies only to CMG configuration)</td>
</tr>
<tr>
<td>PHIFOV</td>
<td>40.0</td>
<td>deg</td>
<td>Maximum range of attitude freedom required to track specific stars (applies only to CMG configuration)</td>
</tr>
<tr>
<td>BTRMX</td>
<td>1.024x10^6</td>
<td>bit/sec</td>
<td>Maximum bit rate</td>
</tr>
<tr>
<td>SCSFL</td>
<td>0.</td>
<td></td>
<td>Special command synchronization flag (0 means no synchronization required, 1 means synchronization required)</td>
</tr>
<tr>
<td>TPRFL</td>
<td>0.</td>
<td></td>
<td>Telemetry processing flag (0 means telemetry processed separately, 1 means otherwise)</td>
</tr>
<tr>
<td>IOPTCM</td>
<td>0</td>
<td></td>
<td>Ranging requirement (0 or 1 for no or yes)</td>
</tr>
<tr>
<td>NET</td>
<td>1</td>
<td></td>
<td>Communications link (0 or 1 for USB or SCLS)</td>
</tr>
<tr>
<td>FREQ(2)</td>
<td>2250., 2250 MHz</td>
<td></td>
<td>Frequency of downlink transmitters (second number refers to separate downlink)</td>
</tr>
<tr>
<td>NADR</td>
<td>0</td>
<td></td>
<td>Nadir coverage flag (0 = no, 1 = yes)</td>
</tr>
<tr>
<td>FREQR</td>
<td>1800.</td>
<td>MHz</td>
<td>Receiver frequency</td>
</tr>
<tr>
<td>COMAT</td>
<td>1000.</td>
<td>baud</td>
<td>Receiver command rate</td>
</tr>
<tr>
<td>BWIDTH(2)</td>
<td>(Calculated)</td>
<td>Hz</td>
<td>Bandwidth for transmitter (default values are flags that cause bandwidth to be computed as a function of bit rate)</td>
</tr>
<tr>
<td>OPTEMP</td>
<td>15.</td>
<td>°C</td>
<td>Battery temperature</td>
</tr>
<tr>
<td>EQPF</td>
<td>5.</td>
<td></td>
<td>Volume sizing factor</td>
</tr>
<tr>
<td>ISBOFG</td>
<td>0</td>
<td></td>
<td>Solar array boom drive requirement (0 means not required, 1 means required)</td>
</tr>
<tr>
<td>XCGSAI</td>
<td>1.</td>
<td></td>
<td>Location of solar paddles (1 means front, 2 means center, 3 means aft end)</td>
</tr>
</tbody>
</table>

* The computer program does not currently possess the ability to design an USB communications link.

** In Subroutine COMM
**Table 8-1. User Input List (Continued)**

**Optional Input Data**

<table>
<thead>
<tr>
<th>FORTRAN Name</th>
<th>Default Value</th>
<th>Units</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>XCGSA3</td>
<td>1.</td>
<td></td>
<td>Location of body mounted solar array (1 means front, 2 means center, 3 means aft end)</td>
</tr>
<tr>
<td>DIAMAX</td>
<td>120.</td>
<td>in.</td>
<td>Maximum satellite diameter</td>
</tr>
<tr>
<td>RFIXED</td>
<td>1.0</td>
<td></td>
<td>Initial system reliability</td>
</tr>
<tr>
<td>KEOPT</td>
<td>1</td>
<td></td>
<td>Expense option indicator (1 means additional redundancy is penalized on the basis of weight; otherwise expense is cost)</td>
</tr>
<tr>
<td>SLBMX</td>
<td>50000.0</td>
<td>lb</td>
<td>Maximum system weight</td>
</tr>
<tr>
<td>ISPT</td>
<td>0</td>
<td></td>
<td>Single point failure requirements option (0 = not in effect, otherwise in effect)</td>
</tr>
<tr>
<td>ISUB</td>
<td>0</td>
<td></td>
<td>Subsystem requirements option ( = at least one subsystem has a reliability spec, otherwise no reliability specs on subsystem)</td>
</tr>
<tr>
<td>SPEC(1)*</td>
<td>(Calculated)</td>
<td></td>
<td>Reliability requirement for the Stabilization and Control subsystem</td>
</tr>
<tr>
<td>SPEC(2)*</td>
<td>(Calculated)</td>
<td></td>
<td>Reliability requirement for the Auxiliary Propulsion subsystem</td>
</tr>
<tr>
<td>SPEC(3)*</td>
<td>(Calculated)</td>
<td></td>
<td>Reliability requirement for the Data Processing subsystem</td>
</tr>
<tr>
<td>SPEC(4)*</td>
<td>(Calculated)</td>
<td></td>
<td>Reliability requirement for the Communication subsystem</td>
</tr>
<tr>
<td>SPEC(5)*</td>
<td>(Calculated)</td>
<td></td>
<td>Reliability requirement for the Electrical Power Subsystem</td>
</tr>
<tr>
<td>CA</td>
<td>10.</td>
<td>g</td>
<td>Axial launch acceleration</td>
</tr>
<tr>
<td>CE</td>
<td>5.</td>
<td>g</td>
<td>Lateral launch acceleration</td>
</tr>
<tr>
<td>FEEPCT</td>
<td>0.07</td>
<td></td>
<td>Contractor's fee percentage</td>
</tr>
</tbody>
</table>

*If SPEC(K) \( \leq 0.00001\), the requirement for the K^{th} subsystem is ignored.
$REQUIR
APOGEE = 0.193229E+05,
EPME = 0.3E+03,
EQM1WT = 0.161E+03,
EQM2WT = 0.0,
IFINT = 3,
MICRO = 0,
NFV = 6,
NQV = 1,
PERIGE = 0.193229E+05,
SPEC1 = 0.36E+02,
SPEC6 = 0.236E+00,
T = 0.6E+02,
$END

Figure 8-1. Input Variables for Test Case
Figure 8-1. Input Variables for Test Case (Continued)
SKDME = 0.0, 0.0, 0.5, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
XMER = 0.323E+089,
XMEU = 0.334E+07,
$END$

Figure 8-1. Input Variables for Test Case (Continued)
$OPTION
ALPHA  =  0.12E+02,
AX     =  0.5E-01,
AY     =  0.5E-01,
AZ     =  0.5E-01,
BTRMX  =  0.1024E+07,
BWIDTH =  -0.1E+11, -0.1E+11,
CA     =  0 .1E+02,
CE     =  0.5E+01,
COMRAT =  0.1E+04,
DIATMX =  0.18E+03,
DPhi   =  0.25E+00,
EA     =  0.1E+00,
EANT   =  0.1E+00,
EP1    =  0.1E-03,
EPQF   =  0.1E+00,
FE     =  0.35E+02,
FEEPCT =  0.7E-01,
FREQ   =  0.225E+04,  0.225E+04,
FREQR  =  0.18E+04,
IEND1  =  1,
IDBUG  =  1,
IEN02  =  2,
IEN03  =  2,
IEND4  =  2,
IEND5  =  2,
IEN06  =  1,
IENDR  =  0,
ICPTCH =  1,

Figure 8-1. Input Variables for Test Case (Continued)
Figure 8-1. Input Variables for Test Case (Continued)
Figure 8-1. Input Variables for Test Case (Continued)
8.3 SAMPLE TEST CASE RESULTS

The test case corresponds to the DSCS-II satellite. This satellite provides for expanded communications service for worldwide military installations and the National Command Authority. The satellite is drum-shaped. Two dish antennas on top of the spacecraft are deployed in orbit to provide narrow beam coverage. Conical horn earth coverage antennas are mounted on top of the spacecraft. An omnidirectional command and telemetry antenna is deployed beneath the main body. Communications equipment is mounted on a mechanically despun platform. Other subsystems are housed in the main body of the spacecraft. The test case results are presented in Figure 8-2.
DEFINITIONS —

CONFIGURATIONS (NCONF)

STABILIZATION AND CONTROL (NCONF(1))
- NCONF(1)=1 IS DUAL SPIN
- NCONF(1)=2 IS YAW SPIN
- NCONF(1)=3 IS MASS EXPULSION
- NCONF(1)=9 IS MASS EXPULSION W/ CMG-S

DATA PROCESSING AND INSTRUMENTATION (NCONF(3))
- NCONF(3)=1 IS GENERAL PURPOSE PROCESSOR
- NCONF(3)=2 IS SPECIAL PURPOSE PROCESSOR

ELECTRICAL POWER (NCONF(5))
- NCONF(5)=1 IS SHUNT REGULATION - PADDLE MTD.
- NCONF(5)=2 IS SHUNT REGULATION - BODY MTD.
- NCONF(5)=3 IS SHNT + DISCH.REG - PADDLE MTD.
- NCONF(5)=4 IS SHNT + DISCH.REG - BODY MTD.
- NCONF(5)=5 IS SERIES LOAD REG. - PADDLE MTD.
- NCONF(5)=6 IS SERIES LOAD REG. - BODY MTD.

MESSAGES (IERR)

STABILIZATION AND CONTROL
- IERR = 0 MEANS NC MESSAGES
- IERR = 1 MEANS MAX ALLOWABLE SYS. ERROR UNSAT.
- IERR = 1X MEANS MAX RATE ERROR TOO SMALL
- IERR = 1XX MEANS 3-AXIS WHEELS ACCEPTABLE

DATA PROCESSING AND INSTRUMENTATION
- IERR = 0 MEANS NO MESSAGES
- IERR = 1 MEANS MAX BUS REQUIRED
- IERR = 10 WORD LENGTH GREATER THAN 256
- IERR = 100 BIT RATE IS TOO LARGE
- IERR = 10000 END OF DATA BASE SENSED

VEHICLE SIZING
- IERR = 0 MEANS NO MESSAGES
- IERR = 1 MEANS BODY MOUNTED SOLAR ARRAY LENGTH EXCEEDS EQUIPMENT BAY LENGTH
- IERR = 10000 END OF DATA BASE SENSED

AUXILIARY PROPULSION (NCONF(2))
- NCONF(2)=1 IS COLD GAS
- NCONF(2)=2 IS MONO PROPELLANT
- NCONF(2)=3 IS BIPROPELLANT

COMMUNICATIONS (NCONF(4))
- NCONF(4)=1 IS SEPARATE UPLINK AND DOWNLINK
- NCONF(4)=2 IS UNIFIED LINK-CO MMON ANTENNAS
- NCONF(4)=3 IS UNIFIED LINK-SEPARATE ANTENNAS
- NCONF(4)=4 IS UNIFIED LINK-COMMON ANT + DOWNLINK
- NCONF(4)=5 IS UNIFIED LINK-SEPARATE ANT + DOWNLINK

RELIABILITY
- REDUNDANCY CONFIGURATION = 0 IS SINGLE SYSTEM
- REDUNDANCY CONFIGURATION = 1 IS DUAL SYSTEM

Figure 8-2. Sample Test Case Results
**STABILIZATION AND CONTROL**

Configuration = Dual Spin
Pointing Accuracy = 0.393000(Deg.)

**AUXILIARY PROPULSION**

Configuration = Monopropellant
Total Impulse = 20276(LB-SEC)

**DATA PROCESSING AND INSTRUMENTATION**

Configuration = Special Purpose Processor (OTU)
Computer Operations Rate = 0.1IPS

<table>
<thead>
<tr>
<th>COPI TABLE</th>
<th>ENGINEERING DATA</th>
<th>MISSION EQUIPMENT DATA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of commands</td>
<td>256 *</td>
<td>0</td>
</tr>
<tr>
<td>Number of main frame words</td>
<td>64 *</td>
<td>0</td>
</tr>
<tr>
<td>Main frame sample rate</td>
<td>1 *</td>
<td>0</td>
</tr>
<tr>
<td>Main frame word length</td>
<td>6 *</td>
<td>0</td>
</tr>
<tr>
<td>Number of subframes</td>
<td>8 *</td>
<td>0</td>
</tr>
<tr>
<td>Subframe length</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>Number of words per subframe</td>
<td>32</td>
<td></td>
</tr>
</tbody>
</table>

**COMMUNICATIONS**

Configuration = Unified Link-Common Antennas
Primary Downlink Data Rate = 1.000(Kbps)
Separate Downlink Data Rate = 0.000(Kbps)

**ELECTRICAL POWER**

Configuration = Shunt - Body Mounted Solar Array
End of Life Power Requirement = 443.40(Watts)
Total Solar Array Area = 183.07(FT**2)
Minimum Installed Battery Capacity = 22.26(AMP-HR)

**VEHICLE SIZING**

Configuration = Cylinder
Vehicle Weight = 1199.01(LBS) Launch Weight = 1199.01(LBS)
Equipment Bay Dimensions Length = 77.70(IN), Height = 108.00(IN), Width = 108.00(IN)
Mission Equipment Length = 48.40(IN), Height = 108.20(IN), Width = 108.20(IN)
Total Satellite Length = 126.10(IN)
Moments of Inertia (LB-FT**2) Ixx = 20764.41 Iyy = 2676386.5 Izz = 2676386.5

Figure 8-2. Sample Test Case Results (Continued)
SAFETY CONFIGURATION - SINGLE SYSTEM
MEAN MISSION DURATION 49.5 (MO)
RELIABILITY 0.5
MISSION LIFETIME 69.8 (MO)

COSTS (ALL AMOUNTS ARE IN DOLLARS)

<table>
<thead>
<tr>
<th>COST CATEGORY</th>
<th>DESIGN ENGINEERING</th>
<th>TEST AND EVALUATION</th>
<th>TOOLLING AND TEST EQUIPMENT</th>
<th>QUALITY CONTROL</th>
<th>SYSTEMS ENGINEERING AND INTEGRATION</th>
<th>PROGRAM MANAGEMENT</th>
<th>INVESTMENT (RECURREING)</th>
</tr>
</thead>
<tbody>
<tr>
<td>DOT+E</td>
<td>853,569.4</td>
<td>501,470.4</td>
<td>0.0</td>
<td>630,595.3</td>
<td>465,285.0</td>
<td>172,265.7</td>
<td>25,465,54.9</td>
</tr>
<tr>
<td>UNIT ENGINEERING</td>
<td>2,196,738.0</td>
<td>2,040,000.0</td>
<td>2,040,000.0</td>
<td>2,040,000.0</td>
<td>2,040,000.0</td>
<td>2,040,000.0</td>
<td>2,040,000.0</td>
</tr>
<tr>
<td>UNIT PRODUCTICN</td>
<td>2,040,000.0</td>
<td>2,040,000.0</td>
<td>2,040,000.0</td>
<td>2,040,000.0</td>
<td>2,040,000.0</td>
<td>2,040,000.0</td>
<td>2,040,000.0</td>
</tr>
<tr>
<td>TOOLING AND TEST EQUIP.</td>
<td>2,040,000.0</td>
<td>2,040,000.0</td>
<td>2,040,000.0</td>
<td>2,040,000.0</td>
<td>2,040,000.0</td>
<td>2,040,000.0</td>
<td>2,040,000.0</td>
</tr>
<tr>
<td>QUALITY CONTROL</td>
<td>630,595.3</td>
<td>630,595.3</td>
<td>630,595.3</td>
<td>630,595.3</td>
<td>630,595.3</td>
<td>630,595.3</td>
<td>630,595.3</td>
</tr>
<tr>
<td>PROGRAM MANAGEMENT</td>
<td>172,265.7</td>
<td>172,265.7</td>
<td>172,265.7</td>
<td>172,265.7</td>
<td>172,265.7</td>
<td>172,265.7</td>
<td>172,265.7</td>
</tr>
<tr>
<td>INVESTMENT OPERATIONS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C.O.C.</td>
<td>12,103.4</td>
<td>12,103.4</td>
<td>12,103.4</td>
<td>12,103.4</td>
<td>12,103.4</td>
<td>12,103.4</td>
<td>12,103.4</td>
</tr>
<tr>
<td>SPACECRAFT</td>
<td>2,196,738.0</td>
<td>2,196,738.0</td>
<td>2,196,738.0</td>
<td>2,196,738.0</td>
<td>2,196,738.0</td>
<td>2,196,738.0</td>
<td>2,196,738.0</td>
</tr>
<tr>
<td>MISSION EQUIPMENT</td>
<td>2,040,000.0</td>
<td>2,040,000.0</td>
<td>2,040,000.0</td>
<td>2,040,000.0</td>
<td>2,040,000.0</td>
<td>2,040,000.0</td>
<td>2,040,000.0</td>
</tr>
<tr>
<td>TOTAL PAYLOAD</td>
<td>4,236,738.0</td>
<td>4,236,738.0</td>
<td>4,236,738.0</td>
<td>4,236,738.0</td>
<td>4,236,738.0</td>
<td>4,236,738.0</td>
<td>4,236,738.0</td>
</tr>
<tr>
<td>QUALIFICATION UNITS</td>
<td>102,945.6</td>
<td>102,945.6</td>
<td>102,945.6</td>
<td>102,945.6</td>
<td>102,945.6</td>
<td>102,945.6</td>
<td>102,945.6</td>
</tr>
<tr>
<td>GSE</td>
<td>121,034.0</td>
<td>121,034.0</td>
<td>121,034.0</td>
<td>121,034.0</td>
<td>121,034.0</td>
<td>121,034.0</td>
<td>121,034.0</td>
</tr>
<tr>
<td>LAUNCH SUPPORT</td>
<td>1,944,938.0</td>
<td>1,944,938.0</td>
<td>1,944,938.0</td>
<td>1,944,938.0</td>
<td>1,944,938.0</td>
<td>1,944,938.0</td>
<td>1,944,938.0</td>
</tr>
<tr>
<td>CONTRACTOR FEE</td>
<td>291,877.1</td>
<td>291,877.1</td>
<td>291,877.1</td>
<td>291,877.1</td>
<td>291,877.1</td>
<td>291,877.1</td>
<td>291,877.1</td>
</tr>
<tr>
<td>PROGRAM TOTALS</td>
<td>6,597,485.8</td>
<td>6,597,485.8</td>
<td>6,597,485.8</td>
<td>6,597,485.8</td>
<td>6,597,485.8</td>
<td>6,597,485.8</td>
<td>6,597,485.8</td>
</tr>
<tr>
<td>INVESTMENT OPERATIONS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TOTAL</td>
<td>6,597,485.8</td>
<td>6,597,485.8</td>
<td>6,597,485.8</td>
<td>6,597,485.8</td>
<td>6,597,485.8</td>
<td>6,597,485.8</td>
<td>6,597,485.8</td>
</tr>
</tbody>
</table>

SCHEDULE

| SCHEDULE COMPONENT DEVELOPMENT TIME | 14.4 (MONTHS) |
| COMPONENT QUALIFICATION TIME       | 14.1 (MONTHS) |
| SUBSYSTEM DEVELOPMENT TIME         | 9.2 (MONTHS)  |
| SUBSYSTEM QUALIFICATION TIME       | 8.1 (MONTHS)  |
| SYSTEM DEVELOPMENT AND FLIGHT READINESS TIME | 9.2 (MONTHS) |
| SCHEDULE DURATION (TO LAUNCH)      | 41.6 (MONTHS) |

Figure 8-2. Sample Test Case Results (Continued)
### Stabilization and Control

**Configuration:** Dual Spin

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>151 252 352 451 551 651 751 801 940</td>
<td>276,900.0</td>
<td>16.82900.0</td>
<td>576,900.0</td>
<td>969,741.2</td>
<td>0.7109</td>
</tr>
</tbody>
</table>

**Equipment Quantities:**

<table>
<thead>
<tr>
<th>Weight</th>
<th>Volume</th>
<th>Power Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>148.13(LB)</td>
<td>5.64(FT**3)</td>
<td>57.1(WATT)</td>
</tr>
</tbody>
</table>

**Schedule:**

- Component Development Time: 14.4 (Month)
- Component Qualification Time: 14.1 (Month)
- Subsystem Development Time: 9.2 (Month)
- Subsystem Qualification Time: 8.1 (Month)
- System Development and Flight Readiness Time: 9.2 (Month)

**IERR:** 0

### Auxiliary Propulsion

**Configuration:** Monopropellant

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>807 807 902 1001 459 201 1102 503 701 1201 601</td>
<td>20.0497.6</td>
<td>16.0699.6</td>
<td>56.0099.6</td>
<td>923,322.8</td>
<td>0.7709</td>
</tr>
</tbody>
</table>

**Equipment Quantities:**

<table>
<thead>
<tr>
<th>Weight</th>
<th>Volume</th>
<th>Power Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>206.0(LB)</td>
<td>5.22(FT**3)</td>
<td>0.0(WATT)</td>
</tr>
</tbody>
</table>

**Dry Weight:** 86.82(LBS)

**Expendable Weight:** 121.25(LBS)

**Schedule:**

- Component Development Time: 4.3 (Month)
- Component Qualification Time: 1.6 (Month)
- Subsystem Development Time: 8.9 (Month)
- Subsystem Qualification Time: 7.3 (Month)
- System Development and Flight Readiness Time: 8.9 (Month)

**IERR:** 11

### Data Processing and Instrumentation

**Configuration:** Special Purpose Processor (DPU)

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>201 12</td>
<td>59651.8</td>
<td>772,983.0</td>
<td>0.8750</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Equipment Quantities:**

<table>
<thead>
<tr>
<th>Weight</th>
<th>Volume</th>
<th>Power Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>20.80(LB)</td>
<td>2.90(FT**3)</td>
<td>6.0(WATT)</td>
</tr>
</tbody>
</table>

**Design Eng. Cost:** 926,989.6

**Test & Eval. Cost:** 644,989.6

**Unit Prod. Cost:** 288,395.2

**Unit Eng. Cost:** 372,322.0

**Reliability:** 0.7709

**Schedule:**

- Component Development Time: 8.9 (Month)
- Component Qualification Time: 2.0 (Month)
- Subsystem Development Time: 2.4 (Month)
- Subsystem Qualification Time: 7.1 (Month)
- System Development and Flight Readiness Time: 2.4 (Month)

**IERR:** 1

---

Figure 8-2. Sample Test Case Results (Continued)
**COMMUNICATIONS CONFIGURATION** -- UNIFIED LINK-GCMON ANTENNAS

<table>
<thead>
<tr>
<th>Equipment Code Identifier</th>
<th>101</th>
<th>201</th>
<th>301</th>
<th>401</th>
<th>502</th>
<th>601</th>
<th>701</th>
<th>702</th>
</tr>
</thead>
<tbody>
<tr>
<td>Equipment Quantities</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Weight (lb)</td>
<td>29.45</td>
<td>7.21</td>
<td>7.82</td>
<td>7.82</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Volume (ft$^3$)</td>
<td>502</td>
<td>601</td>
<td>701</td>
<td>702</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Power Requirement (W)</td>
<td>72.3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DES ENG. COST</td>
<td>181000.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TEST &amp; EVAL. COST</td>
<td>416000.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UNIT PROF. COST</td>
<td>201250.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UNIT ENG. COST</td>
<td>113450.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RELIABILITY</td>
<td>9728</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**SCHEDULE**
- Component Development Time: 16.7 (MONTH)
- Component Qualification Time: 10.0 (MONTH)
- Subsystem Development Time: 2.8 (MONTH)
- Subsystem Qualification Time: 7.2 (MONTH)
- System Development and Flight Readiness Time: 2.8 (MONTH)

**MISSION EQUIPMENT**

<table>
<thead>
<tr>
<th>Equipment Code Identifier</th>
<th>101</th>
<th>201</th>
<th>301</th>
<th>401</th>
</tr>
</thead>
<tbody>
<tr>
<td>Equipment Quantities</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Weight (lb)</td>
<td>181.00</td>
<td>257.41</td>
<td>257.41</td>
<td></td>
</tr>
<tr>
<td>Volume (ft$^3$)</td>
<td>3230000.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Power Requirement (W)</td>
<td>300.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DES ENG. COST</td>
<td>3230000.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TEST &amp; EVAL. COST</td>
<td>334000.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UNIT PROF. COST</td>
<td>334000.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UNIT ENG. COST</td>
<td>334000.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RELIABILITY</td>
<td>47000</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**MISSILE CONFIGURATION** -- SHUNT - BODY MOUNTED SOLAR ARRAY

<table>
<thead>
<tr>
<th>Equipment Code Identifier</th>
<th>111</th>
<th>211</th>
<th>359</th>
<th>1201</th>
</tr>
</thead>
<tbody>
<tr>
<td>Equipment Quantities</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Weight (lb)</td>
<td>183.20</td>
<td>10.93</td>
<td>10.93</td>
<td></td>
</tr>
<tr>
<td>MOSHNESS WEIGHT</td>
<td>372.20</td>
<td>10.93</td>
<td>10.93</td>
<td></td>
</tr>
<tr>
<td>LARGE ARRAY WEIGHT</td>
<td>127.50</td>
<td>10.93</td>
<td>10.93</td>
<td></td>
</tr>
<tr>
<td>DES ENG. COST</td>
<td>2593888.3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TEST &amp; EVAL. COST</td>
<td>1549254.4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UNIT PROF. COST</td>
<td>664986.3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UNIT ENG. COST</td>
<td>665812.5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RELIABILITY</td>
<td>9728</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**SCHEDULE**
- Component Development Time: 16.3 (MONTH)
- Component Qualification Time: 2.9 (MONTH)
- Subsystem Development Time: 4.7 (MONTH)
- Subsystem Qualification Time: 7.2 (MONTH)
- System Development and Flight Readiness Time: 4.7 (MONTH)

**MISSION EQUIPMENT**

<table>
<thead>
<tr>
<th>Equipment Code Identifier</th>
<th>111</th>
<th>211</th>
<th>359</th>
<th>1201</th>
</tr>
</thead>
<tbody>
<tr>
<td>Equipment Quantities</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Weight (lb)</td>
<td>181.00</td>
<td>257.41</td>
<td>257.41</td>
<td></td>
</tr>
<tr>
<td>Volume (ft$^3$)</td>
<td>3230000.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Power Requirement (W)</td>
<td>300.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DES ENG. COST</td>
<td>3230000.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TEST &amp; EVAL. COST</td>
<td>334000.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UNIT PROF. COST</td>
<td>334000.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UNIT ENG. COST</td>
<td>334000.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RELIABILITY</td>
<td>47000</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**SCHEDULE**
- Component Development Time: 0.0 (MONTH)
- Component Qualification Time: 0.0 (MONTH)
- Subsystem Development Time: 0.0 (MONTH)
- Subsystem Qualification Time: 0.0 (MONTH)
- System Development and Flight Readiness Time: 0.0 (MONTH)

Figure 8-2. Sample Test Case Results (Continued)
Thermal Control

Radiator Area 4.2 (FT^2), Battery Radiator Area 1.1 (FT^2)

Heater Power 268.7 (BTU/HR), Battery Heater Power 110.0 (BTU/HR)

Heat Pipe 14768.4 (WATT-IN), Total Heat Conductance H.P. 1560.4 (WATT-IN)

Thermal Control Weight 14.7 (LBS)

DES. ENG. COST $350,906.9, TEST & EVAL. COST $266,219.0
UNIT ENG. COST $763,211.1

Structures

Skin Thickness .007 (IN)
STRINGER NO., THICKNESS, HT. 17, .013 (IN), .370 (IN)
FRAME NO., THICKNESS, HT. 5, .075 (IN), .825 (IN)
ENDCOVER THICKNESS FORWARD 10.21 (IN), CENTER 0.000 (IN), AFT 2.21 (IN)
EQUIPMENT BAY STRUCTURE WT. 175.2 (LBS)
SOLAR ARRAY BOOM AND DRIVE WT. 0.0 (LBS)
ADAPTER WEIGHT 14.2 (LBS)

DES. ENG. COST $1,108,027.7, TEST & EVAL. COST $4,228,074.4
UNIT ENG. COST $1,979,195.6, UNIT ENG. COST $2,468,560.8

Figure 8-2. Sample Test Case Results (Continued)
### OSCS-II ASSEMBLY DESCRIPTIONS - DESIGN NUMBER 1

#### STABILIZATION AND CONTROL

<table>
<thead>
<tr>
<th>IDENT</th>
<th>TYPE</th>
<th>NO.</th>
<th>UNIT UNIT</th>
<th>WEIGHT</th>
<th>VOLUME</th>
<th>POWER</th>
<th>D.E. COST</th>
<th>T.E. COST</th>
<th>VEHICLE</th>
<th>VEHICLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>151</td>
<td>DESPIN MECH+ELECT.</td>
<td>2</td>
<td>30.2</td>
<td>9.9</td>
<td>8.2</td>
<td>146850</td>
<td>862280</td>
<td>27728.5</td>
<td>544594</td>
<td></td>
</tr>
<tr>
<td>352</td>
<td>VALVE DRIVE ASSY.</td>
<td>2</td>
<td>4.9</td>
<td>2.4</td>
<td>1.8</td>
<td>160000</td>
<td>159000</td>
<td>193900</td>
<td>397745</td>
<td></td>
</tr>
<tr>
<td>451</td>
<td>ROLL DAMPER</td>
<td>1</td>
<td>2.4</td>
<td>2.4</td>
<td>1.8</td>
<td>290000</td>
<td>173000</td>
<td>171000</td>
<td>107966</td>
<td></td>
</tr>
<tr>
<td>551</td>
<td>Gimbal Elect. ASSY.</td>
<td>1</td>
<td>5.3</td>
<td>3.9</td>
<td>3.9</td>
<td>250000</td>
<td>175000</td>
<td>175000</td>
<td>107966</td>
<td></td>
</tr>
<tr>
<td>651</td>
<td>CONTROL TIPING ASSY.</td>
<td>1</td>
<td>7.4</td>
<td>4.9</td>
<td>4.9</td>
<td>650000</td>
<td>140000</td>
<td>191525</td>
<td>241467</td>
<td></td>
</tr>
<tr>
<td>751</td>
<td>BIXIAL DRIVE ASSY.</td>
<td>2</td>
<td>14.2</td>
<td>4.3</td>
<td>4.3</td>
<td>650000</td>
<td>800000</td>
<td>665000</td>
<td>241467</td>
<td></td>
</tr>
<tr>
<td>851</td>
<td>COMPRESSOR EARTH SCAN</td>
<td>2</td>
<td>2.7</td>
<td>1.6</td>
<td>1.6</td>
<td>105000</td>
<td>105000</td>
<td>105000</td>
<td>105000</td>
<td></td>
</tr>
<tr>
<td>1401</td>
<td>SAC PWR CONVERTER</td>
<td>2</td>
<td>5.1</td>
<td>1.6</td>
<td>1.6</td>
<td>105000</td>
<td>105000</td>
<td>105000</td>
<td>105000</td>
<td></td>
</tr>
</tbody>
</table>

#### AUXILIARY PROPULSION

<table>
<thead>
<tr>
<th>IDENT</th>
<th>TYPE</th>
<th>NO.</th>
<th>UNIT UNIT</th>
<th>WEIGHT</th>
<th>VOLUME</th>
<th>POWER</th>
<th>D.E. COST</th>
<th>T.E. COST</th>
<th>VEHICLE</th>
<th>VEHICLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>807</td>
<td>THRUSTER TRW MRE-3</td>
<td>6</td>
<td>3.5</td>
<td>1.9</td>
<td>1.9</td>
<td>171800</td>
<td>27000</td>
<td>6000</td>
<td>13351</td>
<td></td>
</tr>
<tr>
<td>902</td>
<td>MONO ISO</td>
<td>2</td>
<td>3.4</td>
<td>1.3</td>
<td>1.3</td>
<td>100000</td>
<td>100000</td>
<td>100000</td>
<td>37462</td>
<td></td>
</tr>
<tr>
<td>1001</td>
<td>MONO FIL3181406100</td>
<td>2</td>
<td>4.6</td>
<td>1.4</td>
<td>1.4</td>
<td>511750</td>
<td>212500</td>
<td>80372</td>
<td>169817</td>
<td></td>
</tr>
<tr>
<td>201</td>
<td>ISO VALVE 272494</td>
<td>2</td>
<td>4.1</td>
<td>1.2</td>
<td>1.2</td>
<td>511750</td>
<td>212500</td>
<td>80372</td>
<td>169817</td>
<td></td>
</tr>
<tr>
<td>1102</td>
<td>MONO SPHER 80156-1</td>
<td>2</td>
<td>4.1</td>
<td>1.2</td>
<td>1.2</td>
<td>511750</td>
<td>212500</td>
<td>80372</td>
<td>169817</td>
<td></td>
</tr>
<tr>
<td>503</td>
<td>PNEUM TANK</td>
<td>1</td>
<td>4.0</td>
<td>1.8</td>
<td>1.8</td>
<td>511750</td>
<td>212500</td>
<td>80372</td>
<td>169817</td>
<td></td>
</tr>
<tr>
<td>701</td>
<td>RELIEF VALVE</td>
<td>1</td>
<td>4.0</td>
<td>1.8</td>
<td>1.8</td>
<td>511750</td>
<td>212500</td>
<td>80372</td>
<td>169817</td>
<td></td>
</tr>
<tr>
<td>1201</td>
<td>MONO FILL</td>
<td>2</td>
<td>4.0</td>
<td>1.8</td>
<td>1.8</td>
<td>511750</td>
<td>212500</td>
<td>80372</td>
<td>169817</td>
<td></td>
</tr>
<tr>
<td>501</td>
<td>FIL+VENT 34659-1</td>
<td>2</td>
<td>2.0</td>
<td>0.0</td>
<td>0.0</td>
<td>511750</td>
<td>212500</td>
<td>80372</td>
<td>169817</td>
<td></td>
</tr>
</tbody>
</table>

#### DATA PROCESSING AND INSTRUMENTATION

<table>
<thead>
<tr>
<th>IDENT</th>
<th>TYPE</th>
<th>NO.</th>
<th>UNIT UNIT</th>
<th>WEIGHT</th>
<th>VOLUME</th>
<th>POWER</th>
<th>D.E. COST</th>
<th>T.E. COST</th>
<th>VEHICLE</th>
<th>VEHICLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>201</td>
<td>SPEC.PURP.PRO.DTU.</td>
<td>2</td>
<td>1.4</td>
<td>3.0</td>
<td>3.0</td>
<td>210000</td>
<td>97000</td>
<td>59851.8</td>
<td>77893</td>
<td></td>
</tr>
</tbody>
</table>

#### COMMUNICATIONS

<table>
<thead>
<tr>
<th>IDENT</th>
<th>TYPE</th>
<th>NO.</th>
<th>UNIT UNIT</th>
<th>WEIGHT</th>
<th>VOLUME</th>
<th>POWER</th>
<th>D.E. COST</th>
<th>T.E. COST</th>
<th>VEHICLE</th>
<th>VEHICLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>161</td>
<td>BASEBAND ASM. UNIT</td>
<td>2</td>
<td>0.9</td>
<td>0.5</td>
<td>0.5</td>
<td>250000</td>
<td>9000</td>
<td>27360</td>
<td>10765</td>
<td></td>
</tr>
<tr>
<td>261</td>
<td>TRANSMITTER</td>
<td>2</td>
<td>5.6</td>
<td>5.6</td>
<td>5.6</td>
<td>100000</td>
<td>150000</td>
<td>150000</td>
<td>10765</td>
<td></td>
</tr>
<tr>
<td>401</td>
<td>RECEIVER</td>
<td>2</td>
<td>4.0</td>
<td>4.0</td>
<td>4.0</td>
<td>650000</td>
<td>27000</td>
<td>65000</td>
<td>10765</td>
<td></td>
</tr>
<tr>
<td>502</td>
<td>COMMS SIG CONN.</td>
<td>2</td>
<td>4.0</td>
<td>4.0</td>
<td>4.0</td>
<td>650000</td>
<td>27000</td>
<td>65000</td>
<td>10765</td>
<td></td>
</tr>
<tr>
<td>601</td>
<td>CPM CONV.</td>
<td>1</td>
<td>1.8</td>
<td>1.8</td>
<td>1.8</td>
<td>100000</td>
<td>6000</td>
<td>66500</td>
<td>22423</td>
<td></td>
</tr>
<tr>
<td>701</td>
<td>CONVERTER (TRANS.)</td>
<td>2</td>
<td>1.8</td>
<td>1.8</td>
<td>1.8</td>
<td>100000</td>
<td>6000</td>
<td>66500</td>
<td>22423</td>
<td></td>
</tr>
<tr>
<td>702</td>
<td>CONVERTER RECEIVER</td>
<td>2</td>
<td>1.8</td>
<td>1.8</td>
<td>1.8</td>
<td>100000</td>
<td>6000</td>
<td>66500</td>
<td>22423</td>
<td></td>
</tr>
</tbody>
</table>

Figure 8-2. Sample Test Case Results (Continued)
### ELECTRICAL POWER

<table>
<thead>
<tr>
<th>IDENT TYPE</th>
<th>NO.</th>
<th>WEIGHT</th>
<th>VOLUME</th>
<th>POWER</th>
<th>D.E. COST</th>
<th>T.E. COST</th>
<th>PROD. COST</th>
<th>ENG COST</th>
</tr>
</thead>
<tbody>
<tr>
<td>101 SHUNT REGULATOR</td>
<td>10</td>
<td>4.2</td>
<td>1.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>211 BATTERY CELL</td>
<td>2</td>
<td>61.6</td>
<td>2.0</td>
<td>0.0</td>
<td>421000.0</td>
<td>143000.0</td>
<td>89264.7</td>
<td>156157.0</td>
</tr>
<tr>
<td>359 BATTERY CHARGER</td>
<td>2</td>
<td>3.3</td>
<td>1.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>1201 POWER CONTROL UNIT</td>
<td>1</td>
<td>10.5</td>
<td>2.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
</tbody>
</table>

### EQUIPMENTS USING COST ESTIMATING RELATIONSHIPS

<table>
<thead>
<tr>
<th>NAME</th>
<th>WEIGHT</th>
<th>D.E. COST</th>
<th>T.E. COST</th>
<th>PROD. COST</th>
<th>ENG COST</th>
</tr>
</thead>
<tbody>
<tr>
<td>SOLAR ARRAY</td>
<td>127.5</td>
<td>703177.7</td>
<td>394111.2</td>
<td>409768.6</td>
<td>167676.7</td>
</tr>
<tr>
<td>HARNESS</td>
<td>97.6</td>
<td>518594.2</td>
<td>437560.9</td>
<td>173363.6</td>
<td>136286.7</td>
</tr>
<tr>
<td>THERMAL CONTROL</td>
<td>14.7</td>
<td>350490.8</td>
<td>266219.0</td>
<td>409590.1</td>
<td>78552.1</td>
</tr>
<tr>
<td>POWER CONVERTERS</td>
<td>17.2</td>
<td>262774.9</td>
<td>280501.1</td>
<td>115211.0</td>
<td>108254.8</td>
</tr>
<tr>
<td>PROPULSION FEED SYS.</td>
<td>279784.9</td>
<td>159789.6</td>
<td>161338.2</td>
<td>62737.4</td>
<td>62737.4</td>
</tr>
<tr>
<td>STRUCTURE</td>
<td>175.2</td>
<td>110082.7</td>
<td>422887.4</td>
<td>197919.5</td>
<td>246838.0</td>
</tr>
<tr>
<td>POWER CONTROL UNITS</td>
<td>60.0</td>
<td>568321.7</td>
<td>300076.2</td>
<td>73375.7</td>
<td>137437.3</td>
</tr>
</tbody>
</table>

Figure 8-2. Sample Test Case Results (Continued)
9. SOURCE CODE LISTING

The following is a listing of the Systems Cost/Performance Computer Program.
COMMON /DBCOM/CATAB(55,100),IDE(30)

COMMON /CHOOSE/ 
  COST(5,60),  DPIA(11,60),  ICHOSE(60), 022575 50
  DPHOSE(60),  REL (6,60),  SKQ(7,60), 022575 51

COMMON /PRTCON/ 
  ACCRCY, AM,  BQ,  BS, 022575 52
  BF,  BS, 022575 53
  BF,  BS, 022575 54
  BF,  BS, 022575 55

C DIMENSION NCONF(6), NCONF(5), IERR(7), IPIC(3), IPIC(2), IPIC(1)

C THE NAMELIST INPUTS ARE BROKEN INTO THREE CATEGORIES. THAT
C IS CATEGORIES OF REQUIRED, DESIRED, AND OPTIONAL PARAMETERS.
C THE FOLLOWING IS A LIST OF THE INPUTS TO THE MODEL:

NAME  REP. VALUE  UNITS  DESCRIPTION
---  ------  ------  ---------------
** REQUIRED INPUT DATA  **  121374
  APOGEE  500.  NMI  ORBIT APOGEE  121374
  EQM1HT  435.  LB  MISS. EQ.M. 1  121374
  EQM2HT  435.  LB  MISS. EQ.M. 2  121374
  IPRINT  1.  ---  1=SYS., 2=ASSEMBLY  121374
  MICRO  0.  ---  MICRO (S/S) FLAG  121374
  NVF  4.  ---  NO. FLITE VEHICLES  121374
  NQV  1.  ---  NO. QUAL. VEHICLES  121374
  PERICE  500.  NMI  ORBIT PERICLE  121374
  SPEC6  0.6  ---  REL. AT EOL  121374
  SPEC1  18.  MO  SYS. MMD REQ.  121374
  T  24.  MO  MISSION LIFETIME  121374
** DESIRABLE INPUT DATA  **  121374
  ARRAY  2.  ---  MISSION DATA FOR UP TO 3 EQ.  121374
  CGEX  2.  ---  LOC. OF EXT. EQ. (FT, CENT, AFT)  121374
  EELOC  3.  ---  LOC. OF EXT. EQ. (RT, LFT, TOP, BOT)  121374
  EEWL +1  ---  EXT. EQ. VOLUMES  121374
  EEQWT  0.  ---  EXT. EQ. WEIGHTS  121374
  EM1YCG  0.  IN  M.E. 1 Y-CG  121374
  EM2YCG  0.  IN  M.E. 2 Y-CG  121374
  EM1ZCG  0.  IN  M.E. 1 Z-CG  121374
  EM2ZCG  0.  IN  M.E. 2 Z-CG  121374
  EQIXL  40.  IN  M.E. 1 LENGTH  121374
  EQIYL  40.  IN  M.E. 1 WIDTH  121374

DIMENSION ITITL(3)

THE FOLLOWING IS A LIST OF THE INPUTS TO THE MODEL:

NAME  REP. VALUE  UNITS  DESCRIPTION
---  ------  ------  ---------------
** REQUIRED INPUT DATA  **  121374
  APOGEE  500.  NMI  ORBIT APOGEE  121374
  EQM1HT  435.  LB  MISS. EQ.M. 1  121374
  EQM2HT  435.  LB  MISS. EQ.M. 2  121374
  IPRINT  1.  ---  1=SYS., 2=ASSEMBLY  121374
  MICRO  0.  ---  MICRO (S/S) FLAG  121374
  NVF  4.  ---  NO. FLITE VEHICLES  121374
  NQV  1.  ---  NO. QUAL. VEHICLES  121374
  PERICE  500.  NMI  ORBIT PERICLE  121374
  SPEC6  0.6  ---  REL. AT EOL  121374
  SPEC1  18.  MO  SYS. MMD REQ.  121374
  T  24.  MO  MISSION LIFETIME  121374
** DESIRABLE INPUT DATA  **  121374
  ARRAY  2.  ---  MISSION DATA FOR UP TO 3 EQ.  121374
  CGEX  2.  ---  LOC. OF EXT. EQ. (FT, CENT, AFT)  121374
  EELOC  3.  ---  LOC. OF EXT. EQ. (RT, LFT, TOP, BOT)  121374
  EEWL +1  ---  EXT. EQ. VOLUMES  121374
  EEQWT  0.  ---  EXT. EQ. WEIGHTS  121374
  EM1YCG  0.  IN  M.E. 1 Y-CG  121374
  EM2YCG  0.  IN  M.E. 2 Y-CG  121374
  EM1ZCG  0.  IN  M.E. 1 Z-CG  121374
  EM2ZCG  0.  IN  M.E. 2 Z-CG  121374
  EQIXL  40.  IN  M.E. 1 LENGTH  121374
  EQIYL  40.  IN  M.E. 1 WIDTH  121374
| **C** | **EQM1ZL** | 4.0 | IN | M.E. 1 HEIGHT | **121374** | 34 |
| **C** | **EQM2ZL** | 4.0 | IN | M.E. 2 LENGTH | **121374** | 39 |
| **C** | **EQM3ZL** | 4.0 | IN | M.E. 3 WIDTH | **121374** | 36 |
| **C** | **IAgency** | 1 | --- | AGENCY TYPE 1=USAF, 2=NASA | **121374** | 36 |
| **C** | **IMET** | 3 | --- | M.E. TYPE, 1=CON, 2=EO, 3=LUN, 4=PL | **121374** | 39 |
| **C** | **ISTATION** | 1 | --- | ORIENT. 1=EO, 2=SO, 3=IO | **121374** | 40 |
| **C** | **MB128** | 1 | --- | M.E. EX SHAPE 1=CYL, 2=BOX | **121374** | 42 |
| **C** | **NIMSEQ** | 1 | --- | NO. M.E. I+G DATA ARRAYS | **121374** | 43 |
| **C** | **NUMEQ** | 0 | --- | NO. EXTR. EQ. | **121374** | 44 |
| **C** | **OPR** | 0.5 | --- | OPERATIONAL INPUT | **121374** | 46 |
| **C** | **PSH** | 0.9 | --- | OPR/SEC | **121374** | 47 |
| **C** | **PHIRX** | 0.75 | DEG | ROLL ACCURACY | **121374** | 45 |
| **C** | **PHIRY** | 0.75 | DEG | PITCH ACCURACY | **121374** | 46 |
| **C** | **PHIRZ** | 0.75 | DEG | YAW ACCURACY | **121374** | 47 |
| **C** | **PI** | 1.0 | --- | PRICE INDEX FACTOR | **121374** | 48 |
| **C** | **REL** | 1.0 | --- | M.E. REL. ELEVATION | **121374** | 49 |
| **C** | **SKME** | 0.0 | --- | M.E. SKED DAILY | **121374** | 50 |
| **C** | **XME** | 0.0 | --- | M.E. DTD+E COST | **121374** | 51 |
| **C** | **XMEU** | 0.0 | --- | M.E. AVG UNIT COST | **121374** | 52 |

**OPTIONAL INPUT DATA**

| **C** | **ALPHA** | 12.0 | DEG | THRSTR OFFSET IN ROLL-YAW | **121374** | 53 |
| **C** | **AX** | 0.85 | DEG | MISEALIGNMENT ERRORS IN | **121374** | 54 |
| **C** | **AY** | 0.85 | DEG | MOUNTING INERTIA UNITS | **121374** | 55 |
| **C** | **AZ** | 0.85 | DEG | (3-AXIS MASS EXP. ONLY) | **121374** | 56 |
| **C** | **BTRM** | 1.024 | E+06BIT/SEC | MAXIMUM BIT RATE | **121374** | 57 |
| **C** | **BWID** | 2.0 | HZ | BANDWIDTH FOR XMTTR(S) | **121374** | 58 |
| **C** | **CS** | 5.0 | --- | LAGRA LAUNCH ACCELERATION | **121374** | 59 |
| **C** | **CMT** | 120.0 | BAUD | RECEIVER COMMAND RATE | **121374** | 60 |
| **C** | **DIA** | 120.0 | IN | MAXIMUM SATELLITE DIAMETER | **121374** | 61 |
| **C** | **DPHI** | 1.0 | DEG | M.E. DEG ALIGN TO THRAX | **121374** | 62 |
| **C** | **EAT** | 1.0 | DEG | ANTENNA ELEVATION (PM ONLY) | **121374** | 63 |
| **C** | **EI** | 0.01 | DEG/SEC | MAX PGM PITCH/OVER RATE(3-AXIS) | **121374** | 64 |
| **C** | **EOPF** | 2.0 | --- | VOLUME SIZING FACTOR | **121374** | 65 |
| **C** | **FEEPT** | 4.0 | --- | CONTRACTOR FEE PERCENTAGE | **121374** | 66 |
| **C** | **FREQ** | 2.0 | HZ | FREQ. OF OCMNLK XMTR(S) | **121374** | 67 |
| **C** | **FREQ** | 18000 | MHz | RECEIVER FREQ. | **121374** | 68 |
| **C** | **IF** | 0.0 | --- | 0=DEBUG OFF, 1=DEBUG ON | **121374** | 69 |
| **C** | **TEND** | 0.0 | --- | LAST ALLOWABLE FOR SANDC | **121374** | 70 |
| **C** | **TEND2** | 3.0 | --- | LAST ALLOWABLE FOR AP | **121374** | 71 |
| **C** | **TEND3** | 2.0 | --- | LAST ALLOWABLE FOR DPI | **121374** | 72 |
| **C** | **TEND4** | 6.0 | --- | LAST ALLOWABLE FOR COM | **121374** | 73 |
| **C** | **TEND5** | 6.0 | --- | LAST ALLOWABLE FOR VESIZE | **121374** | 74 |
| **C** | **TENDR** | 1.0 | --- | LAST ALLOWABLE FOR RELY | **121374** | 75 |
| **C** | **TOPT** | 0.0 | --- | RANGING REQUIREMENT =NO,1= YES | **121374** | 76 |
| **C** | **TCS** | 0.0 | --- | SINGLE PT FAIL REQ. =NO,1= YES | **121374** | 77 |
| **C** | **ISTRT** | 1.0 | --- | FIRST ALLOWABLE FOR SANDC | **121374** | 78 |
| **C** | **ISTRT2** | 1.0 | --- | FIRST ALLOWABLE FOR AP | **121374** | 79 |
| **C** | **ISTRT3** | 1.0 | --- | FIRST ALLOWABLE FOR DPI | **121374** | 80 |
| **C** | **ISTRT4** | 1.0 | --- | FIRST ALLOWABLE COM | **121374** | 81 |
| **C** | **ISTRT5** | 1.0 | --- | FIRST ALLOWABLE FOR EP | **121374** | 82 |
| **C** | **ISTRT6** | 1.0 | --- | FIRST ALLOWABLE FOR VESIZE | **121374** | 83 |
**First Allowable for Reliability**

---

**S/S Rel Flag 1: At least 1 S/S**

---

**Axis Relativity (Dual-Spin)**

---

**Expense Opt Ind**

---

**Link**

---

**Manv**

---

**Nadir Coverage Flag**

---

**Net**

---

**Orb Inc**

---

**PDotAV**

---

**PDotx**

---

**PDoty**

---

**PDotz**

---

**PhifoV**

---

**RFixed**

---

**SBSFL**

---

**SLBMX**

---

**Spec(1)**

---

**Spec(2)**

---

**Spec(3)**

---

**Spec(4)**

---

**Spec(5)**

---

**TACCEL**

---

**Thebix**

---

**Thold**

---

**Triang**

---

**Tprfl**

---

**Tsmall**

---

**Xggsa1**

---

**Xggsa3**

---

**Xn**

---

**Xnn**

---

**Xnnn**

---

**Xnu**

---

**Ym**

---

**Zn**

---

"---"
DATA NEQUIP, NACCEP/6*0/
DATA ITST1, IE01, ISTR2, IEN2, ISTR3, IE03, ISTR4, IE04, ISTR5,
* IE05, ISTR6, IE06, ISTR7, IE07, ISTR8, IE08, ISTR9, IE09, ISTR10,
* IEN10, ISTR11, IE011, ISTR12, IE012, ISTR13, IE013, ISTR14, IE014,
* ISTR15, IE015, ISTR16, IE016, ISTR17, IE017, ISTR18, IE018,
* ISTR19, IE019, ISTR20, IE020, ISTR21, IE021, ISTR22, IE022,
* ISTR23, IE023, ISTR24, IE024, ISTR25, IE025, ISTR26, IE026,
* ISTR27, IE027, ISTR28, IE028, ISTR29, IE029, ISTR30, IE030,
* ISTR31, IE031, ISTR32, IE032, ISTR33, IE033, ISTR34, IE034,
* ISTR35, IE035, ISTR36, IE036, ISTR37, IE037, ISTR38, IE038,
* ISTR39, IE039, ISTR40, IE040, ISTR41, IE041, ISTR42, IE042,
* ISTR43, IE043, ISTR44, IE044, ISTR45, IE045, ISTR46, IE046,
* ISTR47, IE047, ISTR48, IE048, ISTR49, IE049, ISTR50, IE050,
* ISTR51, IE051, ISTR52, IE052, ISTR53, IE053, ISTR54, IE054,
* ISTR55, IE055, ISTR56, IE056, ISTR57, IE057, ISTR58, IE058,
* ISTR59, IE059, ISTR60, IE060, ISTR61, IE061, ISTR62, IE062,
* ISTR63, IE063, ISTR64, IE064, ISTR65, IE065, ISTR66, IE066,
* ISTR67, IE067, ISTR68, IE068, ISTR69, IE069, ISTR70, IE070,
* ISTR71, IE071, ISTR72, IE072, ISTR73, IE073, ISTR74, IE074,
* ISTR75, IE075, ISTR76, IE076, ISTR77, IE077, ISTR78, IE078,
* ISTR79, IE079, ISTR80, IE080, ISTR81, IE081, ISTR82, IE082,
* ISTR83, IE083, ISTR84, IE084, ISTR85, IE085, ISTR86, IE086,
* ISTR87, IE087, ISTR88, IE088, ISTR89, IE089, ISTR90, IE090,
* ISTR91, IE091, ISTR92, IE092, ISTR93, IE093, ISTR94, IE094,
* ISTR95, IE095, ISTR96, IE096, ISTR97, IE097, ISTR98, IE098,
* ISTR99, IE099, ISTR100, IE100, ISTR101, IE101, ISTR102, IE102,
* ISTR103, IE103, ISTR104, IE104, ISTR105, IE105, ISTR106, IE106,
* ISTR107, IE107, ISTR108, IE108, ISTR109, IE109, ISTR110, IE110,
* ISTR111, IE111, ISTR112, IE112, ISTR113, IE113, ISTR114, IE114,
* ISTR115, IE115, ISTR116, IE116, ISTR117, IE117, ISTR118, IE118,
* ISTR119, IE119, ISTR120, IE120, ISTR121, IE121, ISTR122, IE122,
* ISTR123, IE123, ISTR124, IE124, ISTR125, IE125, ISTR126, IE126,
* ISTR127, IE127, ISTR128, IE128, ISTR129, IE129, ISTR130, IE130,
* ISTR131, IE131, ISTR132, IE132, ISTR133, IE133, ISTR134, IE134,
* ISTR135, IE135, ISTR136, IE136, ISTR137, IE137, ISTR138, IE138,
* ISTR139, IE139, ISTR140, IE140, ISTR141, IE141, ISTR142, IE142,
* ISTR143, IE143, ISTR144, IE144, ISTR145, IE145, ISTR146, IE146,
* ISTR147, IE147, ISTR148, IE148, ISTR149, IE149, ISTR150, IE150,
* ISTR151, IE151, ISTR152, IE152, ISTR153, IE153, ISTR154, IE154,
* ISTR155, IE155, ISTR156, IE156, ISTR157, IE157, ISTR158, IE158,
* ISTR159, IE159, ISTR160, IE160, ISTR161, IE161, ISTR162, IE162,
* ISTR163, IE163, ISTR164, IE164, ISTR165, IE165, ISTR166, IE166,
* ISTR167, IE167, ISTR168, IE168, ISTR169, IE169, ISTR170, IE170,
* ISTR171, IE171, ISTR172, IE172, ISTR173, IE173, ISTR174, IE174,
* ISTR175, IE175, ISTR176, IE176, ISTR177, IE177, ISTR178, IE178,
* ISTR179, IE179, ISTR180, IE180, ISTR181, IE181, ISTR182, IE182,
* ISTR183, IE183, ISTR184, IE184, ISTR185, IE185, ISTR186, IE186,
* ISTR187, IE187, ISTR188, IE188, ISTR189, IE189, ISTR190, IE190,
* ISTR191, IE191, ISTR192, IE192, ISTR193, IE193, ISTR194, IE194,
* ISTR195, IE195, ISTR196, IE196, ISTR197, IE197, ISTR198, IE198,
* ISTR199, IE199, ISTR200, IE200, ISTR201, IE201, ISTR202, IE202,
* ISTR203, IE203, ISTR204, IE204, ISTR205, IE205, ISTR206, IE206,
CALL CPTIME(TIMEIN)
DO 285 I=1,9
CALL FILTER(NCONF,ICODE)
IF (ICODE .LT. 0) GO TO 14
IPIC1(I)=0
IPIC1(2)=0
IPIC1(3)=0
DO 23 I=1,9
23 IPIC2(I)=0
IPIC3(I)=0
IPIC3(2)=0
DO 24 I=1,9
24 IPIC4(I)=0
DO 20 I=1,5
20 IPIC5(I)=0
11 CALL INITIL(NCONF,IERERR)
IF (IERERR .EQ. 1) GO TO 14
PRINT 9000,NCONF
PRINT 9000,ICHCSE
PRINT 9000,NCHOSE
DO 10 ITR=1,2
REINDD=1
ITER=ITR-1
CALL READDB(IENDDB)
IF (ITER .NE. I .OR. MICRC .EQ. 1) GO TO 91
IPIC1(I)=0
IPIC1(2)=0
IPIC1(3)=0
IPIC2(I)=0
IPIC3(I)=0
DO 91 I=1,19
91 WRITE (6,1999) II
1999 FORMAT (3H 11,E15.4)
NEQUIP(I)=0
DO 101 I=1,ITEST1
IF (ICHOSI(I).LT. 0) GO TO 14
IF (ICHOSI(I).GT. 0) NEQUIP(I)=NEQUIP(I)+1
101 CONTINUE
WRITE (6,1000) WT,VOL
1000 FORMAT (6,E15.4)
NEQUIP=1
CALL SAVE(ICHOSI,NCHOSI,NOWAT,ITEST1,IENDDB)
CALL READDB(IENDDB)
IF (ITER .NE. I .OR. MICRO .EQ. 2) GO TO 92
DO 28 I=1,9
28 IPIC2(I)=0
CALL AUXPKD(IPIC2,IERR(2),ITER,NCONF,ICHOS2,NCHOS2)
NEQUIP(2)=0
0 102 I=1,ITEST2
IF (ICHOS2(I)*.LT. 0 *AND. MICRO*.LT. 2) GO TO 13
IF (ICHOS2(I)*.LT. 0 *AND. MICRO*.EQ. 2) GO TO 14
IF (ICHOS2(I)*.GT. 0) NEQUIP(2)=NEQUIP(2)+1
102 CONTINUE
WEIGHT(2)=WT-HEIGHT(1)
VOLUME(2)=VOL-VOLUME(1)
POWER(2)=PL-POWER(1)
TEMPWT=WT
TEMPVL=VOL
WRITE (6 ±000) WTVOL
CALL SAVE(ICHOS2,NCHOS2,NOWAT,ITEST2,IENDB)
CALL READOB(IENO6)
IF (ITER*NE. 0 *OR. MICRO*EQ. 3) GO TO 93
IPIC3(I)=0
360 IPIC3(2)=0
93 CALL DPI(IPIC3,IERR(3),ITER,NCONF,ICHOS3,NCHOS3,NOWAT)
NEQUIP(3)=0
DO 103 I=1,ITEST3
IF (ICHOS3(I)*.LT. 0 *AND. MICRO*.LT. 3) GO TO 13
IF (ICHOS3(I)*.LT. 0 *AND. MICRO*.EQ. 3) GO TO 14
IF (ICHOS3(I)*.GT. 0) NEQUIP(3)=NEQUIP(3)+1
103 CONTINUE
WEIGHT(3)=WT-TEMPWT
VOLUME(3)=VOL-TEMPVL
POWER(3)=PL-TEMPPL
TEMPWT=WT
TEMPVL=VOL
WRITE (6 ±1000) WT,VOL
CALL SAVE(ICHOS3,NCHOS3,NOWAT,ITEST3,IENDB)
CALL READDB(IENDB)
IF (ITER*NE. 0 *OR. MICRO*EQ. 4) GO TO 94
DO 29 I=1,9
29 IPIC4(I)=0
94 CALL COMM(IPIC4,IERR(4),ITER,NCONF,ICHOS4,NCHOS4)
NEQUIP(4)=0
DO 104 I=1,ITEST4
IF (ICHOS4(I)*.LT. 0 *AND. MICRO*.LT. 4) GO TO 13
IF (ICHOS4(I)*.LT. 0 *AND. MICRO*.EQ. 4) GO TO 14
IF (ICHOS4(I)*.GT. 0) NEQUIP(4)=NEQUIP(4)+1
104 CONTINUE
WEIGHT(4)=WT-TEMPWT
VOLUME(4)=VOL-TEMPVL
POWER(4)=PL-TEMPPL
TEMPWT=WT
TEMPVL=VOL
WRITE (6 ±1000) WT,VOL
CALL SAVE(ICHOS4,NCHOS4,NOWAT,ITEST4,IENDB)
CALL READDB(IENDB)
IF (ITER*NE. 0 *OR. MICRO*.EQ. 5) GO TO 95
CALL EP(IPIC5,IERR5,ITER,NCONF,ICHOS5,NCHOS5)
WRITE (6,6999) PL,PLMIN
FORMAT (9H PL, PLMIN, 2E15.4)
NEQUIP(5)=1
CO 105 I=1,5
IF (ICHOS5(I) .LT. 0 .AND. MICRO .LT. 5) GO TO 13
IF (ICHOS5(I) .LT. 0 .AND. MICRO .EQ. 5) GO TO 14
IF (ICHOS5(I) .GT. 0) NEQUIP(5)=NEQUIP(5)+1
IF (ICHOS5(I) .EQ. 2) IBTLOC=NOWAT-I
CONTINUE
WEIGHT(5)=WT-TEMPWT
VOLUME(5)=VOL-TEMPVOL
POWER(5)=PL-TEMPPL-EPME
CALL SAVE(ICHOS5,NCHOS5,NOWAT,ITEST5,IONDB)
CALL VESIZE(IERR6,NCONF,ICHOS6)
IF (ICHOS6 .LT. 0) GO TO 13
IF (ITER .GT. 0) GO TO 10
CALL RELY(IRTN,IREL,NEQUIP)
PRINT 3000,IRTN
FORMAT (5H IRTN, I10)
IF (IRTN .LT. 0) GO TO 13
IR1=1
IR2=NEQUIP(1)
CO 31 IR=1,IR2
31 NCHOS4(IR)=NCHOS4(IR)
IR=IR2+1
IR2=NEQUIP(2)
CO 32 IR=1,IR2
32 NCHOS4(IR)=NCHOS4(IR)
IR=IR2+1
IR2=NEQUIP(3)
DO 33 IR=1,IR2
33 NCHOS4(IR)=NCHOS4(IR)
IR=IR2+1
IR2=NEQUIP(4)
CO 34 IR=1,IR2
34 NCHOS4(IR)=NCHOS4(IR)
IR=IR2+1
IR2=NEQUIP(5)
DO 35 IR=1,IR2
35 NCHOS5(IR)=NCHOS5(IR)
IR=IR2+1
CONTINUE
CALL STRUCT(NCONF)
CALL THRML(IERR7,NCONF)
NCHOS5(NOWAT)=0
CALL COSTS(NCONF,NEQUIP)
CALL SKE(NEQUIP,NCONF)
NACCEP=NACCEP+1
CALL PRINT(IERR,NEQUIP,NEACCP,NCONF)
IF (MICRO .GT. 0) GO TO 11
PRINT 9000,NCONF
PRINT 9000,(ICHOS1(I),I=1,NCHAT)
PRINT 9000,(NCHOS1(I),I=1,NCHAT)
9000 FORMAT (10I10)
    CALL CPTIME(TIMEOT)
    TIME=TIMEOT-TIMEIN
    PRINT 9999,TIME
9999 FORMAT(1X,26HC,P. TIME FOR THIS CASE = ,F10.3,8H SECONDS)
14 PRINT 9000,NCONF
    PRINT 9000,ICHOSE
    PRINT 9000,NCHOSE
    CONTINUE
    IF (IREL.EQ. IENDR) GO TO 99
    IREL=IENDR
    GO TO 2
99 STOP
END

REGISTER ALLOCATION
1 REGISTERS ASSIGNED OVER THE LOOP BEGINNING AT LINE 266
1 REGISTERS ASSIGNED OVER THE LOOP BEGINNING AT LINE 427
1 REGISTERS ASSIGNED OVER THE LOOP BEGINNING AT LINE 431
1 REGISTERS ASSIGNED OVER THE LOOP BEGINNING AT LINE 435
1 REGISTERS ASSIGNED OVER THE LOOP BEGINNING AT LINE 439
SUBROUTINE PRESET(IERR)

** THIS SUBROUTINE CALCULATES WHAT USED TO BE EITHER CONSTANTS OR
** INPUT VALUES. IT NOW WILL CALCULATE THE VALUES OF THESE
** CONSTANTS FROM OTHER INPUT VALUES WHERE THEY ARE GIVEN ELSE IT
** WILL USE THE OLD CONSTANT VALUES.

COMMON /USER1/ ALPHA, AX, AY, AZ, OPHI, EA, EANT, EPS, K, MANV,
OMER, PDOTAV, POOTRX, POOTRY, POOTRZ, PDOTIZ,
POOTV, PHIRX, PHYR, PHIRZ, TACCEL, THETMX, THOLC,
TN, TPHIN, TSMALL, XN, XNN, XNNN, XNU, YN,
Z, ZN

COMMON /USER2/ ISPI, ISUB, KEOPT, RFIXED, SLBMX

COMMON /USER3/ APOGEE, COMRAT, DIAMAX, EEQMT(9), EPME,
EOMINI, EQM1XL, EQM1YL, EQM2ZL, EQM2WT, EAGCY, EAGCY,
IDEBUG, ISITOR, MB12SH, OPTEMP, ORBINC, PERIGE,
MICRO, RELME, SPEC(6), SPEC(T), XCGSA1,
XME, XNE

COMMON /BTWN/ ACSSN, ACSHP, ALT, AREA, EATCAP,
BITRAT(2), CLIFE, CONWMT, SATDAM, DI,
DX, DLY, DZ, EQLG, EQLGSID,
FC, HARMN, HPTP, HTT, IBDLOC,
HTP, HTRPRB, HTRPRW, LMBDD, NC,
OMEG, OMGRS, PASSTR, PJ,
PL, FLMIN, POCNMT, PADA, RADA,
RAT, RJ, SABOLG, SAFLG, SATHT,
SATW, SATXCG, SATYCG, SATZCG, SAIXL,
SAIYL, SAIZL, SIDE, SYSLB, TCHMT,
THPRIC, TI, TKWT, TRPRIC, VB,
WCHP, VOL, WATE, WB, WBT,
WT, XJ, XNNZO, YJ, ZJ

** THE FOLLOWING AREA PRESETS THERMAL CONSTANTS

ALT = (APOGEE + PERIGE) / 2.0
IF (ORBINC NE. -360.) GO TO 200
IF (IAGNCY .GT. 1) GO TO 110

IF (ALT .LT. 500.) ORBINC = 80.
IF (ALT .GE. 500. AND ALT .LE. 19000.) ORBINC = 80.
IF (ALT .GT. 19000.) ORBINC = 0.0
GO TO 200

IF (ALT .LT. 500.) ORBINC = 35.
GO TO 200

110 IF (ALT .LT. 500.) ORBINC = 35.
GO TO 200
IF (ALT .GE. 500. AND. ALT .LE. 19000.) ORBINC = 80.
IF (ALT .GT. 19000.) ORBINC = 0.0

*** THE FOLLOWING AREA PRESETS RELIABILITY CONSTANTS ***

200 CONTINUE
BETA = 1.6
B = 1./BETA
RS1 = T/(ALOG(SPEC(6))*B)
RS2 = SPEC1/0.88
A = AMAX(RS1,RS2)
IF (SPEC(1) .EQ. -1.) SPEC(1) = EXP(-(T/(3.0*A))*BETA)
IF (SPEC(2) .EQ. -1.) SPEC(2) = EXP(-(T/(5.3*A))*BETA)
IF (SPEC(3) .EQ. -1.) SPEC(3) = EXP(-(T/(3.3*A))*BETA)
IF (SPEC(4) .EQ. -1.) SPEC(4) = EXP(-(T/(3.0*A))*BETA)
IF (SPEC(5) .EQ. -1.) SPEC(5) = EXP(-(T/(4.5*A))*BETA)
IF (RELME .EQ. -1.) RELME = EXP(-(T/(1.9*A))*BETA)

*** THE FOLLOWING AREA PRESETS VEHICLE SIZING CONSTANTS ***

300 CONTINUE

** - - - DETERMINE PJ AND RJ - - - **

EOMWT = EQM1WT + EQM2WT
DO 305 I = 1,9
305 EOMWT = EQM1WT + EQM2WT
SATWT = 36.9 * EQMWT**.672
EQBVOL = 0.4 * SATWT
SATDAM = (EQBVOL**2201.)*.333
EQBLG = SATDAM
IF (SATDAM .LE. DIAMAX) GO TO 306
SATDAM = DIAMAX
EQBLG = EQBVOL**2201./(SATDAM**SATDAM)

** - - - DETERMINE MISSION EQUIPMENT DIMENSIONS - - - **

IF (EOM1XL .LE. .1. E10. AND. EOM1YL .LE. .1. E10. AND. EOM2XL .LE. .1. E10. AND. EOM2YL .LE. .1. E10.) GO TO 400

EQMDEN = 25.0
V1 = EOM1WT/EQMDEN
V2 = EOM2WT/EQMDEN
IF (MB12SH .LT. 2) GO TO 350
IF (EQM2YL.EQ.1.E10) EQM2YL = (V2*1728./0.6)**.333
IF (EQM2YL.EQ.1.E10) EQM2YL = EQM1YL
IF (EQM2YL.EQ.1.E10) EQM2YL = 0.6 * EQM2YL

115 C
DIAG = EQM1YL/0.707
C
IF (DIAG .LT. DIAMAX) GO TO 310
C
120 C
DIAG = DIAMAX
C
EQM1YL = 0.707 * DIAG
EQM2YL = EQM1YL
EQM1XL = (V1*1728.)/(0.707*DIAG)**2
125 C
310 DIAG = EQM2YL / 0.707
C
IF (DIAG .LT. DIAMAX) GO TO 400
C
130 C
DIAG = DIAMAX
C
EQM2YL = 0.707 * DIAG
EQM2YL = EQM2YL
EQM2XL = (V2 * 1728.) / (0.707 * DIAG)**2
135 C
GO TO 400
C
C
350 IF (EQM1YL.EQ.1.E10) EQM1YL = ((V1*1728.)/0.471)**.333
IF (EQM1YL.EQ.1.E10) EQM2YL = EQM1YL
IF (EQM1YL.EQ.1.E10) EQM1XL = 0.6 * EQM1YL
145 C
IF (EQM2YL.EQ.1.E10) EQM2YL = (V2*1728./0.471)**.333
IF (EQM2YL.EQ.1.E10) EQM2YL = EQM2YL
IF (EQM2YL.EQ.1.E10) EQM2XL = 0.6 * EQM2YL
IF (EQM1YL .LT. DIAMAX) GO TO 360
C
150 C
EQM1YL = DIAMAX
EQM2YL = EQM1YL
EQM1XL = (V1*1728.) / (0.785*EQM1YL)**2
155 C
360 IF (EQM2YL .LT. DIAMAX) GO TO 400,
C
EQM2YL = DIAMAX
EQM2YL = EQM1YL
EQM2XL = EQM2XL = (V2 * 1728.) / (0.785*EQM2YL)**2
C
400 CONTINUE
PJ = EQM1WT * EQM1YL * EQM1YL / 8.0
160 C

*** THE FOLLOWING AREA PRESETS S AND C CONSTANTS **
C

165 C
* * CALCULATE TACCEL * *
C
PDOTMX = AMAX1(PDOTX,PDOTY,PDOTZ)
IF (PDOTMX.EQ.0.) AND. TACCEL.EQ.1.E10) GO TO 491
TMAX = THETMX/PDOTMX
170 C
IF (TACCEL.EQ.1.E10) TACCEL = 0.1 * TMAX 
GO TO 405 

C 401 TACCEL = 0. 

C 405 IF (OMEGR .NE. 1.E10) GO TO 404  
WR1 = 18000./(RJ/4636.8)  
WR2 = 90.-(100.*T)/(7.5*12.)  
WR = OMEGR  
IF(OMEGR.EQ.1.E10) WR = A MAX1(WR1,WR2)  
IF (WR.GE.60.) GO TO 402  
WR = 60. 
GO TO 403 

C 402 IF (WR.LE.90.) GO TO 403 
WR = 90. 

C 403 OMEGR = WR  

C 404 PHIMAX = A MAX1 (PHIRX,PHIRZ)  
HV = RJ/4636.8 * OMEGR * 6.28318 / 60.  
IF (XNN.EQ.1.E10) XNN = 0.0525 * PHIMAX * HV  
IERR = 0  
RETURN 

C  
EN0
SUBROUTINE INITIL(NCONF, IERR)
THIS SUBROUTINE SETS APPROXIMATIONS FOR ALL VALUES IN BTWN NASA 231
C DIMENSION NCONF(6) NASA 232
C WHICH ARE USED BEFORE THEY ARE CALCULATED NASA 233
COMMON /USER1/ ALPHA, AX, AY, AZ, DPHI, NASA 234
OMEGA, POISS, PODAV, POATR, PDRY, PODRZ, NASA 235
POF, PHIRX, PHIRY, PHIRZ, PHIR1, NASA 236
THETX, THOLD, TPHIN, TSMALL, NASA 237
COMMON /BTWN/ ACSSN, ACSNP, ALT, AREA, BATCAP, NASA 238
BITRA(2), CLIFE, CONWRT, D1, DT, EQLD0, EGBSIO, NASA 239
FF, HARNWT, HP1, HTPPPE, ITLOC, NASA 240
LMBD, NC, OMEG, PAASST, PJ, RADA, RABAD, NASA 241
PLMA, POCNM, PRSM, RADGB, RADD, NASA 242
RAT, RJ, SABOLG, SATL, SATTN, NASA 243
SATWT, SATXCG, SATYCG, SATZCG, SAIXL, NASA 244
SID, SISL, TSPW, STCMWI, THCMI, NASA 245
THRT(1), TKWP, TVCP, U, WH, WBI, NASA 246
V, XCM, XCGSA, XHER, XPEU, NASA 247
C 20 COMMON /PRTCOM/ ACCRCY, AM, AN, BF, BS, NASA 248
CODPT, CISTAF, CISTF, DOTE, DOTH, NASA 249
DRIT, EQBSTR, FEER, FF, FEE0PS, FEEEPS, NASA 250
GSE, HREL, IREL, ITRUNC, MBDL, NAME(3,60), NASA 251
OPS, PAY, PAYQL, PAYQ, PE, NASA 252
P, PD, RP, ROL(60), SRMBI, SATAP, NASA 253
QCP, ROLD(60), SATL, SATM, SATT, NASA 254
SATT(6), SUBE(7), SUBT(7), SUBUE(7), SUBUP(7), NASA 255
TAU, TAU(6,6), TE, TE, TCP, NASA 256
TB, TCR, TS, TTTVOLUM, VQL(60), WEIGT(60), NASA 257
XME, XMEWT, XVEST, NASA 258
IERR=0 NASA 231
ACCCY=AMINI(PHIRX, PHIRY, PHIRZ) NASA 232
EQMWT=EQM1WT+EQM2WT NASA 233
00 1 I = 1-9 NASA 234
1 EQMWT=EQM1WT+EQM2WT(I) NASA 235
SUBWT=36.9*EQMWT**.672 NASA 236
55 EQBVOL = .1*SATWT  
56 SATXCG = .500 + EQBLG * .5  
57 SATYCG = 0.  
58 SATZCG = 0.  
59 SATXL = 96.  
60 FS = (EPME+200.)*2.4  
61 SATYL = .1033*PS  
62 SATYI = 1.0  
63 TPRIM=T  
64 N= NCONF(6)  
65 GO TO (20,10,30),N  
66 C HERE IF A BOX  
67 10 EQBLG = (EQBVOL ** 1.56550)**.333  
68 EQBDIA = EQBLG  
69 EQBSID = 70.7*EQBDIA  
70 IF (EQBDIA .LE. DIAMAX) GO TO 11  
71 EQBOIA = DIAMAX  
72 EQBSID = 70.7*EQBDIA  
73 EQBLG = (EQBVOL*1728.)/(EQBSID*EQBSID)  
74 SATINX = (SATWT/6.)*EQBSID*EQBSID  
75 SATINY = (SATWT/12.)*(EQBSID*EQBSID+EQBLG*EQBLG)  
76 SATINZ = SATINY  
77 SATOAM = EQBDIA  
78 GO TO 100  
79 C HERE IF A CYLINDER  
80 20 SATDAM = EQBVOL*2201.0**.333  
81 EQBLG = SATDAM  
82 IF (SATDAM .LE. DIAMAX) GO TO 21  
83 SATDAM = DIAMAX  
84 EQBLG = EQBVOL*2201./(SATDAM*SATDAM)  
85 SATINX = (SATWT*SATDAM*SATC AH/8.)  
86 SATINY = (SATWT*12.)*(75*SATDAM*SATDAM+EQBLG*EQBLG)  
87 SATINZ = SATINX  
88 GO TO 100  
89 C HERE IF A SPHERE  
90 30 SATDAM = EQBVOL*3300.09**.333  
91 SATINX = .1*SATWT*SATDAM*SATDAM  
92 SATINV = .1*SATWT  
93 SATINZ = SATINV  
94 C IF SATDAM TOO BIG STOP PROGRAM  
95 IF (SATDAM .GT. DIAMAX) IERR=1  
96 IF (IERR .GT. 0) RETURN  
97 GO TO 100  
98 C SETS VALUES NEEDED BY S AND C  
99 100 IF (NCONF(1) .NE. 1) GO TO 120  
100 XJ = SATINX  
101 YJ = SATINY  
102 ZJ = SATINZ  
103 DJ = SATDAM/12.  
104 GO TO 200  
105 120 IF (NCONF(1) .NE. 2) GO TO 130  
106 XJ = SATINX  
107 YJ = SATINY  
108 ZJ = SATINZ  
109 DJ = SATDAM  
110 L= 5*SATDAM
OMEGS = 6.28318 / TPRIN
HM = SATINX * OMEGS
IF (HM.LT.500) GO TO 125
IERR = -1
125 IF (NCONF(6),EQ,2) GO TO 200
IF (NCONF(6),EQ,1) DI=.5*EQBLG
IF (NCONF(6),EQ,3) DT=.5*SATDAM
DX=DT
DY=DT
GO TO 200
130 IF(NCONF(1),GT,5) GO TO 200
XJ=SATINX
YJ=SATINY
ZJ=SATINZ
D=SATDAM
IF (NCONF(6),EQ,2) D=EQBEIA
DT=.5*EQBLG
IF (NCONF(6),EQ,3) DT=.5*SATDAM
DX=.5*SATDAM
IF (NCONF(6),EQ,2) DX=.5*EQBLG
DY=DT
DZ=DT
200 CONTINUE
RETURN
END

REGISTER ALLOCATION
1 REGISTERS ASSIGNED OVER THE LOOP BEGINNING AT LINE 52
DATA EA, EA1, ALPHA, TL, TACCEL, XNND, THOLD, POPTAV, POPTST, PHIFOV, NASA 355
* / 0.112, 0.1, 1.E10, 4., 100000., 01, 0.067, 40. /
DATA CLIFE/50000. / NASA 357
DATA MICR, TACGLY / 0.1 /
DATA BTRM, SCFL, TPRL, OPML, ARCN, NMSEQ/1, 1.9246E6, 6.4., 2*106. *, 500., 15.2*280., 1.8., 22*0. / NASA 355
DATA IVOLT, OPTCIP, EMP3, S0/0., 15.6*, 200. /
DATA IOPTCN, LIKE, FREQ, NEL, NDIR, FREQ, COMBAT, BWIDTH, NASA 357
* / 0.1, 1.2*280., 2*0.1801., 1000., 2.1*1.810. /
DATA EQPF, MB12SH, EQR1XL, EQR1YL, EQR1ZL, EQR2XL, EQR2YL / 4.95, 1, 
* 5*1. E10. /
DATA EQM3XL, IS80FG, NUMEEQ, EEQW, EEQVL, EM1YG, EM1ZG, EM2YG, NASA 369
* / 1.110., 200., 1*0. /
DATA EM2ZG, CEEGEX, EELOC, XCGSA1, XCGSA3 / 0., 92., 9, 93., 2*1. / NASA 367
DATA ISATOR, ORBINC, IASTCNV/1., -306.5, 1 /
DATA SKM2/X/1.5, 0 /
DATA CA, CE/10.5, 0 /
DATA EQAMT, EQM2MT, DIAJMN, APOGEE, PERIGE, MICRO/2*435., 120., 2*500., NASA 369
* 0 /
DATA KEPT, RFIXED, ISLMX, ISPT, SPEC, SPEC1, ISUBJ, RELME / 1, NASA 370
* 1., 50000., 0., 0.5*(-1.), 0.6*18., 0.1*1. /
DATA NFv, NQV, XNDR, XNDU, FEI3CP, IMETYP, PI / 4., 1., 0., 0., 0.07, 2., 2.8, 1. / NASA 375
END
CO 2 I=1,60
COMPSE(I)=I.
COMPSP(I)=0.
COMPIT(I)=0.
2 COMPRI(I)=0.
NEQUIP(I)=NEQUIP(1)
DO 3 J=2,5
3 NEQUIP(J)=NEQUIP(J-1)+NEQUIP(J)
X(I) = HAT
X(2) = HARNWT
X(3) = THCMNT
X(4) = CONVNT
X(5) = WEIGHT(2)
X(6) = PASSTR
X(7) = PCCNWT
PRINT 9004,(X(I),I=1,7)
9004 FORMAT (10E11.4)
C
I=1
J=1
C 100 IF (NCHOOSE(I).EQ.0) GO TO 200
C ** COMPUTATIONS FOR CATALOG ITEMS
C IF (I *GT. MEQUIF(J)) J=J+1
ICHSH = ICHOOSE(I)/100
XFP = 1.
XFE = 1.
XFT = 1.
C1 = COST(1,I)
Q =QV + NFV
QP = Q * NCHOOSE(I)
Q5 =5.*NCHOOSE(I)
P5 = COST(3,I)
QREF = COST(4,I)
FQ = NCHOOSE(I)/QREF.
IF (FQ.LT.1) FQ = 1.
FRE = Q0.8875 + 0.125*FQ
FRG = 0.3 + 0.7*FQ
GO TO (110,120,130,140,150),J
C SET SCALE FACTORS FOR S & C CATALOG ITEMS
GO TO (170,170,170,170,170,170,170,170,170,170,115,115,170,170,170,170,170,170,170,170),ICHSH
110 IF (NCONF(1).LE.1) GO TO 170
IF (NCONF(5).EQ.1.OR.NCONF(5).EQ.3.OR.NCONF(5).EQ.5)GO TO 112
XFP = 3.3
GO TO 170
112 XFP = 7.9
GO TO 170
C 115 IF (NCONF(4).LE.1) GO TO 170
IF (NCONF(5).EQ.1.OR.NCONF(5).EQ.3.OR.NCONF(5).EQ.5)GO TO 117
XFE = 13.5
XFT = 2.4
XFP = 2.2
GO TO 170

C
117 XFE = 13.5
XFT = 2.4
XFP = 5.0
GO TO 170

C
120 SET SCALE FACTORS FOR AUXPRO CATALOG ITEMS
121 IF (ITHST (1).GT. 1) GO TO 122
ITHST = ITHST + 1
IF (ITHRM (1).LT.REL (1, I)) XFT = 0.25
GO TO 170

C
122 IF (ITHRM (2).LT.REL (1, I)) XFT = 0.25
GO TO 170

C
130 SET SCALE FACTORS FOR DPI CATALOG ITEMS
131 GO TO (170, 131), ICHS
132 XFT = 1.9
XFP = 3.0
GO TO 136
134 XFT = 1.9
XFP = 4.5
GO TO 170

C
136 IF (IBTFRS .GT. 1) GO TO 138
IBTFRS = IBTFRS + 1
IF (BITRAT (1). GT. 100000.) XFE = 2.7
GO TO 170

C
138 IF (BITRAT (2).GT. 100000.) XFE = 2.7
GO TO 170

C
140 SET SCALE FACTORS FOR COMM CATALOG FACTORS
(NONE NEEDED AT THIS TIME)
141 GO TO 180

C
150 SET SCALE FACTORS FOR E.P. CATALOG ITEMS
151 IF (IBACAP .LE. 15) GO TO 155
IF (NCONF (5).EQ.1.OR.NCONF (5).EQ.3.OR.NCONF (5).EQ.5) XFE = 2.3645

C
155 IF (NCONF (5).EQ.2.OR.NCONF (5).EQ.4.OR.NCONF (5).EQ.6) GO TO 170
XFE = 6.
XFP = 4.
GO TO 170

C
** COMPUTE DESIGN ENGINEERING COST, (DE OR COMPE)
170 COST (1, I) = COST (1, I) * PI * XFE * 1000 * XFE

C
** COMPUTE TEST AND EVALUATION COST, (TE OR COMPT)
C ** SUB-TOTAL ENGINEERING COSTS
C
230 COST(I) = COST(I) + COST(2, I)
C
235 ** COMPUTE COMPONENT AVERAGE UNIT PRODUCTION COST, (PU OR COMPUP)
C
COST(3, I) = 1.277 * P5 * QP**.485 * PI1000.* XFP/Q
C
C ** COMPUTE COMPONENT CM AVG 5 UNIT PROD. COST
C
COMP5P(I) = 200. * P5 * Q5**.485 * PI * 1.277 * XFP
C
C ** COMPUTE COMPONENT AVERAGE PRODUCTION ENGINEERING (PE OR COMPUE)
C
COST(4, I) = C1 * (QP**.485 - 1.0) * FRE * PI * XFE
C
240 GO TO 185
C
180 CONTINUE
C
240 DO 183 JJJ = 1, 5
C
183 COST(JJJ, I) = 0.0
C
185 CONTINUE
C
245 IF (IDEEUG.EQ.0) PRINT 9000, (COST(JJJ, I), JJJ = 1, 4), COMPR(I), COMP5P(I), COMP5E(I)
C
I = I + 1
GO TO 100
C
C
250 ** COMPUTATIONS FOR SUBSYSTEM COSTS BASED ON COST ESTIMATING
C ** RELATIONSHIPS (C.E.R.-S)
C
255 200 M = 0
J = I + 6
ISAVE = I
PRINT 993
C
993 FORMAT(//)
C
260 DO 300 K = I, J
M = M + 1
C
265 FE(M) = 1.
FT(M) = 1.
FP(M) = 1.
GO TO (205, 210, 215, 220, 225, 230, 235), M
C
C SET SOLAR ARRAY CER FACTORS
C
270 205 GO TO (207, 207, 206, 206), IMETYP
206 FE(M) = 4.0
FT(M) = 4.0
C
207 IF (NCONF(5), EQ. 2 .OR. NCONF(5), EQ. 4 .OR. NCONF(5), EQ. 6) GO TO 270
FP(M) = 2.0
GO TO 270
C
C SET WIRING HARNESS CER FACTORS
C
210 FE(M) = 5.0
FT(M) = 5.0
FP(M) = 3.0
GO TO 270
C
```
C SET THERMAL CER FACTORS
215 IF (NCONF(1).NE.1).FE(M)=2.0
   IF (NCONF(5).EQ.1.OR.NCONF(5).EQ.3.OR.NCONF(5).EQ.5) FP(M)=2.0
   GO TO 270
C SET POWER CONVERTERS CER FACTORS
   (NONE NEEDED AT THIS TIME)
   GO TO 270
C SET PROPELLANT FEED SYSTEM CER FACTORS
225 IF (NCONF(1).GT.1) GO TO 226
   RE(M)=123200.
   BE(M)=2.72
   FE(M)=0.507
   FT(M)=0.325
   FP(M)=0.615
   GO TO 270
C 226 RE(M)=545640.
   BE(M)=0.222
   FE(M)=0.268
   FT(M)=0.619
   FP(M)=0.840
   GO TO 270
C SET STRUCTURES CER FACTORS
230 STRF=0.554*SATWT**(-0.168)
   RATIO = PASSTR/SATWT
   IF (STRF.GE.1.0) STF=1.0*(STRF/RATIO)**STF
   IF (STRF.LT.1.0) STF=1.0-(RATIO-STRF)**STF
   PRINT 9001,STF
   FORMAT(C3X,5HSTF =,E23.4)
9001 IF (NCONF(1).EQ.2.OR.NCONF(5).EQ.4.OR.NCONF(5).EQ.6) GO TO 231
   FE(M)=2.5
   FT(M)=3.0
   FP(M)=4.0
   GO TO 270
C SET POWER CONTROL EQUIPMENT CER FACTORS
235 IF (NCONF(5).EQ.2.OR.NCONF(5).EQ.4.OR.NCONF(5).EQ.6) GO TO 270
   FE(M)=3.0
   FT(M)=4.0
   FP(M)=4.0
   GO TO 270
C 270 COMPDEV = RE(M) * X(M)**RE(M) * FE(M)
   IF (M.EQ.11) COMPDEV = (FE(M)*RE(M)*AREA**BE(M))
   = (FT(M)*RT(M)*X(M)**BT(M))
C DESIGN ENGINEERING COSTS (COMPE OR DE)
```
C COST(i,K) = COMP*SF*PI
C TEST + EVALUATION COSTS (COMP OR TE)
C IF(M.EQ.5) X(M) = ORIG 
C COST(2,K) = RT(H)*X(M)**BT(M)*SF*PI*FT(M)
C SUBTOTAL COST
C COMP(K) = COST(i,K) + COST(2,K)
C UNIT PRODUCTION COST
C COST(3,K) = RP(H)*ARE**BP(H)*SF*PI**Q*(M)*FF(H)
C IF (M.EQ.5) COST(3,K) = RP(H)*AREA**BP(H)*SF*PI**Q*(M)*FF(H)
C UNIT ENGINEERING COSTS
C COST(4,K) = COMP*(Q**4.85-1.)*SF*PI/Q
C ** COMPUTE COMPONENT CUM AVG 5 UNIT PRODUCTION COST
C COMP(K) = 0.783*COST(K)*Q**0.152
C ** COMPUTE COMPONENT CUM AVG 5 PRODUCTION COST
C COST(K) = 0.2365*COMP*SF*PI*FE(H)
C SUBTOTAL PRODUCTION
C COMPU(K) = COST(3,K) + COST(4,K)
C IF (DEBUG.EQ.5) PRINT 9000, (COST(J,K), JJ=1,4), COMP(K), COMPU(K),
C GO TO (280,280,281,282,283,286)
C SUBE(5) = SUBE(5) + COST(i,K)
C SUBR(5) = SUBR(5) + COMP(K)
C SUBUE(5) = SUBUE(5) + COST(4,K)
C SUBUP(5) = SUBUP(5) + COST(3,K)
C SUBU(5) = SUBU(5) + COMP(K)
C SUBSE(5) = SUBSE(5) + COMP(K)
C GO TO 300
C ** THERMAL CER SUB-TOTAL
C SUBE(7) = SUBE(7) + COST(i,K)
C SUBR(7) = SUBR(7) + COMP(K)
C SUBUE(7) = SUBUE(7) + COST(4,K)
C SUBUP(7) = SUBUP(7) + COST(3,K)
C SUBU(7) = SUBU(7) + COMP(K)
C SUBSE(7) = SUBSE(7) + COMP(K)
C GO TO 300
C ** AUX PROP CER SUB-TOTAL
C SUBE(2) = SUBE(2) + COST(i,K)
C SUBR(2) = SUBR(2) + COMP(K)
C SUBUE(2) = SUBUE(2) + COST(4,K)
C SUBUP(2) = SUBUP(2) + COST(3,K)
C SUBU(2) = SUBU(2) + COMP(K)
C SUBSE(2) = SUBSE(2) + COMP(K)
C GO TO 300
C ** STRUCTURE CER SUB-TOTAL
C SUBE(6) = SUBE(6) + COST(i,K)
C SUBR(6) = SUBR(6) + COMP(K)
SUBUE(6) = SUBUE(6) + COST(4,K)
SUBUP(6) = SUBUP(6) + COST(3,K)
SUBU(6) = SUBU(6) + COMPU(K)
SUBSE(6) = SUBSE(6) + COMPSE(K)
SUBSP(6) = SUBSP(6) + COMPSP(K)

300 CONTINUE
SUM SUB-TOTALS BY SUBSYSTEMS OF CATALOG ITEMS
IJ = 1
IK = 0
DO 320 J = 1,5, IF(J.NE.1) IJ = IK + 1
IK = IK + NEQUIP(J)
DO 310 I = IJ, IK
SUBE(J) = SUBE(J) + COST(I,I)
SUBT(J) = SUBT(J) + COST(2,I)
SUBR(J) = SUBR(J) + COMPR(I)
SUBU(J) = SUBU(J) + COMPU(I)
SUBSE(J) = SUBSE(J) + COMPSE(I)
SUBSP(J) = SUBSP(J) + COMPSP(I)
310 CONTINUE
320 CONTINUE
DO 9999 I = 1,7
IF(IDEBUG.EQ.1)PRINT 9000,SUBE(I),SUBT(I),SUBR(I),SUBU(I),
* SUBSE(I),SUBUP(I),SUBSP(I)
9999 CONTINUE
9000 FORMAT (8(I2X,F11.0))

** TOTAL COSTS FOR BASIC SPACECRAFT
DO 400 I = 1,7
DE = DE + SUBE(I)
TE = TE + SUBT(I)
SYSR = SYSR + SUBR(I)
PE = PE + SUBU(I)
SYSU = SYSU + SUBU(I)
PSE = PSE + SUBSE(I)
PSP = PSP + SUBSP(I)
400 CONTINUE

COMPUTE TOOLING AND TEST EQUIPMENT
TOOLR = 0.
TOOLU = 0.
TOOL5 = 0.

COMPUTE QUALITY CONTROL
QGR = .015*DE + .14*TE
QCU = 0.015*PE + 0.14*PU
QC5 = 0.015*PSE + 0.14*P5P

COMPUTE SYSTEMS ENGINEERING AND INTEGRATION

SEIR = 0.32*DE + 0.27*TE
SEIP = 0.32*PE + 0.22*PU
SE15 = 0.32*PSE + 0.22*PSP

COMPUTE PROGRAM MANAGEMENT

PMR = 0.19*DE + 0.02*IE
PMP = 0.19*PE + 0.02*PU
PM5 = 0.19*P5E + 0.02*P5P

*** TOTAL SPACE CRAFT COSTS

SATR = SYSR + TOCLR + QCF + SEIR + PMR
SATU = SYSU + TOOLU + QCU + SEIP + PMP

*** TOTAL PAYLOAD COSTS

SATINV = NFV * SATU
XMEINV = NFV * XMEU
PAYU = SATR + XMEU
PAYQL = NOV * (SATU + XMEU)
PAYINV = SATINV + XMEINV

CUMULATIVE AVERAGE COST FOR FIVE(5) SPACECRAFT

PRINT 9003 + SATINV

9002 FORMAT(1X,GHSAT5 = E11.4)

*** COMPUTE GROUND SUPPORT EQUIPMENT COST (DEVEL. AND PROD.)

IF (IMETYP.NE.1) GO TO 420

** SET FACTOR FOR COMSAT

FGSE = 0.409
GO TO 440

420 IF (NCONF(5).EQ.1 .OR. NCONF(5).EQ.3 .OR. NCONF(5).EQ.5) GO TO 430

** SET FACTOR FOR GENERAL

FGSE = 1.0
GO TO 440

** SET FACTOR FOR GENERAL PACOLE

FGSE = 2.121

** SET FACTOR FOR OTHER

GSE = 49.72* DE **.689*FGSE

** COMPUTE LAUNCH COSTS

COMPUTE UNIT LAUNCH COST

XLN = 31.0 * SAT5 **0.588
COMPUTE TOTAL LAUNCH COSTS

\[ XL\text{TOT} = NFV \times XLN \]

**TOTAL DDT+E COST**

\[ D\text{OTE} = PAYR + PAYQUL + GSE + FEER \]

**TOTAL INVESTMENT COST**

\[ FE\text{EINV} = FEER\text{CT} \times \text{SATINV} \]

**TOTAL OPERATIONS COST**

\[ FE\text{EEOPS} = XL\text{TOT} \times FEER\text{CT} \]

\[ \text{OPS} = XL\text{TOT} + FE\text{EEOPS} \]

RETURN

END
SUBROUTINE PRNT

** THIS IS THE OUTPUT SUBROUTINE WHICH CONTROLS THE PRINTED
** OUTPUT OF ANY ACCEPTABLE DESIGN

COMMON /USERI/ AFOGEE, COMMAT, DIAMAX, EEQNT(9), EPME, OPT=2, FTN 4.2+383 03/27/75 21.38.18

SUBROUTINE PRNT (IERRNEQUIPNACCEPNCONF)

COMMON /USERP/ IPRINITITLE

COMMON /BETWN/ ACSN, ACSR, AREA, BATCAP, D, DF, DC, DO, DT, DZ, EQBC, EOBSC, HARTN, HARTPR, HTILOC

COMMON /USERP/ IPRINITITLE

DIMENSION IERR(7), NEQUIP(5), NCONF(6)

DIMENSION ITITLE(13) NASA 878

REAL MNDOLD

MNDOLD=MNDOLD/720.

IF (IPRINT.EQ.0) RETURN

IF (NACCEP.GT.1) GO TO 100 NASA 879

PRINT 9000.
READILY TP TXA42HNCONF(C)=6 IS SERIES LOAD REG.
100 BLX 80 70 S UNIFIED 65 4 ROPPELLAN T 28HNCONF(±)=3

55 NASA C 9010 FORMAT (7Xq44HNCONF(5)=5 IS THAN 256,10X, UCC 7X, 2ZHNCONF(6)=1 IS, 2H (NCONF(2)),/ TX,23HNCONF(i)=1 IS FORMAT (38X,4H)

NASA 94309431 IXXX MEANS OB P, GIMBCMG ACCEPTABLE;25X012 IS-cycle life of thrusters are 5X, 49HNCONF(5)=2 IS SHUNT REGULATION - BODY MT 112174 NASA 951952 NASA 953954 NASA 955956 NASA 957

C PRINT 9013
9013 FORMAT (7X,26HIERR = 0 MEANS NO MESSAGES, 43X,15S SUPPLIED IN PCH, / NASA 958959 NASA 960961 NASA 962963 NASA 964965 NASA 966967 NASA 968969
THIS IS THE BEGINNING OF THE SYSTEM LEVEL OUTPUT

** SYSTEM DESCRIPTION **

100 PRINT 9000
PRINT 9999, TITLE
9999 FORMAT (10X, 13A6)
PRINT 102, NACP
102 FORMAT (47H, ** ** *, SYSTEM DESCRIPTION - - DESIGN NUMBER, 13, 8H *

C

PRINT STABILIZATION AND CONTROL INFORMATION

104 FORMAT (12X, 25HSTABILIZATION AND CONTROL)
104
ICONF = NCONF(1)
GO TO (105, 106, 110, 112, 114), ICONF

C

PRINT DUAL SPIN

106 PRINT 107
107 FORMAT (14X, 27HCONFIGURATION - - DUAL SPIN)
GO TO 116

C

PRINT YAW SPIN

108 PRINT 109
109 FORMAT (14X, 26HCONFIGURATION - - YAW SPIN)
GO TO 116

C

3-AXIS MASS EXPULSION

110 PRINT 111
111 FORMAT (14X, 43HCONFIGURATION - - THREE AXIS MASS EXPULSION)
GO TO 116

C

MASS EXPULSION WITH CONTROL MOMENT GYROS (CMG)

112 PRINT 113
113 FORMAT (14X, 58HCONFIGURATION - - MASS EXPULSION WITH CONTROL MOMENT
1 CYROS)
GO TO 116

C

MASS EXPULSION WITH PITCH MOMENTUM WHEEL

114 PRINT 115
115 FORMAT (14X, 58HCONFIGURATION - - MASS EXPULSION WITH PITCH MOMENTUM
1 WHEEL)
C
1116 PRINT 117, ACCRCY
1117 FORMAT(1X, 20H POINTING ACCURACY = ,F11.6, 6H(DEG.))
1118 PRINT AUXILIARY PROPULSION INFORMATION
1119 PRINT 119
1119 FORMAT(12X, 20H AUXILIARY PROPULSION)
1120 ICONF = NCONF(2)
1120 GO TO (120, 122, 124), ICONF
1121 FORMAT(1X, 26H CONFIGURATION - - COLD GAS)
1122 PRINT 121
1122 FORMAT(14X, 32H CONFIGURATION - - MONOPROPELLANT)
1123 GO TO 126
1124 PRINT 125
1125 FORMAT(14X, 30H CONFIGURATION - - BIPROPELLANT)
1126 PRINT 127, II
1127 FORMAT(16X, 16H TOTAL IMPULSE = ,F11.0, 8H(LB-SEC))
1128 PRINT DATA PROCESSING INFORMATION
1129 PRINT 129
1129 FORMAT(12X, 35H DATA PROCESSING AND INSTRUMENTATION)
1130 ICONF = NCONF(3)
1130 GO TO (130, 132), ICONF
1131 PRINT 131
1131 FORMAT(14X, 43H CONFIGURATION - - GENERAL PURPOSE PROCESSOR)
1132 GO TO 134
1133 PRINT 133
1133 FORMAT(14X, 49H CONFIGURATION - - SPECIAL PURPOSE PROCESSOR (OTU))
1134 PRINT 135, TOTOPS
1135 FORMAT(16X, 27H COMPUTER OPERATIONS RATE = ,F11.0, 5H(IPS))
PRINT COMMUNICATIONS INFORMATION

IF CONF = NCONF(4) THEN
   GO TO (148,150,152,154,156), ICCNF

SEPARATE UPLINK AND DOWNLINK

PRINT 149
FORMAT(14X,46HCONFIGURATION - - SEPARATE UPLINK AND DOWNLINK)
GO TO 158

UNIFIED LINK - COMMON ANTENNAS

PRINT 151
FORMAT(14X,46HCONFIGURATION - - UNIFIED LINK-COMMON ANTENNAS)
GO TO 158

UNIFIED LINK - SEPARATE ANTENNAS

PRINT 153
GO TO 158

PRINT 155
FORMAT(14X,46HCONFIGURATION - - UNIFIED LINK-COMMON ANTENNAS PLUS DOWNLINK)
GO TO 158

UNIFIED LINK - SEPARATE ANTENNAS PLUS DOWNLINK

PRINT 157
FORMAT(14X,62HCONFIGURATION - - UNIFIED LINK-SEPARATE ANTENNAS PLUS DOWNLINK)
PRINT DATA RATES

PRINT BITRAT(1)
FORMAT(16X,3HPRIMAR Y DOWNLINK DATA RATE = ,F14.3,6H(KBPS))

PRINT BITRAT(2)
FORMAT(16X,3HSEPARATE DOWNLINK DATA RATE = ,F14.3,6H(KBPS))

PRINT ELECTRICAL POWER INFORMATION

ICONF = NCONF(5)
GO TO (164,166,168,170,172,174),ICONF

SHUNT - PADDLE MOUNTED
PRINT 165
FORMAT(14X,5H CONFIGURATION - - SHUNT - PADDLE MOUNTED SOLAR ARRAY)
GO TO 176

SHUNT - BODY MOUNTED
PRINT 167
FORMAT(14X,5H CONFIGURATION - - SHUNT - BODY MOUNTED SOLAR ARRAY)
GO TO 176

SHUNT AND DISCHARGE - PADDLE MOUNTED
PRINT 169
FORMAT(14X,7H CONFIGURATION - - SHUNT AND DISCHARGE REGULATION - PADDLE MOUNTED SOLAR ARRAY)
GO TO 176

SHUNT AND DISCHARGE - BODY MOUNTED
PRINT 171
FORMAT(14X,7H CONFIGURATION - - SHUNT AND DISCHARGE REGULATION - BODY MOUNTED SOLAR ARRAY)
GO TO 176

SERIES LOAD REGULATION - PADDLE MOUNTED
PRINT 173
FORMAT(14X,6H CONFIGURATION - - SERIES LOAD REGULATION - PADDLE MOUNTED SOLAR ARRAY)
GO TO 176

SERIES LOAD REGULATION - BODY MOUNTED
PRINT 175
FORMAT(14X,6H CONFIGURATION - - SERIES LOAD REGULATION - BODY MOUNTED SOLAR ARRAY)
PRINT E.P. STATS

PRINT 177, PL

177 FORMAT(16X,35HEND OF LIFE POWER REQUIREMENT =, F11.2,7H(WATTS))

PRINT 179, AREA

179 FORMAT(16X,22HTOTAL SOLAR ARRAY AREA, 12X, 1H=, F11.2, 7H(FT**2))

PRINT 181, CISTAR

181 FORMAT(16X,35HMINIMUM INSTALLED BATTERY CAPACITY =, F11.2,6H(AMP-HR) 1)

PRINT VEHICLE SIZING INFORMATION

183 FORMAT(12X, I4HVEHICLE SIZING)

ICONF = NCONF(6)

GO TO (184, 186, 188), ICCNF

CYLINDER:

184 PRINT 185

185 FORMAT(14X, 26HCONFIGURATION - - CYLINDER)

GO TO 190

BOX

186 PRINT 187

187 FORMAT(14X, 21HCONFIGURATION - - BOX)

GO TO 190

SPHERE

188 PRINT 189

189 FORMAT(14X, 24HCONFIGURATION - - SPHERE)

PRINT VEHICLE WEIGHT AND LAUNCH WEIGHT

190 PRINT 191, SATWT, SATWT


PRINT EQUIPMENT BAY DIMENSIONS

192 FORMAT(16X, 31HVEHICLE BAY DIMENSIONS LENGTH, F11.2, 11H(IN), HEIGHT 1, F11.2, 10H(IN), WIDTH, F11.2, 4H(IN))

PRINT MAXIMUM MISSION EQUIPMENT HEIGHT AND WIDTH AND LENGTH

193 FORMAT(16X, 12HMISSION EQUIPMENT LENGTH , F11.2, 12H(IN), HEIGHT , F11. 12H(IN), WIDTH, F11.2, 4H(IN))
C PRINT 244 PAYQLL
244 FORMAT(16X, 20HQUALIFICATION UNITS, F13.1)

C PRINT GSE
246 FORMAT(16X, 6HG, S.E., 14X, F13.1)

C PRINT LAUNCH SUPPORT
248 FORMAT(16X, 14H LAUNCH SUPPORT, 53X, F13.1)

C PRINT CONTRACTOR FEE
250 FORMAT(16X, 14H CONTRACTOR FEE, 6X, F13.1, 9X, F13.1, 12X, F13.1)

C PRINT PROGRAM TOTALS
252 FORMAT(16X, 14H PROGRAM TOTALS, 6X, F13.1, 9X, F13.1, 12X, F13.1)

C PRINT SCHEDULE INFORMATION
254 FORMAT(12X, 8HSCHEDULE)

C PRINT COMPONENT DESIGN DEVELOPMENT TIME
256 FORMAT(16X, 33H COMPONENT DESIGN DEVELOPMENT TIME, 13X, F5.1, 8H(MONTHS))

C PRINT COMPONENT QUALIFICATION TIME
258 FORMAT(16X, 26H COMPONENT QUALIFICATION TIME, 18X, F5.1, 8H(MONTHS))

C PRINT SUBSYSTEM DEVELOPMENT TIME
260 FORMAT(16X, 26H SUBSYSTEM DEVELOPMENT TIME, 20X, F5.1, 8H(MONTHS))

C PRINT SUBSYSTEM QUALIFICATION TIME
262 FORMAT(16X, 26H SUBSYSTEM QUALIFICATION TIME, 16X, F5.1, 8H(MONTHS))

C PRINT SYSTEM DEVELOPMENT AND FLIGHT READINESS TIME
264 FORMAT(16X, 46H SYSTEM DEVELOPMENT AND FLIGHT READINESS TIME, F5.1, 18H(MONTHS))
PRINT SCHEDULE DURATION TO LAUNCH

PRINT 266,SKTAU(6)

26E FORMAT(IX,D29HSCHEOULE DURATION (TO LAUNCH),17X,F5.1,8H(MONTHS))

END OF SYSTEM PRINT-OUT.  --- BEGIN SUBSYSTEM IF REQUIRED  ---

FRINT 9000
IF (IPRINT.LE.1) RETURN
PRINT 9999,ITITLE
PRINT 1000,ITITLE
1000 FORMAT(9H* * * SUBSYSTEM DESCRIPTIONS -- DESIGN NUMBER ,I3,8
1H * * * *)
PRINT 104
ICONF = NCONF(1)
GO TO (1004,1006,1008,1010,1012),ICONF
DUAL SPIN
1004 PRINT 107
GO TO 1014
C YAW SPIN
1006 PRINT 109
GO TO 1014
C 3-AXIS MASS EXPULSION
1008 PRINT 111
GO TO 1014
C M.E. W/ CMG
1010 PRINT 113
GO TO 1014
C M.E. W/ P.M.W.
1012 PRINT 115

CONTINUE

FRINT S + C EQUIPMENT IDENTIFIERS
IK = NEQUIP(I)
FRINT 1020, (ICHOOSE(I),I=1,IK)
1020 FORMAT(14X,25HEQUIPMENT CODE IDENTIFIER,14(1X,I4))
FRINT 1022, (NCFOST(I),I=1,IK)
1022 FORMAT(14X,25HEQUIPMENT QUANTITIES ,14(1X,I4))
FRINT 1024,WEIGHT(I),VOLUME(I),POWER(I)
1A REQUIREMENT ,F11.1,6H(WATT))
FRINT S+ C COSTS
FRINT 1026, SUBE(1), SUBT(1), SUBUP(1), SUBUE(1)
1026 FORMAT(24X,15HCES. ENG, COST ,F11.1,10X,17TEST + EVAL. COST,4X,F
11.1,/,24X,15HUNIT PROF.COST ,F11.1,10X,17HUNIT ENG. COST,4X,F
11.1)
625  PRINT S+C REL
   1028 FORMTF(24x,11HRELIABILITY,5X,F11.4)
630  PRINT S+C SCHEDULE
635  PRINT 1030,(TAU(J),J=1,5)
   1030 FORMAT(24X,8HSCHEDULE,7X,26X,7HCOMPONENT DEVELOPMENT TIME ,F5.1,3
*H(MONTH) COMPONENT QUALIFICATION TIME ,F5.1,7H(MONTH) /,26X,7H(S)YSTEM DEVELOPMENT TIME ,F5.1,7H(MONTH) SUBSYSTEM QUALIFICATION TIME ,F5.1,7H(MONTH) SUBSYSTEM QUALIFICATION TIME ,F5.1,7H(MONTH) SUBSYSTEM QUALIFICATION TIME ,F5.1,7H(MONTH) SUBSYSTEM QUALIFICATION TIME ,F5.1,7H(MONTH)
640  PRINT 1032,IERR(I)
   1032 FORMAT(24X,4HIERR,5X,10)
645  PRINT 119
   119 ICONF = NCONF(2)
   GO TO (1034,1036,1038), ICONF
650  COLD GAS
   1034 PRINT 121
   121 GO TO 1040
655  MONOPROPellant
   1036 PRINT 123
   123 GO TO 1040
660  EIPROPellant
   1038 PRINT 125
665  IJ = IK + 1
   1040 IK = NEQUIP(2) + IK
   1SH(LBS))
   1042 PRINT 1026,SUBE(2),SUBT(2),SUBUP(2),SUBUE(2)
670  PRINT 1028,SSREL(2)
   1028 PRINT 1030,(TAU(J),J=1,5)
675  PRINT 1032,IERR(2)
   1032 PRINT DPI S/S INFO
PRINT 129
C
    ICONF = NCONF(3)
    GO TO (1042,1044), ICONF
    GEN. PURP. PROC.
1042 PRINT 131
    GO TO 1046
C
1044 PRINT 133
C
1046 IJ = IK + 1
    IK = NEQUIP(3) + IK
 C
    PRINT 1020, (ICHOSE(I), I=IJ, IK)
    PRINT 1022, (INCHOOSE(I), I=IJ, IK)
    PRINT 1024, WEIGHT(3), VOLUME(3), POWER(3)
    PRINT 1025, SUBE(3), SUBT(3), SUBUP(3), SUBUE(3)
    PRINT 1028, SSREL(3)
    PRINT 1030, (TAU(3, J), J=1,5)
    PRINT 1032, IEQ(3)
C
    PRINT COMM S/S INFO
C
710 PRINT 9000
    PRINT 146
C
    ICONF = NCONF(4)
    GO TO (1048, 1050, 1052, 1054, 1056), ICONF
C
1048 PRINT 149
    GO TO 1058
C
1050 PRINT 151
    UNIF. COM. ANT.
    GO TO 1058
C
1052 PRINT 153
    UNIF. SEP. ANT.
    GO TO 1058
C
1054 PRINT 155
    UNIF. COM. ANT. + DOWN
    GO TO 1058
C
1056 PRINT 157
C
C
1058 IJ = IK + 1
    IK = NEQUIP(4) + IK
C
    PRINT 1020, (ICHOSE(I), I=IJ, IK)
    PRINT 1022, (INCHOOSE(I), I=IJ, IK)
    PRINT 1024, WEIGHT(4), VOLUME(4), POWER(4)
    PRINT 1025, SUBE(4), SUBT(4), SUBUP(4), SUBUE(4)
PRINT 1028,SSREL(4)
PRINT 1030,(TAU(4,J),J=1,5)
PRINT 1032,IERR(4)
PRINT E,P. S/S INFO.
PRINT 162
ICONF = NCONF(5)
GO TO (1060,1062,1064,1066,1068,1070),ICONF
SHNT / PADDLE
GO TO 1072
SHNT / BODY
GO TO 1072
SHNT + DSCHG / PADDLE
GO TO 1072
SHNT + DSCHG / BODY
GO TO 1072
SER, LD. / PADDLE
GO TO 1072
SER, LD. / BODY
GO TO 1072
1072 IJ = IK + 1
IK = NEQUIP(5) + IK
PRINT 1028,(ICHOOSE(I),I=IJ,IK)
PRINT 1024,WEIGHT(5),VOLUME(5),POWER(5)
PRINT 1074,HARNWT,MATE
PRINT 1026,SUBE(5),SUBT(5),SUBUP(5),SUBUE(5)
PRINT 1028,SSREL(5)
PRINT 1030,(TAU(5,J),J=1,5)
1074 FORMAT(24X,14HARNESS WEIGHT,F11.1,25H(LBS), SOLAR ARRAY -WEIGHT,F1
*1.1,5H(LBS))
PRINT MISSION EQUIPMENT INFORMATION
PRINT 1076
1076 FORMAT(12X,17HMISSION EQUIPMENT)
PRINT 1024,XMEV,XMREV,EPHE
PRINT 1078,XMEK,XMEU
1078 FORMAT(24X,10H00+6 COST,6X,F11.1,9X,17H AVERAGE UNIT COST,3X,F11.1
*)
PRINT 1028,SSREL(6)
PRINT 2010
2010 FORMAT(39X,9HUNIT,3X,4HUNIT,3X,4HUNIT,29X,7HVEHICLE,7X,7HVEHICLE/,1X,9HIDENT TYPE NO. WEIGHT VOLUME POWER D.E. COST)
4X, T.E. COST PROD. COST ENG. COST)
I J = 1
I K = NEQUIP(I)
C PRINT S+G
CO 2100 I = IJ, I K
2100 PRINT 2110, ICHOSE(I), (NAME(J, I), J=1, 3), NCHOSE(I), REL(I, I), VQL(I),
PWR(I), (COST(J, I), J=1, 4)
11X, F11, 11X, F11, 11X
C PRINT A.P.
I J = IK + 1
I K = NEQUIP(2) + IK
C PRINT 9001
PRINT 119
C PRINT 2010
CO 2200 I = IJ, I K
2200 PRINT 2110, ICHOSE(I), (NAME(J, I), J=1, 3), NCHOSE(I), REL(I, I), VQL(I),
PWR(I), (COST(J, I), J=1, 4)
11X, F11, 11X, F11, 11X
C PRINT DPI
I J = IK + 1
I K = NEQUIP(3) + IK
C PRINT 9001
PRINT 129
C PRINT 2010
DO 2300 I = IJ, I K
2300 PRINT 2110, ICHOSE(I), (NAME(J, I), J=1, 3), NCHOSE(I), REL(I, I), VQL(I),
PWR(I), (COST(J, I), J=1, 4)
11X, F11, 11X, F11, 11X
C PRINT COMM
I J = IK + 1
I K = NEQUIP(4) + IK
C PRINT 9001
PRINT 146
C PRINT 2010
DO 2400 I = IJ, I K
2400 PRINT 2110, ICHOSE(I), (NAME(J, I), J=1, 3), NCHOSE(I), REL(I, I), VQL(I),
PWR(I), (COST(J, I), J=1, 4)
C PRINT E.P.
I J = IK + 1
I K = NEQUIP(5) + IK
C PRINT 9000
PRINT 162
C PRINT 2010
DO 2500 I=I,J,IK
ICH = ICH0E(I)/100
IF (ICH.EQ.2) REL(I,J) = 0
PRINT 2110, ICH0E(I), (NAME(J,I), J=1,3), NCH0E(I), REL(J,I), VQI(I)
1 PRINT 9001
PRINT 2500
2600 FORMAT(15X,46HEQUIPMENTS USING COST ESTIMATING RELATIONSHIPS)
PRINT 2610
2610 FORMAT(88X,7VEHICLE, 7X, 7VEHICLE, /, 12X, 4HNAME, 21X, 6WEIGHT, 17X, 48
* HD. E. COST T. E. COST PROD. COST ENG. COST)
C I = IK + 1
C PRINT 2620, HATE, (COST(J,I), J=1,4)
C I = I + 1
C PRINT 2630, HARNW, (COST(J,I), J=1,4)
C I = I + 1
C PRINT 2640, THCMW, (COST(J,I), J=1,4)
C I = I + 1
C PRINT 2650, CONVW (COST(J,I), J=1,4)
C I = I + 1
C PRINT 2660, PROPS, (COST(J,I), J=1,4)
C I = I + 1
C PRINT 2670, PASS, (COST(J,I), J=1,4)
C I = I + 1
C PRINT 2680, P0C, (COST(J,I), J=1,4)
C I = I + 1
C PRINT 9000
C ******************************************************************************
** ENQ-OF-PRINT ROUTINE **

RETURN
ENO
SUBROUTINE FILTER(NCONF,ICODE)

FILTER CHECKS FOR COMPATIBLE COMBINATIONS OF CONFIGURATIONS
A MINUS 1 IS RETURNED FOR UNACCEPTABLE COMBINATIONS
ICODE IS CODE RETURNED

DIMENSION NCONF(6)

C CHECKS AND IF (POOTRX .LT. .0 AND NCONF(1) .EQ. 1) ICODE = -1
C IF (POOTRX .LT. .0 AND NCONF(1) .EQ. 2) ICODE = -1
C IF (AMIN(PHIRX,PHIRY,PHIRZ) .LT. .0 AND NCONF(1) .EQ. 2) ICODE = -1
C MANEUVERABILITY IS MANV AND IS VALUES 1-4
C IF (MANV .EQ. 4 AND NCONF(1) .EQ. 1) ICODE = -1
C PAYLOAD YAW IS 0 OR 1
C IF (NCONF(1) .EQ. 4 AND NCONF(3) .EQ. 2) ICODE = -1
C IOPTCM(1) IS RANGING (1-YES), IOPTCM(2) IS SEPARATE LINK, AND
C IOPTCM(3) IS SEPARATE ANTENNA
C IF (IOPTCM(1) .LT. 0 AND NCONF(4) .EQ. 1) ICODE = -1
C IF (NCONF(4) .LE. 3) GO TO 45
DO 43 I=1,11
DO 45 J=1,3
IF (ARRAYN(I,J) .GT. 0) GO TO 45
CONTINUE
ICODE=-1
CONTINUE

43 CONTINUE
ICODE=-1
CONTINUE

45 CONTINUE
IF (NCONF(5) .EQ. 1 AND NCONF(1) .LT. 3) ICODE = -1
IF (NCONF(5) .EQ. 3 AND NCONF(1) .LT. 3) ICODE = -1
IF (NCONF(5) .EQ. 5 AND NCONF(1) .LT. 3) ICODE = -1
IF (NCONF(5) .EQ. 2 AND NCONF(1) .LT. 3) ICODE = -1
RETURN
END
SUBROUTINE READDS

C THIS READS THE DATABASE FOR ONE SUBSYSTEM AT A TIME
C IDDB IS SET AS THE DATABASE IS READ BY SCANNING EQUIP NUMBERS
DIMENSION STORE(55)
COMMON /DBCOM/ DATAB(55,100), IDDB(30)

5 C DATA STORE/5E0.
IF (IDDB .LE. 1) GO TO 2
IF (STORE(I) .EQ. 0.) GO TO 2
DO 1 J=1,55
DATAB(J,I)=STORE(J)
1 GO TO 3
I=1
IDOLD=0
II=DB=1
GO TO 3
2
I=1
IDOLD=0
II=DB=1
3 READ (I,100) (DATAB(J,I), J=1,55)
IF (EOF(I)) 90,110,1
IF (ID .LE. IDCLD) GO TO 60
IF (ID .EQ. IDCLD) GO TO 4
ID=DB=I-1
II=DB+I
IDOLD=ID
GO TO 3
C TEST FOR ENH OF SUBSYSTEM
C IF (ID .LT. IDOLD) GO TO 60
C TEST FOR NEW EQUIP TYPE
C IF (ID .EQ. IDCLD) GO TO 4
ID=DB=I-1
II=DB+I
IDOLD=ID
GO TO 3
C HERE WHEN SWITCHING SUBSYSTEMS
DO 5 J=1,55
STORE(J)=DATAB(J,I)
5 ID=DB=I-1
ENDDB=I-1
RETURN
DO 6 J=1,55
STORE(J)=0
ID=DB=I-1
REWIND
6 J=1
ENDDB=I-1
RETURN
END
SUBROUTINE SAVE(NWAT,TEST,ENDB)

THIS SUBROUTINE SAVES THE CHOSEN PORTIONS OF THE DATABASE REQUIRED BY THE SUBSYSTEMS OR ROUTINES.

DIMENSION II(15), NIN(15)

COMMON/DATCOM/DATAB(55,100), IDB(3)
COMMON/CHOSE/ COST(5,60), NCHOSE(50), REL(6,60), SKO(7,60),
                THM(4,60), THM(2,60), THM(3,60), THM(4,60),
                SATINV, SSREL(6), SATR, SEIP, SEIR,
                SKTAU(6), SUBE(7), SUBT(7), SUBUE(7), SUBUP(7),
                TOTOPS, TRUNC, TVOLUME(6), VQL(60), WEIGHT(6),
                XLTOT, XMEH, XMEINV, XMEL, XMEVL,
                XVEST

DIMENSION IDATAB(55,90)

EQUIVALENCE (IDATAB, DATAB)

DO 1 I=1,TEST
  IF (II(I) .LE. 0) GO TO 1
  ICHOOSE(NWAT)=II(I)
  NCHOSE(NWAT)=I
  DO 3 J=1,ENDB
    IF (DATAB(I*J) .NE. II(I)) GO TO 3
    DATAB(I*J)=I
    DO 2 KKK=1,5
      COST(KKK,NWAT)=DATAB(45+KKK,J)
      REL(1,NWAT)=DATAB(25,J)
      DO 4 KKK=1,5
        REL(KKK,NWAT)=DATAB(34+KKK,J)
        DO 5 KKK=1,11
          THM(1,NWAT)=DATAB(17,J)
          THM(2,NWAT)=DATAB(18,J)
          THM(3,NWAT)=DATAB(27,J)
          THM(4,NWAT)=DATAB(28,J)
          SKO(1,NWAT)=DATAB(49,J)
          SKO(2,NWAT)=DATAB(47,J)
          DO 6 KKK=3,7
            SKO(KKW,NWAT)=DATAB(48+KKK,J)
            VOL(NWAT)=DATAB(24,J)
            TRUNC(NWAT)=DATAB(16,J)
            DO 7 KKK=1,3
              NAME(KKW,NWAT)=DATAB(KKW+2,J)
              CONTINUE
            CONTINUE
          CONTINUE
  CONTINUE
  I CONTINUE

SUBROUTINE SAVCIIbNINNOWATITESTIENDUB)

THIS SUBROUTINE SAVES CHOSEN PORTIONS OF THE DATABASE REQUIRED BY THE SUBSYSTEMS OR ROUTINES.

DIMENSION II(15), NIN(15)

COMMON/DATCOM/DATAB(55,100), IDB(3)
COMMON/CHOSE/ COST(5,60), NCHOSE(50), REL(6,60), SKO(7,60),
                THM(4,60), THM(2,60), THM(3,60), THM(4,60),
                SATINV, SSREL(6), SATR, SEIP, SEIR,
                SKTAU(6), SUBE(7), SUBT(7), SUBUE(7), SUBUP(7),
                TOTOPS, TRUNC, TVOLUME(6), VQL(60), WEIGHT(6),
                XLTOT, XMEH, XMEINV, XMEL, XMEVL,
                XVEST

DIMENSION IDATAB(55,90)

EQUIVALENCE (IDATAB, DATAB)

DO 1 I=1,TEST
  IF (II(I) .LE. 0) GO TO 1
  ICHOOSE(NWAT)=II(I)
  NCHOSE(NWAT)=I
  DO 3 J=1,ENDB
    IF (DATAB(I*J) .NE. II(I)) GO TO 3
    DATAB(I*J)=I
    DO 2 KKK=1,5
      COST(KKK,NWAT)=DATAB(45+KKK,J)
      REL(1,NWAT)=DATAB(25,J)
      DO 4 KKK=1,5
        REL(KKK,NWAT)=DATAB(34+KKK,J)
        DO 5 KKK=1,11
          THM(1,NWAT)=DATAB(17,J)
          THM(2,NWAT)=DATAB(18,J)
          THM(3,NWAT)=DATAB(27,J)
          THM(4,NWAT)=DATAB(28,J)
          SKO(1,NWAT)=DATAB(49,J)
          SKO(2,NWAT)=DATAB(47,J)
          DO 6 KKK=3,7
            SKO(KKW,NWAT)=DATAB(48+KKK,J)
            VOL(NWAT)=DATAB(24,J)
            TRUNC(NWAT)=DATAB(16,J)
            DO 7 KKK=1,3
              NAME(KKW,NWAT)=DATAB(KKW+2,J)
              CONTINUE
            CONTINUE
          CONTINUE
  CONTINUE
  I CONTINUE

END
SUMMARY OF CHANGES MADE BY THE OPTIMIZER

13 WORDS OF INVARIANT RLIST REMOVED FROM THE LOOP BEGINNING AT LINE 33
13 WORDS OF INVARIANT RLIST REMOVED FROM THE LOOP BEGINNING AT LINE 36
13 WORDS OF INVARIANT RLIST REMOVED FROM THE LOOP BEGINNING AT LINE 38
13 WORDS OF INVARIANT RLIST REMOVED FROM THE LOOP BEGINNING AT LINE 46

REGISTER ALLOCATION
1 REGISTERS ASSIGNED OVER THE LOOP BEGINNING AT LINE 33
1 REGISTERS ASSIGNED OVER THE LOOP BEGINNING AT LINE 36
1 REGISTERS ASSIGNED OVER THE LOOP BEGINNING AT LINE 38
1 REGISTERS ASSIGNED OVER THE LOOP BEGINNING AT LINE 46
SUBROUTINE THRML (IERR, NCONF)

COMMON /USER1/ APOGEE, COMRAT, DIAAMAX, EQWT(9), EPME, 022575 315
EGM1W, EGM1XL, EGM2YL, EGM2YL, EGM3W, 317
EGM2XL, EGM2YL, EGM3W, EGM4YL, EGM4YL, 348
IAGNCY, NCONF, ORBING, PERIGE, 349

COMMON /BITWN/ ACSSN, ACSNP, ALT, AREA, BATCAP, 350
BITRAT(2), CLIFE, CONVWT, D, DT, 351
DZ, EOBL, EOBSD, EOBT, EOBSD, 352
HTPI, HTRPRB, HTRPRD, IBLTCL, PJ, 353
LMBDO, LG, LMEES, PASSR, RADA, RADA, 354
PL, PLMT, PGNMT, RADAB, RADAB, 355
RAT, RAJ, RABOLG, SAQ, SAQ, SAQ, 356
SATMI, SATXCG, SATYCG, SATZCG, SAIL, 357
SATYI, SAIL, SAIL, SAIL, SAIL, 358
THRUS(2), TRX, TRX, TRPRB, TRPRB, 359
VCHP, VOL, WITE, MB, WBJ, 360
WBI, XJ, XZERO, YJ, ZJ 361

COMMON /CHOSE/ COST(5,60), DPIA(11,60), ICHOSE(60), 362
NCHOSE(60), REL(6,60), SKD(7,60), 363
THRME(460), NASA 1260

DIMENSION NCONF(6) NASA 1270
REAL LENGTH NASA 1271
DATA SIGMA/0.1744E-06/, QS/44.20/, EMISS/60.0/0, ALBDO/155.0/, CONST/1. 1272
PI/3.1415926535/ 1273

SUBROUTINE THRML USES A METHODOLOGY FOR SIZING THE THERMAL 1274
CONTROL SUB-SYSTEM FOR A VARIETY OF SPACECRAFT. THIS METHODOLOGY 1275
DETERMINES SIZE AND PERFORMANCE OF THE THERMAL SUB-SYSTEM 1276
A GLOSSARY OF VARIABLES FOLLOWS --

** CODE IS AS FOLLOWS --
** U = USER INPUT, DB = DATA BASE, INT = INTERNAL
** O = OUTPUT, I = INPUT FROM MAIN OR OTHER S/S

** Var. Name  Code  Units  Description
** ALBDO  INT  155 BTU/(HR*FT**2)  ALBEDC
** ALPHA  INT  0.30 (DIMENSIONLESS)  CONV.FAD.CONST
** ALT  U  N.MI.  ALTITUDE

C$
** QS INT 442.0 BTU/(HR*FT**2) SOLAR CONST.**

** RADA 0 (FT**2) RADIATOR AREA**

** RADAB 0 FT**2 BATT.RAD.AREA**

** RAT 0 FT**2 TOTAL RAD. AREA**

** SATLG I (VS) INCHES SAT. LENGTH**

** SATRAD I (VS) INCHES SAT. RADIUS**

** SIGMA INT 0.1714E-8 BTU/(HR*FT*2*R**4) BOLTZMANN CONST**

** TLOAD I THERMAL DATA BASE**

** Tmax INT (DEG) DEGREES R MAX TEMPERATURE**

** TMAXB INT**

** TMIN INT (DEG) DEGREES R MIN TEMPERATURE**

** THMINT**

** VCHP 0 VAR. COND. HEAT PIPE**

** THRHOB I THERMAL DATA BASE**

** TMAX INT (DEG) DEGREES R MAX TEMPERATURE**

** TMAXB INT**

** TMIN INT (DEG) DEGREES R MIN TEMPERATURE**

** THMINT**

** VCHP 0 VAR. COND. HEAT PIPE**

** IERR = X X X X X X X X X X X**

WHERE 0 MEANS NO SUCH REQUIREMENT, OR 1 MEANS WE HAVE THIS REQUIREMENT.
**INITIALIZE FOLLOWS --**

RAOA=0.
RAOB=0.
RAT=0.0
HTPRM=0.
HTPRB=0.
HPTP=0.
HPTPE=0.
VCHP=0.
HPT=0.
TMAX=1.0E+20
TMIN=-1.0E20
PMAX=0.
FMIN=0.0
ETAT=1.0
I=0.
SATRA=0.5*0.
TMAX=I*E20
THIN=-I.E20
PNAX=G.
FNIN=0.0
ETAT=I.0
I=0.
BD
I0 I:1+t
C
SATRA=.4
C
SEARCH FOR MIN(TEMP) AND MAX(MIN TEMP), AND
ACCUMULATE THE POWER (EXCLUDING XMTS AND BATTERIES)

IF (ICHOOSE(I).LE.0) GO TO 50
IF (I.EQ.IBTLOC) GO TO 20

TNAX=AMINI(TMTHRDB(3,I))
TMIN=AMAXI(TMIN,THRMDB(4,I))
PMIN=THRMDB(2,I)+PMIN
PMAX=THRMDB(1,I)+PMAX
GO TO 10

HERE IF WE HAVE THE BATTERY

TNAX=THRMDB(4,I)+60.
TNAX=THRMDB(3,I)+60.

GO TO 10

50 CONTINUE
QMIN=PMAX*3.41
IF (PMAX*3.6GT. PMIN) PMAX=PMAX*5.
QMAX=PMAX*3.41
TMAX=THRMDB(460.
TMIN=THRMDB(460.
IGNF=NGCONF(6)

GOTO (60,70,30), ICCNF

**SAT TEL LITE LENGTH IN INCHES (MUST CONVERT TO CM) (FROM VS)
(CYLINDER)**

LENGTH=SATLG*2.54*0.75
GO TO 90
SATELLITE LENGTH IN INCHES (MUST CONVERT TO CM) (FROM VS) ** NASA 1469
LEN=SATL*2.54*0.75 NASA 1470
GO TO 90 NASA 1471

SATELLITE LENGTH IN INCHES (MUST CONVERT TO CM) (FROM VS) ** NASA 1472
LEN=PIE*SATRAD*2.54 NASA 1473
GO TO 90 NASA 1474

CONTINUE NASA 1475
IF (ALT.GT.19000.) GO TO 300 NASA 1476
IF (ALT.LT.50.) GO TO 160 NASA 1477
GO TO (130,160,100), ISATOR NASA 1478
ICN=ICN+1 NASA 1479
GO TO (120,120,110,110,110), ICONF NASA 1480

** ORBITS GT 500 BUT LT 19000 AND, ** NASA 1481
** SOLAR ORIENTED AND, ** NASA 1482
** 3-AXIS STABILIZED  (EQUATION 3.3.1.1) ** NASA 1483

ALPHA=0.30 NASA 1484
EPSL=0.75 NASA 1485

* DETERMINE RADIATOR AREA NASA 1486
RADA=QMAX/(SIGMA*EPSLON*TMAX**4-(EMISS*EPSLON)) NASA 1487

* DETERMINE HEATER POWER NASA 1488
HTRPWR=1.25*((SIGMA*EPSLON*RADA*TMIN**4)-(QMIN)-(EMISS*EPSLON)) NASA 1489

* DETERMINE HEAT PIPE NASA 1490
HTPIPE=(QMAX*LEN)/3.41 NASA 1491
IERR=10111 NASA 1492
GO TO 380 NASA 1493

** ORBITS GT 500 BUT LT 19000 AND, ** NASA 1494
** SOLAR ORIENTED AND, ** NASA 1495
** SPIN STABILIZED  (EQUATION 3.3.1.2) ** NASA 1496

ALPHA=0.30 NASA 1497
EPSL=0.75 NASA 1498

* DETERMINE RADIATOR AREA NASA 1499
RADA=QMAX/(SIGMA*EPSLON*TMAX**4-(EMISS*EPSLON)) NASA 1500

* DETERMINE HEATER POWER NASA 1501
HTRPWR=1.25*((SIGMA*EPSLON*RADA*TMIN**4)-(QMIN)-(EMISS*EPSLON)) NASA 1502

* DETERMINE HEAT PIPE NASA 1503
HTPIPE=(QMAX*LEN)/3.41 NASA 1504

IERR=10111
GO TO 360

C
130 ICONF=NCONF(1)
GO TO (140,140,150,150,150), ICONF

C
290 ** ORBITS GT 500 BUT LESS THAN 19000 AND,
** EARTH ORIENTED AND,
** SPIN STABILIZED (EQUATION 3.4.1.2)

140 ALPHA=0.08
EPSLON=0.73

C
295 * DETERMINE RADIATOR AREA
RADA=QMAX/((SIGMA*EPSLON*TMAX**4)-(QS*ALPHA))

C
300 * DETERMINE HEATER POWER
HTRPWR=1.25*((SIGMA*EPSLON*RADA*TMIN**4)-(QMIN))

C
305 * DETERMINE PCM
PCM=(0.26*ALPHA*QS*RADA*CONST)/40.

C
310 * DETERMINE HEAT PIPE
HTPIPE=(QMAX*LNGTH)/3.41

C
315 IERR=10011011
GO TO 360

C
320 ** ORBITS GT 500 BUT LESS THAN 19000 AND,
** EARTH ORIENTED AND,
** 3-AXIS STABILIZED (EQUATION 3.4.1.1)

150 ALPHA=0.08
EPSLON=0.73

C
325 * DETERMINE RADIATOR AREA
RADA=QMAX/((SIGMA*EPSLON*TMAX**4)-(QS*ALPHA))

C
330 * DETERMINE HEATER POWER
HTRPWR=1.25*((SIGMA*EPSLON*RADA*TMIN**4)-(QMIN))

C
335 * DETERMINE PCM
PCM=(0.26*ALPHA*QS*RADA*CONST)/40.

C
335 * DETERMINE HEAT PIPE
HTPIPE=(QMAX*LNGTH)/3.41

C
IERR=10011011
GO TO 360
IF (ORBINC.GT.30.) GO TO 230
GO TO (170,200,200), ISATOR
ICONF=CONF(1)
GO TO (180,180,190,190,190), ICONF
** ORBIT LT 500, ORBITAL INCLINATION LE 30 DEGREES AND, **
** EARTH ORIENTED AND, **
** SPIN STABILIZED (EQUATION 2.1.2.2) **
ALPHA=0.08 EPSLON=0.73
* DETERMINE RADIATOR AREA
RADA=QMAX/((SIGMA*EPSLON*TMAX**4)-(EMISS*EPSLON/PIE)-(QS+ALBDO)*ALPHA/PIE)
HTRPWR=1.25*((SIGMA*EPSLON*RADA*TMIN**4)-(QMIN)-((EMISS*EPSLON)/PIE))
IERR=I011 GO TO 380
** ORBIT LT 500, ORBITAL INCLINATION LE 30 DEGREES AND, **
** EARTH ORIENTED AND, **
** 3-AXIS STABILIZED (EQUATION 2.1.2.1) **
ALPHA=0.08 EPSLON=0.73
* DETERMINE RADIATOR AREA
RADA=QMAX/((SIGMA*EPSLON*TMAX**4)-(ALPHA*QS))
DTERMINE HEATER POWER
HTRPWR=1.25*((SIGMA*EPSLON*RADA*TMIN**4)-QMIN)
* DETERMINE PCM MASS
PCM=(0.26*QS*RADA*ALPHA*CONST)/40.
* DETERMINE ISOETHERMALIZER HEAT PIPE
HTPPIPE=(QMAX*LNGTH)/3.41
IERR=11011011 GO TO 380
200  ICONF=NCONF(1)
      GO TO (210,210,220,220,220), ICONF

400  ** ORBIT LT 500, ORBITAL INCLINATION LE 30 DEGREES AND,  **
    ** SUN ORIENTED AND,  **
    ** SPIN STABILIZED  **
      (EQUATION 2.1.1.2)

405  210  ALPHA=0.08
      EPSLON=0.73
      C  DETERMINE RADIATOR AREA

410  RADA=QMAX/((SIGMA*EPSLON*TMAX**4)-(EMISS*EPSLON)-(0.5*ALBDO*ALPHA))
      C  DETERMINE HEATER POWER

415  HTRPWR=1.25*((SIGMA*EPSLON*RADA*TMIN**4)-QMIN)
      C  DETERMINE HEAT PIPES

420  HTPipe=(QMAX*LNGTH)/3.41
      C  IERR=10011
      GO TO 380

425  ** ORBIT LT 500, ORBITAL INCLINATION LE 30 DEGREES AND,  **
    ** SUN ORIENTED AND,  **
    ** 3-AXIS STABILIZED  **
      (EQUATION 2.1.1.1)

430  220  ALPHA=0.08
      EPSLON=0.73
      C  DETERMINE RADIATOR AREA

435  RADA=QMAX/((SIGMA*EPSLON*TMAX**4)-(EMISS*EPSLON)-(ALBDO*ALPHA))
      C  DETERMINE HEATER POWER

440  HTRPWR=1.25*((SIGMA*EPSLON*RADA*TMIN**4)-QMIN)
      C  DETERMINE HEAT PIPES

445  HTPipe=(QMAX*LNGTH)/3.41
      C  IERR=10011
      GO TO 380

450  WHERE IF ORBITNG GT 30
      GO TO (240,270,270), ISATOR

450  240  ICONF=NCONF(1)
      GO TO (250,250,260,260,260), ICONF
** ORBIT LT 500, ORBITAL INCLINATION GT 30 DEGREES AND, **

** EARTH ORIENTED AND, **

** SPIN STABILIZED **

(EQUATION 2.2.3.2)

C

** ORBIT LT 500, ORBITAL INCLINATION GT 30 DEGREES AND, **

** EARTH ORIENTED AND, **

** 3-AXIS STABILIZED **

(EQUATION 2.2.3.1)

C

** ORBIT LT 500, ORBITAL INCLINATION GT 30 DEGREES AND, **

** SUN ORIENTED AND, **

** SPIN STABILIZED **

(EQUATION 2.2.2.2)

C

** ORBIT LT 500, ORBITAL INCLINATION GT 30 DEGREES AND, **

** SUN ORIENTED AND, **

** SPIN STABILIZED **

(EQUATION 2.2.2.2)
* DETERMINE RADIATOR AREA
RADA=QMAX/((SIGMA*EPSLCN*TMAX**4)-(EMISS*EPSLN)-(ALBDC*ALPHA))

* DETERMINE HEATER POWER
HTRPWR=1.25*((SIGMA*EPSLN*RADA*TMIN**4)-QMIN-(EMISS*EPSLN))

* DETERMINE PCB MASS
FCM=(0.26*ALPHA*ALBDO*RADA*CONST)/40.

* DETERMINE HEAT PIPES
HTPIPE=(QMAX*LNGTH)/3.41
IERR=10011011
GO TO 380

** ORBIT LT 500, ORBITAL INCLINATION GT 30 DEGREES
** SUN ORIENTED AND,
** 3-AXIS STABILIZED (EQUATION 2.2.2.1)

** HERE IF ORBIT GT 19000
GO TO (340,310,310), ISATOR
ICONF=ICONF(1)
GO TO (320,320,330,330,330), ICONF

** ORBIT GT 19000 AND
** SOLAR INERTIALLY ORIENTED AND,
** SPIN STABILIZED (EQUATION 1.1.1.2)
ALPHA=0.30
EPSLON=0.75

* DETERMINE RADIATOR AREA
RADA=QMAX/(SIGMA*EPSLON*TMAX**4)

* DETERMINE HEATER POWER
HTRPWR=1.25*((SIGMA*EPSLON*RADA*TMIN**4)-QMIN)

* DETERMINE HEAT PIPES
HTPIPE=(QMAX*LENGTH)/3.41

IERR=10111
GO TO 380

** ORBIT GT 19000 AND
** SOLAR INERTIALLY ORIENTED AND
** 3-AXIS STABILIZED (EQUATION 1.1.2)

* DETERMINE RADIATOR AREA
RADA=(2.*QMAX)/(SIGMA*EPSLON*TMAX**4)

* DETERMINE HEATER POWER
HTRPWR=1.25*((SIGMA*EPSLON*RADA*TMIN**4/2.)-QMIN)

* DETERMINE DIODE HEAT PIPE (2 REG-D)
HTPIPE=(QMAX*LENGTH)/3.41

IERR=110111
GO TO 380

ICCNF=NCNF(1)
GO TO (350,360,370,370,360), ICNF

** ORBIT LT 19000 AND
** EARTH ORIENTED AND
** DUAL OR NORMAL SPIN STABILIZED (EQUATION 1.2.3)

* DETERMINE RADIATOR AREA
RADA=QMAX/(SIGMA*EPSLON*TMAX**4)

* DETERMINE HEATER POWER
C HTRPWR=1.25*((SIGMA*EPSLON*RADA*TMIN**4)-QMIN)
C * DETERMINE HEAT PIPES
C HPIPE=(QMAX*LENGTH)/3.41
C IERR=10111
GOTO 380

C *** ORBIT GT 19000 AND ** YAW SPIN STABILIZED
T(EQUATION 1.2.2)
C
C 360 ALPHA=0.08
C EPSLON=0.73
C
C * DETERMINE RADIATOR AREA
C RADA=QMAX/((SIGMA*EPSLON*TMAX**4)-(QS*ALPHA/PIE))
C
C * DETERMINE HEATER POWER
C HTRPWR=1.25*((SIGMA*EPSLON*RADA*TMIN**4)-QMIN)
C IERR=10111
GOTO 380

C *** ORBIT GT 19000 AND ** EARTH ORIENTED AND **
T(EQUATION 1.2.1)
C 370 ALPHA=0.30
C EPSLON=0.75
C
C * DETERMINE RADIATOR AREA
C RADA=(2.*QMAX)/(SIGMA*EPSLON*TMAX**4)
C
C *** DETERMINE HEATER POWER
C HTRPWR=1.25*((SIGMA*EPSLON*RADA*TMIN**4/2.)-QMIN)
C
C * DETERMINE DIODE HEAT PIPE (2 REG-D)
C HPIPE=(QMAX*LENGTH)/3.41
C IERR=11011
GOTO 380

C *** HERE WE WILL SIZE THE BATTERY THERMAL CONTROL NETWORK

C 380 CA=.5
C BV=1.5
C ALPHA=0.08
C EPSLON=0.73
QMAXB=NC*CA*9V*3.41
QMINB=0.

* DETERMINE RADIATOR AREA FOR BATTERY
RADAB=QMAXB/((SIGMA*EPSLON*(TMAXB-30.)*4)-(QS*ALPHA))

* DETERMINE HEATER POWER FOR BATTERY
HTRPRB=1.25*(SIGMA*EPSLON*RADAB*(TMINB)**4-QMINB)

* DETERMINE VARIABLE CONDUCTANCE HEAT PIPE
VCHP=QMAXB*LNGTH/3.41
IERR=IERR+1100000000

HTPIPE = HTPIPE / 2.54
VCHP = VCHP / 2.54
RAT=RADA+RADAB
HPT=HTRPWR+HTRPRB
HTPT=HTPIPE+VCHP

RETURN

END
55
PM(1) = 0.
PM(2) = 0.
ISPFLG(1) = 0.
ISPFLG(2) = 0.

3 IF (ITER.EQ.0 .AND. IPIC(4).EQ.0) IC=0
NCNF = MCONF(4)
DO 1 I=1,2
1 BITRAT(I) = .011 * BITRAT(I)
SSS=0.
IF (MCONF(1).EQ.1) SSS = 1
INX=1.
IF (NCONF.EQ.4 .OR. NCONF.EQ.5) INX=2.
C INITIALIZATION OF IPIC AND ICHOOSE.
IC=IC+1.
IF (ITER.NE.0) IC=1.
IF (ITER.EQ.0 .AND. IPIC(4).GT.0) IC=1.
DO 10 I=ICC,9
10 KCHOOSE(I)=0.
DO 20 I=1,9
KPIC(I)=IPIC(I).
20 IF (ITER.EQ.0 .AND. IPIC(4).EQ.0) KPIC(I)=RESET(I)
F1=0.
IF (NCONF.EQ.2 .OR. NCONF.EQ.4) F1=1.
IF (F1.EQ.1) KPIC(6)=0.
IF (F1.EQ.0) KPIC(6)=0.
IF (ITER.NE.0) I=ICC,9.
IF (IC.NE.0) GO TO 700.
LIMPIC(1)=IDB(1).
LIMPIC(2)=IDB(2).
LIMPIC(3)=IDB(3).
LIMPIC(4)=IDB(4).
LIMPIC(5)=IDB(5).
LIMPIC(6)=IDB(6).
CALL BESS(BETA,BESSJ,1)
CONTINUE.
30 IF (NCONF.GE.4 .OR. BITRAT(2).EQ.0) GO TO 40
BITRAT(1)=(BITRAT(1)+BITRAT(2))*1.3.
40 CONTINUE.
RATE1=IBER(1).
RATE2=0.
DO 50 I=1,13.
50 IF (BITRAT(1).GT.IBER(1)) RATE1=IBER(1+1).
IF (BITRAT(2).GT.IBER(1)) RATE2=IBER(1+1).
CONTINUE.
100 IF (NCNF.EQ.1) GO TO 90
IERR=1.
IF (LINK.EQ.0) GO TO 770.
C SGLS BBAU SELECTED
IERR=0.
IC=1.
C ONE HOUSKEEPING BIT STREAM ONLY (THIS SEMESTER)
C IS SGLS 2 IS USB
60 IF (DATAB(IB1, J1).EQ.1) GO TO 70
   J1=J1+1
   IF (J1.GT.IDB(1)) GO TO 760
   GO TO 60
70 IF (BITRAT(I).GT. .28.) GO TO 80
   IF (ABS(DATAB(IB2, J1) - 1.024) *LT. .01) J=J1+1
   IF (J1.GT.IOB(I)) GO TO 760
   GO TO 60
80 IERR=2
   IF (BITRAT(I), GT. ±28.) GO TO 770
   IERR=0
120 IF (OATAB(I2, J1).EQ.1) GO TO 760 
   GO TO 60
C END OF BBAU SELECTION
130 IC=2
   KXMTR=1
   GO TO 110
100 IC=3
   KXMTR=2
   GO TO 120
135 CONTINUE
C ANTENNA SELECTION
J2=KPIC(IDC)
   IF (ISFFLG(KXMTR), .EQ. 1) GO TO 251
   IF (SEC.GT. 1) GO TO 250
   IF (ALT.GT.12000) GO TO 140
   IF (ALT.GT.12000) GO TO 130
   IC=2
   KXMTR=1
   GO TO 110
140 IC=3
   KXMTR=2
135 CONTINUE
C OMNI (8 OMNI)
120 IF (OATAB(I2, J1).EQ.11) GO TO 299
   J2=J2+1
   IF (J2.GT.IOB(2)) GO TO 760 
   GO TO 120
140 IERR=30
   IF (ALT.GT. 20000.) GO TO 770
   IERR=0
C CONICAL SPIRAL (F2)
160 IF (OATAB(I2, J1).EQ.13) GO TO 299 
   J2=J2+1
   IF (J2.GT.IOB(2)) GO TO 760
   GO TO 160
C MONOPOLE (F1)
180 IF (OATAB(I2, J1).EQ.15) GO TO 299
   J2=J2+1
   IF (J2.GT. IOB(2)) GO TO 760
   GO TO 180
200 IF (GRP.EQ.0.) GO TO 180
C END OF BBAU SELECTION
130 IC=2
   KXMTR=1
   GO TO 110
100 IC=3
   KXMTR=2
   GO TO 120
135 CONTINUE
C MONOPOLE (F1)
200 IF (OATAB(I2, J1).EQ.51) GO TO 299
   J2=J2+1
   IF (J2.GT.IOB(2)) GO TO 760
   GO TO 200

NASA 2051
NASA 2052
NASA 2053
NASA 2054
NASA 2055
NASA 2056
NASA 2057
NASA 2058
NASA 2059
NASA 2060
NASA 2061
NASA 2062
NASA 2063
NASA 2064
NASA 2065
NASA 2066
NASA 2067
NASA 2068
NASA 2069
210 IF (ALT GT 12000.) GO TO 230

C HELIX (F3)
220 IF (DATAB(IA1,J2) EQ .31.) GO TO 299
J2=J2+1
IF (J2 GT IDB(2)) GO TO 760
GO TO 220
230 IERR=30
IF (ALT GT 20000.) GO TO 770
IERR=0

C PARABOLA (B HIGH GAIN)
240 IF (DATAB(IA1,J2) EQ .1.) GO TO 299
J2=J2+1
IF (J2 GT IDB(2)) GO TO 760
GO TO 240

C STEERABLE PARABOLA OPTION WILL BE INCLUDED NEXT SEMESTER
250 IF (ISPFLG(KXMTR) EQ .1) GO TO 251
IF (BITRAT(KXMTR) LT 10) GO TO 120
251 IERR=1001
255 IF (DATAB(IA1,J2) EQ .1.) GO TO 299
J2=-J2+1
IF (J2 GT IDB(2)) GO TO 760
GO TO 255

KXMTR=1
GO TO 280
KXMTR=2
GO TO 280
CONTINUE
LUNI=0

IF (NCONF GT .1 AND KXMTR EQ 1) LUNI=1

C SPACE LOSS
IF (SLANT EQ -1.E+10) SLANT=SQRT(APOGEE*APOGEE+6880)
SLOSS=37.6*E20*ALOG10(FREQ(KXMTR)*SLANT)

C G TO T
IF (GTOT NE .1.E+10) GO TO 320
IF (GR NE .1.E+10 AND T NE .1.E+10) GO TO 310
IF (GR NE .1.E+10 AND GR NE .1.E+10) GO TO 300

C NET EQ .0 FOR AFSOL NET NE .0 FOR NASA
IF (NET EQ .0) GO TO 290
GR=46
T=110
GO TO 310

GR=47.5
T=120
GO TO 310

GT(KXMTR) = DATAB(IT1,J2)
GO TO 690

T=(*10.**(NF/10)-1)*290.

GO TO 320
CONTINUE

C TRANSMITTER CIRCUIT LOSS
IF (TCLOSS(KXMTR) NE .0) GO TO 330
TCLOSS(KXMTR)=1.0
IF (LUNI.EQ.1.AND.(NCONF.EQ.2.OR.NCONF.EQ.4)) TCLOSS(KXMTR)=1.5
330 CONTINUE

C MODULATION LOSS
MODLOS=0
IF (LUNI.EQ.0) GO TO 340
IERR=10
IF (LINK.EQ.0) GO TO 770
IERR=0
C BESSJ(2)=J1(BETA) / BESSJ(1)=J0(BETA)
MODLOS=ABS(ALOG10(2*BESSJ(2)*COS(GAMMA))**2))
340 CONTINUE

C HARDWARE DEGRADATION LOSS
IF (LUNI.EQ.1) GO TO 360
DO 350 I=1,14
IF (BITRAT(KXMTR).EQ.I) MODLOS=MODLOS+BER(I.3)
350 CONTINUE

C BANDWIDTH IN DB
IF (BITRAT(KXMTR).EQ.0) BWIDTH(KXMTR)=BITRAT(KXMTR)*1000
B=10*ALOG10(BWIDTH(KXMTR))
C CALCULATION OF ERP
ERP=SIGNOI(KXMTR)+SLOSS+B-GTOT+LMARG(KXMTR)+TCLOSS(KXMTR)+POLLOSS
1NTLOSS+MODLOS+HLOSS=2.28-6
PW(KXMTR)=10.**((ERP-GET(KXMTR))/10.)
PRINT 385, KXMTR, PW(KXMTR)
385 FORMAT(4H PW(KXMTR),4H =1E11.4)
C TRANSMITTER SELECTION
IF (PW(KXMTR).LT.20.6) GO TO 388
265 IF (PW(KXMTR).LT.20.6) GO TO 388
IC = KXMTR + 1
ISPFLG(KXMTR) = 1
IF (J2SAVE(KXMTR).GT.0) J2 = J2SAVE(KXMTR)
GO TO 250
388 KNSTRA = IC
IC=KXMTR+3
J3=KPIC(IC)
390 IF (LUNI.EQ.0) GO TO 440
IF (DATAB(I1.I2).EQ.1.) GO TO 388
J3=J3+1
390 IF (LUNI.EQ.0) GO TO 440
400 IF (DATAB(I1.I2).EQ.1.) GO TO 410
J3=J3+1
J3=J3+1
410 IF (DATAB(I3.I1).EQ.0) GO TO 760
GO TO 420
410 IF (DATAB(I3.I1).EQ.0) GO TO 760
KNSTRA=1
420 IF (DATAB(I3.I1).EQ.0) GO TO 760
255 IF (BWIDTH(KXMTR).EQ.0) GO TO 388
6=10*ALOG10(BWIDTH(KXMTR))
GO TO 400
285 IF (DATAB(IT2,J3).EQ.0) GO TO 430
J3=J3+1
IF (J3.GT.IODB(3)) GO TO 760
GO TO 400
430 IF (DATAB(IB2,J1).EQ.DATAB(IT3,J3).OR.DATAB(IB2,J1) .GT. DATAB(IT4,J3)) GO TO 460
J3=J3+1
IF (J3.GT.IODB(3)) GO TO 760
GO TO 400
440 CONTINUE
C NON UNIFIED TRANSMITTER
295 CONTINUE
450 IF (BITRT(KXMTR)/1000.LE.DATAB(IT6,J3)) GO TO 460
J3=J3+1
IF (J3.GT.IODB(3)) GO TO 760
GO TO 450
460 IF (LINK.EQ.0) GO TO 470
IF (DATAB(IT5,J3).EQ.1.) GO TO 470
J3=J3+1
IF (J3.GT.IODB(3)) GO TO 760
GO TO 440
C FREQUENCY, POWER, AND MODULATION COMPATIBILITY
470 IF (FREQX(KXMTR).GE.DATAB(IT7,J3).AND.FREQX(KXMTR).LE.DATAB(IT8,J3)) GO TO 480
J3=J3+1
IF (J3.GT.IODB(3)) GO TO 760
GO TO 390
480 IF (PMX(KXMTR).LE.DATAB(IT9,J3)) GO TO 490
J3=J3+1
IF (J3.GT.IODB(3)) GO TO 760
GO TO 390
490 IF (LINK.EQ.1) MODX(KXMTR) = 1
IF (MODX(KXMTR).EQ.0) GO TO 500
IF (MODX(KXMTR).EQ.0) GO TO 500
J3=J3+1
IF (J3.GT.IODB(3)) GO TO 760
GO TO 390
C MODULATION PHASE=1, FREQUENCY=2, AMPLITUDE=3
500 GO TO 690
510 CONTINUE
C REceiving Antenna Selection ****************************************************
IC=6
325 IF (F1.EQ.0) GO TO 690
IF (SSS.EQ.1) GO TO 520
IF (SSS.EQ.0) GO TO 530
330 IF (DATAB(I1,J7).EQ.11) GO TO 650
J7=J7+1
IF (J7.GT.IODB(2)) GO TO 760
GO TO 520
530 IF (NADIR.EQ.0) GO TO 550
540 IF (DATAB(I1,J7).EQ.41) GO TO 690
J7=J7+1
IF (J7.GT.IODB(2)) GO TO 760
GO TO 540
550 IF (GRF.EQ.0) GO TO 540
IF (OATAB(IA1,J7).EQ.51) GO TO 690
J7=J7+1
IF (J7.GT.IDB(2)) GO TO 760
GO TO 560
C END RECEIVER ANTENNA SELECTION
C RECEIVER SELECTION
IC=7
IERR=10
IF (LINK.EQ.0) GO TO 770
IERR=0

IF (DATAB(IR1,J4).EQ.I) GO TO 590
J4=J4+1
IF (J4.GT.IDB(4)) GO TO 580
GO TO 560
IF (DATAB(IR2,J4).GE.COMRAT) GO TO 690
J4=J4+1
IF (J4.GT.IDB(4)) GO TO 580
GO TO 560
IC=8
C COMMAND SIGNAL CONDITIONER
C RECEIVER CONSTRAINT TESTED
C LINK SGLS OR USB
J6=J6+1
IF (J6.GT.IDB(6)) GO TO 610
GO TO 600
C DIPLEXER SELECTION
IC=9
IF (F1.EQ.0) GO TO 690
C LINK SGLS OR USB
J7=KPIC(4)
IF (DATAB(IO2,J6).GE.DATAB(IT9, JT)) GO TO 690
J6=J6+1
IF (J6.GT.Datab(6)) GO TO 690
J6=J6+1
IF (IN.NE.0) KCHOSE(IC)=Datab(1,IN)
WRITE (6,2000) IN, KCHOSE(IC), IC
2000 FORMAT (4H 690, 3110)
700 CONTINUE
IF (IC.GE.9 .AND. ITER.NE.0) GO TO 740
IF (IC.GE.9 .AND. IPIC(I) .EQ. 0) GO TO 740
IF (IN .EQ. 0) GO TO 710
KPIC(IC) = IC
IC = IC+1
GO TO (30, 90, 100, 260, 270, 510, 570, 600, 660), ICX
710 ICX=ICX+1
GO TO (30, 90, 100, 260, 270, 510, 570, 600, 660), ICX
410 IF (IC.EQ.0 .OR. IC.EQ.3) KPIC(IC)=J2
IF (IC.EQ.4 .OR. IC.EQ.5) KPIC(IC)=J3
CONTINUE
415 WRITE (6,2000) IN, KCHOSE(IC), IC
420 IF (IC.GE.9 .AND. ITER.NE.0) GO TO 740
IF (IC.GE.9 .AND. IPIC(I) .EQ. 0) GO TO 740
IF (IC.GE.9 .AND. IPIC(I) .EQ. 0) IC=IC+1
ICX=ICX+1
GO TO (30, 90, 100, 260, 270, 510, 570, 600, 660), ICX
720 IF (KPIC(II).EQ.0) GO TO 725
KPIC(II)=RESET(II)
CONTINUE
730 CONTINUE
740 IF (J7.GT.1) GO TO 752
IF (J7.GT.1) KCHOSE(1).NE.-1.
CONTINUE
750 CONTINUE
760 IF (J7.GT.1) GO TO 752
CONTINUE
770 CONTINUE
ICHOOSE(J) = KCHOOSE(I)
CONTINUE
ICHOOSE(ICONV) = 0
ICHOOSE(ICONV+1) = 0
ID86 = ID86(6)
DO 730 I = 1,9
    DO 730 J = I, ID86
    IF (DATAB(1, J) .NE. ICHOOSE(I)) GO TO 780
    IF (DATAB(22, J) .EQ. 0.) GO TO 780
    IF (ICHOOSE(ICONV) .EQ. DATAB(22, J) .OR. ICHOOSE(ICONV+1) .EQ. 
        * DATAB(22, J)) GO TO 780
    IF (ICHOOSE(ICONV) .NE. 0) GO TO 781
    ICHOOSE(ICONV) = CATAB(22, J)
    GO TO 780
781 CONTINUE
ICHOOSE(ICONV+1) = DATAB(22, J)
780 CONTINUE
DO 756 J = 1, ID87
    DO 756 I = 1, 11
    DO 756 J = 1, ID87
    IF (DATAB(1, J) .NE. ICHOOSE(I)) GO TO 756
    IF (ICHOOSE(I) .GE. ICONV) 
        CONVT = CONVT + DATAB(23, J) * NCHOOSE(I)
        WT = WT + DATAB(22, J) * NCHOOSE(I)
        VOL = VOL + DATAB(24, J) * NCHOOSE(I)
        PL = PL + DATAB(16, J) * NCHOOSE(I)
        PLMIN = PLMIN + DATAB(16, J) * NCHOOSE(I)
    GO TO 757
756 CONTINUE
CCONTINUE
C 757 CONTINUE
C 757 CONTINUE
WRITE (6, 3000) ICHOOSE
3000 FORMAT (4H757, 11I10)
RETURN
CONTINUE
760 CONTINUE
IF (IC .EQ. 2 .OR. IC .EQ. 3) KPIC(IC) = J2
IF (IC .EQ. 4 .OR. IC .EQ. 5) KPIC(IC) = J3
KONT = 1
GO TO 710
CONTINUE
770 CONTINUE
WRITE (6, 4000) KCHOOSE
4000 FORMAT (4H770, 11I10)
ICHOOSE(1) = -1
RETURN
END

REGISTER ALLOCATION
1 REGISTERS ASSIGNED OVER THE LOOP BEGINNING AT LINE 61
SUBROUTINE BESS (X, BESJ, NMAX)

DIMENSION BESJ(1), TJ(200)
EULER=0.577215664901533
PI=2.0/3.141592653589793

NU22=20

10 IF (10,-X) 10,10,20

10 HATN=3.5*(X+25.)

GO TO 30

20 HATN=35.*(3.5-ALOG(X))

30 NU=HATN

TJ(NU+1)=0.0

DO 40 J=1,NU

K=NU+1-J

40 TJ(K)=FK*TJ(K+1)/X-TJ(K+2)

SUM=0.0

DO 50 J=3,NU,2

SUM=SUM+TJ(J)

50

SUM=SUM

TK=1./(TJ(1)+SUM)

END
INTEGER FUNCTION RESET(K)

COMMON /DBCOM/DATAB(55,100),ID8(30)

C

IF (K.EQ.1) RESET=1
IF (K.EQ.2 .OR. K.EQ.3 .OR. K.EQ.6) RESET=IDB(1)+1
IF (K.EQ.4 .OR. K.EQ.5) RESET = IDB(2) + 1
IF (K.EQ.7 .OR. K.EQ.8 .OR. K.EQ.9) RESET = ID8(K-4) + 1
RETURN
END
IC=IC+2+MCONF(IS)
R=REDUN*1.125
TSUB(3)=CONF(IC,2)+CONF(IC,5)*R+CONF(IC,1)*C**.6667
TSUB(4)=CONF(IC,3)+CONF(IC,4)*R*C
TSUB=TSUB(2)

IF (TSUB(2) .LT. TSAVE(3)+TSUB(4)) TSDQ=TSUB(3)+TSUB(4)
TSUB(5)=CONF(IC,1)*FK*TSLB(3)
TSUB(6)=TSUB(1)+TSDQ
DO 10 J=1,6
10 TAU(J)=TSUB(J)

IF (TSUB(5) .LT. TSAVE(5)) TSAVE(5)=TSUB(5)
IF (TSUB(5) .GT. TSAVE(5)) TSAVE(5)=TSUB(5)
DO 3 J=1,6
3 TSAVE(J)=TSUB(J)

ISSAVE=IS
CONTINUE
NOW DO MISSION EQUIP
DO 5 J=1,3
DO 5 I=1,7
5 CONTINUE
J=0
6 IF (J .EQ. 0) GO TO 20
TSUB(1)=0.
TSUB(2)=0.
C=0.
DO 8 J=1,7
TSUB(3)=CONF(22,2)+CONF(22,5)*CONF(22,1)*C**.6667
TSUB(4)=CONF(22,3)+CONF(22,4)*C
TSUB(5)=CONF(22,2)+CONF(22,5)*CONF(22,1)*C**.6667
TSUB(6)=TSUB(1)+TSDQ
DO 11 J=1,6
11 TAU(J)=TSUB(J)

IF (TSUB(5) .LT. TSAVE(5)) TSAVE(5)=TSUB(5)
IF (TSUB(5) .GT. TSAVE(5)) TSAVE(5)=TSUB(5)
DO 9 J=1,6
9 TSAVE(J)=TSUB(J)
20 TSAVE(6)=TSAVE(6)+TSAVE(5)
RETURN
END
SUBROUTINE SANDG(IPIC, IERR, ITER, NCONF, ICHOOSE, NCHOSE) NASA 2552
ICHOOSE(10) IS SELECTED EQUIP AS FOURTH EQUIP = -- MANF = NASA 2553
ICHOOSE IS CONFIGURATION NUMBER, ITER IS NUMBER OF THIS ITERATION NASA 2554
IERR IS A MULTIPLE MESSAGE ERROR FLAG, IPIC IS THE LAST NASA 2555
SET OF SUBSCRIPTS CHOSEN NASA 2556
COMMON USER INPUT PARAMETERS NASA 2557
COMMON BTWN LISTS NECESSARY COMMUNICATION BETWEEN SUBROUTINES NASA 2558
COMMON DATA HAS LAST SUBSCRIPT FOR EACH PIECE OF EQUIP, AND NASA 2559
THE NECESSARY PIECE OF THE DATA BASE NASA 2560
DIMENSION ICHOOSE(9), IPIC(3), ES(6), C(5), DMA(2), G(3), F(9), NCHOSE(9) NASA 2561
DIMENSION NCONF(6) NASA 2562
COMMON / USER1/ ALPHA, AX, AY, AZ, OPHI, NASA 2563
15 EA, EANT, EPSI, K, MANV, NASA 2564
PHIE, PHIRX, PHIRY, PHIRZ, TACCEL, NASA 2565
THEMX, THOD, IL, TPHIN, ISMAL, NASA 2566
COMMON / USER2/ X, XN, XNU, YN, NASA 2567
20 ZN NASA 2568
COMMON / USER3/ APOGEE, COMRAT, DIAMAX, EEQNT(9), EPME, NASA 2569
EQMHT, EQMHTL, EQMHTL, EQN2NT, NASA 2570
EQM2XL, EQMZYL, EQM2XL, NASA 2571
IAGNCY, NASA 2572
DEBUG, ISAT, NBI2SH, ORBINO, PERIGE, NASA 2573
MICRO, RELME, SPEC(6), SPEC, NASA 2574
COMMON /TNZ/ ACSSN, ACSWP, NASA 2575
ALT, AREA, BACAP, NASA 2576
BITRAT(2), CLIFE, CONVT, D, DT, NASA 2577
DX, DO, FC, FF, NASA 2578
EQBQ, FBQ, HARNET, HPT, HTPIPE, NASA 2579
HTRPB, HTRPRB, HTPNPR, IBLOC, NASA 2580
LMADD, NC, OMEG, PASSTR, PJ, NASA 2581
PL, PGC, PLMN, POCNT, RADA, RADAB, NASA 2582
RAIN, RJ, SABOLG, SATLG, SATTWT, NASA 2583
SAXM2L, SATXCG, SATYCG, SATZCG, SATXCL, NASA 2584
SAX2L, SAI2L, SIDE, SYSLB, THCHNT, NASA 2585
THRUST, TI, TRIM, TPRIM, TPRM, NASA 2586
VCHP, VOL, WATE, WB, WT, NASA 2587
WT, WJ, XN, XZERO, YJ, ZJ NASA 2588
COMMON /DBCOM/DATAB(55,100), IDb(30) NASA 2589
DATA XMD, YMD, ZMD, D1, XNO2, YNO2, ZMC2/3*.0003,.03,.3* .04 NASA 2590
ACSSN=2, IF (NCONF(1) .EQ. 1) GO TO 10 NASA 2591
DT=D1/12, NASA 2592
DQ=DQ/12, NASA 2593
DX=DX/12, NASA 2594
DY=DY/12, NASA 2595
CZ=CZ/12, NASA 2596
XJ=XJ/4636.8 NASA 2597
YJ=YJ/4636.8 NASA 2598
ZJ=ZJ/4636.8 NASA 2599
RJ=1.
CONVWT=0.

POMO = RJ/4636.8
PJ = PJ/4636.8
WT=0.

PL=0.*
PLMIN=0.
DB = 0.* * PHIRX
IF (DB .LT. 0.05) DB = 0.05
ETA = 0.85
EPSLN = 0.7 EPSA = 0.2
OHEGO = 1.1864E8/(20.920E6 + 6076.*ALT)**1.5
RE = 3441.66
OHEGE = 7.2927E-5
MO = 9.4E-8
XX = 3*OHEGO*OHEGO/(57.3)**2
YCG = 1.12.* (SATXCG - 500.*5.*EQBLG)
ZCG = 1.12.* SATXCG
A = 5.* (2.* RE + APOGEE + PERIGE)
EQBLG = EQBLG/12.

IF (NCONF(6) .EQ. 2) EQBSID = EQBSID/12.

SIDE = 0
XLNTH= EQBLG
GO TO 14.
SIDE = EQBSID
XLNTH= EQBLG
GO TO 14.
SIDE = 0
XLNTH= D

IF (EQM1WT .GT. 0.) GO TO 15

EOM1YL = 0.
EOM1ZL = 0.
EOM2YL = 0.
EOM2ZL = 0.

CONTINUE

IF (EQM2WTC .GT. 0.) GO TO 16

CONTINUE

ICNF = NCONF(1)
GO TO (17,18,19,19,19),ICNF

INOUSE = 1
TEMPIN = XJ
XJ = YJ
YJ = TEMPIN
GO TO 20.

INOUSE = 2
TEMPIN = XJ
XJ = ZJ
ZJ = TEMPIN
GO TO 20
19 INOSE = 3
GO TO 20
NO TRANSPOSITION REQUIRED
20 CONTINUE
TGX = XK * (ZJ-YJ) * PHIRX / 57.3
IGY = XK * (ZJ-XJ) * PHIRY / 57.3
TGZ = 0.*
ICONF = NCONF(5)
GO TO (21,26,21,26,21,26) / ICONF
21 XREFSP = SAIXL & SAIYL &
I = XGSA1
GO TO (22,23,24), I
22 XREFSP = .5 * XLNTH
GO TO 25
23 XREFSP = 0.0
GO TO 25
24 CONTINUE
DELXSP = 2.0 * (XCG-XREFSP)
DELYSP = 2.0 * YCG
DELZSP = 2.0 * ZCG
GO TO 27
25 XREFSP = 0.
DELYSP = 0.
DELZSP = 0.
GO TO 27
26 XREFSP = 0.
DELYSP = 0.
DELZSP = 0.
GO TO 27
27 IF (PERIGE .LT. 65.) GO TO 28
IF (PERIGE .GT. 500.) GO TO 29
TX = 0.
TY = 0.
TZ = 0.
TAUX = 0.
TAUY = 0.
TAUZ = 0.
GO TO 40
28 IERR = -1
ICHOSE(1) = -1
RETURN
29 TAX = 0.
TAY = 0.
TAZ = 0.
TAUXA = 0.
TAUYA = 0.
TAUZA = 0.
AP = (SIDE**XLNTH) + (EQM1XL*EQM1YL/144.) + (EQM2XL*EQM2YL/144.)
ACP = (EQM1XL-EQM2XL)/24.
YCP = 0.0
SIDE12 = SIDE**12.
ZCP = AMAX1(SIDE12,EQM12L,EQM2ZL) /(-24.)
XLX = XCG - XCP
XLY = YCG - YCP
XLZ = ZCG - ZCP
165 R = REEA
S = 1.02 * ASIN(R)
TS = 2.0 * (3.14159-S)/OMEGA
GO TO (30,31,32), INOSE
30 TEMPX = XLX
XLX = XLX* (1. + EPSLN) * XLY
TSX = HO * AP * (1. + EPSLN) * (-XLX)
TSY = HO * AP * (1. + EPSLN) * (XLY)
TSZ = 0.
GO TO 38

31 TEMPX = XLX
XLX = -XLZ
XLZ = TEMPX
TSX = 0.
TSY = HO * AP * (1. + EPSLN) * (XLY)
TSZ = HO * AP * (1. + EPSLN) * (-XLX)
GO TO 38

32 TSX = HO * (AP*(1. + EPSLN)*XLX + SREFSP*(1. + EPSA)*DELYSP)
TSY = HO * (AP*(1. + EPSLN)*XLX + SREFSP*(1. + EPSA)*DELXSP)
TSZ = 0.

38 TAUXS = TS
TAUYS = TS
TAUZS = TS
GO TO 60

190 C

40 WEDJO = OMEGE / OMEGA
SINIC = SIN(OERING)
BETA = ATAN(WEDJO*SINIC) + DB
ALFA = DB
ALBAR=ACOS(COS(BETA)/(SQRT(1.+COS(BETA)**2 * TAN(ALFA)**2)))
CA = 2.012 * COS(ALBAR)**2
COY = -SIN(2.*BETA)
CN = SIN (2.*ALFA)

200 IF (PERIGE *GE.300.) GO TO 42
210 IF (PERIGE *GE.100.) GO TO 41
RHO = 1.5E-9 * 02**(PERIGE-65./35.)
GO TO 43
41 RHO = 3.1E-11 * 4.333E-4**(PERIGE-100.)/200.
GO TO 43
42 RHO = 1.3E-14 * 1.538E-2**(PERIGE-300.)/200.
GO TO 43
43 Q = 3.6E10 * (RHO/A)
GO TO 44, 64, 65, 66, 67, INOIE
44 SREF = SIDE*XLNTH + EQM1XL*EQM1YL/144. + EQM2XL*EQM2YL/144.
XREF = (EQM1XL-EQM2XL) / 24.
SIDE12 = SIDE * 12.
YREF = AMAXI (SIDE12,EQM1YL,EQM2YL) / 24.
ZREF = 0.
DELY = XCG - XREF
DELT = YCG - YREF
DELTZ = ZCG - ZREF
DELTX = DELT
DELY = -TEMPX
GO TO 50

220 C

46 SREF = SIDE*XLNTH + EQM1XL*EQM1YL/144. + EQM2XL*EQM2YL/144.
XREF = (EQM1XL-EQM2XL) / 24.
YREF = 0.
SIDE12 = SIDE * 12.
ZREF = AMAXI (SIDE12,EQM1YL,EQM2YL) / (-24.)
GO TO 60

190 COMPUTE SOLAR TORQUES
DELTX = XCG - XREF
DELY = YCG - YREF
DELZ = ZCG - ZREF
TEMPX = DELTX
DELTX = TEMPX
GO TO 50

C 48
SIDE12 = SIDE * 12.
SREF = AMAX1(SIDE12, EQM1YL, EQM2YL) * AMAX1(SIDE12, EQM1ZL, EQM2ZL) / 144.
XREF = XLTH/2. * EQM1XL/12.
YREF = 0.
ZREF = 0.
DELTX = XCG - XREF
DELTY = YCG - YREF
DELZ = ZCG - ZREF

C 50
TAX = Q * SREF * (DELTX * CY + DELTY * CN) + Q * SREFSP * (DELZSP * CY + DELYSP * CN)
TAY = Q * SREF * (DELTX * CA - DELTY * CN) + Q * SREFSP * (DELZSP * CA - DELYSP * CN)
TAZ = Q * SREF * (-DELTX * CA + DELTY * CN) + Q * SREFSP * (-DELZSP * CA + DELYSP * CN)

P = 17.6 / (APOGEE - PERIGEE + 18.75) + 0.05
TAUXA = P * T * 2.592E6
TAYA = P * T * 2.592E6
TAUZA = P * T * 2.592E6

C 60
TAX = ABS (TAX)
TAY = ABS (TAY)
TAZ = ABS (TAZ)
TGX = ABS (TGX)
TCY = ABS (TCY)
TGZ = ABS (TGZ)
TSX = ABS (TSX)
TSY = ABS (TSY)
TSZ = ABS (TSZ)

C 230
IF (Iter .LT. 0) DX = DX / 12.
DO 100 I = 1, 7
II = I + 1
IF (Iter .EQ. 0) NCHOOSE(I) = 1
WT = WT + DATA(T23, II) * NCHOOSE(I)
NASA 25912
340 J1=J1+1
PRINT 9000,RJ,PJ
9000 FORMAT (1X,4HEJ= E11.4,4HPJ= E11.4)
PRINT 9001,EX,EY,EZ
9001 FORMAT (1X,4HEJ= E11.4,4HEY= E11.4,4HEZ= E11.4)
IF (J1*LE.*J1E) GO TO 103
C LAST ONE CHECKED AND NONE FOUND
118 ICHOOSE (8)=1
RETURN
C ACCEPTABLE DEVICE SELECTED
350 IPIC(1)=J1
PRINT 9000,RJ,PJ
PRINT 9001,E*,EY,EZ
IPIC(2)=0
ICHOOSE(8)=DATAB(8,11)
WT=WT+DATAB(23,J1)*NCHOOSE(9)
VOL=VOL+DATAB(24,J1)*NCHOOSE(6)
FL=PL+DATAB(16,J1)*NCHOOSE(3)
FLMIN=FLMIN+DATAB(16,J1)*NCHOOSE(8)
TI=267.*TPRIM+FE*TSMALL
FF=14.1E-9*FF.*0X*DI/(RJ*PHIRX)
CLIFE=37.E6*TPRIM*FF.*DX/ DI/(RJ*PHIRX)
RETURN
C YAW SPIN CONFIG
365 C INITIALIZE SKIPPING SOME IF ITERATING
200 IERR=0
TEMPIN = XJ
XJ = 2J
2J = TEMPIN
11=IDB(8)+1
ICHOOSE(1)=DATAB(1,11)
12=IDB(9)+1
ICHOOSE(2)=DATAB(1,12)
13=IDB(10)+1
ICHOOSE(3)=DATAB(1,13)
14=IDB(11)+1
ICHOOSE(4)=DATAB(1,14)
15=IDB(12)+1
ICHOOSE(5)=DATAB(1,15)
16=IDB(13)+1
ICHOOSE(6)=DATAB(1,16)
17=IDB(14)+1
ICHOOSE(7)=DATAB(1,17)
IF (ITER .GT. 0) GO TO 203
DO 202 I=1,9
202 NCHOOSE(1)=1
203 WT=WT*NCHOOSE(1)*DATAB(23,11)*NCHOOSE(2)*DATAB(23,12)*NCHOOSE(3)*DATAB(23,13)*NCHOOSE(6)*DATAB(23,14)*NCHOOSE(7)*DATAB(23,17)
CONVT=DATAB(23,11)*NCHOOSE(6)
VOL=VOL*NCHOOSE(1)*DATAB(24,11)*NCHOOSE(2)*DATAB(24,12)*NCHOOSE(3)*DATAB(24,13)*NCHOOSE(6)*DATAB(24,14)*NCHOOSE(7)*DATAB(24,17)
PL=PL*NCHOOSE(1)*DATAB(16,11)*NCHOOSE(2)*DATAB(16,12)*NCHOOSE(3)*DATAB(16,13)*NCHOOSE(6)*DATAB(16,14)*NCHOOSE(7)*DATAB(16,17)
FLMIN=FLMIN*NCHOOSE(1)*DATAB(16,11)*NCHOOSE(2)*DATAB(16,12)+NCHOOSE(3)*DATAB(16,13)*NCHOOSE(6)*DATAB(16,14)*NCHOOSE(7)*DATAB(16,17)
18=IDB(15)+1
ICHOOSE(8)=DATAB(1,18)
19=IDB(16)+1
ICHOOSE(9)=DATAB(1,19)
IF (PHIRX .LT. 1.25) IERR=1
GO TO 200
DB=PHIRX*.4  
IF (DB .LT. .05) DB=.05 
C CALCULATE F VALUES  
F(1)=(DPHI*DT/57.3+.04*D)*FE/DY  
F(2)=(DPHI*DT/57.3+.04*D)*FE/DZ  
F(3)=2.*.04*DPHI/57.3*FE/DX.  
F(4)=XMD/DX  
F(5)=YMD/DY  
FMX=AMAX1(F(1),F(2),F(3),F(4),F(5))  

C IERR 1x & MAX RATE ERROR TOO SMALL  
IF (FMX .LT. 2.*FMIN) IERR=IERR+10  
FF=2.*FMIN  
IF (FMX .LT. FF) FF=FMX  
TMD=AMAX1(XMD,YMD)  
E=540.*TMD/(DX*FF)+.12=08  
IF ( E .LT. 0.) E=0.  
C SELECT EARTH SENSOR WITH PHIX<=PHIRX  
IF (IPIC(1) .GE. 0) GO TO 204  
GO TO 206  
204 J1=IPIC(1)  
IF (IPIC(2) .GE.IOB(13) .AND. .ITER.EQ.0) GO TO 211  
206 J1=IOB(12)  
E1=DATAB(6,J1)  
E2=IOB(10)  
P=SQRT(DATAB(7,J1)*2+DATAB(8,J1)*2)+DB+DATAB(11,J1)+E  
IF (DATAB(8,J1) .LT. 08) GO TO 211  
IF (PHIX .LT. PHIRX) GO TO 211  
ICHOOSE(4)=DATAB(11,J1)  
ICHOOSE(1)=J1  
C EARTH SENSOR SET  
GO TO 212  
211 J1=J1+1  
IF (J1 .LE. J1E) GO TO 206  
C MINUS ONE FLAG FOR NOT FOUND  
ICHOOSE(4)=-1  
ICHOOSE(5)=0  
RETURN  
C HERE WHEN ACCEPTABLE EARTH SENSOR FOUND  
212 H=J1 * OMEGS  
C SELECT REACTION WHEEL WITH MOMENTUM GRTR THAN H  
J2=IPIC(2)  
IF (J2 .GE. IOB(13) .AND. .ITER .EQ. 0) IPIC(2)=0  
IF (IPIC(2) .EQ. 0) J2=IOB(12)+1  
IF (ITER .EQ. 0 .AND. IPIC(2) .NE. 0) J2=J2+1  
J2=IOB(13)  
445 214 H1=DATAB(6,J2)  
IF ( H1 .GT. H) GO TO 218  
J2=J2+1  
IF ( J2 .LE. J2E) GO TO 214  
GO TO 211  
C ACCEPTABLE COMBINATION FOUND  
218 ICHOOSE(5)=DATAB(11,J2)  
IPIC(2)=J2
WT = WT + DATAB(23, J2) * NCHOSE(5) + DATAB(23, J1) * NCHOSE(4)
VOL = VOL + DATAB(24, J2) * NCHOSE(5) + DATAB(24, J1) * NCHOSE(4)
PL = PL + DATAB(16, J2) * NCHOSE(5) + DATAB(16, J1) * NCHOSE(4)
PHMIN = PHMIN + DATAB(18, I2) * NCHOSE(5) + DATAB(18, I1) * NCHOSE(4)

XI = 37000000. * TPRIH * OX * DF / (JX * 4) * PHIRX

YI = 37000000. * TPRIH * OY / (JX * 4) * PHIRY

ZI = 37000000. * TPRIH * OZ / (JX * 4) * PHIRZ

RETURN

C 3-AXIS N CONFIG
C CHOSEN AS:
C ATTITUDE REF
C ASC
C GYROS
C EARTH SENSOR
C
INITIALIZE

IF (ITER > GO TO 303

NCHOSE = 1

NCHOSE = NCHOSE(1) * DATAB(23, I1) + NCHOSE(2) * DATAB(23, I2) + DATAB(24, I14) + NCHOSE(3)

CONVT = DATAB(23, I1) * NCHOSE(1) + DATAB(24, I1) * NCHOSE(2) + DATAB(24, I14) * NCHOSE(3)

PH(1) = (OPHI / 57.3 * DT + O4 * DT) / FE * TSMA/L/DY

PH(2) = (OPHI / 57.3 * DT + O4 * DT) / FE * TSMA/L/DY

PH(3) = (OPHI / 57.3 * DT + O4 * DT) / FE * TSMA/L/DY

RETURN

300 IERR = 0

I1 = I0B(I4) + 1

I2 = I0B(I5) + 1

I2 = I0B(I5) + 1

I2 = I0B(I5) + 1

I2 = I0B(I5) + 1

IF (ITER > GO TO 303

DO 301 I = 1, 9

302 NCHOSE(I) = 0

303 WT = WT + NCHOSE(I) * DATAB(23, I1) + NCHOSE(2) * DATAB(23, I2) + DATAB(24, I14) + NCHOSE(3)

NCHOSE(3) = NCHOSE(1) * DATAB(23, I1) + NCHOSE(2) * DATAB(23, I2) + DATAB(24, I14) + NCHOSE(3)

CONVT = DATAB(23, I1) * NCHOSE(1) + DATAB(24, I1) * NCHOSE(2) + DATAB(24, I14) * NCHOSE(3)

PH(1) = (OPHI / 57.3 * DT + O4 * DT) / FE * TSMA/L/DY

PH(2) = (OPHI / 57.3 * DT + O4 * DT) / FE * TSMA/L/DY

PH(3) = (OPHI / 57.3 * DT + O4 * DT) / FE * TSMA/L/DY

RETURN

C CALCULATE F VALUES
F(1) = (OPHI / 57.3 * DT + O4 * DT) / FE * DZ
F(2) = (OPHI / 57.3 * DT + O4 * DT) / FE * DZ
F(3) = (OPHI / 57.3 * DT + O4 * DT) / FE * DZ
F(4) = (OPHI / 57.3 * DT + O4 * DT) / FE * DZ
F(5) = (OPHI / 57.3 * DT + O4 * DT) / FE * DZ
F(6) = (OPHI / 57.3 * DT + O4 * DT) / FE * DZ

NASA 2763
NASA 2764
NASA 2765
NASA 2766
NASA 2767
NASA 2768
NASA 2769
NASA 2770
NASA 2771
NASA 2772
NASA 2773
NASA 2774
NASA 2775
NASA 2776
NASA 2777
NASA 2778
NASA 2779
NASA 2780
NASA 2781
NASA 2782
NASA 2783
NASA 2784
NASA 2785
NASA 2786
NASA 2787
NASA 2788
NASA 2789
NASA 2790
NASA 2791
NASA 2792
NASA 2793
NASA 2794
NASA 2795
NASA 2796
NASA 2797
NASA 2798
NASA 2799
NASA 2800
NASA 2801
NASA 2802
NASA 2803
NASA 2804
NASA 2805
NASA 2806
NASA 2807
NASA 2808
NASA 2809
NASA 2810
NASA 2811
NASA 2812
NASA 2813
NASA 2814
NASA 2815
NASA 2816
NASA 2817
NASA 2818
NASA 2819
NASA 2820
FMIN = AMAX1(F(1), F(2), F(3), F(4), F(5), F(6))

F(7) = POOTRX*YL/(DI*DX)
F(8) = POOTRY*YL/(DI*DY)
F(9) = POOTRZ*YL/(DI*DZ)

FMAX = AMAX1(F(7), F(8), F(9))
IERR = 0

C IERR 1x : MAX RATE ERROR TOO SMALL
IF (FMAX .LT. 2.*FMIN) IERR = IERR + 10
FF = 2.*FMIN

C IERR 1x : FF .LT. 2.*FMIN
IF (IERR .LT. 2) FF = FMIN

DBX = 0.4*PHIRX
DBY = 0.4*PHIRY
DBZ = 0.4*PHIRZ
IF (DBX .LT. 0.05) DBX = 0.05
IF (DBY .LT. 0.05) DBY = 0.05
IF (DBZ .LT. 0.05) DBZ = 0.05

R1 = 2.*POOTRX
R2 = 2.*POOTRY
R3 = 2.*POOTRZ
R = AMAX1(R1, R2, R3)

C SELECT 3 GYROS
IF (IPIC(1) .GE. 0) GO TO 304
J1 = I0B(16) + 1
GO TO 306

C SELECT EARTH SENSOR
J2 = IPIC(2)
IF (J2 .GE. I0B(18)) AND. ITER .EQ. 0) IPIC(2) = 0
IF (IPIC(2) .EQ. 0) J2 = I0B(17) + 1
IF (ITER .EQ. 0) AND. IPIC(2) .NE. 0) J2 = J2 + 1
J2 = I0B(18)

C
FPHIN = DATAB(6, J2)*DATAB(6, J2)/DATAB(13, J2)*
+ ATAN(DATAB(13, J2)/DATAB(9, I1))
FOMEN = DATAB(6, J2)*DATAB(6, J2)/DATAB(13, J2)*.02
C IERR = IXXX MEANS 3-AXIS WHEELS ACCEPTABLE

IERR = IXXX MEANS DOUBLE GIMBAL CMG*IS ACCEPTABLE

IF (TREQ/HREQ .LT. 1) IERR = 100

C 403 IF (TREQ/HREQ .LT. 1) IERR = 100

NL = LIMIT(J1, QJ1, QJ2, QJ3)

C SET FIXED EQUIPMENT: ELECTRONICS PROCESSOR, VALVE DRIVER,

C SENSOR(SUN OR HORIZON)

II = I03(I1) +

I2 = I0B(I1) +

IF (ISAT .EQ. 1) I3 = I0B(I1) +

C 404 IF (TACCEL*LT. 20.) IERR = IERR + 1

C 405 ICHOSE(1) = DATAB(1, I1)

C 406 ICHOSE(2) = DATAB(1, I2)

C 407 ICHOSE(3) = DATAB(1, I3)

C 408 ICHOSE(7) = 0

C 409 ICHOSE(8) = 0

C 410 ICHOSE(9) = 0

C 411 IF (ITER .GT. 0) GO TO 407

C 412 GO 409 I1 = 9

C 413 ICHOSE(I1) = 1

C 414 WT = WT + NCHOSE(1) * DATAB(23, I1) + NCHOSE(2) * DATAB(23, I2) + DATAB(23, I3)

C 415 VOL = VOL + NCHOSE(1) * DATAB(24, I1) + NCHOSE(2) * DATAB(24, I2) + DATAB(24, I3)

C 416 PL = PL + NCHOSE(1) * DATAB(16, I1) + NCHOSE(2) * DATAB(16, I2) + DATAB(16, I3)

C 417 SELECT CMG

C 418 GAMMA = ATAN(HS/(XNNN-2.))/(HL*XNNN)

C 419 H = HS / (XNNN * SIN(GAMMA))

C 420 IF (IFIC(1) .ST. 0) J1 = IFIC(1)

C 421 IF (IFIC(2).GE.IOB(17).AND.IFIC(3).GE.IOB(21).AND.ITER.EQ.0)

C 422 * GO TO 440

C 423 C RETURNS HERE TO TEST NEW CMG

C 424 IF (DATAB(6, J1) .LT. 1) GO TO 414

C 425 SOOTH = 2 * T/H * (TACC* DATAB(6, J1))

C 426 TMAX = DATAB(6, J1) * PDATION/57.3

C 427 IF (SOOTH .LE. DATAB(6, J1) .AND. TMAX .LE. DATAB(6, J1)) GO TO 417

C 428 J1 = J1 + 1

C 429 IF (J1 .LE. IOB(20)) GO TO 410

C 430 ICHOSE(4) = 1

C 431 ICHOSE(5) = 0

C 432 ICHOSE(6) = 0

C 433 RETURN

C 434 CMG SELECTED

C 435 ICHOSE(4) = DATAB(1, J1)

C 436 NCHOSE(I) = XNNN

C 437 XKK = DATAB(7, J1) * PDATION/57.3

C 438 W = -32. +(0.068+.29*XKK)*DATAB(6, J1) + 360.

C 439 F = (0.013 * 0.265 + 0.062 * XKK)*DATAB(6, J1) + 1430.

C 440 V = 7.45 + 7.45 + DATAB(24, J1) = W

C 441 DATAB(23, J1) = W

C 442 DATAB(24, J1) = V
DATA(18, J1) = P

GYRO NEXT

IF (IPIC(2) * GT. 0) J2 = IFIC(2)
IF (IPIC(2) * EQ. 0) J2 = ID(16) + 1

420 ICHOSE(5) = DATA(1, J2)

SELECT STAR SENSOR

PHIR = AMIN1(PHIRX, PHIRY, PHIRZ)

IF (IPIC(3) * GE. ID(21) * AND. ITER * EQ. 0) IPI2C(3) = 0

IF (IPIC(3) * EQ. 0) J3 = ID(20) + 1

422 IF (DATA(16, J3) * GT. 1. * AND. POOTST * GT. 2.) GO TO 440

424 IF (PHIE * LT. PHIR) GO TO 450

440 J3 = J3 + ±

IF (J3 * LE. ID(21)) GO TO 422

IF (J3 * LE. ID(21)) GO TO 422

IF (J3 * LE. ID(17)) GO TO 420

IF (J2 * LE. ID(17)) GO TO 420

450 ICHOSE(6) = DATA(1, J3)

IF (IPIC(1) * J1)

IF (IPIC(2) * J2)

IF (IPIC(3) * J3)

VT = WT * XNNN * NCHOOSE(5) * DATA(23, J2) + NCHOOSE(6) * DATA(23, J3)

VOL = VOL * XNNN * NCHOOSE(5) * DATA(24, J2) + NCHOOSE(6) * DATA(24, J3)

PL = PL * XNNN * NCHOOSE(5) * DATA(16, J2) + NCHOOSE(6) * DATA(16, J3)

11N = PMT * XNNN * NCHOOSE(5) * DATA(18, J2) + NCHOOSE(6) * DATA(18, J3)

NOW THRUST AND IMPULSE

F1 = (DPHI * DT + 57.3 + 0.04 * DUE) / FE * DY
F2 = (DPHI * DT + 57.3 + 0.04 * DUE) / FE * DZ
F3 = 2.573 * 10.04 * DPHI * FE * DX

FF = AMAX1(F(1); F(2); F(3))

TI = F(1) * F(2) * F(3) / TSHALL

* XNNP * POOTB * FT7 * DX + Y * J * DT + Z * J

* TAX + TAXA + 2.592E6 * TGX + TXS * TAYAXS) / DX

* TAY + TAXA + 2.592E6 * TGY + TSY * TAYA Y) / DY
FLMIN=PLMIN+NCHOOSE(3)*CATAB(18,J1)+NCHOOSE(4)*DATAB(18,J2)

DX = 0.5 * D * COS(ALPHA/57.3)
FMAX=DATAE(6,J2)*SORT(XJ/ZJ)/(DI*DX)
OZ = 0.5 * D * SIN(ALPHA/57.3)
OY = 5*D

F(1)=(DPHI/57.3*DT+.04*D)*FE/DY
F(2)=(DPHI/57.3*DT+.04*D)*FE/DZ
F(3)=2.*57.3*.04*D*DPHI*FE/DX

FMIN=AMAX1(F(1),F(2),F(3))

FF=2.*FMIN

IF (FF < LT. FMIN) FF=FMIN

DELH = 2. * DATAB(6,J2) * (PHIRX/57.3) * TAN(ALPHA/57.3)

TI = (F(1)+F(2)+F(3)) * TSMALL

CLIFE = (TI - FE * TSMALL) / (FF * DI * 3.0)

FC = 1.3889E-12 * CLIFE / T

RETURN
END
SUBROUTINE STRUCT

DIMENSION NCONF(6)
COMMON /USER9/ CA, CE

COMMON /USER1/ APACFGE, COMRM, DIAJAX, EQWMT(9), EPHE
COMMON /USER2/ EQMTX, EQMTX, EQWMT(9), EQWMT
1 COMMON /USER3/ IDEBUG, ISATOR, MBL2SH, OPTEMP, ORBINC, PERIGE
4 COMMON /USER4/ MICRO, RELNEK, SEP(6), XDUM1, XCGSA1
5 COMMON /USER5/ ACSSN, ACSWP, ALT, AREA, BATCAP
6 COMMON /USER6/ BITRAT(2), CLIFE, CONMV, SATDA4, DT
7 COMMON /USER7/ CLIFE, CONVNT, SATDA4, DT
8 COMMON /USER8/ SATDA4, SATWL, XCH, ZJ
9 COMMON /USER9/ AM, AN, BF, BS
10 COMMON /BTWN/ ACSSN, ACSWP, ALT, AREA, BATCAP
11 COMMON /BTWN/ BITRAT(2), CLIFE, CONMV, SATDA4, DT
12 COMMON /BTWN/ CLIFE, CONVNT, SATDA4, DT
13 COMMON /BTWN/ SATDA4, SATWL, XCH, ZJ
14 COMMON /BTWN/ AM, AN, BF, BS
15 COMMON /BTWN/ ACSSN, ACSWP, ALT, AREA, BATCAP
16 COMMON /BTWN/ BITRAT(2), CLIFE, CONMV, SATDA4, DT
17 COMMON /BTWN/ CLIFE, CONVNT, SATDA4, DT
18 COMMON /BTWN/ SATDA4, SATWL, XCH, ZJ
19 COMMON /BTWN/ AM, AN, BF, BS
20 COMMON /BTWN/ ACSSN, ACSWP, ALT, AREA, BATCAP
21 COMMON /BTWN/ BITRAT(2), CLIFE, CONMV, SATDA4, DT
22 COMMON /BTWN/ CLIFE, CONVNT, SATDA4, DT
23 COMMON /BTWN/ SATDA4, SATWL, XCH, ZJ
24 COMMON /BTWN/ AM, AN, BF, BS

DATA E,XNU,SIGY,PI/1.77.33,3.143.1416/
25 DATA E,XNU,SIGY,PI/1.77.33,3.143.1416/
26 NASA 3165
27 NASA 3165
28 NASA 3165
29 NASA 3165
30 NASA 3165
31 NASA 3165
32 NASA 3165
33 NASA 3165
34 NASA 3165
35 NASA 3165
36 NASA 3165
37 NASA 3165
38 NASA 3165
39 NASA 3165
40 NASA 3165
41 NASA 3165
42 NASA 3165
43 NASA 3165
44 NASA 3165
45 NASA 3165
46 NASA 3165
47 NASA 3165
48 NASA 3165
49 NASA 3165
50 NASA 3165

IF(NCONF(5).EQ.1.OR.NCONF(5).EQ.3.OR.NCONF(5).EQ.5).AND.
*XCGSA1.EQ.2) ICHECK= 2

ICHECK= 2 MEANS THAT SOLAR ARRAYS ARE PADDLES AND MOUNTED AT
CENTER OF VEHICLE. ICHECK= 1 MEANS OTHERWISE.

IF(ICHECK.EQ.2) XL="EGBLG"

SOLAR ARRAYS ARE PADDLES
APPLIED LOAD (ONLY BENDING MOMENT)

COMPUTE WEIGHT PER PADDLE (ASSUME 2 PADDLES)
WE = .5*SOART

NOMINAL TUBE RADIUS
R = (SABOLG**4*XMA/(PI**5*E))**.1428

TUBE WALL THICKNESS
TW = 2.*SQRT(XMA/(PI*E*R))

CHECK FOR APPLICABILITY OF EULER COLUMN STABILITY
FAC1 = (PI*E**2*XMA/(8.*R*SABOLG**2))**.3333
IF(SIGY-FAC1*CE*0.) GO TO 1

EULER COLUMN STABILITY NOT APPLICABLE

VOLUME OF SOLAR BOOM
VARAY = R*TH*SABOLG

SIZING OF EQUIPMENT BAY STRUCTURE
1 CONTINUE

P = CA*SATWT

BENDING MOMENT
XM = .75*CE*EQBLG*SATWT
IF(ICHECK.EQ.1) XM = CE*EQBLG*SATWT
IF(NCONF(6).NE.1) GO TO 5

EQUIVALENT AXIAL LOAD
RR = .5*SATDAM
XM = P/(2.*PI**RR) + XM/(PI**RR*RR)

SIZING OF EQUIVALENT MONOCOGHE CYLINDER
TH = .672*(XXNU*XN*XL*XL/E)**.3333
FAC2 = XL**2*SQRT(XXNU)/(RR*TH)
IF(FAC2.LE.31) GO TO 3
TH = 2.76*SQRT(SQRT(XXNU)*XN**RR/E)
2 CONTINUE

3 CONTINUE
EQUIVALENT THICKNESS OF STIFFENED CYLINDER

\[
T_{\text{BAR}} = 0.267 T_M
\]

SIZING OF SKIN-STRINGER ASSEMBLY

\[
T = 0.44 T_{\text{BAR}}
\]
\[
T_S = 0.64 T_S \sqrt{E T_{\text{BAR}}/(X N U \times X N)}
\]
\[
N = 1.0 + 2.0 \pi R R / \theta
\]
\[
N = N
\]
\[
B = 2.0 \pi R R / A N
\]
\[
A_{\text{LPHA}} = 0.745 / X N U \times 0.25
\]

SIZING OF CYLINDER FRAMES

\[
A = E \times A LPHA \times 2 \times T_{\text{BAR}} \times 2 / X N
\]
\[
R H O F = 0.54 \times (R R / 2 / A) \times (X N \times A LPHA \times 2 / (E \times A)) \times 0.25
\]
\[
A F = 0.0075 \times X N \times R R \times 4 \times (E \times R H O F \times 2 / A)
\]
\[
B F = 3.464 \times R H C F
\]
\[
T F = A F / B F
\]
\[
N = 1.0 + X / A
\]
\[
M = N
\]
\[
A = X L / A M
\]

SIZING OF END COVERS

\[
T C = 352 \times \sqrt{C A \times S A T W T / S I G Y}
\]
\[
T A = T C
\]
\[
X L D = R R
\]

VOLUME OF EQUIPMENT BAY STRUCTURE

\[
V E Q = E Q B L G \times (T + (T S \times B S / B) + (T F \times B F / A))
\]

IF (ICHECK.EQ.0) GO TO 4

MID-SECTION BULKHEAD IS REQUIRED

\[
W L = 0.55 \times C A \times S A T W T / X L D \times 2
\]
\[
T B = 0.59 \times X L D \times \sqrt{W L / S I G Y}
\]
\[
V E Q = 2.0 \times V E Q + 0.219 \times T B \times R R
\]

4 CONTINUE

TOTAL STRUCTURE WEIGHT

\[
V E Q = V E Q + R R \times T C
\]
\[
S T R W A T = 2.0 \times \pi \times 0.1 \times (R R \times V E Q + 4.0 \times V A F A V)
\]

RETURN

5 CONTINUE

IF (NCNF(6).NE.2) RETURN

W = 0.707 \times S A T D A M
EQUIVALENT AXIAL LOAD

\[ XN = 0.25 \frac{P}{W} + 0.75 \frac{XN}{W^{*}2} \]

SIZING OF EQUIVALENT MONOCOQUE BOX

\[
\begin{align*}
\text{IF}(XL/W \leq 0.5) & \quad TM = 1.068 \times \left( \frac{XXN \times XN \times XL}{XL/E} \right)^{0.3333} \\
\text{IF}(XL/W > 0.5) & \quad TM = 0.72 \times \left( \frac{XXN \times XN \times W/E}{XL} \right)^{0.3333}
\end{align*}
\]

EQUIVALENT THICKNESS OF STIFFENED BOX

\[ TBAR = 0.267 \times TM \]

SIZING OF SKIN STRINGER ASSEMBLY

\[
\begin{align*}
T &= 0.44 \times TBAR \\
TS &= 1.99 \times T \\
BS &= 0.64 \times TS \times \sqrt{E \times TBAR / (XXN \times XN)} \\
B &= 1.49 \times BS \\
N &= 1 + \frac{W}{B} \\
AN &= N \\
A &= \frac{W}{AN} \\
ALPHA &= 0.745 / XXNU ** 0.25
\end{align*}
\]

SIZING OF FRAMES

\[
\begin{align*}
A &= E \times ALPHA ** 2 \times TBAR ** 2 / XN \\
RHOF &= 0.405 \times (W**2/A) \times (XXN \times ALPHA ** 2 / (E*A))^{0.25} \\
AF &= 0.041 \times XN \times W**4 / (E \times RHOF**2 \times A) \\
EF &= 3.164 \times RHOF \\
TF &= AF / BF \\
M &= 1 + \frac{XL}{A} \\
AM &= M \\
A &= XL / AM
\end{align*}
\]

SIZING OF END COVERS

\[
\begin{align*}
TC &= 0.303 \times \sqrt{CA \times SATWT / SIGY} \\
TA &= TC \\
XLD &= 0.5 \times W
\end{align*}
\]

VOLUME OF EQUIPMENT BAY STRUCTURE

\[
\begin{align*}
VEQ &= 2 \times (T + (TS \times BS / 8) + (TF \times BF / A)) \\
\text{IF}(ICHECK = \text{EQ} \times 1) & \quad \text{GO TO 6}
\end{align*}
\]

MID-SECTION BULKHEAD IS REQUIRED

\[
\begin{align*}
W &= 0.455 \times CA \times SATWT / XLD^{**2} \\
TB &= 0.859 \times XLD \times \sqrt{W / SIGY} \\
VEQ &= 2 \times VEQ + 0.219 \times W \times TB
\end{align*}
\]

TOTAL STRUCTURAL WEIGHT
C  VEQ=2.0*EQVLG*VEQ+2.0*TC*(H**2)
    STRWAT=0.1*(VEQ+0.1*PI*VARAY)
230  RETURN
    END

50
SUBROUTINE VESIZE(IERR, NCONF, ICHOOSE)

DIMENSION NCONF(6), EE10(6), EEYCC(9), EEZCG(9), EEINX(9), EEINY(9),
* EEINZ(9), EEEXG(9)

COMMON /USER6/ CEEEX(9), EELOC(9), EEYCC(9), EEZCG(9),
EEINX(9), EEINY(9), 
* EEINZ(9), EEEXG(9), NASA 3325

COMMON /USERI/ APOGEE, COMRAI, DIAMAX, EEQWT(9), EPME,
* EQM1HT, EQM1YL, EQM1ZL, EQM2WT, NASA 3330

COMMON /BTWN/ ACGSN, ACSWP, ALT, SOAREA, BATCAP,
* DX, DY, DZ, EQBLG, EQBSID, NASA 3335

COMMON /PRTCOM/ CDP1(17), CISTAR, CITOT, DOTE, DE,
* DR, EQBSR, FEIN, FEOCS, FER, NASA 3340

COMMON /IFMT/ ESSREL(6), SUBE(7), SUBT(7), SUBUE(7), SUBUP(7),
* IF, TOOLG, TOOL(6), TOTOPS, TRUNC,

ISHAPE = NCONF(6)
ISPIN = 0

RLD = 0.600
XMEW = EEq1XL*EQM2XL
XMEW = AMAX1(EQ1YL, EQM2YL)
XMEW = AMAX1(EQ1ZL, EQM2ZL)
EOMST = 0.0
EOMST = 0.0

IF(NCONF(1).EQ.1.OR.NCONF(1).EQ.2) ISPIN = 1
TEQTYP = 1

D DETERMINE EQUIPMENT BAY EQUIPMENT WEIGHT AND VOLUME
EQWT = 1.025*STINHT

1956-76 OPT=2
FTN 4.2+383 03/27/75 21.38.45
EQVOL = 1.025 * SATVOL
THCMWT = 0.25 * STINWT
THE THERMAL CONTROL SUBSYSTEM IS ACCOUNTED FOR BY THE 1.025 FACTOR
NOTE THAT VOLUMES ARE IN FT**3

ACCOUNT FOR PACKING FACTOR
EQVOL = 1728 * EQVOL * EQPF

DETERMINE EQUIPMENT BAY LENGTH

IF (ISHDPE > 2) GO TO 4

SATDAM = (EQBVOL / (0.785 * RLDf)) ** 0.333
EQLG = RLD * SATDAM
IF (SATDAM .LE. DIAMAX) GO TO 4
SATDAM = DIAMAX
EQLG = EQBVOL / (0.785 * SATDAM ** 2)
GO TO 4

EQBLG = EQBVOL ** 0.333
SATDAM = 1.4162 * EQBLG
IF (SATDAM .LE. DIAMAX) GO TO 4
SATDAM = DIAMAX
EQLG = 2 * EQBVOL / SATDAM ** 2
GO TO 4

SATDAM = (EQBVOL / 0.524 ** 0.333)
EQLG = SATDAM
IF (SATDAM .LE. DIAMAX) GO TO 4
ICHOOSE = -1
THAT IS, THIS IS NOT AN ACCEPTABLE MACRO CONFIGURATION
RETURN

CONTINUE

DETERMINE SOLAR ARRAY DIMENSIONS

IERR = 0
GO TO 11
SURF = 3.1416 * SATDAM * EQBLG
IF (SURF .GE. SAAREA) GO TO 20
SATDAM = SQRT(SAAREA / 1.88496)
EQBLG = 0.6 * SATDAM
EQBVOL = 0.785 * SATDAM ** 2 * EQBLG
IF (SATDAM .LE. DIAMAX) GO TO 20
SATDAM = DIAMAX
EQBLG = SAAREA / (3.14159 * SATDAM)
EQBVOL = 0.785 * SATDAM ** 2 * EQBLG
GO TO 20

SURF = 1.5708 * SATDAM * EQBLG
IF (SURF .GE. SAAREA) GO TO 20
SATDAM = SQRT(SAAREA / 94248)
EQBLG = 0.6 * SATDAM
EQBVOL = 0.785 * SATDAM ** 2 * EQBLG
IF (SATDAM .LE. DIAMAX) GO TO 20
SATDAM = DIAMAX
EQBLG = 2 * SAAREA / (3.14159 * SATDAM)
EQBVOL = 0.785 * SATOAM**2 * EQBLG
GO TO 20

C

115 14 GO TO (15,20), IEOTYP
15 IF (ISPIN.EQ.1) GO TO 20
   SURF = SATDAM * EQBLG / 1.4142
   IF (SURF .GE. SAAREA) GO TO 20
   SATDAM = SQRT(SAAREA)* 1.4142
   EQBLG = SATDAM
   EQBVOL = 0.500 * SATDAM**2 * EQBLG
   GO TO 20

120 16 IF (SATDAM .LE. DIAMAX) GO TO 20

125 17 SATDAM = DIAMAX
   EQBLG = 1.4142 * SAAREA / SATDAM
   EQBVOL = 0.500 * SATDAM**2 * EQBLG
   GO TO 20

C

130 18 GO TO (19,20), IEOTYP
19 IF (ISPIN.EQ.0) GO TO 20
   SURF = 3.14159 * SATDAM**2
   IF (SURF .GE. SAAREA) GO TO 20
   SATDAM = SQRT(SAAREA/3.14159)
   EQBLG = 0.524 * SATDAM**2
   IF (SATDAM .LE. DIAMAX) GO TO 20
   ICHOSE = -1
   THAT IS WE CANNOT LENGTHEN A SPHERE
   RETURN

140 20 SATDAM = SATDAM
   EQBLG = 1.4142 * SAAREA / SATDAM
   EQBVOL = 0.500 * SATDAM**2 * EQBLG
   GO TO 20

C

150 21 CONTINUE
C
C DETERMINE SATELLITE LENGTH

155 51 SA3XL = SAAREA/SATDAM
   IF (SA3XL.LE.EQBLG) GO TO 55
   IERR=IERR+1
   GO TO 55

160 52 SA3XL= 1.4142*SAAREA/SATDAM
   IF (SA3XL.LE.EQBLG) GO TO 55
   IERR=IERR+1
   GO TO 55

165 53 SA3XL= SQRT(1.2734*SAAREA)
   IF (SA3XL.LE.SATDAM) GO TO 55
   IERR=IERR+1
   GO TO 55

54 SA1YL = 0.05208*SAAREA
   SA1XL = 90.
   SA12L = 1.

NASA 3384
NASA 3385
NASA 3386
NASA 3387
NASA 3388
NASA 3389
NASA 3390
NASA 3391
NASA 3392
NASA 3393
NASA 3394
NASA 3395
NASA 3396
NASA 3397
NASA 3398
NASA 3399
NASA 3400
Determine Equipment Bay Structural Weight

```
170 55 CONTINUE
175 DETERMINE EQUIPMENT BAY STRUCTURAL WEIGHT
     EQBSTR = .218 * EQWT ** .986 * (EQL / SATOAM) ** .263
     IF (IEQTY .LT. 2) EQBSTR = .99 * EQBSTR
     EQBSTR = EQBSTR + 0.1 * EQWT

180 DETERMINE STRUCTURAL THERMAL PROTECTION WEIGHT
     STTPS = 0.025 * EQBSTR
     EQQHT = EQQHT + STTPS

185 DETERMINE MISSION EQUIPMENT BAY TOTAL VOLUME
     IF (HBI2SH .EQ. 2) GO TO 150
     EQMVOL = .785 * EQM1VL * EQM2VL
     EQMWT = EQMWT + EQMVOL
     GO TO 151

190 CONTINUE
     EQMVOL = EQM1VL * EQM2VL
     EQMWT = EQMWT + EQMVOL
     GO TO 151

195 DETERMINE SA1WT, SA2WT, SA3WT
     SA3WT = SOARWT
     IF (IEQTY .LT. 1) GO TO 152
     SA1WT = .5 * SOARWT
     SA2WT = SA1WT

200 CONTINUE

205 DETERMINE BOOM AND MECHANISM WEIGHT
     SABOLG = 0.
     SABOOM = 0.
     SADRIV = 0.
     IF (ISPIN .LT. 1) GO TO 23
     SABOLG = 24.
     SABOOM = 15.2
     IF (ISPOG .LT. 0) GO TO 23
     SADRIV = 166 * (SA1WT + SA2WT)

210 CONTINUE
     SABMWT = SABOOM + SADRIV

215 CALCULATE HARNESS AND STRUCTURAL TPS WEIGHT

220 FIRST NEED MISSION EQUIPMENT WEIGHT AND EXTERNAL EQUIPMENT WEIGHT AND VOLUME
     EQQMT = EQM1WT + EQM2WT
     EEQMT = 0.
     EEQVOL = 0.
```
IF(NEQEQ.EQ.0) GO TO 232
DO 231 IS1=1,NEQEQ
EEQTWT = EEQTWT + EEQT(I)
EEQVOL = EEQVOL + EEQV(I)
231 CONTINUE
DO 232 I=1,NEQEQ
XHWT = EQHWT + EEQT(I)
XHVL = (EQHVOL + EEQVOL) / 1728.
232 CONTINUE
HARWHT = 0.013 * (STINHT-WHT-ACSWP+5*EEQT(I)+5*EQHWT) ** 1.31
+ (EEQVOL/1728.) ** 0.16

DETERMINE SATELLITE DRY WEIGHT LESS AUXILIARY PROPULSION DRY WEIGHT

FIRST DETERMINE SOLAR ARRAY WEIGHT

SOARWT = 0.
IF(IEQTYP.EQ.2) GO TO 233
SOARWT = SA1WT + SA2WT
233 GO TO 234
234 CONTINUE

PASSTR = EQBSTR + EQM1ST + EQM2ST + SABOOM + SADRIV
SUBWT1 = EQHWT + EQM1ST + EQM2ST + SOARWT + EEQT(I)
SUBWT = SUBWT1
CHRWT = SUBWT

SYSLB = EQHWT + EQM1ST + EQM2ST + EEQT(I)
+ EQBSTR + SOARWT + SABMWT + HARWHT

DETERMINE SATELLITE GROSS WEIGHT

SATWT = DRYWT + ACSWP

DETERMINE ADAPTER WEIGHT

SATAADP = 0.012 * SATWT

DETERMINE SATELLITE LAUNCH WEIGHT

SATWT = SATWT + SATAADP

CENTER OF GRAVITY CALCULATIONS

EBXCG = 500. + 5*EQBXL
EBYCG = 0.
EBZCG = 0.

MISSION EQUIPMENT AND MISSION EQUIPMENT BAY STRUCTURE C.G.

EM1XCG = 500. + EQBXL + 5*EQM1XL
EM2XCG = 500. - 5*EQM2XL

EQUIPMENT BAY STRUCTURE C.G.
EXTERNAL EQUIPMENT C.G.
IF(NUMEEQ.EQ.0) GO TO 240
CO 239 I=1,NUMEEQ
EESIO(I)=(1728*EEQVL(I))**.333
EEXC(I)= 500. + EQLGGO
GO TO 239
EEXC(I)= 500. + *5*EQBLG
GO TO 239
EEXC(I)= 500. 
236 EEXC(I)= 500. + .5*EQBLG
GO TO 239 
237 EEXC(I)= 500. 
238 CONTINUE
EEYCG(I)= 0.
EEXCG(I)= 0.
230 IF(IEEQTY.P.EQ.1.) EEYCG(I)= .5*(SATDAM + EESIO(I))
IF(IEEQTY.P.EQ.2.) EEYCG(I)= -.5*(SATDAM + EESIO(I))
IF(IEEQTY.P.EQ.3.) EEYCG(I)= -.5*(SATDAM + EESIO(I))
IF(IEEQTY.P.EQ.4.) EEYCG(I)= .5*(SATDAM + EESIO(I))
239 CONTINUE
240 CONTINUE
SOLAR ARRAY C.G.'S
IF(IEEQTY.P.EQ.2) GO TO 244
SA3YCG= 0.
SA3ZCG= 0.
241 SA3XCG= 500. + EQBLG
GO TO 243
242 SA3XCG= 500. + .5*EQBLG
GO TO 243 
243 SA3XCG= 500. 
244 CONTINUE
IF(XCGSA3-2.) 241,242,243
244 CONTINUE
245 SA1XCG= 500. + EQBLG
GO TO 247
246 SA1XCG= 500. + .5*EQBLG
GO TO 247 
247 SA1XCG= 500. 
248 CONTINUE
SA2XCG= SA1XCG
SA1YCG= 24.* + .5*(SATDAM + SA1YL)
SA2YCG= SA1YCG
SA2ZCG= 0.
249 CONTINUE
SA2YCG= SA1YCG
SA2ZCG= 0.
SA2XCG= SA1XCG
SA2YCG= SA1YCG
SA2ZCG= 0.
249 CONTINUE
SATELLITE CENTER OF GRAVITY CALCULATIONS
FIRST DETERMINE CONTRIBUTION OF SOLAR ARRAYS
C IF(IEQTYPE.EQ.2) GO TO 250  
SAX= SA3WT*SA3XCG  
SAY= SA3WT*SA3YCG  
SAZ= SA3WT*SA3ZCG  
GO TO 251  
CONTINUE  
SAX= SA1WT*SA1XCG + SA2WT*SA2XCG + SABMWT*SABXCG  
SAY= SA1WT*SA1YCG + SA2WT*SA2YCG + SABMWT*SABYCG  
SAZ= SA1WT*SA1ZCG + SA2WT*SA2ZCG + SABMWT*SABZCG  
GO TO 251  
CONTINUE  
C  
NEXT DETERMINE CONTRIBUTION OF EXTERNAL EQUIPMENT  
C  
EEX=0.  
EEY=0.  
EEZ=0.  
IF(INUEEC.EQ.0) GO TO 253  
GO 252 I=1, NUMEEQ  
EEX= EEX + EEQWT(I)*EEXCG(I)  
EEY= EEY + EEQWT(I)*EEYCG(I)  
EEZ= EEZ + EEQWT(I)*EEZCG(I)  
252 CONTINUE  
253 continue  
C  
SATXCG= (EQBSTR*STRXCG + EQWT*EBXCG + (EQM1ST + EQM1WT)*EM1XCG +  
1 (EQM2ST + EQM2WT)*EM2XCG + SAX +  
2 (HARNWT+STTPS)*EBXCG)/SATWT  
111274  
C  
SATYCG= (EQBSTR*STRYCG + EQWT*EBYCG + (EQM1ST + EQM1WT)*EM1YCG +  
1 (EQM2ST + EQM2WT)*EM2YCG + SAY +  
2 (HARNWT+STTPS)*EBYCG)/SATWT  
111274  
C  
SATZCG= (EQBSTR*STRZCG + EQWT*EBZCG + (EQM1ST + EQM1WT)*EM1ZCG +  
1 (EQM2ST + EQM2WT)*EM2ZCG + SAZ +  
2 (HARNWT+STTPS)*EBZCG)/SATWT  
111274  
C  
CALCULATE MOMENTS OF INERTIA  
C  
FIRST DETERMINE EQUIPMENT BAY STRUCTURE AND EQUIPMENT BAY  
EQUIPMENT INCREMENTAL INERTIA  
C  
SATRAD= .5*SATDAM  
IF(ISHAPE.EQ.0) GO TO 69  
66 STRINX= EOBSTR*(SATRAD**2 + .167*EQBLG**2)  
STRINZ= STRINX  
EQINK= .5*EQMT*SATRAD**2  
EQINK= .5*EQMT*SATRAD**2  
EQINK= .5*EQMT*SATRAD**2 + EQBLG**2  
EQINZ= EQINK  
SIDE = SATDAM  
GO TO 69  
67 EOBSID= .708*SATDAM  
STRINX= .333*EOBSTR*EOBSID**2  
STRINY= .333*EOBSTR*EOBSID**2 + EQBLG**2  
STRINZ= STRINX  
EQUINX= .167*EQWT*EOBSID**2

340 345 350 355 360 365 370 375 380 385 390 395
EQINY = .0833*EQWTSATRO'2
EQINZ = EQINY
SIDE = EQBSID
GO TO 69
68 SIRINX = .167*EQBSTR*SATRAD*2
SIRINZ = SIRINX
405 EQINX = .0833*EQWTSATRO'2
EQINZ = EQINX
SIDE = SATRO
GO TO 69
69 CONTINUE IF (NUMEEQ *EQ. 0) GO TO 71
410 EXTERNAL EQUIPMENT INCREMENTAL INERTIA (BOX SHAPE)
DO 70 I=1,NUMEEQ
EEINX(I) = .167*EEQWT(I)*EESID(I)*E2
EEINZ(I) = EEINX(I)
70 CONTINUE
71 CONTINUE
420 SOLAR ARRAY INERTIAL CALCULATIONS
IF (IEQTYP *EQ. 2) GO TO 37
SAIINX = SAI1WT*SAITAD*2
SAIINZ = SAI1WT*(SAITAD*2 + .167*SAI3XL*2)
GO TO 38
37 SAIINX = .0833*SAI1WT*(SA1YL*2 + SA1ZL*2)
SAIINZ = .0833*SAI1WT*(SA1YL*2 + SA1ZL*2)
GO TO 38
38 CONTINUE
435 MISSION EQUIPMENT BAY INCREMENTAL INERTIA
IF (MB12SH *EQ. 2) GO TO 39
EMIINX = .5*(EQM1ST + EQM1WT) * S1RAD*2
EMIINY = .0833*(EQM1ST + EQM1WT)*(SC1S1RAD*2 + EQM1XL*2)
EMIINZ = EMIINY
EMIINZ = EMIINZ
EMI2INX = .5*(EQM2ST + EQM2WT)*S2RAD*2
EMI2INY = .0833*(EQM2ST + EQM2WT)*(SC2S2RAD*2 + EQM2XL*2)
EMI2INZ = EMI2INY
EMI2INZ = EMI2INZ
GO TO 40
39 TEM1 = .0833*(EQM1WT + EQM1WT)
TEM2 = .0833*(EQM2ST + EQM2WT)
EMI1INX = TEM1*(EQM1YL*2 + EQM1ZL*2)
EMI1INY = TEM1*(EQM1YL*2 + EQM1ZL*2)
EMI1INZ = TEM1*(EQM1YL*2 + EQM1ZL*2)
EMI2INX = TEM2*(EQM2YL*2 + EQM2ZL*2)
EMI2INY = TEM2*(EQM2YL*2 + EQM2ZL*2)
EMI2INZ = TEM2*(EQM2YL*2 + EQM2ZL*2)
40 CONTINUE
SATINX = STRINX + EQBSTR* ((SATYCG-STRYCG)**2 + (SATZCG-STRZCG)**2)
 1 + EMINI X + EQM1TO* ((SATYCG-EM1YCG)**2 + (SATZCG-EM1ZCG)**2) + SAIX + EEIX
 2 + EM2INI X + EQM2TO* ((SATYCG-EM2YCG)**2 + (SATZCG-EM2ZCG)**2) + SAIX + EEIX
 3 + E0IN X + EQMT* (SATYCG**2 + SATZCG**2) + SAIX + EEIX

SATINY = STRINY + EQBSTR* ((SATZCG-STRZCG)**2 + (SATXCG-STRXCG)**2)
 1 + EMINI Y + EQM1TO* ((SATZCG-EM1ZCG)**2 + (SATXCG-EM1XCG)**2) + SAIY + EEIY
 2 + EM2INI Y + EQM2TO* ((SATZCG-EM2ZCG)**2 + (SATXCG-EM2XCG)**2) + SAIY + EEIY
 3 + E0INY + EQMT* (SATZCG**2 + SATXCG**2) + SAIY + EEIY

SAITN = STRINZ + EQBSTR* ((SATYCG-STRYCG)**2 + (SATXCG-STRXCG)**2)
 1 + EMINI Z + EQM1TO* ((SATYCG-EM1YCG)**2 + (SATXCG-EM1XCG)**2) + SAIZ + EEIZ
 2 + EM2INI Z + EQM2TO* ((SATYCG-EM2YCG)**2 + (SATXCG-EM2XCG)**2) + SAIZ + EEIZ
 3 + E0IN Z + EQMT* (SATYCG**2 + SATXCG**2) + SAIZ + EEIZ

COMPUTE DISTANCE FROM C.G. TO MAIN ENGINE(0T), GAS JET LEVER ARMS.
ON ROLL, PITCH, AND YAW AXES, RESPECTIVELY, (DX, DY, DZ). THE
CONVERSION TO UNITS OF FT IS DONE IN SUBROUTINE SANDC.

515 IFISHAPE-2) 45, 48, 46
 45 DT= SATXCG - 500.
 46 DT= SATXCG - 500.
 47 DT=.5*EQBLG
 48 DT=.5*EQBLG
 49 T= SATXCG - 500.
 50 OX=.5*SATDAM
 51 OY= DX
 52 OZ= DX
 53 OY=.5*SATDAM
 54 OX=.5*SATDAM
 55 OZ= DX
 56 OY=.5*SATDAM
 57 OX=.5*SATDAM
 58 OZ= DX
 59 DT=.5*EQBLG
 60 OX=.5*EQBSID
 61 OY= DT
 62 OZ= DT
 63 FJ= EMINX
 64 RJ= SATINX-PJ
 65 WRITE (G, 1000) EQBLG, ACSWP, THCMWT, EQBSTR, SOARWT, SABMWT,
* HARNWT, SATWT,
1000 FORMAT(1X, 8E15.4)
RETURN
END

REGISTER ALLOCATION
2 REGISTERS ASSIGNED OVER THE LOOP BEGINNING AT LINE 227
2 REGISTERS ASSIGNED OVER THE LOOP BEGINNING AT LINE 358
1 REGISTERS ASSIGNED OVER THE LOOP BEGINNING AT LINE 413
**SUBROUTINE EP**

**OPT=2**

NASA 3809

---

**SUBROUTINE EP (IFIC, IERR, ITER, NCONF, ICHOSE, NCHOSE)**

**C WILL SELECT AND SIZE THE ELECTRICAL SUBSYSTEM WHICH WILL BE**

**THESE CONFIGURATIONS AS FOLLOWS**

**NCONF (1) = 1 IS DUAL SPIN**

**NCONF (1) = 2 IS YAW SPIN**

**NCONF (1) = 3 IS MASS EXPULSION**

**NCONF (1) = 4 IS PITCH MOMENTUM BIAS**

**NCONF (1) = 5 IS SHUNT - PADDLE**

**NCONF (1) = 6 IS SHUNT - BODY**

**NCONF (5) = 1 IS SERIES PADDLE**

**NCONF (5) = 2 IS SERIES BODY**

**NCONF (5) = 3 IS SERIES PADDLE**

**NCONF (5) = 4 IS SERIES BODY**

**NCONF (5) = 5 IS SERIES PADDLE**

**NCONF (5) = 6 IS SERIES BODY**

---

**A LIST OF THE VARIABLES FOLLOWS**

**A VARIABLE HOW USED FROM TO DEFAULT DESCRIPTION**

| A1 | INT | EPS | EP | FT | ME + HP |
| A32 | INT | EPS | EP | A**(3/2) |
| ALT | INT | USER | EPS | HT ALTITUDE |
| AREA | INT | EPS | EPS | FT**2 ARRAY AREA |
| CAPMAX | INT | DB | EPS | A-H MIN REQ CP |
| GCELL | INT | DB | EPS | A-H CAP SEL CL |
| CHMIN | INT | EPS | EPS | 2.0 HRS HECH IN |
| CHISTAR | INT | EPS | EPS | A-H MIN INST CP |
| CR | INT | EPS | EPS | A-HCAP SEL SEL |
| DATAB | INT | 0 | EPS | EPS | DATA BASE |
| DELF | INT | EPS | EPS | .03 XMIS LOSS |
| DELI | INT | EPS | EPS | .02 FAB LOSS |
| DELM | INT | EPS | EPS | .01 MISDIFF LOSS |
| DELT | INT | EPS | EPS | .05 OR .3 RAD DEG FAC |
| ETAC | INT | DB | EPS | 1.0 EFF CHCR |
| ETAD | INT | DB | EPS | 0.85 EFF DISCH |
| EFT | INT | EPS | EPS | 0.65 EFF BATT |
| ETALR | INT | DB | EPS | 0.10 EFF MOD |
| ETAR | INT | EPS | EPS | 1.0 PHR DIST LS |
| FN | INT | EPS | EPS | SIZE FACT |
| HE | INT | EPS | EPS | HT FACTOR |
| HEDA | INT | EPS | EPS | 20.902E6FT RADIUS EARTH |
| HP | INT | EPS | EPS | HE/A |
| I | INT | EPS | EPS | FT PERIGEO |
| INT INDEX | INT | EPS | EPS | NASA 3863 |
** INITIALIZATION **

IF (ITER.NE.0) GO TO 140

LMBD=0.05

MEAN RADIUS OF EARTH IN FEET

HE=20.92E6

MU=3.907645E16

HP=9076. * PERIGEE

A=HP+HE

A32=A**1.5

HEDA=HE/A

S = 1.02 * ASIN(HEDA)

N=SORT(MU)/A32

TEOT=S/(PIE-S)

TE=2.*S/N

RFD=.000

ICHOSE(I)=0

IERR=0

LMBD MUST GO TO REL

NB=0

DO 10 I=1,5

II=1

NCHOSE(I)=0

NIR=2

WATE=0.0

AREA=0.0

** SET UP DELTA-R AND DELTA-T (RADIATION DEGRADATION AND TEMPERATURE CORRECTION FACTORS)

DELR=.05

IF (ALT.GT.400.) DELR=.3

**
** ALTIMETER IS GREATER THAN 5000 NAUTICAL MILES **

** SHUNT REGULATION DESIGN **

** NOW WE WILL BE DOING THE EPR MACRO SELECTION (S, SANDO-SLR) **

** NOW WE WILL BE DOING THE EPR MACRO SELECTION (S, SANDO-SLR) **
IF (ISR1.GE.ISR1E) GO TO 180

ISR1=IPIC(1)+1
ICELL=IPIC(2)
ICHGR=IPIC(3)
GO TO 210

C

IF (ICELL.GE.ICELLE) GO TO 190

ISR1=1
ICELL=IPIC(2)+1
ICHGR=IPIC(3)+1
GO TO 210

C

IF (ICHGR.GE.ICHGRE) GO TO 200

ISR1=i
ICELL=IDB(i) +
ICHGR=IPIC(3)+i
GO TO 210

C

ICHOOSE(1)=-1
ICHOOSE(2)=-1
ICHOOSE(3)=-1
RETURN

** COMPUTE SELECTION PARAMETERS FOR SHUNT REGULATION DESIGN --
** THIS IS FOR SHUNT REGULATORS, BATTERY AND BATTERY CHARGER --

** DETERMINE NUMBER OF SHUNT REGULATORS REQUIRED

NSR=1

** DETERMINE EXCESS ARRAY POWER FOR REGULATION

PS = (PL/ETAR)*(1+TEST)*(1/(ETAD*ETAC*ETAIC))

PBO1=PBOl-FLMIN

PEXCES=PBO1-FLMIN

CAPMAX = 0.0

IF (PEXCES .GE. 720.) CAPMAX = 120.

IF (PEXCES .GE. 1440.) CAPMAX = 240.

C

IF (FLMIN.GE.PEXCES) GO TO 230

NSR = PEXCES / DATAB(6,ISR1) + 0.9

IF (NSR.LE.0) NSR=1

CONTINUE

** NOTE -- ADD SPECIAL EQUIPMENT (AS NECESSARY)

** SET VOLTAGES FOR THIS DESIGN

VOB=27.
VSH=25.

** BATTERY ALGORITHM
DETERMINE REQUIRED CAPACITIES

\[ CR = \frac{(PL \cdot TE/3600)}{(LM \cdot BDD \cdot ETAO)} \]

IF (\( CR \) \ LT. 2300.0) GO TO 238

\( NB = 4 \)

IF (\( CR \) \ LT. 4700.0) GO TO 238

\( NB = 6 \)

IF (\( CR \) \ LT. 7050.0) GO TO 238

\( NB = 8 \)

IF (\( CR \) \ LT. 9400.0) GO TO 238

\( NB = 10 \)

\( NB = \text{AMAX0(NB, NCHOOSE(2))} \)

\[ CA = \frac{CR}{VOB} \]

DETERMINE MINIMUM INSTALLED CAPACITY

\[ CI = CA \cdot RF \]

** DETERMINE NUMBER OF CELLS IN SERIES (TO BE SUPPLIED TO REL)**

\[ NC = \frac{VBM}{VC} \]

DETERMINE SELECTION PARAMETERS ON CELLS

\[ \text{CISTAR} = \frac{C1}{NB} \]

** DETERMINE CHARGE CURRENT RATING REQUIRED FOR THE BATTERY CHARGER **

\[ \text{CELL} = \frac{\text{CISTAR}}{ICCH} \]

\[ \text{ICH} = \frac{\text{CELL}}{ICHR} \]

** COMPARE THE HARDWARE PARAMETER TO THE SELECTION PARAMETER **

IF (\( \text{DATAB(6, ISR1)} \) \ GE. \ CAPMAX.AND. \( \text{DATAB(6, ICELL)} \) \ GE. \( \text{CISTAR} \) .AND. \( \text{DATAB(6, ICHGR)} \) \ GE. \( \text{ICHR} \)) GO TO 270

IF (\( \text{ISR1} \) \ GE. \( \text{ISR1E} \)) GO TO 240

\( \text{ISR1} = \text{ISR1} + 1 \)

GO TO 220

IF (\( \text{ICELL} \) \ GE. \( \text{ICELLE} \)) GO TO 250

\( \text{ICELL} = \text{ICELL} + 1 \)

GO TO 220

IF (\( \text{ICHR} \) \ GE. \( \text{ICHRE} \)) GO TO 260

\( \text{ICHR} = \text{ICHR} + 1 \)

GO TO 220

\( \text{CHOOSE(1)} = -1 \)

\( \text{CHOOSE(2)} = -1 \)

\( \text{CHOOSE(3)} = -1 \)

RETURN
C
270 WCELL=DATAB(24,ICELL)
    WCELL=DATAB(25,ICELL)
    ETAE=DATAB(7,ICELL)
C
    WB=NC*WCELL*K2
    WB=NC*WCELL*K1
C
405 WBT=WB*NB
    VBT=VB*NB
C
410 BATCAP=NB*DATAB(6,ICELL)
    ETA=ETA=DATAB(7,ICHGR)
    NC=NB
C
415 IF (NCHOSE(1) .GE. NSR) GO TO 271
    NCHOSE(1)=NSR
271 IF (NCHOSE(2) .GE. NB) GO TO 272
    NCHOSE(2)=NB
272 IF (NCHOSE(3) .GE. NCH) GO TO 273
    NCHOSE(3)=NCH
273 IF (NCHOSE(4) .GE. NPCU) GO TO 274
    NCHOSE(4)=NPCU
274 NCHOSE(5)=0
C
420 ICHOSE(1)=DATAB(1,ISR1)
    ICHOSE(2)=DATAB(1,ICELL)
    ICHOSE(3)=DATAB(1,ICHGR)
    ICHOSE(4)=DATAB(1,IPCU)
    ICHOSE(5)=0
C
425 IPIC(1)=ISR1
    IPIC(2)=ICELL
    IPIC(3)=ICHGR
    IPIC(4)=IPCU
    IPIC(5)=0
C
430 WT=NSR*DATAB(23,ISR1)+WBT*NCH*DATAB(23,ICHGR)+NPCU*DATAB(23,IPCU)
    * + WT
C
440 VOL=NSR*DATAB(24,ISR1)+VBT*NCH*DATAB(24,ICHGR)+NPCU*DATAB(24,IPCU)
    * + VOL
C
445 GO TO 590
C
** SHUNT AND DISCHARGE DESIGN **
C
480 ICONF=NC CNF(5)
    IDRE=IDB(4)
    ISRE2=IDB(5)
    IICELLE=IDB(2)
    ICHGR2=IDB(6)
    NCU=1
IF (IPIC(i).NE.0) GO TO 290
IDR=IDB(3)+1
ISR2=IDB(4)+1
ICELL=IDB(1)+1
ICHR=IDB(5)+1
ICCU=IDB(6)+1
ETAD=0.85
ETA=0.05
ETAE=0.05
ETAR=1.0
GO TO 360
IF (ITER.EQ.0) GO TO 300
IDR=IPIC(1)+1
ISR2=IPIC(2)
ICELL=IPIC(3)
ICHR=IPIC(4)
ICCU=IPIC(5)
GO TO 360
IF (ISR2.GE.ISR2E) GO TO 320
IDR=IDB(3)+1
ISR2=IDB(4)+1
ICELL=IDB(1)+1
ICHR=IDB(5)+1
ICCU=IDB(6)+1
ETAD=0.85
ETA=0.05
ETAE=0.05
ETAR=1.0
GO TO 360
IF (ICELL.GE.ICELLE) GO TO 330
IDR=IDB(3)+1
ISR2=IDB(4)+1
ICELL=IDB(1)+1
ICHR=IDB(5)+1
ICCU=IDB(6)+1
ETAD=0.85
ETA=0.05
ETAE=0.05
ETAR=1.0
GO TO 360
IF (ICHR.GE.ICHGRE) GO TO 340
IDR=IDB(3)+1
ISR2=IDB(4)+1
ICELL=IDB(1)+1
ICHR=IDB(5)+1
ICCU=IDB(6)+1
ETAD=0.85
ETA=0.05
ETAE=0.05
ETAR=1.0
GO TO 360
DO 350 I=1,5
dO 350 I=1,5
ICHOOSE(I)=-1
RETURN
** COMPUTE SELECTION PARAMETERS FOR SHUNT AND DISCHARGE REGULATION -
** THIS IS FOR DISCHARGE REGULATOR, SHUNT REGULATOR, BATTERY, BATTERY CHARGER AND SIZING THE CENTRAL CONTROL UNIT -

** DETERMINE NUMBER OF DISCHARGE REGULATORS REQUIRED

NO=NB

** DETERMINE EXCESS ARRAY POWER FOR REGULATION

PS=(PL/ETAR)*(1+TEDT)*(1/(ETAO*ETAC*ETAE))

FBOL=PS/(1-DEL)^2*(1-DEL)*(1-DELT)/(1-DELM)

PEXCES=PFOL-ALMIN

IF (ITER .GE. 1 .AND. NCHOSE(1) .GE. NO) ND=NCHOSE(1).

PD=PL/(ND*ETAD)

** DETERMINE NUMBER OF SHUNT REGULATORS REQUIRED

370 CAPMAX = 0.0

IF (PEXCES .GE. 720.) CAPMAX = 120.

IF (PEXCES .GE. 1440.) CAPMAX = 240.

380 CONTINUE

** SET VOLTAGES FOR SHUNT AND DISCHARGE DESIGN

VDB=21.
VN=19.

** SET UP BATTERY SELECTION PARAMETER

** DETERMINE REQUIRED CAPACITIES

CR=(PL*TE/3600.)/(LM8600*ETAD)

IF (CR .LT. 1827.0) GO TO 322

IF (CR .LT. 3654.0) GO TO 322

IF (CR .LT. 5481.0) GO TO 322

322 NB = MAX0(NB,NCHOSE(3))

CA=CR/VDB

** DETERMINE MINIMUM INSTALLED CAPACITY

CI=CA*RFO

** DETERMINE NUMBER OF CELLS IN SERIES (TO BE SUPPLIED TO REL)
NC=VBM/VC

C CISTAR IS SELECTION PARAMETER ON CELLS

C CISTAR=C1/AB

** CHARGER SELECTION

* DETERMINE CHARGE CURRENT RATING REQUIRED FOR THE BATTERY CHARGER *

CCELL=CISTAR
ICH=CCELL/CHNINT

** COMPARISON THE HARDWARE PARAMETER TO THE SELECTION PARAMETER **

IF (OATAB(6,ISR2).GE.PD.AND.OATAB(6,ISR2).GE.CAPMAX.AND.OATAB(6,ICE1L).GE.CISTAR.AND.OATAB(6,ICHGR).GE.ICH) GO TO 440

IF (IDR.GE.IORE).GO TO 390
GO TO 370

IF (ISR2.GE.ISR2E) GO TO 400
ISR2=ISR2+1
GO TO 370

IF (ICELL.GE.ICELLE) GO TO 410
IDR=IDB(3)+1
ISR2=IDB(4)+1
ICELL=ICELL+1
GO TO 370

IF (ICHGR.GE.ICHRG) GO TO 420
IDR=IDB(3)+1
ISR2=IDB(4)+1
ICELL=IDB(4)+1
ICHGR=ICHGR+1
GO TO 370

DO 430 T=1.5
ICHSE(I)=-1
RETURN

ETA0=OATAB(7,ISR)
ETAE=OATAB(7,ICEL)
ETAC=OATAB(7,ICHGR)
VCELL=OATAB(23,ICEL)
WB=NC*WCELL*K2
VB=NC*VCELL*K1
WBT=WB*NB
VBT = VB \times NB
\text{NCH} = NB
\text{BATCAP} = NB \times \text{DATAB}(6, \text{ICELL})

\text{ICHOSE}(1) = \text{DATAB}(1, \text{IDR})
\text{ICHOSE}(2) = \text{DATAB}(1, \text{ISR2})
\text{ICHOSE}(3) = \text{DATAB}(1, \text{ICELL})
\text{ICHOSE}(4) = \text{DATAB}(1, \text{ICHGR})
\text{ICHOSE}(5) = \text{DATAB}(1, \text{ICCU})

\text{IF} (\text{ICHOSE}(1) \geq \text{NO}) \text{GO TO 451}
\text{ICCHOSE}(1) = \text{NO}

\text{IF} (\text{ICHOSE}(2) \geq \text{NSR}) \text{GO TO 452}
\text{ICCHOSE}(2) = \text{NSR}

\text{IF} (\text{ICHOSE}(3) \geq \text{NB}) \text{GO TO 453}
\text{ICCHOSE}(3) = \text{NB}

\text{IF} (\text{ICHOSE}(4) \geq \text{NCH}) \text{GO TO 454}
\text{ICCHOSE}(4) = \text{NCH}

\text{IF} (\text{ICHOSE}(5) \geq \text{NCCU}) \text{GO TO 455}
\text{ICCHOSE}(5) = \text{NCCU}

\text{IPIC}(1) = \text{IDR}
\text{IPIC}(2) = \text{ISR2}
\text{IPIC}(3) = \text{ICELL}
\text{IPIC}(4) = \text{ICHGR}
\text{IPIC}(5) = \text{ICCU}

\text{WT} = \text{NO} \times \text{DATAB}(23, \text{IDR}) + \text{NSR} \times \text{DATAB}(23, \text{ISR2}) + \text{VBT} \times \text{NCH} \times \text{DATAB}(23, \text{ICHGR}) + \text{NCCU} \times \text{DATAB}(23, \text{ICCU}) + \text{WT}

\text{VOL} = \text{NO} \times \text{DATAB}(24, \text{IDR}) + \text{NSR} \times \text{DATAB}(24, \text{ISR2}) + \text{VBT} \times \text{NSR} \times \text{DATAB}(24, \text{ICHGR}) + \text{NCCU} \times \text{DATAB}(24, \text{ICCU}) + \text{VOL}

\text{GO TO 590}

** SERIES LOAD REGULATION DESIGN **

TCONF = NCONF(5)
\text{ILRE} = \text{IDB}(8)
\text{ICELL} = \text{IDB}(2)
\text{ICHGR} = \text{IDB}(9)
\text{ISPDE} = \text{IDB}(10)
\text{IPDE} = \text{IDB}(11)

\text{IF} (\text{IPIC}(1) \neq 0) \text{GO TO 460}
\text{ILR} = \text{IDB}(7) + 1
\text{ICELL} = \text{IDB}(1) + 1
\text{ICHGR} = \text{IDB}(8) + 1
\text{ISPDE} = \text{IDB}(9) + 1
\text{IPDE} = \text{IDB}(10) + 1
\text{ETALR} = 0.9
\text{ETAE} = 0.65
\text{ETAD} = 1.0
\text{ETAD} = 1.0
\text{NSPD} = 1
NPO=1
GO TO 520

C 460 IF (ITER.EQ.0) GO TO 470
ILR=IPIC(1)
ICELL=IPIC(2)
ICHGR=IPIC(3)
ISPD=IPIC(4)
IPD=IPIC(5)
GO TO 520

C 470 IF (ILR.GE.ILRE) GO TO 480
ILR=IPIC(1)+1
ICELL=IPIC(2)+1
ICHGR=IPIC(3)+1
ISPD=IPIC(4)
IPD=IPIC(5)
GO TO 520

C 480 IF (ICELL.GE.ICELLE) GO TO 490
ILR=IDB(7)+1
ICELL=IPIC(2)+1
ICHGR=IPIC(3)
ISPD=IPIC(4)
IPD=IPIC(5)
GO TO 520

C 490 IF (ICHGR.GE.ICHGRE) GO TO 500
ILR=IOB(7)+1
ICELL=IDB(1)+1
ICHGR=IPIC(3)+1
ISPD=IPIC(4)
IPD=IPIC(5)
GO TO 520

DO 510 I=1,5
ICHOOSE(I)=-1
RETURN

GO TO 520

** COMPUTE SELECTION PARAMETERS FOR SERIES LOAD REGULATION
** THIS IS FOR THE LOAD REGULATORS, BATTERY, BATTERY CHARGER AND
** SIZING THE SOLAR POWER DISTRIBUTOR AND POWER DISTRIBUTOR.

NLR IS THE NUMBER OF LOAD REGULATORS REQUIRED
NLR=2

** DETERMINE EXCESS ARRAY POWER FOR REGULATION
PS=(PL/ETAR)*(1.+TEDTS*(1./ETAD*ETAC*ETAE))
PBOB=PS/((1.-DELR)*(1.-DELF)*(1.-DELT)*(1.-DELI)*(1.-DELM))
PEXCES=PBOB-FLMIN

** DETERMINE SELECTION PARAMETERS FOR LOAD REGULATORS
IF (ITER .GE. 1 .AND. NCHOSE(1) .GE. NLR) NLR=NCHOSE(1)
FLR=PL/(ETALR*NLR)

SET VOLTAGES FOR THIS DESIGN
VDB=27.
VBM=23.

** SET UP BATTERY SELECTION PARAMETERS

DETERMINE REQUIRED CAPACITIES

CR=(PL*TE/3600.)/(LMEDO*ETAD)
IF (CR .LT. 2300.0) GO TO 522
NB = 4
IF (CR .LT. 4700.0) GO TO 522
NB = 6
IF (CR .LT. 7050.0) GO TO 522
NB = 8
IF (CR .LT. 9400.0) GO TO 522
NB = 10

522 NB = AMAX0(NB, NCHOSE(2))
CA=CR/VDB

DETERMINE MINIMUM INSTALLED CAPACITY
CI=CA*VFD

DETERMINE NUMBER OF CELLS IN SERIES (TO BE SUPPLIED TO REL)
NC=VBM/VC

CISTAR IS SELECTION PARAMETERS ON CELLS
CISTAR=CI/NB

** CHARGER SELECTION PARAMETER

* DETERMINE CHARGE CURRENT RATING REQUIRED FOR THE BATTERY CHARGER *

CCELL=CISTAR
ICH=CCELL/CHMINT

**

** COMPARE THE HARDWARE PARAMETER TO THE SELECTION PARAMETER **

IF (DATA(6,ILR).GE.PLR.AND.DATA(6,ICELL).GE.CISTAR.AND.DATA(6,ICH).GE.ICH) GO TO 580

IF (ILR.GE.ILRE) GO TO 540
ILR=ILR+1
GO TO 530
C540  IF (ICELL.GE.ICELLE) GO TO 550
    ILR=IDB(7)+1
    ICELL=ICELL+1
    GO TO 530
550  IF (ICHGR.GE.ICHGRE) GO TO 560
    ILR=IDB(7)+1
    ICELL=IDB(1)+1
    ICHGR=ICHGR+1
    GO TO 550
560  DO 570 I=1,5
    ICHOOSE(I)=-I
    RETURN
570  ETA=0ATAB(7,ILR)
    ETA=ETA+1
    FLRD=PL*(1./ETA*ILR-1.)
    ETAE=OATAB(7,ICELL)
    ETAC=OATAB(7,ICHGR)
    VCELL=OATAB(25,ICELL)
    WCELL=OATAB(23,ICELL)
    WB=NC*WCELL*K2
    VB=NC*VCELL*K1
    WBT=WB+NB
    WBT=WBT+NB
    WCH=WB
    BATCAP=NB*OATAB(6,ICELL)
780  ICHOOSE(1)=-OATAB(1,ILR)
    ICHOOSE(2)=-OATAB(1,ICELL)
    ICHOOSE(3)=-OATAB(1,ICHGR)
    ICHOOSE(4)=-OATAB(1,ISP)
    ICHOOSE(5)=-OATAB(1,IPD)
581  IF (NCHOOSE(1).GE. NLR) GO TO 581
    NCHOOSE(1)=NLR
582  IF (NCHOOSE(2).GE. NB) GO TO 582
    NCHOOSE(2)=NB
583  IF (NCHOOSE(3).GE. NCH) GO TO 583
    NCHOOSE(3)=NCH
584  IF (NCHOOSE(4).GE. NSPD) GO TO 584
    NCHOOSE(4)=NSPD
585  IF (NCHOOSE(5).GE. NPD) GO TO 585
    NCHOOSE(5)=NPD
840  IPIC(1)=ILR
    IPIC(2)=ICELL
    IPIC(3)=ICHGR
    IPIC(4)=ISP
    IPIC(5)=IPD
850  WT=NLR*DATAB(23,ILR)+WBT+NB*DATAB(23,ICELL)+NSPD*DATAB(23,ISP)+...
* NPD*DATAB(23,IPC)+WT
VOL=NLR*DATAB(24,ILR)+WBT+NCH*DATAB(24,ICHGR)+NSPD*DATAB(24,ISPD)
* +NPD*DATAB(24,IPD)+VOL

** SOLAR ARRAY SIZING
ICNF=NCONF(5)
FCONW=WT-WTIN-WST
GO TO (600,610,600,610,600,610), ICONF

** ORIENTED PADDLE SOLAR ARRAY (NON-SPINNING)
FW=7.3
LMBDG=1,
GO TO 670

** BODY MOUNTED, BOX SHAPE, NON-SPINNING
FW=3.4
LMBDG=1.,
GO TO 670

** BODY MOUNTED CYLINDER SPINNING
FW=3.4
LMBDG=1./PIE
GO TO 670

** BODY MOUNTED CYLINDER NON-SPINNING
FW=3.4
LMBDG=2./PIE
GO TO 670

** BODY MOUNTED SPHERE SPINNING
FW=3.4
LMBDG=25
GO TO 670

IF (ICCNF *GE. 3) GO TO 640

IF (ICCNF *GE. 3) GO TO 660

IF (ICCNF *GE. 3) GO TO 680

IF (ICCNF *GE. 3) GO TO 700

** BODY MOUNTED SPHERE NON-SPINNING

FW=3.4
LMBDG=.5

COMPUTE ENERGY BALANCE EQUATION
PS=PL/ETAR*(1.+TETS*(1./(ETAD*ETAC*ETA)))

COMPUTE SIZING FACTOR
FS=LMBDG*LMBDP*((1.-DELR)*(1.-DELF)*(1.-DELT)*(1.-DELI))

COMPUTE ARRAY AREA
Ai=PS/(FS*SOL*ETAI)

COMPUTE ARRAY WEIGHT
WATE=A1*FW

CONVERT TO ENGLISH FROM METRIC
AREA=A1*10.76426265
WATE=WATE*2.20462

RETURN

END
SUBROUTINE AUXPRC

COMMON /USERII/ APOGEE, CUSRAT, DMAX, EEW(9), EPM, 022875
EQM2XL, EQM2YL, EQM3L, EQM3LT, ITHST, IAGNCY, 025675
IDEBUG, ISATOR, MB12SH, OPTEMP, ORBINNC, PERIGE, 022575
MICR, RELME, SPEC(6), XDOM1, XGSAI, 022575
OPT=2
FTN 4.2.1+383 03/27/75 21.38.52

DATA XMR /1.5/

PRINT 9000, ACTHST, THST
FORMAT (1X,9HACTHIT, = ,Eii.4v1X,8HTTHST = ,*4I.4)

IF(NCONF(2).LT.0) GO TO 1

THIS IS COLD GAS CONFIGURATION

DETERMINE MAXIMUM THRUST FROM SANDC

FMAX= AMAX1(ACTHST,THST)

IF(FMAX.LT.50.0 AND. TOITM.PT.50800.) GO TO 1
THIS IS NOT AN ACCEPTABLE CONFIGURATION

ICHOOSE(1) = -1
RETURN

1 CONTINUE

IF (ITER .NE. 0) GO TO 3

INITIALIZE ICHOOSE, NCHOOSE, IERR AND SELECT HARDWARE NOT Sized
I.E., THE FILL AND VENT Valve AND RELIEF Valve

2 NCHOOSE(1) = 0
ICHOOSE(1) = 0
IERR = 0
NCHOOSE(2) = 2
NCHOOSE(3) = 4
NCHOOSE(4) = 9
DO 299 I = 5, 8
299 NCHOOSE(I) = 1

II = ICB(5) + 1
JJ = ICB(6) + 1
ICHOOSE(7) = ICAB(1, II)
ICHOOSE(8) = ICAB(1, JJ)

THRUSTER SELECTION

FIRST CHECK TO SEE IF THERE IS AN ACCEPTABLE THRUSTER IN THE DATA BASE

JIE = ICB(1)
J1 = 1
THrust = ICAB(6, J1)
IF (THRUST .GE. FMAX) GO TO 12
IF (J1 .EQ. JIE) GO TO 11
J1 = J1 + 1
GO TO 10

NO ACCEPTABLE THRUSTERS

11 ICHOOSE(1) = -1
RETURN

AT LEAST ONE ACCEPTABLE THRUSTER

SELECT PNEUMATIC ATTITUDE AND CONTROL THRUSTERS

FIRST DETERMINE SET OF ALL THRUSTERS WHICH SATISFY THE INEQUALITY
THRUST GE ACTHST
I= 1
J1= 1
13 THRUST= DATAB(6,J1)
   IF(THRUST.GE.ACTHST) GO TO 15
14. IF(J1.EQ.JIE) GO TO 16
   J1= J1 + 1
   GO TO 13
15 IACCPT(I)= J1
   I= I + 1
   GO TO 14
16 CONTINUE
   IMAX= I - 1
125 CHOOSE THAT THRUSTER FROM THE ABOVE SET WHICH MINIMIZES THE
   QUANTITY, ABS(THRUST - ACTHST)
130 I= I ACCPT(I)
   THRUST= DATAB(6,J1)
   DIFOLD=ABS(THRUST - ACTHST)
17 ICHOOSE(1)= DATAB(1,J1)
   JSAVE=J1
   IF(I.EQ.IMAX) GO TO 20
18 I= I + 1
   J1= I ACCPT(I)
   THRUST= DATAB(6,J1)
   CIFNEW= ABS(THRUST - ACTHST)
   IF(CIFNEW.LE.DIFOLD) GO TO 19
   I= L1,IMAX) GO TO 18
19 DIFOLD= CIFNEW
   GO TO 17
20 J1=JSAVE
   Ti(1)=DATAB(6,J1) 
145 SELECT PNEUMATIC TRANSLATIONAL THRUSTERS USING ABOVE PROCEDURE
150 I= 1
160 J2= I ACCPT(I)
   THRUST= DATAB(6,J2)
   IF(THRUST.GE.ITHST) GO TO 23
17 IF(J2.EQ.JIE) GO TO 24
   J2= J2 + 1
   GO TO 21
22 I= I + 1
   GO TO 22
23 IACCPT(I)= J2
24 CONTINUE
   IMAX= I - 1
25 ICHOOSE(2)= DATAB(1,J2)
   JSAVE=J2
   IF(I.EQ.IMAX) GO TO 28
26 I= I + 1
J2 = IACPT(I)
THUST = DATAB(6, J2)
CIFNEW = ABS(THUST - TTHST)
IF(CIFNEW.LE.DIFOL) GO TO 27
IF(I.I.EJ MAX) GO TO 28
GO TO 26
27 DIFOLO = DIFNEW
GO TO 25
28 J2 = JSAVE
T(2) = DATAB(6, J2)

THUSTERS HAVE BEEN SELECTED
SET NUMBER OF EACH TYPE OF THRUSTER

CHECK TO SEE IF CYCLE LIFE REQUIREMENT IS SATISFIED
IFERR = 0
IF(DATAB(7, J1).LT.CLIFE) IERR = 1
IF(DATAB(7, J2).LT.CLIFE) IERR = IERR + 10
IERR = 1 IMPLIES THAT THE CYCLE LIFE OF THE ATTITUDE AND CONTROL
THUSTERS IS TOO SHORT, IERR = 10 IMPLIES THAT THE CYCLE LIFE OF
THE TRANSLATIONAL THUSTERS IS TOO SHORT, IERR = 11 IMPLIES THAT
THE CYCLE LIVES OF BOTH THRUSTERS ARE TOO SHORT.

PRELIMINARY CALCULATIONS FOR SELECTION OF PNEUMATIC ISOLATION
VALVES AND FILTERS
PTI = DATAB(8, J1)
RHO = 1.02E-7*PTI
WOOTPR = (3*ACTHST + 2*TTHST)/65.
CDAISO = WOOTPR/SQRT(200*RHO/1.29E-3)
RMAX = 200/WOOTPR**2

SET LAST EQUIPMENT INDICES
J3E = IDB(2)
J4E = IDB(3)
J5E = IDB(4)
J6E = IDB(5)

DETERMINE HARDWARE INDICES
TO 30 I = 1
IF(IPIC(I).NE.0) GO TO 31
JO 4

30 CONTINUE
GO TO 5
IF(IPIC(1).LT.J3E) GO TO 6
IF(IPIC(2).LT.J4E) GO TO 7
IF(IPIC(3).LT.J5E) GO TO 8
IF(IPIC(4).LT.J6E) GO TO 9

NO ACCEPTABLE COMBINATIONS

Preliminary calculations for selection of pneumatic isolation valves and filters:
ICHOSE(1) = -1
RETURN

4 CONTINUE
J3 = IDB(1) + 1
J4 = IDB(2) + 1
J5 = IDB(3) + 1
J6 = IDB(4) + 1
GO TO 1200

5 CONTINUE
J3 = IPIC(1)
J4 = IPIC(2)
J5 = IPIC(3)
J6 = IPIC(4)
GO TO 1200

6 CONTINUE
J3 = IPIC(1) + 1
J4 = IPIC(2)
J5 = IPIC(3)
J6 = IPIC(4)
GO TO 1200

7 CONTINUE
J3 = IDB(1) + 1
J4 = IDB(2) + 1
J5 = IPIC(3) + 1
J6 = IPIC(4)
GO TO 1200

8 CONTINUE
J3 = IDB(1) + 1
J4 = IDB(2) + 1
J5 = IDB(3) + 1
J6 = IPIC(4) + 1
GO TO 1200

1200 CONTINUE
THE HARDWARE INDICES ARE SET

32 IF (DATAB(7, J3) .LT. CDAISO .OR. DATAB(7, J4) .GT. RMAX) GO TO 33

33 ISOLATION VALVE AND FILTER ARE ACCEPTABLE

DELPI = (1.29E-3/RHO)*(WDOTPR/DATAB(7, J3))**2
DELPFI = DATAB(7, J4)*WDOTPR**2
ICHOSE(3) = DATAB(1, J3)
ICHOOSE(4) = DATAB(1, J4)

Preliminary Calculations for Selection of Regulator and Tank

PREG = PTI + 2 * DELPS + DELPFI
CDAREG = VDOTPR/SORT(5600, *PREG/1.27E4)

WPR = 1.4 * CDAREG/65.

ACSWP = WPR
VPRT = 3.4 * WPR/28.

IF (PREG < DATAB(8, JS). OR. PREG > DATAB(9, JS). OR. DATAB(7, JS) < CDAREG. OR. DATAB(6, JS) < VPRT) GO TO 33

REGULATOR AND TANK ARE ACCEPTABLE
ICHOOSE(5) = DATAB(1, J5)
ICHOOSE(6) = DATAB(1, J6)

TNKWT = DATAB(23, J6)

SIZE PLUMBING AND CONNECTORS

PCWATE = 2 * DATAB(23, J6) * NCHOOSE(6)

STORE LAST INDICES ACCEPTABLE

IPIC(1) = J3
IPIC(2) = J4
IPIC(3) = J5
IPIC(4) = J6

N(7) = II
N(9) = JJ
N(1) = J1
N(2) = J2
N(3) = J3
N(4) = J4
N(5) = J5
N(6) = J6

DO 322 I = 1, 8
J = N(I)

WT = WT + NCHOOSE(I) * DATAB(23, J)

VOL = VOL + NCHOOSE(I) * DATAB(24, J)

PL = PL + NCHOOSE(I) * DATAB(16, J)

PLMIN = PLMIN + NCHOOSE(I) * DATAB(18, J)

CONTINUE

322 CONTINUE

ORINT = WT + PCWATE + WEIGHT(I)

WT = WT + ACSWP + PCWATE

RETURN

HARDWARE SELECTION NOT ACCEPTABLE - INCREMENT HARDWARE INDICES

33 CONTINUE

IF (J3 < J3E) GO TO 34
IF (J4 < J4E) GO TO 35
IF (J5 < J5E) GO TO 36
IF (J6 < J6E) GO TO 37

NO ACCEPTABLE HARDWARE COMBINATION
ICHOSE(1) = -1
RETURN

J3 = J3 + 1
GO TO 32
J3 = IDB(1) + 1
J4 = J4 + 1
GO TO 32
J3 = IDB(1) + 1
J4 = IDB(2) + 1
J5 = J5 + 1
GO TO 32
J3 = IDB(1) + 1
J4 = IDB(2) + 1
J5 = IDB(3) + 1
J6 = J6 + 1
GO TO 32
CONTINUE

IF(NCONF(2).EQ.3) GO TO 62

THIS IS MONOPROPELLANT CONFIGURATION
Determine maximum thrust from ANDC
FMAX = AMAX1(ACTHST,TTHST)
IF(FMAX.LT.10000..AND.TOTIMP.LT.20000..AND.TOTIMP.GE.10000.) GO TO 39
THIS IS NOT AN ACCEPTABLE CONFIGURATION
ICHOSE(1) = -1
RETURN

CONTINUE
IF(ITER.NE.0) GO TO 42

INITIALIZE ICHOOSE,NCHOOSE,IERR AND SELECT HARDWARE NOT SIZED
IE., THE RELIEF VALVE,FILL AND VENT VALVE AND FILL AND DRAIN VALVE

DO 40 I = 1,14
ICHOSE(I) = 0
NCHOOSE(I) = 0
IERR = 0
NCHOOSE(1) = 6
NCHOOSE(2) = 9
NCHOOSE(3) = 4
NCHOOSE(4) = 9
DO 41 I = 5,11
NCHOOSE(I) = 1
II = IDB(5) + 1
JJ = IDB(6) + 1
KK = IDB(11) + 1
ICHOSE(9) = DATAB(1, JJ)
ICHOSE(10) = DATAB(1, KK)
ICHOSE(11) = DATAB(1,II)

CONTINUE

THRUSTER SELECTION
FIRST CHECK TO SEE IF THERE IS AN ACCEPTABLE THRUSTER IN THE DATA EASE

J1E= IDB(8)
J1 = IDB(7) + 1
THUST= DATAB(6,J1)
IF(THUST.GE.MAX) GO TO 120
IF(J1.EQ.J1E) GO TO 110
J1 = J1 + 1
GO TO 100

NO ACCEPTABLE THRUSTERS

ICHOSE(11) = -1
RETURN

SELECT AT LEAST ONE ACCEPTABLE THRUSTER

FIRST DETERMINE SET OF ALL THRUSTERS WHICH SATISFY THE INEQUALITY THRUST GE ACTHST

I= 1
J1 = IDB(7) + 1
THUST= DATAB(6,J1)
IF(THUST.GE.ACTHST) GO TO 150
IF(J1.EQ.J1E) GO TO 160
J1 = J1 + 1
GO TO 130

IACCPT(I) = J1
I = I + 1
GO TO 140

CONTINUE
IMAX= I - 1

CHOOSE THAT THRUSTER FROM THE ABOVE SET WHICH MINIMIZES THE QUANTITY, ABS(THRUST - ACTHST)

I=1
J1 = IACCPT(I)
THUST= DATAB(6,J1)
DIFOLO= ABS(THRUST - ACTHST)
ICHOSE(I1) = DATAB(I,J1)
SAVE=J1
IF(I.EQ.IMAX) GO TO 200
I = I + 1
J1 = IACCPT(I)
THUST= DATAB(6,J1)
DIFNEW = ABS(THRST - ACSTST)
IF(DIFNEW .LE. DIFOLD) GO TO 190
IF(I .LT. IMAX) GO TO 180
GO TO 200
DIFOLD = DIFNEW
GO TO 170
200 J1 = JSAVE
T(1) = DATAB(6, J1)
GO TO 210

SELECT PNEUMATIC TRANSLATIONAL THRUSTERS USING ABOVE PROCEDURE

I = 1
J2 = I0B(7) + 1
THRST = DATAB(6, J2)
IF(THRST .GE. TTHST) GO TO 230
J2 = J2 + 1
GO TO 210
230 IACCEPT(I) = J2
I = I + 1
GO TO 220
240 CONTINUE
IMAX = I - 1
I = 1
J2 = IACCEPT(I)
THRST = DATAB(6, J2)
DIFOLD = ABS(THRST - TTHST)
250 ICHOOSE(2) = DATAB(1, J2)
JSAVE = J2
IF(I .EQ. IMAX) GO TO 280
I = I + 1
GO TO 220
260 J2 = IACCEPT(I)
THRST = DATAB(6, J2)
DIFNEW = ABS(THRST - TTHST)
IF(DIFNEW .LE. DIFOLD) GO TO 270
IF(I .LT. IMAX) GO TO 250
GO TO 280
270 DIFOLD = DIFNEW
GO TO 250
280 J2 = JSAVE
T(2) = DATAB(6, J2)

THURSTERS HAVE BEEN SELECTED
SET NUMBER OF EACH TYPE OF THRUSTER

CHECK TO SEE IF CYCLE LIFE REQUIREMENT IS SATISFIED

IERR = 0
IF(DATAB(7, J1) .LT. CLIFE) IERR = 1
IF(DATAB(7, J2) .LT. CLIFE) IERR = IERR + 10

IERR = 1 IMPLIES THAT THE CYCLE LIFE OF THE ATTITUDE AND CONTROL
THRUSTERS IS TOO SHORT. IERR = 10 IMPLIES THAT THE CYCLE LIFE OF
THE TRANSLATIONAL THRUSTERS IS TOO SHORT. IERR = 11 IMPLIES THAT.
THE CYCLE LIVES OF BOTH THRUSTERS ARE TOO SHORT

PRELIMINARY CALCULATIONS FOR SELECTION OF MONOFROPPELLANT ISOLATION VALVES AND FILTERS

\[ \text{RHOF} = 0.36 \]
\[ \text{WDOF} = (3 \ast \text{ACTHST} + 2 \ast \text{TTTST}) / 200 \]
\[ \text{IF} (\text{NCONF}(1), \text{EQ}, 1, \text{WDOF} = \text{TTTST} / 200) \]
\[ \text{CDAYS} = \text{WDOF} / \text{SORT}(5 \ast \text{RHOF} / 1.29 \times 10^{-3}) \]
\[ \text{MAX} = 5 \ast \text{WDOF} \times 2 \]

SET LAST EQUIPMENT INDICES

\[ \text{J3E} = \text{IDB}(9) \]
\[ \text{J4E} = \text{IDB}(10) \]
\[ \text{J5E} = \text{IDB}(4) \]
\[ \text{J6E} = \text{IDB}(2) \]
\[ \text{J7E} = \text{IDB}(11) \]
\[ \text{J8E} = \text{IDB}(5) \]

DETERMINE HARDWARE INDICES

\[ \text{DO } 43 \text{ I=1,9} \]
\[ \text{IF} (\text{IPIC}(1), \text{NE}, 0) \text{ GO TO 44} \]
\[ \text{CONTINUE} \]
\[ \text{GO TO 45} \]
\[ \text{IF} (\text{ITER}, \text{NE}, 0) \text{ GO TO 46} \]
\[ \text{IF} (\text{IPIC}(1), \text{LT}, \text{J3E}) \text{ GO TO 47} \]
\[ \text{IF} (\text{IPIC}(2), \text{LT}, \text{J4E}) \text{ GO TO 48} \]
\[ \text{IF} (\text{IPIC}(3), \text{LT}, \text{J5E}) \text{ GO TO 49} \]
\[ \text{IF} (\text{IPIC}(4), \text{LT}, \text{J6E}) \text{ GO TO 50} \]
\[ \text{IF} (\text{IPIC}(5), \text{LT}, \text{J7E}) \text{ GO TO 51} \]
\[ \text{IF} (\text{IPIC}(6), \text{LT}, \text{J8E}) \text{ GO TO 52} \]

NO ACCEPTABLE COMBINATIONS

\[ \text{ICHOSE}(1) = -1 \]
\[ \text{RETURN} \]

CONTINUE

\[ \text{J3} = \text{IDB}(6) + 1 \]
\[ \text{J4} = \text{IDB}(9) + 1 \]
\[ \text{J5} = \text{IDB}(3) + 1 \]
\[ \text{J6} = \text{IDB}(1) + 1 \]
\[ \text{J7} = \text{IDB}(10) + 1 \]
\[ \text{J8} = \text{IDB}(4) + 1 \]
\[ \text{GO TO 53} \]

CONTINUE

\[ \text{J3} = \text{IPIC}(1) \]
\[ \text{J4} = \text{IPIC}(2) \]
\[ \text{J5} = \text{IPIC}(3) \]
\[ \text{J6} = \text{IPIC}(4) \]
J8 = IPIC(6)
GO TO 53

C
47 J3 = IPIC(1) + 1
J4 = IPIC(2)
J5 = IPIC(3)
J6 = IPIC(4)
J7 = IPIC(5)
J8 = IPIC(6)
GO TO 53

C
48 J3 = IDB(8) + 1
J4 = IPIC(2) + 1
J5 = IPIC(3) + 1
J6 = IPIC(4)
J7 = IPIC(5)
J8 = IPIC(6)
GO TO 53

C
49 J3 = IDB(8) + 1
J4 = IDB(9) + 1
J5 = IPIC(3) + 1
J6 = IPIC(4)
J7 = IPIC(5)
J8 = IPIC(6)
GO TO 53

C
50 J3 = IDB(8) + 1
J4 = IDB(9) + 1
J5 = IDB(3) + 1
J6 = IPIC(4) + 1
J7 = IPIC(5)
J8 = IPIC(6)
GO TO 53

C
51 J3 = IDB(8) + 1
J4 = IDB(9) + 1
J5 = IDB(3) + 1
J6 = IDB(1) + 1
J7 = IPIC(5) + 1
J8 = IPIC(6)
GO TO 53

C
52 J3 = IDB(8) + 1
J4 = IDB(9) + 1
J5 = IDB(3) + 1
J6 = IDB(1) + 1
J7 = IDB(10) + 1
J8 = IPIC(6) + 1

C
53 CONTINUE
C
THE HARDWARE INDICES ARE SET

C
54 IF(DATAB(7,J3).LT.COAISO.OR.DATAB(7,J4).GT.RMAX) GO TO 55
C
FUEL CIRCUIT ISOLATION VALVES AND FILTERS ARE ACCEPTABLE
PRELIMINARY CALCULATIONS FOR SELECTION OF PNEUMATIC REGULATOR

\[ \text{DELPIS} = (1.29 \times 10^{-3}/\text{RHOF}) \times \text{WDOTF}/\text{DATAB}(7,J3) \times 2 \]
\[ \text{DELPFI} = \text{DATAB}(7,J4) \times \text{WDOTF} \times 2 \]
\[ \text{ICHOSE(3)} = \text{DATAB}(1,J3) \]
\[ \text{ICHOSE(4)} = \text{DATAB}(1,J4) \]
\[ \text{IPIC(1)} = J3 \]
\[ \text{IPIC(2)} = J4 \]

IF (\text{PREG} \lt \text{DATAB}(8,J5) \text{ OR } \text{PREG} \gt \text{DATAB}(9,J5) \text{ OR } \text{DATAB}(7,J5) \text{ LT } \text{COAREG}) \text{ GO TO 55}

REGULATOR IS ACCEPTABLE
\[ \text{ICHOSE(5)} = \text{DATAB}(1,J5) \]
\[ \text{IPIC(3)} = J5 \]

PRELIMINARY CALCULATIONS FOR SELECTION OF PNEUMATIC ISOLATION VALVE

\[ \text{RHOPR} = 3000 \times 1.02 \times 10^{-7} \]
\[ \text{COAISO} = \text{WDOTP/QRRT(200, \times \text{RHOPR} / 1.29 \times 10^{-3})} \]
IF (\text{DATAB}(7,J6) \lt \text{COAISO}) \text{ GO TO 55}

PNEUMATIC ISOLATION VALVE IS ACCEPTABLE
\[ \text{ICHOSE(6)} = \text{DATAB}(1,J6) \]
\[ \text{IPIC(4)} = J6 \]

PRELIMINARY CALCULATIONS FOR SELECTION OF FUEL TANK AND PNEUMATIC TANK

\[ \text{WF} = 1.1 \times \text{TOTIMP} / 200. \]
\[ \text{VF} = \text{WF} / 0.36 \]
\[ \text{VFT} = 1.1 \times \text{VF} \]
\[ \text{WPRT} = \text{PFT(3000) - 2 \times \text{PFT}} \]
\[ \text{WPRT} = 0.0085 \times \text{WPRT} \]
\[ \text{ACSNP} = \text{WF} + \text{WPRT} \]

SELECT FUEL TANK.
IF (\text{DATAB}(7,J7) \lt \text{PFT}) \text{ GO TO 55}

IF (\text{MICRO} = 2) \text{ GO TO 5005}
IF (\text{DATAB}(7,J7) \lt \text{GE.PFT}) \text{ AND } \text{DATAB}(6,J7) \lt \text{GE.VFT) GO TO 5001}

AT LEAST ONE FUEL TANK IN DATA BASE WHICH SATISFIES PRESSURE DESIGN CRITERIA.
\[ \text{J7SAVE} = J7 \]
VFTMAX = DATAB(6,J7)

CHECK TO SEE IF THERE IS AT LEAST ONE TANK IN DATA BASE WHICH SATISFIES BOTH PRESSURE AND VOLUME DESIGN CRITERIA.

JJ7 = IDDB(10) + 1

5000 IF (DATAB(6,JJ7) .GE. VFT AND DATAB(7,J7) .GE. PFT) GO TO 55

IF (JJ7 .EQ. J7E) GO TO 5002

JJ7 = JJ7 + 1

GO TO 5000

C 5002 CONTINUE

NO TANK IN DATA BASE WHICH SATISFIES BOTH THE PRESSURE AND VOLUME DESIGN CRITERIA. SELECT TANK WITH LARGEST VOLUME WHICH SATISFIES PRESSURE DESIGN CRITERIA.

JJ7 = IDDB(10) + 1

5003 IF (DATAB(7,JJ7) .LT. PFT OR DATAB(6,JJ7) .LT. VFTMAX) GO TO 5004

VFTMAX = DATAB(6,JJ7)

J7SAVE = JJ7

C 5004 JJ7 = JJ7 + 1

IF (JJ7 .LT. J7E + 1) GO TO 5003

J7 = J7SAVE

5005 NCHOOSE(7) = VFT / DATAB(6,J7) + 5

IF (NCHOOSE(7) .LT. J7) NCHOOSE(7) = 1

GO TO 5005

C 5001 CONTINUE

NCHOOSE(7) = DATAB(1,J7)

IPIC(5) = J7

TNKWT = DATAB(23,J7)

SELECT PNEUMATIC TANK

IF (DATAB(6,J8) .LT. VPRT) GO TO 55

NCHOOSE(8) = DATAB(1,J8)

IPIC(6) = J8

SELECT PLUMBING AND CONNECTORS

PCWATE = .2*(DATAB(23,J7)*NCHOOSE(7) + DATAB(23,J8)*NCHOOSE(8))
CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE
THIS IS A BIPROPellant CONFIGURATION

IF(TOTIHP.GE.50800.) GO TO 63

THIS IS NOT AN ACCEPTABLE CONFIGURATION

ICHOSE(1) = -1
RETURN

CONTINUE

IF(ITER.NE.0) GO TO 65

INITIALIZE ICHOSE,NCHOSE,IERR AND SELECT HARDWARE NOT SIZED
I=1, IERR = 0
I=0, I=0, FILL AND DRAIN VALVES, FILL AND VENT VALVE AND RELIEF VALVE

DO 64 I=1,14
ICHOSE(I) = 0
NCHOSE(I) = 0
IERR = 0
NCHOSE(1) = 6
NCHOSE(2) = 2
NCHOSE(3) = 3
NCHOSE(4) = 3
NCHOSE(5) = 4
NCHOSE(6) = 4
DO 65 I=7,11
NCHOSE(I) = 1
NCHOSE(12) = 2
NCHOSE(13) = 1
NCHOSE(14) = 1
II = I0B(5) + 1
JJ = I0B(6) + 1
KK = I0B(16) + 1
ICHOSE(12) = DATAB(1,KK)
ICHOSE(13) = DATAB(1,II)
ICHOSE(14) = DATAB(1,JJ)

CONTINUE

THRUSTER SELECTION

FIRST CHECK TO SEE IF THERE IS AN ACCEPTABLE THRUSTER IN THE DATA BASE

FMAX=A MAX1(AC THST,THST)
JJ = I0B(13)
J1 = I0B(12) + 1
101 THRUST= DATAB(6,J1)
IF(THRUST.GE.FMAX) GO TO 121
IF(J1.EQ.J1E) GO TO 111
J1 = J1 + 1
GO TO 101

NO ACCEPTABLE THRUSTERS
C
111 ICHOOSE(1) = -1
RETURN

121 CONTINUE

SELECT PNEUMATIC ATTITUDE AND CONTROL THRUSTERS

FIRST DETERMINE SET OF ALL THRUSTERS WHICH SATISFY THE INEQUALITY
THRUST GE ACTHST

I= 1
J1 = IDB(12) + 1
THRUST= DATAB(6, J1)
IF(THRUST .GE. ACTHST) GO TO 151

141 IF(J1 .EQ. JIE) GO TO 161
J1 = J1 + 1
GO TO 131

151 IACCP(1) = J1
I = I + 1
GO TO 141

161 CONTINUE

IMAX = I - 1

CHOOSE THAT THRUSTER FROM THE ABOVE SET WHICH MINIMIZES THE
QUANTITY, ABS(THRUST - ACTHST)

I=1
J1 = IACCP(I)
THRUST= DATAB(6, J1)
DIFOLD = ABS(THRUST - ACTHST)

171 ICHOOSE(1) = DATAB(1, J1)
JSAVE = J1
IF(I .EQ. IMAX) GO TO 201

181 I = I + 1
J1 = IACCP(I)
THRUST= DATAB(6, J1)
DIFNEW = ABS(THRUST - ACTHST)
IF(DIFNEW .LE. DIFOLD) GO TO 191
IF(I .LT. IMAX) GO TO 181
GO TO 201

191 DIFOLD = DIFNEW
GO TO 171

201 J1 = JSAVE
T(I) = DATAB(6, J1)

SELECT PNEUMATIC TRANSLATIONAL THRUSTERS USING ABOVE PROCEDURE

I=1
J2 = IDB(12) + 1
THRUST= DATAB(6, J2)
IF(THRUST .GE. TTHST) GO TO 231

221 IF(J2 .EQ. JIE) GO TO 241
J2 = J2 + 1
GO TO 211
IACCPT(I) = J2
I = I + 1
GO TO 221

CONTINUE
IMAX = I - 1
I = 1
J2 = IACCPT(I)
THRUST = DATAB(6, J2)
DIFOLD = ABS(THRUST - TTHST)
ICHOOSE(2) = DATAB(1, J2)
JSAVE = J2
IF (I = EG) GO TO 281

I = I + 1
J2 = IACCPT(I)
THRUST = DATAB(6, J2)
DIFNEW = ABS(THRUST - TTHST)
IF (I .EQ. IMAX) GO TO 281
GO TO 281

IF (DIFNEW .LE. DIFOLO) GO TO 271
IF (I .LT. IMAX) GO TO 261
GO TO 281

CIFOLD = DIFNEW
C
C THRUSTERS HAVE BEEN SELECTED
C SET NUMBER OF EACH TYPE OF THRUSTER
C
C CHECK TO SEE IF CYCLE LIFE REQUIREMENT IS SATISFIED

IERR = 0
IF (DATAB(7, J1) .LT. CLIFE) IERR = 1
IF (DATAB(7, J2) .LT. CLIFE) IERR = IERR + 10

IERR = 1 IMPLIES THAT THE CYCLE LIFE OF THE ATTITUDE AND CONTROL
THRUSTERS IS TOO SHORT. IERR = 10 IMPLIES THAT THE CYCLE LIFE OF
THE TRANSLATIONAL THRUSTERS IS TOO SHORT. IERR = 11 IMPLIES THAT
THE CYCLE LIVES OF BOTH THRUSTERS ARE TOO SHORT

PRELIMINARY CALCULATIONS FOR SELECTION OF BIPROPELLANT ISOLATION
VALVES AND FILTERS

RHOF = .032
RHOD = .054
WDOTF = (3 * ACHST + 2 * TTHST) / (260 * (1 + XMR))
WDOTO = WDOTF * XMR
CDAISO = WDOTF / SQRT(50 * RHOF / 1.29E-3)
CDAISF = WDOTO / SQRT(50 * RHCO / 1.29E-3)

C
C SET LAST EQUIPMENT INDICES
C
J3E = IDB(14)
J4E = IDB(14)
DETERMINE HARDWARE INDICES

DO 66 I=1,9
   IF(IPIC(I).NE.0) GO TO 67
   CONTINUE
GO TO 66
66 CONTINUE
67 IF(ITER.NE.0) GO TO 69
   IF(IPIC(1).LT.J3E) GO TO 70
   IF(IPIC(2).LT.J5E) GO TO 71
   IF(IPIC(3).LT.J7E) GO TO 72
   IF(IPIC(4).LT.J9E) GO TO 73
   IF(IPIC(5).LT.JIE) GO TO 74
   IF(IPIC(6).LT.J6E) GO TO 75
   IF(IPIC(7).LT.J9E) GO TO 76
   IF(IPIC(8).LT.J10E) GO TO 77
   IF(IPIC(9).LT.JI1E) GO TO 78
   GO TO 79
69 CONTINUE
   J3= IPIC(1) + 1
   J4= IPIC(2) + 1
   J5= IPIC(3) + 1
   J6= IPIC(4) + 1
   J7= IPIC(5) + 1
   J8= IPIC(6) + 1
   J9= IPIC(7) + 1
   J10= IPIC(8) + 1
   J11= IPIC(9) + 1
   GO TO 79
70 CONTINUE
   J3= IPIC(1) + 1
   J4= IPIC(2) + 1
   J5= IPIC(3) + 1
   J6= IPIC(4) + 1
   RETURN
J7 = IPIC(5)
J6 = IPIC(6)
J9 = IPIC(7)
J10 = IPIC(8)
J11 = IPIC(9)
GO TO 79

C

71 J3 = IDB(13) + 1
J4 = IPIC(2) + 1
J5 = IPIC(3) + 1
J6 = IPIC(4)
J7 = IPIC(5)
J8 = IPIC(6)
J9 = IPIC(7)
J10 = IPIC(8)
J11 = IPIC(9)
GO TO 79

72 J3 = IDB(13) + 1
J4 = IDB(13) + 1
J5 = IPIC(3) + 1
J6 = IPIC(4)
J7 = IPIC(5)
J8 = IPIC(6)
J9 = IPIC(7)
J10 = IPIC(8)
J11 = IPIC(9)
GO TO 79

C

73 CONTINUE

J3 = IDB(13) + 1
J4 = IDB(13) + 1
J5 = IDB(14) + 1
J6 = IPIC(4) + 1
J7 = IPIC(5)
J8 = IPIC(6)
J9 = IPIC(7)
J10 = IPIC(8)
J11 = IPIC(9)
GO TO 79

C

74 J3 = IDB(13) + 1
J4 = IDB(13) + 1
J5 = IDB(14) + 1
J6 = IDB(14) + 1
J7 = IPIC(5) + 1
J8 = IPIC(6)
J9 = IPIC(7)
J10 = IPIC(8)
J11 = IPIC(9)
GO TO 79

C

75 CONTINUE

J3 = IDB(13) + 1
J4 = IDB(13) + 1
J5 = IDB(14) + 1
J6 = IDB(14) + 1
J7 = ID8(3) + 1
J8 = IPIC(5) + 1
J9 = IPIC(7)
J10 = IPIC(8)
J11 = IPIC(9)
GO TO 79

C 76 CONTINUE
J3 = ID8(13) + 1
J4 = ID8(13) + 1
J5 = ID8(14) + 1
J6 = ID8(14) + 1
J7 = ID8(3) + 1
J8 = ID8(1) + 1
J9 = IPIC(7) + 1
J10 = IPIC(8) + 1
J11 = IPIC(9)
GO TO 79

C 77 CONTINUE
J3 = ID8(13) + 1
J4 = ID8(13) + 1
J5 = ID8(14) + 1
J6 = ID8(14) + 1
J7 = ID8(3) + 1
J8 = ID8(1) + 1
J9 = IPIC(7) + 1
J10 = IPIC(8) + 1
J11 = IPIC(9)
GO TO 79

C 78 CONTINUE
J3 = ID8(13) + 1
J4 = ID8(13) + 1
J5 = ID8(14) + 1
J6 = ID8(14) + 1
J7 = ID8(3) + 1
J8 = ID8(1) + 1
J9 = IPIC(7) + 1
J10 = IPIC(8) + 1
J11 = IPIC(9) + 1

C 79 CONTINUE

THE HARDWARE INDICES ARE SET

IF (DATAB(7, J3) .LT. CDAISF OR DATAB(7, J4) .LT. CDAISO OR DATAB(7, J5) .LT. RMAXF OR DATAB(7, J6) .GT. RMAXO) GO TO 81

FUEL CIRCUIT AND OXIDIZER CIRCUIT ISOLATION VALVES AND FILTERS ARE ACCEPTABLE

LPISF = (1.29E-3/ RHOF) * (HDOTF/DATAB(7, J3))**2
LPISO = (1.29E-3/ RHOO) * (HDOTO/DATAB(7, J4))**2
LPFIF = DATAB(7, J5) * HDOTF
LPFI0 = DATAB(7, J6) * HDOTO
ICH0SE(3) = DATAB(1, J3)
PRELIMINARY CALCULATIONS FOR SELECTION OF PNEUMATIC REGULATOR

\[
F_T = \text{DATAB}(8,J1)
\]
\[
P_F = P_T + 2.5 \times \text{DLP}_S + 2.5 \times \text{DLP}_F
\]
\[
P_O = P_T + 2.5 \times \text{DLP}_S + 2.5 \times \text{DLP}_F
\]
\[
W_{\text{DTPR}} = 1.55 \times 10^{-17} \times 28 \times \text{PREG} \times (\text{DOTT} / \text{RHOF} + \text{DOTO} / \text{ROH})
\]
\[
C_{\text{DAREG}} = \text{DOTT} / \text{SORT}(9600, \times \text{PREG} / 1.27E4)
\]

IF (PREG < \text{DATAB}(8,J7) OR PREG > \text{DATAB}(9,J7) OR \text{DATAB}(6,J8) < PREG / \text{CIAREG}) GO TO 81

REGULATOR IS ACCEPTABLE

PRELIMINARY CALCULATIONS FOR SELECTION OF PNEUMATIC ISOLATION VALVE

\[
\text{R}_{\text{OPR}} = 1.02E-7 \times 3000
\]
\[
C_{\text{DAISO}} = W_{\text{DTPR}} / \sqrt{200 \times \text{R}_{\text{OPR}} / 2.9E3}
\]

IF (\text{DATAB}(7,J8) < C_{\text{DAISO}}) GO TO 81

PNEUMATIC ISOLATION VALVE IS ACCEPTABLE

PRELIMINARY CALCULATIONS FOR SELECTION OF FUEL TANK, OXIDIZER TANK AND PNEUMATIC TANK

\[
W_P = 1.5 \times \text{TOTIMP} / 260
\]
\[
W_F = W_P / (1.8 + XMR)
\]
\[
V_F = W_F / \text{RHOF}
\]
\[
V_T = 1.5 \times V_F
\]
\[
W_O = W_F / XMR
\]
\[
A_{\text{CSWP}} = W_F + W_O
\]
\[
V_O = W_O / \text{ROH}
\]
\[
V_T = 1.5 \times V_O
\]
\[
V_{\text{PR}} = P_T \times (V_F + V_O) / (3000 \times 2.5 \times P_F)
\]

IF (\text{DATAB}(6,J9) LT V_F OR \text{DATAB}(7,J9) LT P_F OR \text{DATAB}(6,J10) LT V_O OR \text{DATAB}(7,J10) LT P_O OR \text{DATAB}(6,J11) LT V_{\text{PR}}) GO TO 81

FUEL TANK, OXIDIZER TANK AND PNEUMATIC TANK ARE ACCEPTABLE

CHASE(9) = \text{DATAB}(4,J9)
CHASE(10) = \text{DATAB}(1,J10)
ICHOOSE(11) = DATAB(1,J11)
TNKW = DATAB(23,J9) + DATAB(23,J10)

SIZE PLUMBING AND CONNECTORS

PCWATE = 2*DATAB(23,J9)*NCHOSE(9) + DATAB(23,J10)*NCHOSE(10)
+ DATAB(23,J11)*NCHOSE(11)

IPIC(7) = J9
IPIC(8) = J10
IPIC(9) = J11
N(12) = K
N(13) = II
N(14) = J1
N(15) = J2
N(16) = J3
N(17) = J4
N(18) = J5
N(19) = J6
N(20) = J7
N(21) = J8
N(22) = J9
N(23) = J10
N(24) = J11
DO 810 I = 1,14
J = N(I)
WI = WT + NCHOSE(I)*DATAB(23,J)
VOL = VOL + NCHOSE(I)*DATAB(23,J)
PL = PL + NCHOSE(I)*DATAB(16,J)
PLMIN = PLMIN + NCHOSE(I)*DATAB(16,J)

CONTINUE

IF(J3.LT.J3E) GO TO 82
IF(J4.LT.J4E) GO TO 83
IF(J5.LT.J5E) GO TO 84
IF(J6.LT.J6E) GO TO 85
IF(J7.LT.J7E) GO TO 86
IF(J8.LT.J8E) GO TO 87
IF(J9.LT.J9E) GO TO 88
IF(J10.LT.J10E) GO TO 89
IF(J11.LT.J11E) GO TO 90

NO ACCEPTABLE HARDWARE
ICHOOSE(1) = -1
RETURN

J3 = J3 + 1
GO TO 60
83 J3 = IDB(13) + 1
J4 = J4 + 1
GO TO 80
1255
84 J3 = IDB(13) + 1
J4 = IDB(13) + 1
J5 = J5 + 1
GO TO 80
1260
85 J3 = IDB(13) + 1
J4 = IDB(13) + 1
J6 = J6 + 1
GO TO 80
1265
86 J3 = IDB(13) + 1
J4 = IDB(14) + 1
J5 = IDB(14) + 1
J6 = IDB(14) + 1
J7 = J7 + 1
GO TO 80
1270
87 J3 = IDB(13) + 1
J4 = IDB(13) + 1
J5 = IDB(14) + 1
J6 = IDB(14) + 1
J7 = IDB(3) + 1
J8 = J8 + 1
GO TO 80
1275
88 J3 = IDB(13) + 1
J4 = IDB(13) + 1
J5 = IDB(14) + 1
J6 = IDB(14) + 1
J7 = IDB(3) + 1
J8 = IDB(1) + 1
J9 = J9 + 1
GO TO 80
1280
89 J3 = IDB(13) + 1
J4 = IDB(13) + 1
J5 = IDB(14) + 1
J6 = IDB(14) + 1
J7 = IDB(3) + 1
J8 = IDB(1) + 1
J9 = IDB(15) + 1
J10 = J10 + 1
GO TO 80
1285
90 J3 = IDB(13) + 1
J4 = IDB(13) + 1
J5 = IDB(14) + 1
J6 = IDB(14) + 1
J7 = IDB(3) + 1
J8 = IDB(1) + 1
J9 = IDB(15) + 1
J10 = IDB(15) + 1
J11 = J11 + 1
GO TO 80
1290
91 J3 = IDB(13) + 1
J4 = IDB(13) + 1
J5 = IDB(14) + 1
J6 = IDB(14) + 1
J7 = IDB(3) + 1
J8 = IDB(1) + 1
J9 = IDB(15) + 1
J10 = IDB(15) + 1
J11 = IDB(15) + 1
J12 = IDB(15) + 1
GO TO 80
1295
92 J3 = IDB(13) + 1
J4 = IDB(13) + 1
J5 = IDB(14) + 1
J6 = IDB(14) + 1
J7 = IDB(3) + 1
J8 = IDB(1) + 1
J9 = IDB(15) + 1
J10 = IDB(15) + 1
J11 = IDB(15) + 1
J12 = IDB(15) + 1
GO TO 80
1300
93 J3 = IDB(13) + 1
J4 = IDB(13) + 1
J5 = IDB(14) + 1
J6 = IDB(14) + 1
J7 = IDB(3) + 1
J8 = IDB(1) + 1
J9 = IDB(15) + 1
J10 = IDB(15) + 1
J11 = IDB(15) + 1
J12 = IDB(15) + 1
GO TO 80
1305 END

REGISTER ALLOCATION
4 REGISTERS ASSIGNED OVER THE LOOP BEGINNING AT LINE 318
FUNCTION GAM(X)
  IF (X.GT.1) GO TO 2
  Z=X
1  IF (Z.GT.0.) GO TO 3
  Z=Z+1.
  GO TO 1
2  Z=X-1.
3  T1=Z+.5
  TZ=Z1+T1
  T1=EXP(-TZ)**T1*2.50662327465
  GAMZ=T1*(1+76*18009173/(Z+1.))**2**4.0532033/(Z+2.)+24.01409822/
  G (Z+3.)**2.31739516/(Z+4.)**5.1206580083E-2/(Z+5.)**.5363826E-5/(Z+6.))
  IF (X.GT.1) GO TO 5
4  GAMZ=GAMZ/Z
  IF (Z.EQ.X) GO TO 5
  Z=Z-1.
  GO TO 4
5  GAM=GAMZ
  RETURN
END

C
FUNCTION CERF

DIMENSION B(28), A(26), AA(17), BB(19)

IF(Y.GT.4.0)GO TO 2

DATA AZERO / 3.88730365/
DATA A(1) / -1.36163142/
DATA A(2) / -0.437316404/
DATA A(3) / -0.108931024/
DATA A(4) / 0.1386797472/
DATA A(5) / -0.59247450/
DATA A(6) / 0.23591751/
DATA A(7) / 0.0884736263/
DATA A(8) / 0.0388666171/
DATA A(9) / -0.01000386351/
DATA A(10) / 0.0039746328343/
DATA A(11) / -0.82610836-04/
DATA A(12) / 0.236450986-04/
DATA A(13) / 0.60791002E-05/
DATA A(14) / 1.46597217E-05/
DATA A(15) / 0.033151593E-06/
DATA A(16) / 0.007280679E-06/
DATA A(17) / 0.00105791E-06/
DATA A(18) / 0.67094742E-08/
DATA A(19) / 0.95021279E-09/
DATA A(20) / 0.10113162E-09/
DATA A(21) / 0.17586041E-10/
DATA A(22) / 0.029403813E-10/
DATA A(23) / 0.4659254E-12/
DATA A(24) / 0.7162152E-13/
DATA A(25) / 0.1863745E-13/
DATA A(26) / 0.152467E-14/
DATA A(27) / 0.0/
DATA B(28) / 0/
DATA AAZERO / 1.970705272/
DATA AA(1) / -0.0143374271775/
DATA AA(2) / 0.002973616220261/
DATA AA(3) / -0.960351605E-05/
DATA AA(4) / 0.04331334E-05/
DATA AA(5) / 0.2362130E-07/
DATA AA(6) / 0.15154536E-08/
DATA AA(7) / -0.1084936E-09/
DATA AA(8) / 0.90425594E-11/
DATA AA(9) / -0.80947054E-12/
DATA AA(10) / 0.763856E-13/
DATA AA(11) / 0.817918E-14/
DATA AA(12) / -0.30715E-15/
DATA AA(13) / 0.1546E-15/
DATA AA(14) / 0.01315E-15/
DATA AA(15) / 0.00170E-15/
DATA AA(16) / 0.0023E-15/
DATA AA(17) / 0.0003E-15/
DATA BB(18) / 0/
DATA BB(19) / 0/

x=7/4
COEFF=4.*X*X-2.
55  GO TO 1, 26
1  B(J) = COEFF*B(J+1) - B(J+2) + A(J)
   BZERO = COEFF*B(1) - B(2) + AZERO
   CERF = X/2.0*(BZERO - B(2))
   RETURN
2  X = 4.0/Y
   COEFF = 4.0*X*X - 2.0
   DO 3 I = 1, 17
      J = 18-I
      BB(J) = COEFF*BB(J+1) - BB(J+2) + A(J)
      BBZERO = COEFF*BB(1) - BB(2) + AZERO
      CERF = (BBZERO - BB(2))/((2.0*Y*EXP(Y*Y))**.564189583547756)
   RETURN
3  END

C ***********************************************************************


SUBROUTINE RELY

COMMON /USERI/  ISpT, ISUB, KEOPT, RFIXED, SLBMX
COMMON /USERR/  ISpT,.
C022575
570

COMMON /USERI/  APOGEE, COMMAT, DIAMAX, EEQWT(9), EPHE, 
COMMON /USERR/  EOM1N, EOM1XL, EOM2L, EOM2XL, EOM2LAT, 

COMMON /BTWN/  ACSSN, ACSPW, ALT, AREA, BATCAP,

COMMON /DBCOM/  R(31), NR(60), RT(31,60), Z(31), RO(31), ROWNM(31), 

COMMON /CHOSE/  COSTM(5,60), DIPA(11,60), ICLOSE(60), 

COMMON/PRTCOM/  ACCRCY, AM, AN, BF, BS, 

DIMENSION N(5), NEQUIP(5)
SUBROUTINES CALLED

QSF - INTEGRATION BY SIMPSON'S RULE (SSP)
RIMOD- RELIABILITY MODELS CALCULATION

*** PROGRAM INITIALIZATIONS ***

DO 110 I=1,60
NR(I)=0
*** USER INPUTS ***

KEOPT  EXPENSE OPTION INDICATOR
        1  WEIGHT
        0  COST

RFIXED  INITIAL SYS RELIABILITY

SYSLB  INITIAL WEIGHT (POUNDS)

SLMX  MAX SYS WEIGHT

TRUNC  MISSION LENGTH (HRS)

ITRUNC  NUM OF TIME POINTS

ISUB  REQUIREMENTS OPTION
        1  AT LEAST ONE SUB-SYS SPEC
        0  OTHERWISE NO SUB-SYS SPEC

ISPT  SINGLE POINT FAILURE REQUIREMENTS OPTION
        0  REQ NOT IN EFFECT
        1  OTHERWISE REQ IN EFFECT

SPEC1  MMD SYS REQUIREMENT (HRS)

SPEC(K)  R(ITRUNC) SUB-SYS REQ; K=1,NSS
        DEFAULT VALUE IS 0.0

SPEC(NSS+1)  R(ITRUNC) SYS REQ DEFAULT VALUE IS 0.0

N(K)  CUMULATIVE NUM OF MODULES THRU SUB-SYS K

TRUNC=TPRIM*730*

IF (RFNL <= 0.999) ALPHA=TRUNC/((-ALOG(RFNL))**.625)

SPEC1=CONS*730

ITRUNC=31

SYSLB=SATTWT

C  SET NUM OF SUB SYS

NSS=5

N(I)=NEQUIP(I)

100  N(I)=NEQUIP(I)+N(I-1)

*** SIS INPUTS ***

ACSWP  INITIAL EXPENDABLES WEIGHT (POUNDS)

EMU  EXPENDABLES INITIAL MEAN LIFETIME (HRS)

ESIG  EXPENDABLES INITIAL STD. DEV. (HRS)

MAXEXP  MAX NUM OF EXPENDABLE INCREMENTS

NZERO  ORBITAL MEAN MOTION (RAO/HRS)

OC  DUTY CYCLE
**NASA**

**170**

```plaintext
Tb = BATTERY TEMP (DEGREES KELVIN)  OTHER NASA  6166
D = DEPTH OF DISCHARGE (BETWEEN 0 AND 100)  OTHER NASA  6177
NC = TOTAL NUM OF CELLS (ALL BATTERIES)  OTHER NASA  6178

PARAMETERS NECESSARY TO COMPUTE THE CYCLES/HR FACTOR S=C NASA  6179
NOW FIXED AT 4.0E-11, REF MODEL 5 NASA  6180

EMU=TRUNC
ESTG=TRUNC/6.
MAXEXP=20
DC=1
TB=TB1+273.
```

**180**

```plaintext
*** FIXED CONSTANTS ***
SENSE/SWITCH FAILURE RATE NASA  6184
PAYOFF THRESHOLD, R(TRUNC) NASA  6185
PAYOFF THRESHOLD, MM NASA  6186
EXPENDABLES LIFE INCRA. NASA  6187
EXPENDABLES STD. DEV. INCRA. NASA  6188
```

**190**

```plaintext
LAMS=120.
RHO1=0.00001
RHO2=0.1
DELMU=2190.
DELSIG=365.
```

**200**

```plaintext
```

**205**

```plaintext
JMIN=N(2)+1
JMAX=N(NSS)
```

```plaintext
DO 130 I=1,6
Z(I)=DATAB(I,JMIN)
NZ=NCHOSE(JMIN)
DO 140 J=JMIN,JMAX
NY=NCHOSE(J+1)
NCHOSE(J+1)=NZ
N=NY
DO 140 I=1,6
```

**210**

```plaintext
R(I)=DATAB(I,J+1)
DATAB(I,J+1)=Z(I)
Z(I)=R(I)
```

**220**

```plaintext
CONTINUE
```

**225**

```plaintext
DATAB(I,JMIN) = 3.*ACSWF+TNKWT)/TPRIM
```

**9-154**

---

**SYSGM**
DATAB(2, JMIN) = 4,
DATAB(3, JMIN) = ENU
DATAB(4, JMIN) = ESIG
DATAB(5, JMIN) = DELMU
DATAB(6, JMIN) = DELSIG

C
NMX(JMIN) = MAXEXP
SAVMX(JMIN) = NMX(JMIN)

C
DO 150 K = 2, NSS
150 N(K) = N(K) + 1
JMAX = JMAX + 1

C
DO 160 J = 1, JMAX
MODL = INT(DATAB(2, J) + 1)
C
IF (MODL .EQ. 4) GO TO 160

C
NMX(J) = DATAB(6, J) + 1
NMX(J) = NMX(J) - NCHOSE(J)
GO TO (151, 152, 153, 160, 155), MODL

C
151 DATAB(4, J) = LAMS*1.0E-09
DATAB(3, J) = DATAB(3, J)*1.0E-09
DATAB(6, J) = DC
GO TO 160

C
152 DATAB(3, J) = DATAB(3, J)*1.0E-09
DATAB(4, J) = DATAB(4, J)*1.0E-09
GO TO 160

C
153 DATAB(6, J) = NC/NCHOSE(J)
THOPI = 0.2631853
DATAB(5, J) = NZER*C*3600./THOPI
DATAB(4, J) = TB
DATAB(3, J) = D
GO TO 160

C
155 DATAB(6, J) = DC
DATAB(3, J) = DATAB(3, J)*FC
DATAB(4, J) = LAMS*1.0E-09
CONTINUE

C
DELH = TRUNC/FLOAT(ISTRUNC - 1)
C
LIN = N(NSS)
IF (IDEBUG .EQ. 0) GO TO 165
WRITE(6, 3000) NMX(J), J = 1, LIN
WRITE(6, 3000) NCHOSE(J), J = 1, LIN
3000 FORMAT(6(I17, 1X))
WRITE(6, 4000)(DATAB(I, J), I = 1, 6, J = 1, LIN)
4000 FORMAT(6(I16, 8X))
165 DO 180 J = 1, LIN
IADD = 0
CALL RMOD(J, DELH, ITRUNC, ITRUNC, IADD, 0)
DO 170 I=1,ITRUNC
RI(I,J)=R(I)
CONTINUE

IF (IDEBUG.EQ.1) PRINT 4000,(R(I),I=1,ITRUNC)

C SET EXPENSE OPTION

IF (KEOPT.NE.1) GO TO 195
DO 190 J=1,LIM
C COST(J)=DATA(1,J)
GO TO 200
C COMPUTE COST

C DO 196 J=1,LIM
195 COST(J)=COSTM(1,J)+COSTM(2,J)+COSTM(3,J)

*** MAIN PROGRAM ***

R(ITRUNC) MODE

LIM=NSS+1

SAVNSR=0
GO 270 K=1,LIM

CK FOR ANY SUB-SYS USER SPEC.

IF (ISUB .NE. 1 .AND. K .NE. LIM) GO TO 270

SELECT JMIN AND JMAX

IF (K .NE. LIM) GO TO 210
JMIN=1
JMAX=N(NSS)
ISR=SAVNSR
GO TO 220
JMIN=JMAX+1
JMAX=N(K)

CALCULATE MAX NUM SYS RED.

DO 230 L=JMIN,JMAX
NSM=NSM+SAVAX(L)

CK FOR SUB-SYS USER SPEC

IF (SPEC(K).LE.RHO1) GO TO 269

SET PARAMETERS FOR REDAP ENTRY

PHOTH=RHO1
NT=1
IRTN=1

CALCULATE INITIAL SUB-SYS RELIABILITY

ROLD(ITRUNC)=RFIXED*RFNL
GO 240 J=JMIN,JMAX
340 240 ROLD(ITRUNC) = ROLD(ITRUNC) * RI(ITRUNC, J)  
       CK RELIABILITY AGAINST SPEC. ENTER REDAP 

345 C RLDH0 = 1.0  
       RDI(ITRUNC) = 1.0 - (1.0 - ROLD(ITRUNC)) ** 2  
       IF (IDS .EQ. 0) GO TO 245  
       IF (RD(ITRUNC) .LT. SPEC(K)) GO TO 390  
       GO TO 255  

350 245 IF (ROLD(ITRUNC) .LT. SPEC(K)) GO TO 390 

355 C 255 IF (K .EQ. LIM) GO TO 269  
       DO 250 J = JMIN, JMAX  

260 NMX(J) = NR(J)  

269 SAVNSR = SAVNSR + NSR  

270 CONTINUE 

C 280 DO 280 J = 1, JMAX 

280 NMX(J) = SAVMX(J)  

285 ENTRY TO MMD DETERMINATION 

370 C 300 IF (ISFT .EQ. 0) GO TO 300  

375 C 305 DO 290 J = 1, JMAX 
       IF ((NMX(J) .LE. 0) .OR. (NR(J) .GT. 0)) GO TO 290 
       MODL = DATAB(2, J) + 1  
       L = 1  
       IF (MODL .EQ. 5) L = NCHOS(J)  
       NSR = NSR + L 
       NR(J) = NR(J) + L 
       IADD = 0 
       CALL RINOD(J, DELN, ITRUNC, ITRUNC, IADD, 0)  
       DO 285 I = 1, ITRUNC  

285 RII(J) = RI(J)  

290 CONTINUE 

C

390 C 300 RHOH = RHO2 

395 C NT = ITRUNC 

I RTN = 2  

C  

INITIALIZATION OF PARAMETERS BEFORE ENTRY TO THE REDUNDANCY ALLOCATION PROCEDURE 

COMPUTE INITIAL RELIABILITY
C FOR SINGLE AND DOUBLE STRING SYSTEMS

GO 320 I=1,ITRUNC
RLOD(I)=RI(I,1)
IF (RFNL .GE. (.999) GO TO 305
RLOD(I)=RLOD(I)*EXP(-((DELH*FLOAT(I-1))/ALPHA)**1.6)
DO 310 J=2,NMAX
310 RLOD(I)=RLOD(I)*RI(I,J)
R(D)=1.0-(1.0-RLOD(I))**2
CONTINUE
C CALL QSF (DELH, RLOD, Z, ITRUNC)
MMDLOT=RFIXED*Z(ITRUNC)
CALL QSF (DELH, RLOD, Z, ITRUNC)

OLDRHO=-1.0
IF (IDS .EQ. 0) GO TO 350
IF (MMDLOT .LT. SPEC1) GO TO 390
GO TO 351
350
351 IRTN=0
C COMPUTE INITIAL MMD VALUE
JMAX=0
DO 353 K=1,NSS
JMIN=JMAX+1
JMAX=N(K)
SSREL(K)=1.0
DO 352 J=JMIN,JMAX
SSREL(K)=SSREL(K)*RI(ITRUNC,J)
352 CONTINUE
C MISSION EQUIPMENT RELIABILITY
353
SSREL(6)=RFNL
JMAX=N(2)-1
DO 370 J=1, JMIN
NCHOSE(J)=NCHOSE(J)+NR(J)
370 JMIN=N(2)
JMAX=N(NSS)
DO 380 J=JMIN, JMAX
NCHOSE(J)=NCHOSE(J)+NR(J+1)
380
C COMPUTE SUBSYS RELIABILITIES
TPRIM=TPRIM+FLCAT(3*NR(JMIN))
PRINT 1000,TRIM
1000 FORMAT (1X,6F10.4)
PRINT 1100,TRIM
1100 FORMAT (1X,2X,E11.4)
TRIM=ITRUNC
IF (IDS .EQ. 0) GO TO 385
MMDLOT=MMDLOT=DSMD
RLOD(ITRUNC)=ROD(ITRUNC)
DO 381 J=1,NMAX
381 NCHOSE(J)=NCHOSE(J)*2
385 RETURN
*** MAIN REDUNDANCY ALLOCATION PROCEDURE ***

IF MAX NUM RED EXCEEDED, RETURN
OTHERWISE CONTINUE PROCEDURE

IF (NSR .GE. NSMX) GO TO (490, 510), IRTN

SELECT MODULE TO ADD A RED, IF J .GE. JMAX GO TO SYS UPDATE PROCEDURE.

IF (NR(J) .GE. NMX(J)) GO TO 440
ADD A RED TO MODULE AND COMPUTE THE RELIABILITY FNC

CALL RIMOD(J, DELH, ITRUNC, NT, IADD, 1)

CALL QSF(DELH, RNEW(J), ITRUNC) MMONEW = Z(ITRUNC)*RFIXED

IF (NT .NE. 1) GO TO 410

CALL QSF(DELH, RNEW(Z), ITRUNC) MMONEW = Z(ITRUNC)*RFIXED

IF (RHOL .LT. OLDRHO) GO TO 440

JSAVE = J
OLDRHO = RHOL

RHO = (ABS(RNEW(J) - ROLD(J)))/COST(J)

IF (RHO .LT. OLDRHO) SAVE CURRENT RELIABILITY DATA, MODULE NUM, AND VALUE OF RHO.

*** END REDAP ***
*** SYSTEM RELIABILITY UPDATE PROCEDURE ***
(SYRUP)

MODEL = DATAB(2, JSAVE) + 1

IF (MODEL .LE. 0.5) L = NCHOSE(JSAVE)

NSR = NSR + L
NR(JSAVE) = NR(JSAVE) + L
IF (MODEL .NE. 4) GO TO 449
SYSLB = SYSLB + DATAB(1, JSAVE) / (TPRM + FLOAT(3*NR(JSAVE)))
GO TO 450

449 SYSLB = SYSLB + DATAB(1, JSAVE) * FLOAT(L)
IF (MODEL .NE. 3) GO TO 450
XNR = FLOAT(NR(JSAVE) + NCHOSE(JSAVE))
D = 0.5 * (XNR - 1) / XNR
DATAB(3, JSAVE) = D
NR(JSAVE+1) = NR(JSAVE+1) + 1
SYSLB = SYSLB + DATAB(1, JSAVE+1)
IADD = 0
CALL RIMOD(JSAVE+1, DELH, ITRUNC, ITRUNC, IADD, 0)
GO 452 I = I + ITRUNC
452 IF (I .LE. JSAVE) = ITRUNC
GO 450 I = I + ITRUNC
450 CALL RIMOD(JSAVE, DELH, ITRUNC, ITRUNC, IADD, 0)
GO 451 I = I + ITRUNC
451 SAVR(I) = R(I)
GO 460 I = I + ITRUNC
460 R(I, JSAVE) = SAVR(I)
CONTINUE
EXIT IF SYS WEIGHT EXCEEDED.

IF (SYSLB .GE. SLBMAX) GO TO (500, 520), IRTN
BRANCH TO START ANOTHER PASS
THRU REDAP(
MODE NT STMNT NUM
R(ITRUNC) 1 250
MMD ITRUNC 330
)

IF (NT .NE. 1) GO TO 330
GO TO 250

*** END SYRUP ***

*** PROGRAM RETURNS ***
SUMMARY OF CHANGES MADE BY THE OPTIMIZER

13 WORDS OF INVARIANT RLIST REMOVED FROM THE LOOP BEGINNING AT LINE 477
13 WORDS OF INVARIANT RLIST REMOVED FROM THE LOOP BEGINNING AT LINE 502
SUBROUTINE RIMOD

76/76

OPT=2

FTN 4.2+383

03/27/75 21.39.03

SUBROUTINE RIMCO(J, DELH, ITRUNC, NT, IADD, IOPT)
COMMON /DBCOM/ R(31, NR(60), RI(31, 60), R(31), RO(60), RDUM(31),
M(31), RD(31), RNEW(31), NHX(60), SAVMRX(60),
COST(60), OUM(31, 613), NASA 022575

COMMON /CHOSE/ COSTM(51, 60), OPTIMIZE(11, 60), ICH(60),
INOPT(60), SYSPAR(6, 60), SAVMR, NASA 022575

COMMON/PRTCOM/ ACCRCY, AM, AN, BF, BS, CT, CR, NASA 022575

DPIAC, Ill(60), ICHCSE(60), NASA 022575

COMMON /CHOUSE/ ACCRCY, AM, AN, BF, BS, CT, CR, NASA 022575

DPIAC, Ill(60), ICHCSE(60), NASA 022575

COMMON/PRTCOM/ ACCRCY, AM, AN, BF, BS, CT, CR, NASA 022575

DPIAC, Ill(60), ICHCSE(60), NASA 022575

COMMON /CHOUSE/ ACCRCY, AM, AN, BF, BS, CT, CR, NASA 022575

DPIAC, Ill(60), ICHCSE(60), NASA 022575

COMMON/PRTCOM/ ACCRCY, AM, AN, BF, BS, CT, CR, NASA 022575

DPIAC, Ill(60), ICHCSE(60), NASA 022575

COMMON /CHOUSE/ ACCRCY, AM, AN, BF, BS, CT, CR, NASA 022575

DPIAC, Ill(60), ICHCSE(60), NASA 022575

COMMON/PRTCOM/ ACCRCY, AM, AN, BF, BS, CT, CR, NASA 022575

DPIAC, Ill(60), ICHCSE(60), NASA 022575

COMMON /CHOUSE/ ACCRCY, AM, AN, BF, BS, CT, CR, NASA 022575

DPIAC, Ill(60), ICHCSE(60), NASA 022575

COMMON/PRTCOM/ ACCRCY, AM, AN, BF, BS, CT, CR, NASA 022575

DPIAC, Ill(60), ICHCSE(60), NASA 022575

COMMON /CHOUSE/ ACCRCY, AM, AN, BF, BS, CT, CR, NASA 022575

DPIAC, Ill(60), ICHCSE(60), NASA 022575

COMMON/PRTCOM/ ACCRCY, AM, AN, BF, BS, CT, CR, NASA 022575

DPIAC, Ill(60), ICHCSE(60), NASA 022575

REAL LAM, LAMBAR, LAMS

SUBROUTINE RIMOD

PURPOSE
TO COMPUTE THE RELIABILITY FUNCTION FOR MODULE J AFTER
REDUCTIONS ARE ADDED TO THE MODULE.

USAGE
CALL RIMCO(J, NR, J, DELH, ITRUNC, NT, IADD, IOPT)

DESCRIPTION OF PARAMETERS
J -INPUT MODULE NUM

DELH -DELTA TIME, THE TIME INCREMENT

ITRUNC -THE NUM OF TIME POINTS

NT -INPUT OPTION PARAMETER

IADD -INPUT OPTION PARAMETER

IOPT -INPUT OPTION PARAMETER

REMARKS
OPTION PARAMETER VALUE ACTION
NT 1 ONLY COMPUTE RELIABILITY AT
TRUNCATION TIME. RETURN VALUE IN

IADD 0 ADD NO REDUNDANCIES BEFORE COM-
PUTING THE RELIABILITY FUNCTION. ADD
REDUNDANCIES BEFORE COMPUTING
C

IOFT 0  UNCOUPLE MODELS 1 AND 3.
OTHER COUPLE MODELS 1 AND 3.

GLOBAL VARIABLES PASSED THOUGH COMMON
R - THE RESULTING RELIABILITY FUNCTION
NR - INPUT VECTOR OF THE NUMBER OF REDUNDANCIES BY MODULE
NCHOS - INITIAL NUMBER OF ELEMENTS IN MODULES
SYSPAR - MATRIX OF MODEL PARAMETERS
SYSPAR(2,J) = MODEL ID FOR J-TH MODULE
FOR FURTHER DESCRIPTION SEE COMMENTS PRECEEDING THE
PARTIAL MODEL OF INTEREST.

SUBROUTINES AND SUBPROGRAMS REQUIRED
FORTRAN SYS FCNS: EXP, FLOAT, INT, SQRT
EXTERNAL FCNS: GAM=Gamma FCN, CERF=Error FCN
SUBROUTINES: NONE

*....
ROOT2=SQRT(2)+1
MOD=INT(SYSPAR(2,J)+1)
GO TO (10,90,120,160,10), MOD

MODELS 1 AND 5

VARIABLES SIZE ORIGIN DEFN

LAMS 1 INT SIZE/STATE FAILURE RATE
LAM 1 INT FAILURE RATE
Q 1 INT DORMANCY FACTOR
DC 1 INT MODULE DUTY CYCLE
MI 1 INT NUMBER OF STANDBY ELEMENTS
NI 1 INT NUMBER OF ACTIVE ELEMENTS
SYSPAR I,J GLOBAL MODEL PARAMETERS FOR J-TH
MODULE

END... END

LAM=SYSPAR(3,J)
LAMS=SYSPAR(4,J)
Q=SYSPAR(5,J)
DC=SYSPAR(6,J)
NREQ=NCHOSE(J)
NRED=NR(J)

IF (MOD .EQ. 1) GO TO 15  CK MODEL TYPE
NREQ=1
NRED=NRED/NCHOSE(J)
LAM=LAM*FLOAT(NCHOSE(J))

CK INCR MODE TQ=1 ACTIVE
OTHERWISE STOBY
\[ Z = \exp(-q \lambda \text{TIME} \times \text{ARG1}) \]

\[ \text{SUM2} = \text{SUM2} + 2/CJ \]

\[ \text{SUM2} = \text{SUM1} + EC1 \times \text{SUM2} \]

\[ \text{ARG1} = \text{FLOAT}(\text{NI}) \]

\[ \text{ARG2} = 1.0 + \text{FLOAT}(K) \]

\[ \text{ARG3} = 1.0 + \text{ARG1}/\text{QBAR} \]

\[ \text{AK} = \Gamma(\text{ARG1}+1.0) \times \Gamma(\text{ARG3}+\text{FLOAT}(\text{MI}))/\Gamma(\text{ARG2}) \times \Gamma(\text{ARG3}) \]

\[ \text{IF} ((K-2.0)/(K/2)).EQ.1) \]

\[ \text{AK} = -\text{AK} \]

\[ \text{SUM0} = \text{SUM0} - \text{AK} \times \text{SUM2} \]

\[ \text{CONTINUE} \]

\[ \text{IF} (\text{MOD}.EQ.3) \]

\[ \text{SUM0} = \text{SUM0} + R(I) \]

\[ R(I) = \text{SUM0} \]

\[ \text{CONTINUE} \]

\[ \text{IF} (\text{MOD}.EQ.3) \]

\[ J = J - 1 \]

\[ \text{RETURN} \]

**Model 2**

**Variables**

- **FMU**
- **FSIG**
- **SYSPAR**

**Definition**

- **FMU**
- **FSIG**
- **SYSPAR**

**Origin**

- **LOCAL**
- **GLOBAL**

**Mean Unit Life**

**Model Parameters for J-th Module**

- **I=3** Value of FMU
- **I=4** Value of FSIG

**Total Num of Elements**

**Incr Redund.**

\[ \text{DO 110 I=1,NT} \]

\[ K = \text{TRUNC} + 1.0 - 1 \]

\[ Z = ((\text{DELH} \times (K-1)) - \text{FMU}) / (\text{ROOT2} \times \text{FSIG}) \]

\[ \text{AN} = \text{ERF}(\text{ABS}(Z)) \]

\[ \text{IF} (\text{ABS}(Z) \lt 4.0) \]

\[ \text{AN} = 1.0 - \text{AN} \]

\[ \text{AN} = 0.5 \times (1.0 - \text{AN}) \]

\[ \text{IF} (Z \lt 0.0) \]

\[ \text{AN} = 1.0 - \text{AN} \]
COMPUTATION OF CUMULATIVE BINOMIAL PROBABILITIES

230   Z=AN
231   IN=AN**NI
232   SUM=AN
233   IF (LIM.EQ.0) GO TO 110
234   Z=(1.-Z)/Z
235   LLLIM=LIM2+1
236   DO 100 LLL=1, LLLIM
237       L=L-1
238       AN=AN*(FLOAT(NI-L)/FLOAT(L))*Z
239   100   SUM=SUM+AN
240   RETURN

MODEL3( VARIABLES SIZE ORIGIN DEFN
245 D    SC LOCAL DEPTH OF DISCHARGE
250 TB    SC LOCAL BATTERY TEMPERATURE
255 AB    SC LOCAL BATTERY CELL Constant
260 BB    SC LOCAL BATTERY CELL Constant
265 BCYC  SC LOCAL CYCLE RATE OF BATTERY
270 NI    SC LOCAL TOTAL NUM OF BATTERIES
275 NC    SC LOCAL NUM OF CELLS IN BATTERY
280 SYSPAR I,J  GLOBAL MODEL PARAMETERS FOR J-TH MODULE
285 I=3 VALUE OF D
290 I=4 VALUE OF TB
295 I=5 VALUE OF BCYC
300 I=6 VALUE OF NC

120 BCYC = SYSPAR(5,J)
125 NC = SYSPAR(6,J) + 1
130 INCR REDUND.
135 LIM = NR(J) + IADD
140 NI = LIM + NCCHOOSE(J)
145 LIM2 = LIM - 1
150 D = SYSPAR(3,J) * FLOAT(NI-IADD) / FLOAT(NI)
155 TB = SYSPAR(4,J)
160 LIM3 = NC/2
165 NC = NC + LIM3
170 AB = EXP(-1.*3.89958238995921*TB + 64986593*D - .0050646174*TB*TB + .0019619976*D + .0011242688*TB*D)
175 BB = EXP(-138.1033249927089*TB + 18740227*D - .0016717786*TB*TB + .0019619976*D + .0011242688*TB*D)
180 AB = AB * EXP(-.0019619976*D - .0011242688*TB*D)
185 BB = BB * EXP(-.0019619976*D - .0011242688*TB*D)
190 AB = AB + BB
195 COMPUTE NEW RELIABILITIES

240 DO 140 I=1, NI
245   K=ITRUNC(I-1)
250   Z=(DELH*K-1-87660.)/(17532.*ROOT2)
AN = \text{GERF}(\text{ABS}(Z))

\text{IF} (\text{ABS}(Z) > 0) \text{ AN} = i - \text{AN}

\text{AN} = 0.5 (i - \text{AN})

\text{IF} (Z < 0.0) \text{ AN} = i - \text{AN}

Z = \exp(-((\text{BCYC4} (\text{DELH4LK} - i))/\text{AB})*s_EB)

LLLIM = LIM3 \text{ PA}' = Z*NC

\text{LLLIM} = LIM2 + i

\text{LLL} = \text{LLLIM} - i

\text{AN} = \text{AN}(\text{FLOAT}(\text{NI} - \text{LLL}))/\text{FLOAT}(i - \text{LLL})

\text{SUM} = \text{AN}

\text{IF} (\text{LIM} = 0) \text{ GO TO} 140

Z = (i - Z)/Z

\text{RETURN}

\text{CK COUPLING OPTION}

\text{COMPUTE NEW RELIABILITIES}

\text{COMPUTATION OF CUMULATIVE BINOMIAL PROBABILITIES}

\text{MODEL VARIABLES SIZE CRIGIN} SC I, J

\text{GLOBAL \text{FSIG} = S\text{ORT}(\text{FSIG} + Z \ast \text{SYSPAR}(6, J))}

\text{LOCAL \text{FMU} = \text{SYSPAR}(3, J) + Z \ast \text{SYSPAR}(5, J)}

\text{INCR \text{REO}LNO. (6, J) ++}

\text{INCREMENT \text{VALUE OF MUC I = 4}

\text{INCREMENT \text{VALUE OF SIG I = 5}

\text{INCREMENT \text{VALUE OF REO}LNO. I = 6}
340 DO 170 I=1,NT
  K=ITRUNC+1-I
  Z=(({DELH*(K-1)})-FMU)/(ROOT2*FSIG)
  R(K)=ERF(ABS(Z))
  IF (ABS(Z)>0.0) R(K)=1.0-R(K)
  R(K)=0.5+(1.-R(K))
  IF (Z>LT 0.0) R(K)=1.-R(K)
  CONTINUE
170 CONTINUE
RETURN
END
SUBROUTINE QSF

SUBROUTINE QSF(H,Y,Z,NDIM)

PURPOSE
TO COMPUTE THE VECTOR OF INTEGRAL VALUES FOR A GIVEN
EQUIDISTANT TABLE OF FUNCTION VALUES.

USAGE
CALL QSF (H,Y,Z,NDIM)

DESCRIPTION OF PARAMETERS
H - THE INCREMENT OF ARGUMENT VALUES.
Y - THE INPUT VECTOR OF FUNCTION VALUES.
Z - THE RESULTING VECTOR OF INTEGRAL VALUES. Z MAY BE
IDENTICAL WITH Y.
NDIM - THE DIMENSION OF VECTORS Y AND Z.

REMARKS
NO ACTION IN CASE NDIM LESS THAN 3.

SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED
NONE

METHOD
BEGINNING WITH Z(1)=0, EVALUATION OF VECTOR Z IS DONE BY
MEANS OF SIMPSONS RULE TOGETHER WITH NEWTONS 3/8 RULE OR A
COMBINATION OF THESE TWO RULES. TRUNCATION ERROR IS OF
ORDER H**5 (I.E. FOURTH ORDER METHOD). ONLY IN CASE NDIM=3
TRUNCATION ERROR OF Z(2) IS OF ORDER H**4.

FOR REFERENCE SEE
(1) F.B.HILDEBRAND, INTRODUCTION TO NUMERICAL ANALYSIS,
MCGRAW-HILL, NEW YORK/TORONTO/LONDON, 1956, PP.71-76.
(2) R.ZURMUEHL, PRAKTISCHE MATHEMATIK FUR INGENIEURE UND
PHYSIKER, SPRINGER, BERLIN/GOETTINGEN/HEIDELBERG, 1963,
PP.214-221.

DIMENSION Y(1),Z(1)

HT=.3333333*H
IF(NDIM-5)7,8,1

1 SUM1=Y(2)+Y(2)
SUM1=SUM1+SUM1
SUM1=HT*Y(1)+SUM1+Y(3)
AUX1=Y(4)+Y(4)
AUX1=AUX1+AUX1

...
INTEGRATION LOOP

DO 4 I = 7, NDim, 2
  SUM1 = AUX1
  SUM2 = AUX2
  AUX1 = Y(I - 1) + Y(I - 1)
  AUX1 = AUX1 + AUX1
  AUX1 = SUM1 + HT*Y(I - 2) + AUX1 + Y(I)
  Z(I - 2) = SUM1
  AUX2 = Y(I) + Y(I)
  AUX2 = AUX2 + AUX2
  AUX2 = SUM2 + HT*Y(I - 1) + AUX2 + Y(I + 1)
  Z(I - 1) = SUM2
  IF(NDim - 1) = AUX1
  END OF INTEGRATION LOOP

IF(NDim = 3, 12, 11, 9)
  NDIM IS EQUAL TO 4 OR 5
  SUM2 = 1.125*HT*Y(1) + Y(2) + Y(3) + Y(4)
  SUM1 = Y(2) + Y(3)
  SUM1 = SUM1 + SUM1
  SUM1 = HT*Y(1) + SUM1 + Y(3)
  Z(1) = 0
  AUX1 = Y(3) + Y(3)
  AUX1 = AUX1 + AUX1
  Z(2) = SUM2 - HT*Y(2) + AUX1 + Y(4)
  IF(NDim = 5) = 10, 9, 9
  AUX1 = Y(4) + Y(4)
  AUX1 = AUX1 + AUX1
  Z(5) = SUM1 + HT*(Y(3) + AUX1 + Y(5))
  RETURN

IF(NDim IS EQUAL TO 3)
  NDIM IS EQUAL TO 3
  SUM2 = HT*(1.25*Y(1) + Y(2) + Y(2) + 1.25*Y(3))
  SUM2 = Y(2) + Y(2)
  SUM2 = SUM2 + SUM2
  RETURN
$Z(3) = HT \cdot (Y(1) + \text{SUM}2 + Y(3))$

$Z(1) = 0$

$Z(2) = \text{SUM1}$

115  RETURN

END
SUBROUTINE DPI

OPT=2

DIMENSION IPIC(2),ICHOOSE(2),NCONF(5),NCHOOSE(2),NOWAT)

COMMON /USER3/ARRAY(11,3),BIRMX,NMSEQ,OPSM,SCSFL,SCSFL)

1

TPRFL

COMMON /USER3/

APOGEE,COMAT,DIAMAX,EEQHT(9),EPME

COMMON /USER3/

EQMTH,EQMH,L,EQMYL,EQMY2L,EQMY3L,FE,IAGNCY

COMMON /USER3/

IDEBUG,ISATOR,M12SH,OPIEPI,ORBING,PERIGE

COMMON /USER3/

MICRO,RELME,SPCSL,SCSFL,T,XCOSA1

10

COMMON /USER3/

DIMENSION IPIC(2),ICHOOSE(2),NOWAT)

COMMON /USER3/ARRAY(11,3),BIRMX,NMSEQ,OPSM

COMMON /USER3/

APOGEE,COMAT,DIAMAX,EEQHT(9),EPME

COMMON /USER3/

EQMTH,EQMH,L,EQMYL,EQMY2L,EQMY3L,FE,IAGNCY

COMMON /USER3/

IDEBUG,ISATOR,M12SH,OPIEPI,ORBING,PERIGE

COMMON /USER3/

MICRO,RELME,SPCSL,SCSFL,T,XCOSA1

C

COMMON /BTWN/

ACSSN,ACSM,ALT,AREA,BATCAP

COMMON /BTWN/

DX,DT,DY,DT,DZ

COMMON /BTWN/

HARMS,HPT,HPT,HTPIPE

COMMON /BTWN/

OEGHT,OMEG,PASSTR,PJ Vin

COMMON /BTWN/

RADA,RADAB

COMMON /BTWN/

SABOLG,SATLG,SATTNT

COMMON /BTWN/

SATXCG,SATYCG,SATZCG,SAILXL,SAILYL

COMMON /BTWN/

IDEBUG,ISATOR,M12SH,OPIEPI,ORBING,PERIGE

COMMON /BTWN/

MICRO,RELME,SPCSL,SCSFL,T,XCOSA1

COMMON /BTWN/

DIMENSION HSFT(60),TLPTH(60),GRANH(60),XSRT(60),TLPTL(60)

DATA ACSRT,ACSOF,COMOPOFREO

INPUTS FOR DATA PROCESSING SUBSYSTEMS - DPI

C

INPUT CPIT T D SOURCE UNITS DESCRIPTION

C

VAR. IN.
C GRAN RY ALL S/S GRANULARITY HIGH RATE TABLE NASA 7033
C HSRT 35 R Y ALL S/S SPS SAMPLE RATE HIGH TABLE NASA 7034
C TLTH 34+35 R Y ALL S/S NO OF ANOL AND DIG POINTS HIGH NASA 7035
C GEAL 40 R Y ALL S/S GRANULARITY LOW RATE TABLE NASA 7036
C XSR 39 R Y ALL S/S SAMPLE RATE LOW TABLE NASA 7037
C TLPL 37+38 R Y ALL S/S NO OF ANOL AND DIG POINTS LOW NASA 7038
C SCPL U SPECIAL COMMAND SYNC FLAG NASA 7039
C TOTC 30+32 R DB TOTAL NO OF COMMANDS NASA 7040
C COMY 32 R MACRO NCONF(3) - SPEC OR GEN COMPUTER FLAG NASA 7041
C TICPL 32 R U TIME TAG COMMAND FLAG NASA 7042
C TRPL R U SC SUM OF ACS SENSOR NASA 7043
C CMRT R U COMM COMMAND RATE NASA 7044
C OPSMS R U SEC-1 MISSION OPS NASA 7045
C MISPO R U MISSION DATA PROC. FLAG NASA 7046
C DO 1 I=1,7
C 1 CDPL(I,J)=0.0
C IERR=0
C ICHOSE(1)=0
C ICHOSE(2)=0
C IF (ITER *EQ. 0) NCHOSE(1)=1
C IF (ITER *EQ. 0) NCHOSE(2)=1
C BITRAT(2)=0.
C TOPS=0.
C IERR1=0
C IERR2=0
C IERR3=0
C IERR4=0
C IERR5=0
C NEWFL=0
C IF (NCONF(4) *GT. 3)
C CALL MIS(IPICIERRITERNCONFICHOSENCHOSE)
C IF (NCONF(4) *GT. 3) GO TO 110
C NEWFL=1
C CONTINUE
C ANOLH=0.
C ANOLL=0.
C MUX=0
C COMPUTE TABLES
C TOTC=0
C TICPL=0
C NTAH=0
C NTA=0
C NTA=0
C NTAB=0
C ********************************************************
C ********************************************************
C NTAB= NO MAT = 1
C ********************************************************
C ********************************************************
C K= -1
C DO 170 I=1,NTAB
TOTC = (ARRAY(K+2, I) + ARRAY(K+3, I) + ARRAY(K+4, I)) * NCHOSG(I) + TOTCM
TTCFL = TTCFL + ARRAY(K+3, I) * NCHOSG(I)

IF (ARRAY(K+7, I) .EQ. 0) GO TO 150
NTAB = NTAB + 1
HSRT(NTAB) = ARRAY(K+7, I)
GRANH(NTAB) = ARRAY(K+8, I)
TLPTH(NTAB) = (ARRAY(K+5, I) + ARRAY(K+6, I)) * NCHOSG(I)

150 IF (ARRAY(K+6, I) .NE. 0) MUX = 1
IF (ARRAY(K+11, I) .EQ. 0) GO TO 160
NTABL = NTABL + 1
XSRT(NTABL) = ARRAY(K+11, I)
GRANL(NTABL) = ARRAY(K+12, I)
TLPTL(NTABL) = (ARRAY(K+5, I) + ARRAY(K+6, I)) * NCHOSG(I)

120 IF (ARRAY(K+10, I) .NE. 0) MUX = 1
ANCLH = ANOLH + ARRAY(K+5, I) * NCHOSG(I)
ANOLL = ANOLL + ARRAY(K+5, I) * NCHOSG(I)
IF (NEWFL .EQ. 0) GO TO 240
NTABN = NTABN + 1

C ******************** WE NEED NTABN ********************

130 NTABN = NHTAB

C ********************

K = 1
DO 230 I = 1, NTABN
TOTC = TOTC + ARRAY(K+2, I) + ARRAY(K+3, I) + ARRAY(K+4, I)
TTCFL = TTCFL + ARRAY(K+3, I) * NCHOSG(I)

230 IF (ARRAY(K+7, I) .EQ. 0) GO TO 210
NTAB = NTAB + 1
HSRT(NTAB) = ARRAY(K+7, I)
GRANH(NTAB) = ARRAY(K+8, I)
TLPTH(NTAB) = ARRAY(K+5, I) + ARRAY(K+6, I)

210 IF (ARRAY(K+11, I) .EQ. 0) GO TO 220
NTABL = NTABL + 1
XSRT(NTABL) = ARRAY(K+11, I)
GRANL(NTABL) = ARRAY(K+12, I)
TLPTL(NTABL) = ARRAY(K+5, I) + ARRAY(K+6, I)

220 IF (ARRAY(K+10, I) .NE. 0) MUX = 1
ANCLL = ANOLL + ARRAY(K+5, I) * NCHOSG(I)
ANCLH = ANOLH + ARRAY(K+5, I) * NCHOSG(I)

240 CONTINUE

C ANOH -- NO OF ANOL PTS IN HIGH TAB
ANOL -- NO OF ANOL PTS IN LOW TAB

C ********************

150 ANOH = ANOH + IERR1
ANOLL = ANOLL + IERR1
IERR = IERR + IERR1

155 C COUNT NUMBER OF POINTS OF ALL TABLES
SUHTLP = 0.
C ********************

JL = 1
BTRFL = 0
IF (TRMX .NE. 24E5) BTRFL = 1

160 IF (NTAB + 1 .EQ. 0) GO TO 280
CONTINUE

C ORDER TELEM POINTS BY SAMPLE RATE - HIGH
CALL ORDER (NTABH, HSRT, TLPTH, GRANH, XM2, MEDIAN)
JL = JL + 1

165 IF (JL .EQ. 2) GO TO 280
DO 270 I = 1, MEDIAN

260 IF (HSRT(I) .LE. XL) GO TO 270
HSRT(I) = HSRT(I) / 2.
TLPTH(I) = 2.*TLPTH(I)
GO TO 260
CONTINUE
GO TO 250
NAA
280 SSR=HSRT(1)

SSR = MAIN FRAME RATE
C

290 CONTINUE
CALL ORDER (NTABH,GRANH,TLPTH,+SRT,XM2,MAIN)

270 C

300 JL=I-1,MEDIAN
IF (GRANH(I).LE.XM2) GO TO 310
GRANH(I)=GRANH(I)/2
TLPTH(I)=2.*TLPTH(I)
GO TO 300
310 CONTINUE
GO TO 290
320 SUMWH=S
DO 330 I=1,NTABH
SUMWH=SUMWH+TLPTH(I)
C
SUMWH = NUMBER OF WORDS
SUMWH=SUMWH+1.2
IF (SUMWH.LE.XM2) GO TO 340
ICHOOSE(1)=-1
IERR2=10
IERR=IERR+IERR2
RETURN
340 POWER=16.
DO 350 NN=5,8
C
350 IF (POWER. GE. SUMWH) GO TO 360
CONTINUE
360 TLNW=POWER
DO 365 I=1,NTABH
TP=GRANH(I) .GT. WOLMAX)
WOLMAX=WOLMAX+GRANH(I)
C
WOLMAX = WORD LENGTH TO MAX REQUIRED LENGTH
C
365 CONTINUE
CDP1(2,1)=TLNW
CDP1(3,1)=SSR
CDP1(4,1)=WOLMAX
BIRATE=WOLMAX*TLNW*MAIN
C
PRINT 1000,BIRATE

1000 FORMAT (13H BIRATE (1) = ,E11.4)
DO 370 NN=1,10
C
370 TT=2.**N*7.6125
C
IF (TT. GE. BIRATE) GO TO 380
CONTINUE
ICHOOSE(1)=-1
IERR3=100
IERR=IERR+IERR3
C
ICHOOSE = 100 BIT RATE TOO LARGE
RETURN
BIRATE=TT
BIRAT(1)=BIRATE
JL=0

ORDER LOW SAMPLE RATE

IF (NTABL.EQ.1) GO TO 420
CONTINUE
CALL ORDER (NTABL,XSRT,TLPTL,GRANL,XM2,MEDIAN)
JL=JL+1
IF (JL.EQ.2) GO TO 420
DO 410 I=1,MEDIAN
IF (XSRT(I).LE.XM2) GO TO 410
TLPTL(I)=TLPTL(I)*2.
GO TO 400
CONTINUE

SFR=XSRT(1)
SFR=HIGHEST RATE IN LOW RATE TABLE
C
C
SFL=XSRT/I
N=5
IF (SFL.LE.2.*N) GO TO 440
N=7
IF (SFL.LE.2.*N) GO TO 440
CSRT(I)=CSRT(I)/2*
TLPTL(I)=TLPTL(I)*2*
GO TO 400
CONTINUE

SFL=2.*N
N=NTABL
DO 450 I=1,NTABL
SUMWL=SUMWL+TLPTL(I)
SUMNL=SUMWL*SFL
NSUBFR=SUMWL/SFL
COPI(5,1)=NSUBFR
COPI(6,1)=SFR
COPI(?,?)=SFL
C
C
C
C
C
C
C
C
SPECIAL COMMAND SYNC FLAG

IF (SCSFLNE.) IERR4=±Q Om
IERR=IERR IERR4
TOTCM=TOTCMi5
CO

111874 78
NASA 7237

IF (TOTCM.LE.2.*N) GO TO 470
CONTINUE

TOM=2.*N
COPI(1,1) = TOTCM
CWDL=N
K=NCONF(3)
GO TO (480, 500), M
TLMOPS=0
IF (TPRFL .NE. 0.) TLMOPS=TT*OPREQ/WDLNAX
IF (TPRFL .NE. 0.) GO TO 490
J1=IDB(1)+1
IPIC(2)=J1
ICH0SE(2)=DATAB(1,J1)
IF (ITER.EQ.0) NCHOSE(2)=1
MT=WT+NCHOSE(2)*DATAB(23,J1)
VOL=VOL+NCHOSE(2)*DATAB(24,J1)
PL=PL+NCHOSE(2)*DATAB(16,J1)
FLMIN= PLMIN + NCHOSE(2)*DATAB(18,J1)
CONTINUE
ACSOPS=ACSOP1+ACSOP2+ACS0P3
CMOPS = COMAT + CMOP2
TOTOPS=TLMOPS+ACSOPS+CMOPS+OPSMS
IF (ITER.EQ.0) GO TO 510
J1=IPIC(L)
GO TO 540
 troch0=ICH0SE(1)=0
L=1
J1=IDB(M)
JL = 1
IF (M .LT. 1) J1 = IDB(1) + 1
IF (IPIC(L).NE.0) GO TO 520
GO TO 540
IF (ITER.EQ.0) GO TO 530
J1=IPIC(L)
GO TO 540
HARPAR=DATAB(6,J1)*1000.
GO TO (560, 550), M.
IPIC(L)=J1
ICH0SE(L)=DATAB(1,J1)
IF (ITER.EQ.0) NCHOSE(L)=1
MT=MT+NCHOSE(L)*DATAB(23,J1)
VOL=VOL+NCHOSE(L)*DATAB(24,J1)
PL=PL+NCHOSE(L)*DATAB(16,J1)
FLMIN= PLMIN + NCHOSE(L)*DATAB(18,J1)
RETURN
HARPAR=DATAB(6,J1)*1000.
IF (TOTOPS.LE.HARPAR) GO TO 550
J1=J1+1
GO TO 560
IF (J1.GE.J1E) GO TO 570
HARPAR=DATAB(6,J1E)*1000.
IF (TOTOPS.LE.HARPAR) GO TO 550
J1=J1+1
GO TO 560
ERR5=1000
ERR=ERR+ERR5
ICH0SE(L)=-1
RETURN
END
SUBROUTINE ORDER

SUBROUTINE ORDER (NA, B, C, XM2, MEDIAN)
DIMENSION A(I), B(I), C(I)

IF (N .EQ. 1) GO TO 50
MEDIAN=N/2
KK=MEDIAN+2

IF (KK .NE. N) KKK=1

IF (KKE .NE. N) KKK = 1 / ODD NUMBER OF POINTS
DO 20 I=1, N

IF (A(I) .EQ. 0.) GO TO 20

XLG=A(I)
JJ=I

DO 10 J=I, N

IF (XLG .GE. A(J)) GO TO 10
XLG=A(J)
JJ=J

CONTINUE

IF (I .EQ. JJ) GO TO 20
AS=A(I)
BS=B(I)
CS=C(I)

A(JJ)=A(I)
B(JJ)=B(I)
C(JJ)=C(I)

CONTINUE

GO TO (30, 40), KKK

XM2=A(MEDIAN)+2

RETURN

CONTINUE

XM2=A(MEDIAN)+A(MEDIAN+1)
RETURN

END
C COMTY  R MACRO NCONF(3) - SPEC OR GEN COMPUTER FLAG NASA  7366
55  C  TICPL  32  R  TIME TAG COMMAND FLAG NASA  7367
C TPRFL  R  TELEM PROCESS FLAG NASA  7368
C ACSR  R  SUM OF ACS SENSOR NASA  7369
C COMRT  R  COMMAND RATE NASA  7370
C CPSMS  R  MISSION OPS NASA  7371
C MISPSD  I  MISSION DATA PROC. FLAG NASA  7372
40  C  ERROR FLAGS
C IERR = 10  WORD LENGTH GREATER THAN 256 NASA  7373
C IERR = 100  BIT RATE IS TOO LARGE NASA  7374
C IERR = 1000  SPECIAL COMMAND SYNC FLAG IS NOT EQUAL TO ZERO NASA  7375
C IERR = 10000  J1 .GE. J1E NASA  7376
70  C  IERR = 0
C IERR1 = 0 NASA  7377
C IERR2 = 0 NASA  7378
C IERR3 = 0 NASA  7379
C IERR4 = 0 NASA  7380
C IERR5 = 0 NASA  7381
C ANOLH = 0 NASA  7382
C ANOLL = 0 NASA  7383
C MUX = 0 NASA  7384
80  C  COMPUTE TABLES
C TOTCH = 0 NASA  7385
C TTCFL = 0 NASA  7386
C NTABH = 0 NASA  7387
C NTABL = 0 NASA  7388
C NTABN = NMSEQ NASA  7389
C NTABN = NTABN NASA  7390
C ************************************************************ 7391
85  C  WE NEED NTABN ************************************************************ 7392
C NTABN = NTABN NASA  7393
C ************************************************************ 7394
K = -1 NASA  7395
DO 60  I = 1, NTABN NASA  7396
60  TOTCH = TOTCH + ARRAYN(K+2, I) + ARRAYN(K+3, I) + ARRAYN(K+4, I) NASA  7397
C TTCFL = TICFL + ARRAYN(K+4, I) NASA  7398
C IF (ARRAYN(K+7, I).EQ.0) GO TO 40 NASA  7399
C NTABH = NTABH + 1 NASA  7400
C HSRT(NTABH) = ARRAYN(K+7, I) NASA  7401
C GRANH(NTABH) = ARRAYN(K+8, I) NASA  7402
C TLPL(NTABH) = ARRAYN(K+5, I) + ARRAYN(K+6, I) NASA  7403
C IF (ARRAYN(K+11, I).EQ.0) MUX = 1 NASA  7404
C IF (ARRAYN(K+11, I).NE.0) MUX = 1 NASA  7405
C IF (ARRAYN(K+11, I).EQ.0) GO TO 50 NASA  7406
C NTABL = NTABL + 1 NASA  7407
C XSR(NTABL) = ARRAYN(K+11, I) NASA  7408
C GRANL(NTABL) = ARRAYN(K+12, I) NASA  7409
C TLPL(NTABL) = ARRAYN(K+9, I) + ARRAYN(K+10, I) NASA  7410
C IF (ARRAYN(K+10, I).NE.0) MUX = 1 NASA  7411
C ANOLH = ANOLH + ARRAYN(K+5, I) NASA  7412
C ANOLL = ANOLL + ARRAYN(K+9, I) NASA  7413
C C ANOLH = NO OF ANOL PTS IN HIGH TAB NASA  7414
C ANOLL = NO OF ANOL PTS IN LOW TAB NASA  7415
C IF (MUX .NE. 0) IERR2 = I NASA  7416
100  IF (IERR = IERR + IERR2) NASA  7417
C COUNT NUMBER OF POINTS OF ALL TABLES NASA  7418
C SUMLP = 0 NASA  7419
110  JL = 0 NASA  7420
IF (NTABH .EQ. 0) GO TO 140
CALL ORDER (NTABH, GRANH, TLPTH, HSRT, XM2, MEDIAN)
JL = JL + 1
IF (JL .EQ. 2) GO TO 140
DO 130 I = 1, MEDIAN
   IF (GRANH(I) .LE. XM2) GO TO 130
   GRANH(I) = GRANH(I) / 2.
   TLPTH(I) = 2. * TLPTH(I)
GO TO 120
130 CONTINUE
GO TO 100

SSR = HSRT(1)
C SSR = MAIN FRAME RATE
IF (NTABH .EQ. 1) GO TO 140
JL = 0
CONTINUE
CALL ORDER (NTABH, GRANH, TLPTH, HSRT, XM2, MEDIAN)
JL = JL + 1
IF (JL .EQ. 2) GO TO 140
DO 130 I = 1, MEDIAN
   IF (GRANH(I) .LE. XM2) GO TO 130
   GRANH(I) = GRANH(I) / 2.
   TLPTH(I) = 2. * TLPTH(I)
GO TO 120
130 CONTINUE
GO TO 100

SUMWH = 0
DO 150 I = 1, NTABH
   SUMWH = SUMWH + TLPTH(I)
ENDDO
C SUMWH = NUMBER OF WORDS
SUMWH = SUMWH * 2
IF (SUMWH .LE. 256) GO TO 160
IHCHE(I) = -1
IERR = IERR + IERR2
RETURN
POWER = 16
DO 170 N = 5, 8
   POWER = POWER * 2
ENDDO
C POWER = POWER * 2
IF (POWER .GE. SUMWH) GO TO 180
CONTINUE
TLMD = POWER
C MAIN FRAME LENGTH - TLMD
WOLMAX = WORD LENGTH TO MAX REQUIRED LENGTH
WOLMAX = 0
DO 185 II = 1, NTABH
   IF (GRANH(II) .GT. WOLMAX) WOLMAX = GRANH(II)
ENDDO
CONTINUE
C
COPI(I, 2) = TOTCH(I)
COPI(2, 2) = TLMD
COPI(3, 2) = SSR
COPI(4, 2) = WOLMAX
170 BIRATE=WDMAX*TLWDM*SSR
180 PRINT 1000,BIRATE
190 FORMAT (13H BIT RATE (2) = ,E11.4)
200 DO 190 MM=1,18
210 IF (MM.NE.1) THEN
220 TT=2.**MM*7.8*125
230 CONTINUE
240 ICHOS(I)=-1
250 IERR3=100
260 IERR=IERR+IERR3
270 C IERR=100 BIT RATE TOO LARGE
280 RETURN
290 BIRATE=TT
300 BITRAT(2)=BIRATE
310 IF (NTABL.NEQ.1) GO TO 240
320 JL=0
330 C ORDER LOW SAMPLE RATE
340 CONTINUE
350 CALL ORDER (NTABL,XSRT,TLFTL,GRANL,XM2,MEDIAN)
360 JL=JL+1
370 IF (JL.EQ.2) GO TO 240
380 DO 230 I=1,MEDIAN
390 IF (XSRT(I).LE.XM2) GO TO 239
400 XSRT(I)=XSRT(I)/2.
410 TLFTL(I)=TLFTL(I)'2.
420 GO TO 220
430 CONTINUE
440 GO TO 210
450 SFR=XSRT(1)
460 C SFR = HIGHEST RATE IN LOW RATE TABLE
470 N=SUB FRAME LENGTH
480 IF (SFL.LE.2.***N) GO TO 260
490 N=7
500 IF (SFL.GE.2.***N) GO TO 260
510 DO 250 N=9,7
520 N=N+1
530 IF (SFL.GE.2.***N.AND.SFL.LE.2.***NP1) GO TO 260
540 CONTINUE
550 SFL=2.***N
560 SUMWL=0
570 GO 270 I=1,NTABL
580 SUMWL=SUMWL+TLFTL(I)
590 SUMWL=SUMWL*1.2
600 NSUBFR=SUMWL/SFL+1
610 C (1) BIT RATE
620 C (2) WORD LENGTH
630 C (3) NUMBER OF H/F WORDS
640 C (4) NUMBER OF SUBFRAMES
650 C (5) NUMBER OF WORDS PER S/F
660 C (6) NEED FOR DIGITAL MUX
670 CDPI(S+2) = NSUBFR
680 CDPI(6+2) = SFR
690 CDPI(7+2) = SFL
700 IF (NCONF(3) .EQ. 1) GO TO 280
C
J1=108(1)+1
IPIC(2)=J1
ICHOSE(2)=DATAB(1,J1)
IF(ITER.EQ.0) NCHOSE(2)=1
WT= WT + NCHOSE(2)*DATAB(23,J1)
VOL= VOL + NCHOSE(2)*DATAB(24,J1)
PL= PL + NCHOSE(2)*DATAB(16,J1)
FLMIN= FLMIN + NCHOSE(2)*DATAB(18,J1)

230 CONTINUE
280 RETURN
END

REGISTER ALLOCATION
1 REGISTERS ASSIGNED OVER THE LOOP BEGINNING AT LINE 143
1 REGISTERS ASSIGNED OVER THE LOOP BEGINNING AT LINE 212
10. DETAILED FLOW CHARTS

The following are detailed flow charts of the entire model.
THIS IS THE MAIN DRIVER IT SEQUENCES ALL SEGMENTS OF CODING HANDLES I/O SETS CONFIGURATIONS

PROGRAM NASA

COMMON /USER1/
  ALPH,
  AX,
  AY,
  AZ,
  OPHI,
  EA,
  EANT,
  EPI,
  K,
  MANV,
  OMEGR,
  POOTAY,
  POOTRX,
  POOTRY,
  POOTRZ,
  POOTST,
  POOTX,
  POOTY,
  POOTZ,
  POOTO,
  PHIFOV,
  PHIRX,
  PHIRY,
  PHIRZ,
  TACCEL,
  THEMX,
  TAHOLD,
  TL,
  TPMIN,
  TSMALL,
  TX,
  TXN,
  TXNN,
  TXNZ,
  ZNH,

COMMON /USER3/ARRAY(11,3), BTRMX,

COMMON /USER4/BWIDTH(2), FRED(2), FREDR, I0PTCM(3), LINK,

COMMON /USER6/ CGEEXI9), EELOCI9), EEOVI9), EMIYCO, EMIZCO,

COMMON /USER8/SKDONE(7,3)

COMMON /USER9/ CA, CE

COMMON /USERR/ ISPT, ISUB, KEOPT, RFIXED, SLBMMX

COMMON /USERP/ FEEDPT, IMETYP, NFV, NOV, PI

COMMON /USERO/SKOME(7,3)

COMMON /USER/ APOGEE, COMRAT, DIAMAX, EEQWTI9), EPME,

COMMON /BTWN/ ACSN, ACSNP, ALT, AREAL, BATCAP,

BITRAT1(2), CLIFE, CONVWI, D, DT,

DX, DY, DZ, EOBG, EOBSID,

FC, FF, HARNWT, HPT, HTPPIPE,

HTPI, HTPRPB, HIRPWR, IBTLOC,

LMGDD, NC, OMEG5, PASSPR, P,

PL, PLMIN, POCW, RADA, RADAB,

RAT, RJ, SABOLG, SALTG, SATTWI,

SANT, SATXCO, SATYCO, SATZCO, SAI,XL,

SAIYL, SAIZL, SIDE, SYSLB, THCMMW,

THRUST1(2), TI, TNKWT, TPRIM, YB,

VCHP, VOL, WATE, WB, WBT,

WT, XJ, XNZERO, YJ, YJ

CONT. ON PG 2
### Name List Inputs

The name list inputs are broken into three categories. That is, categories of required, desired, and optional parameters.

The following is a list of the inputs to the model.

#### Required Input Data

<table>
<thead>
<tr>
<th>NAME</th>
<th>REP VALUE</th>
<th>UNITS</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>APOGEE</td>
<td>500.</td>
<td>NMI</td>
<td>Orbit Apogee</td>
</tr>
</tbody>
</table>

#### Mission Equip Power Req.

- **PEM**: 300. **Watts**
- **EOM1WT**: 435. **LB**
- **EOM2WT**: 435. **LB**
- **IPRINT**: 1
- **NOV**: 1
- **SPEC6**: 0.6
- **SPEC1**: 18. **MD**
- **T**: 24. **MD**

#### Mission Data for Up to 3 EQ.

- **CODEX**: 2. **LOC. OF EXT. EQ. (FT, CENT, AFT)**
- **EELOC**: 3. **LOC. OF EXT. EQ. (RT, LFT, TOP, BOT)**

---

Cont. on pg 3
| C | IEND2 | 3 | --- | LAST ALLOWABLE FOR AP |
| C | IEND3 | 2 | --- | LAST ALLOWABLE FOR DPN |
| C | IEND4 | 5 | --- | LAST ALLOWABLE FOR COMM |
| C | IEND5 | 6 | --- | LAST ALLOWABLE FOR EP |
| C | IEND6 | 3 | --- | LAST ALLOWABLE FOR VESIZE |
| C | IENDR | 1 | --- | LAST ALLOWABLE FOR RELY |

| C | IOPTRM | 0 | --- | RANGING REQUIREMENT O=NO, 1=YES |
| C | JSBOFG | 0 | --- | SAFETY BOOM OR VR DEQ O=NO, 1=YES |
| C | ISOPT | 0 | --- | SINGLE PT FAIL RED O=NO |
| C | ISTR1 | 1 | --- | FIRST ALLOWABLE FOR SANDSC |
| C | ISTR2 | 1 | --- | FIRST ALLOWABLE FOR AP |
| C | ISTR3 | 1 | --- | FIRST ALLOWABLE FOR DPN |
| C | ISTR4 | 1 | --- | FIRST ALLOWABLE FOR COMM |
| C | ISTR5 | 1 | --- | FIRST ALLOWABLE FOR EP |

| C | ISTR6 | 1 | --- | FIRST ALLOWABLE FOR VESIZE |
| C | ISTRTR | 0 | --- | FIRST ALLOWABLE FOR RELY |
| C | JSUB | 0 | --- | S/S REL FLAG EARLEAST 1 S/S |
| C | KEOPT | 1 | --- | AXIS RELATIVITY (DUAL-SPIN) |
| C | KLINK | 1 | --- | EXPENSE OPT IND |
| C | RMM | 1 | --- | COMM LINK (O=USB, I=SGLS) |
| C | NADIR | 1 | --- | VEH SKewing FLAG |
| C | NADIR | 1 | --- | NADIR COVERAGE FLAG |

| C | NET | 1 | --- | O=AFSCF NET, O=NASA NET |
| C | ORMCR | 60 | RPM | SPIN RATE OF ROTOR |
| C | OTEMP | 15 | DEG C | BATT. TEMP. |
| C | ORBCNC | 28.5 | DEG | ORBITAL INCLINATION |
| C | POCTR | .01 | DEG/SEC | AVG BODY RATE LO ORBIT CMG ONL |
| C | POCTR | .012 | DEG/SEC | RED SYS RATE ACC. X |
| C | POCTR | .012 | DEG/SEC | RED SYS RATE ACC. Y |
| C | POCTR | .012 | DEG/SEC | RED SYS RATE ACC. Z |

| C | POI | .0667 | DEG/SEC | MAX RATE STAR RATE INFO (CMG) |
| C | POI | 1.0 | DEG/SEC | MAX MANY RATE X |
| C | POI | 1.0 | DEG/SEC | MAX MANY RATE Y |
| C | POI | 1.0 | DEG/SEC | MAX MANY RATE Z |
| C | PHIFOV | 40.0 | DEG | MAX RNC ATT FROM TRK STB (CMG) |
| C | RFIXED | 1 | --- | INITIAL SYSTEM RELIABILITY |
| C | SCFL | 0 | --- | SPEC. CMD SYNC FLG O=NO, 1=YES |

| C | SLBMX | 50000 | LB | MAXIMUM SYSTEM WEIGHT |
| C | SPECO(1) | .9 | --- | SANDC S/S REL. REQ. |
| C | SPECO(2) | .9 | --- | AP S/S REL. REQ. |
| C | SPECO(3) | .9 | --- | DPN S/S REL. REQ. |
| C | SPECO(4) | .9 | --- | COMM S/S REL. REQ. |
| C | SPECO(5) | .9 | --- | EP S/S REL. REQ. |
| C | TACCEL | 20.0 | SEC | ACCEL TIME FOR MANY (CMG) |
| C | TETRX | 180.0 | DEG | MAX MANY ANGLE (CMG ONLY) |

| C | THOLD | 100000 | MIN | TIME VEH. INERT HOLD (CMG) |
| C | TL | 1.0 | --- | TIME B/W UNLOAD WHL MANT (CMG) |
| C | TPMIN | 1.0 | SEC | MIN P/L SCAN PERIOD |
| C | TPRFL | 0 | --- | TLMR PROD FLG O=SEPARATE |

CONT. ON PG 5

PG 4 OF 21
DATA ITEST1, ITEST2, ITEST3, ITEST4, ITEST5/9, 14, 2, 11, 5/
READ 6, ITITLE

FORMAT (13A6)
DEBUG = 0
SPEC6 = SPEC6
DO 6 I = 1, 5
SPEC1(I) = -1.
READ (5, REQUIR)
READ (5, DESIRE)
READ (5, OPTION)
SPEC6) = SPEC6
THST=FE
ISEQ=ISATOR
IREL=ISATOR
ISAT=ISATOR
CALL PRESET (IERRI)

IF (IERRI .LT. 0) THEN
  T
GO TO 99
ELSE
WRITE (6, REQUIR)
WRITE (6, DESIRE)
WRITE (6, OPTION)
PRINT 9500
END IF
CONT. ON PG 7
CALL SAVE(CHOS2,NCHOS2,NOWAT,TEST2,ENDDB)

CALL READD(B,ENDDB)

IF (ITR .NE. 0 .OR. MICRO .EQ. 3) THEN
   GO TO 93
ELSE
   IF (ITR .GE. 0) THEN
      CALL DPII(IP3,(ERR(3),ITR,MCONF,CHOS3,NCHOS3,NOWAT)
   END IF
END IF

NEQUIP(3) = 0
DO 103 I = 1, ITEST
CONT. ON PG 13

PG 17 D 21

10-13
IF (ITER .NE. 0 .OR. MICRO .EQ. 4)

GO TO 94

00 29 I=1,9

94

CALL COMM (PIC4, IERR(4), ITER, NCONF, ICHOS4+NCHOS4)

NEQUIP(4)=0

DO 104 I=1,TEST4 > 15

IF (ICHOS4(I) .LT. 0 .AND. MICRO .LT. 4)

GO TO 13

CONT. ON PG 15

PG 10F 21

10-15
CALL RELY(IRTN,IREL,NEQUIP)

PRINT 3000,IRTN

3000 FORMAT (5H IRTN,II.O)

IF (IRTN.LT.0) THEN

GO TO 13

IR1=1

IR2=NEQUIP(1)

DO 31 IR=1,IR2

NCHOS1(IR)=NCHOSE(IR)

IR1=IR2+1

IR2=NEQUIP(2)

DO 32 IR=1,IR2

NCHOS2(IR)=NCHOSE(IR1)

CONT. ON PG 19
IR2 = NEQUIP(3)
DO 33 IR = 1 * IR2
NCHOS3(IR) = NCHOS(EIRI)

33
IR1 = IR1 + 1
IR2 = NEQUIP(4)
DO 34 IR = 1 * IR2
NCHOS4(IR) = NCHOS(EIR1)

34
IR1 = IR1 + 1
IR2 = NEQUIP(5)
DO 35 IR = 1 * IR2
NCHOS5(IR) = NCHOS(EIR1)

35
IR1 = IR1 + 1
CONTINUE
CALL STRUCT(NCONF)
CALL THRML(ERR(7), NCONF)
NCHOS(NOWAT) = 0
CALL COSTS(NCONF, NEQUIP)
CONT. ON PG 20
SUBROUTINE PRESET(IERR)

C ** THIS SUBROUTINE CALCULATES WHAT USED TO BE EITHER CONSTANTS OR
C ** INPUT VALUES. IT NOW WILL CALCULATE THE VALUES OF THESE
C ** CONSTANTS FROM OTHER INPUT VALUES WHERE THEY ARE GIVEN ELSE-IT
C ** WILL USE THE OLD CONSTANT VALUES.

COMMON /USERI/  ALPHA, AX, RX, AZ, OMEGA,
                POTA, POTA, POTA, POTA,
                PHIT, PHIT, PHIT, PHIT,
                THEM, THMD, TL, TMIN, TSMALL,
                XN, XWW, XNN, XN, XN,
                ZN, NULL

COMMON /USERR/  ISPRT, ISUB, KEQPT, RFIXED, SLBMX

COMMON /USERI/  APOGEE, COMRAI, DIAMAX, EEEWTA(9), EPME,
                ECM1WT, ECM1XL, ECM1YL, ECM2L, ECM2WT,
                ECM2XL, ECM2YL, ECM2ZL, FE, IAGNYC,
                IDEBUG, ISATOR, MB12SM, OECDR, OREINC, PERIGE,
                MICRO, RELME, SPEC1, T, XCSAI

COMMON /BTWN/   ACSSN, ACMP, ALT, AREA, BATCAP,
                BITRAT, CLI, CMNW, SATAM, DT,
                DX, DY, DZ, EQBGL, EQSSID,
                FC, FF, FHTN, HWT, HTFIPR,
                HTPT, HTRPRB, HTRPI, IET1PRC,
                LMBDD, NC, OMEGS, PASSR, PJ,
                PL, PLMIN, POCNWT, RADA, RADA,
                RA1, RJA, SABOLG, SALTG, SATW

CONT. ON PG 2

PG 1 OF 15

10-23
305'.

NOTION = 0.0

A AIRRSR.

II CONT

RS2: PEI/S

F S

T TE

00. ANIO-LTE.19000

202

C ** THE FOLLOWING AREA PRESETS RELIABILITY CONSTANTS**

C

---------

2 A5

GO TO 200

2 A4

110

IF (ALT .LT. 500.)

T

ORBINC = 35.

F

IF (ALT .GE. 500. .AND. ALT .LE. 19000.)

T

ORBINC = 80.

F

IF (ALT .GT. 19000.)

T

ORBINC = 0.0

CONT. ON PG 4

PG 3 OF 15

10-25

BETA = 1.6

B = 1./BETA

RS1 = T/(1-ALOG(SPEC1/1.0))**B

RS2 = SPEC1/0.88

A = AMAX1(RS1,RS2)
IF (SPEC(1) .EQ. -1.)

SPEC(1) = EXP(-T/(1.43))**BETA)

IF (SPEC(2) .EQ. -1.)

SPEC(2) = EXP(-T/(1.53))**BETA)

IF (SPEC(3) .EQ. -1.)

SPEC(3) = EXP(-T/(1.33))**BETA)

IF (SPEC(4) .EQ. -1.)

SPEC(4) = EXP(-T/(1.30))**BETA)

CONT. ON PG 5
RETURN

THE FOLLOWING AREA PRESETS VEHICLE SIZING CONSTANTS

A7

CONTINUE

= = = DETERMINE PJ AND RJ = = =

EQMWT = EQMWT + EQM2WT

DD 305 I = 1.9

EQMWT = EQMWT + EQM2WT

SATWT = 36.9 = EQMWT = 0.672
EQQYOL = 0.1 = SATWT
SATDAM = (EQQYOL = 2201.) = 0.333
EDBLO = SATDAM

(FISATDAM.LE.DIAMAX) F

T

GO TO 306

SATDAM = DIAMAX

(EQBLG = EQBYOL = 2201. / (ISATDAM = SATDAM))

306

SATINX = (SATWT = SATDAM = SATDAM / 8.)

RJ = SATINX

CONT. ON PG 7

PG 60F 15

10-28
C

- DETERMINE MISSION EQUIPMENT DIMENSIONS -

IF 1 EQM1XL NE 1.E10 AND EQM1YL NE 1.E10 AND EQM1ZL NE 1.E10 AND
   EQM2XL NE 1.E10 AND EQM2YL NE 1.E10 AND EQM2ZL NE 1.E10
T

GO TO 400

F

EQMDEN = 25.0

V1 = EQM1WT/EQMDEN
V2 = EQM2WT/EQMDEN

IF I MB12SH LT 2)
T

GO TO 350

F

IF (EQM1YL EQ 1.E10)
T

EQM1YL = (V1*1728/.06)*.333

IF (EQM1ZL EQ 1.E10)
F

CONT. ON PG 9

GO TO 400

F

GO TO 350

F

CONT. ON PG 9

GO TO 350

CONT. ON PG 9

GO TO 400

F

CONT. ON PG 9

GO TO 350

CONT. ON PG 9

GO TO 400

F

CONT. ON PG 9

GO TO 350

CONT. ON PG 9

GO TO 400

F

CONT. ON PG 9

GO TO 350

CONT. ON PG 9

GO TO 400

F

CONT. ON PG 9

GO TO 350

CONT. ON PG 9

GO TO 400

F

CONT. ON PG 9

GO TO 350

CONT. ON PG 9

GO TO 400

F

CONT. ON PG 9

GO TO 350

CONT. ON PG 9

GO TO 400

F
THE FOLLOWING AREA PRESETS S AND C CONSTANTS

CONTINUE

F (POOTMX.EQ.0. AND. TACCEL.EQ.1.E10)

IF (TACCEL.EQ.1.E10)

TACCEL = 0.1 * TMAX

CONT. ON PG 13
GO TO 405

12 B8

GO TO 405

401

TACCEL = 0.

CALCULATE OMEGR

405

IF (OMEGR .NE. 1.E10) T

GO TO 404

WR1 = 18000./IRJ/4636.8.

WR2 = 90.-(30.*T)/(7.5*12.)

WR = OMEGR

IF (OMEGR.EQ.1.E10) T

WR = RMAXI(WR1,WR2)

IF (WR.GE.60.) F

CONT. ON PG 14

PG 139F 15

10-35
GO TO 402

WR = 60.
GO TO 403

IF (WR .LE. 90.)

WR = 90.
OMEGR = WR

CALCULATE XNN

PHIMAX = AMAX1 (PHIRX, PHIRZ)

HV = RJ/4636.8 = OMEGR * 6.28318 / 60.

IF (XNN .EQ. 1.0) ?

XNN = 0.0525 * PHIMAX * HV

CONT. ON PG 15

PG 1 OF 15
C3

IERR = 0

RETURN

END
SUBROUTINE INITILINCONF(IERR)

THIS SUBROUTINE SETS APPROXIMATIONS FOR ALL VALUES IN BTWN WHICH ARE USED BEFORE THEY ARE CALCULATED

DIMENSION NCONF(6)

COMMON /USER1/ ALPHA, AX, AY, AZ, OPHI
EA, EANT, EP, MANV
OMEGAO, PDDAY, APOTX, PDDAY, PDDOY, PDDOY
PDDOZ, PDDOZ, PDDOZ, PDDOZ
PHIFOY, PHIRX, PHIR, PHIZ, TACCEL
THE.TAX, THOLD, TL, TPRIN, TSMALL
XW, XNN, XNNN, XN, YW

COMMON /USER1/ APOGEE, COMMAT, DIAMAX, EEOWT(9), EPME,
EDINW, EOMX, EOMXL, EOM1ZL, EOM2ZL, FEA,
IDEBUG, ISATOR, MBIZSH, OPTEMP, ORBINC, PERIGE
MICRO, RELME, SPECI61, SPECI, T, XCOIR
XMER, XYED

COMMON /BTWN/ ACSSN, ACSW, ALT, AREA, BATCAP
BITRAT(2), CLIFE, CONWAT, O, DZ
FC, FE, HARMIT, HPT, HTPRO,
HTPT, HTRPRB, HTRPRM, IBTLOC
LMBD, NC, OMES, PASST, PJ
PL, PLMT, POCNM, RACR, RADA,
RAT, RJ, SABOLG, SATL, SATWI

SATINT, SATXCO, SATYCO, SATZCO, SAI, XL
SAIYL, SAIYL, SIDE, SYSL, THCHNT.
THRUST(2), TI, TNAIT, TPRIN, VB
VCHP, VOL, WATE, WB, WBT
WT, XJ, XNZERO, YJ, ZJ

COMMON/PRTCOM/ ACCRCY, AN, AN, BF, BS
CDPL(7,2), CISTAR, CDT, DOTE, DE
ORINW, EOBSTR, FEEN, FEOPS, FER,
GSE, IREL, ITRUNC, MDOLD, NAME(3,60),
DPS, PAYINY, PAYUL, PAYR, PE,
PMP, PMR, POWER(6), PU, PWI(60),
QCP, QCR, ROLI(60), SABMWT, SATDOP,
SATINV, SATR, SEIP, SEIR, SKIUA(6)

SSREL(16), SUBE(7), SUBT(7), SUBUE(7), SUBUP(7),
TA, TAU1(6), TB, TC, TE
TF, TOOLR, TOLOU, TOLOPS, TRAN
TS, TTVOLME(8), VOL(60), WEIGHT(16),
XLTOT, XMEH, XMEINV, XME, XMEV
XMEX, XMEIW, XVEST

IERR=0

CONT. ON PG 2
HERE IF A SPHERE

\[ SATDAM = (EQVOL = 3300.9)^{0.333} \]

\[ SATINX = SATNX \]
\[ SATINY = SATINX \]
\[ SATINZ = SATINX \]

IF SATDAM TOO BIG STOP PROGRAM

IF (SATDAM .GT. DIAMAX)

T

[IERR = 1]

F

IF (IERR .GT. 0)

T

RETURN

GO TO 100

SETS VALUES NEEDED BY S AND C

IF (INCONF(1) .NE. 1)

T

CONT. ON PG S

10-41
IxJ = SATINX
YJ = SATINY
ZJ = SATINZ
RJ = SATINX

GO TO 200

GO TO 120

IF (NCONF11) .NE. 2

T

GO TO 130

XJ = SATINX

YJ = SATINY
ZJ = SATINZ

GO TO 130

IF (HW.LT.500)

F

6

A7

T

GO TO 125

CONT. ON PG 6

PG 5 OF 8

10-42
SUBROUTINE COSTS (NCONF, NEQUIP)

*** THIS SUBROUTINE COLLECTS COSTS FOR CATALOG ITEMS AND CALCULATES ***
*** COSTS FOR CER ITEMS AND STORES THEM FOR OUTPUTTING ***

COMMON /USERC/ FEEPCT, IMETYP, NFV, NOV, PI

COMMON /USER1/ APOGEE, COMMAT, DIAVAX, EEQWT(19), EPME,

EQMWT, EQMXL, EQM1YL, EQM1ZL, EQM2WT,

EQM2XL, EQM2YL, EQM2ZL, XDOM1, IADMCY,

IDEBUG, (SATOR, MB12SH, OPMAY, ORBINC, PERIDE),

MICRO, RELME, SPEC(6), SPEC1, XDUM2, XCSAI,

XMER, XMEU

COMMON /BTWN/ ACSSN, ACSUP, ALT, AREA, BATCAP,

BITRAT(2), CLIFE, CONWIT, D, DT,

OX, OY, OZ, EQLG, EBOSID,

FC, FF, HARNWIT, HPT, HPIPE,

HTP, HTIRPB, HTIRPW, IBTLQC,

LMBOD, NC, OMECT, PASSR, PJ,

PL, PLMIN, POOWIT, RADA, RADAB,

RAT, RJ, SABOLD, SATLC, SATTWT,

SAYL, SATYC, SATYC, SATZCW, SAIXL,

STHRT(2), TI, TINUM, TRIPM, V8,

VCHP, VOL, WATC, WB, WBT,

WT, XJ, XNZERD, YJ, ZJ

COMMON /CHOOSE/ COST(5.60), DPIA(11.60), ICHOOSE(60),

NCHOOSE(60), REL(6.60), SKO17(60),

THM(4.60)

COMMON /PRI/ ACCRCY, AM, AN, BF, BS,

CDP(17.2), CISTAR, CTOT, DOTE, DE,

DRINT, EOBSTK, FEEINV, FEEOPS, FEER,

GSE, IREL, ITRUNC, MGOOL, NAME(360),

OPS, PAYINV, PAVOL, PAVK, PE,

PMP, PMR, POWER(6), PI, PMR(60),

QCU, OCR, ROLD(60), SABMWT, SATADV,

SATINV, SATR, SEIP, SEIR, SKTRU(6),

SSREL(6), SUBE(7), SUBT(7), SUBUE(7), SUBUP(7),

TA, TAN(6.6), TB, TC, TE,

TF, TODE, TOOLU, TOTOPS, TRUNC,

TS, TIT, VOLUME(6), VOL(60), WEIGH(6),

XLIT, XMEH, XMEINV, XME, XMEV,

XMEM, XMENT, XVEST

DIMENSION RE(17), RT(7), RP(7), BE(7), BT(7), BP(7),

X(7), FP(7), FT(7), FE(7), NCONF(7), NEQUIP(5))
```
<table>
<thead>
<tr>
<th>DATA</th>
</tr>
</thead>
<tbody>
<tr>
<td>FP</td>
</tr>
<tr>
<td>FT</td>
</tr>
<tr>
<td>FE</td>
</tr>
<tr>
<td>RE</td>
</tr>
<tr>
<td>RT</td>
</tr>
</tbody>
</table>

| RP   | /42670.2050.9400.14870.14000.53545.36660. |
| BE   | /627.715.500.620.272.393.587/ |
| BT   | /500.585.500.620.675.410.301/ |
| BP   | /444.745.566.738.668.263.182/ |
| SF   | /1.0/ |

| SEIR | 0. |
| OCR  | 0. |
| PMR  | 0. |
| SUMTD | 0. |
| TOOLR | 0. |
| SEIP  | 0. |
| OCU  | 0. |
| PKP  | 0. |
| SUPE  | 0. |
| TOTSUM | 0. |
| SATR  | 0. |
| SATINV | 0. |
| XMENW | 0. |
| PAYR  | 0. |
| PAYINV | 0. |
| PAYOUL | 0. |
| GSE  | 0. |
| XLTOT | 0. |
| CLOT | 0. |
| FEER | 0. |
| FeeINV | 0. |
| DOTE | 0. |
| XVEST | 0. |
| OPS | 0. |
| DE  | 0. |
| TE  | 0. |
| PE  | 0. |
| PU  | 0. |
| SYSR | 0. |
| SYSU | 0. |
| Q5 | 0. |
| PSP | 0. |
```
PSE = 0.
1BTFRS = 1
IMRST = 1

DO 1 I = 1 * 7

SUBSI(I) = 0.
SUBSEI(I) = 0.
SUBE(I) = 0.
SUBT(I) = 0.
SUBRI(I) = 0.
SUBUEI(I) = 0.
SUBUPII(L) = 0.

SUBUI(I) = 0.

DO 2 I = 1 * 60

COMPSII(I) = 0.
COMPSPI(I) = 0.
COMPUII(I) = 0.
COMPRI(I) = 0.

MEQUIII(I) = MEQUIII(I)

DO 3 J = 2 * 5

MEQUIII(J) = MEQUIII(J-1) + MEQUIII(J)

X(1) = WATE
X(2) = HARNWT
X(3) = THCMWT
X(4) = CONWWT
X(5) = HEIGHII(2)
X(6) = PASSTR
X(7) = POCHWT
PRINT 9004, (X(I), I = 1 * 7)

9004
FORMAT (10E11.4)

CONT. ON PG 4

PG 3 OF 25
10-50
IF IHOOSE(I).EQ.0)
T
GO TO 200

C COMPUTATIONS FOR CATALOG ITEMS

 IF (I.GT. REQUIP(J))
T
J=J+1

ICHS = ICHOSE(I)/100
XFP = 1.
XE = 1.
XFT = 1.
C1 = COST(I+1)
Q = NOV + NFV
DP = Q * NCHOOSE(I)
QS = S * NCHOOSE(I)
P5 = COST (3*I).
QREF = COST(4*I)
FO = NCHOOSE(I)/QREF

IF (FO.LT.1.)
F
CONT. ON PG 5

REPRODUCIBILITY OF THIS ORIGINAL PAGE IS POOR
C. SET SCALE FACTORS FOR AUXPRO CATALOG ITEMS

GO TO 170

XFE = 13.5
XFT = 2.4
XFP = 5.0

GO TO 170

TO 170

IF (ITHRST .GT. 1)

ITHRST = ITHRST + 1

IF (ITHRST(1).LT.REL(1,1))

XFT = 0.25

CONT. ON PG 8

PG 7 OF 25

10-54
COST(4:I) = C1*(OP\[485\] - 1)*FRE*(1000\[XFE/0\]

C ** COMPUTE COMPONENT CUM AVG 5 PROD. ENG. COST

COMP5E(I) = 2000*C1*(0.485 - 1)*FRE*PI*XFE

COMP(I) = COST(3:I) + COST(4:I)

GO TO 185

189 CONTINUE

DO 183 JJJ = 1, 5

183 COST(JJJ:1) = 0.0

185 CONTINUE

IF (DEBUG.EQ.1) THEN

PRINT 9000*(COST(JJJ:1), JJ = 1, 4), COMPr(I), COMPl(I),

COMPSP(I), COMP5E(I)

1 = 1 + 1

GO TO 100

C ** COMPUTATIONS FOR SUBSYSTEM COSTS BASED ON COST ESTIMATING

C ** RELATIONSHIPS (C.E.E.R.-S)

CONT. ON PG. 13

PG 1206. 25

10-59

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR.
"C SET WIRING HARNESS CER FACTORS"

"C SET THERMAL CER FACTORS"

SET POWER CONVERTERS CER FACTORS
(WONE NEEDED AT THIS TIME)

SET PROPELLANT FEED SYSTEM CER FACTORS

IF (INCONF1) GT 1
T
GO TO 226

IF (INCONF1) LT 1
F

GO TO 226

RE(M) = 129200
BE(M) = 272
FE(M) = 0.507
FT(M) = 0.325
FP(M) = 0.615

GO TO 270

RE(M) = 545640
BE(M) = 0.222
FE(M) = 0.268
FT(M) = 0.619
FP(M) = 0.840

GO TO 270

CONT. ON PG 16
C SET STRUCTURES CER FACTORS

230

STRF = 0.5054 \# SATMT # (1.0 - 160)

RATIO = PASS/STRF/SATMT

IF (STRF .GE. RATIO)

T

STF = 1.0 - (STRF - RATIO) / STRF

IF (STRF .LT. RATIO)

F

STF = 1.0 - (RATIO - STRF) / RATIO

IF (NCONF5 .EQ. 2 .OR. NCONF5 .EQ. 4 .OR. NCONF5 .EQ. 6)

F

T

GO TO 231

IF (FT (M) .LE. 2.5) GO TO 231

CONT. ON PG 17

PG 18 OF 25

10-63
DESIGN ENGINEERING COSTS (COMPE OR DE)

COST(1*K) = COMPER * SF * PI

TEST EVALUATION COSTS (COMPT OR TE)

IF (M.EQ.5)

T

X(M) = ORNL

COST(2*K) = RT(IN)*X(IN)*BT(M)*SF*PI*FT(IN)

SUBTOTAL

COMPRI(K) = COST(1*K) + COST(2*K)

UNIT PRODUCTION COST

COST(3*K) = RP(M)*X(M)*BP(M)*SF*PI*Q**(-.152)*FP(M)**1.277

IF (M.EQ.1)

T

C UNIT ENGINEERING COSTS

C COST(4*K) = COMPER*(Q**(-.485-1.))*SF*PI/D

COMPUTE COMPONENTCUM AVG 5 UNIT PROD. COST

CONT. ON PG 19
C ** COMPUTE COMPONENT CUM AVG 5 PROD. ENG COST

\[
COMPSEIK) = 0.2365 \times \text{COMPRESSFPI#FE(M)}
\]

C ** SUBTOTAL PRODUCTION

\[
COMPUI) = \text{COST}(3+K) \times \text{COST}(4+K)
\]

IF (IDBUGQ_Equals_1)

\[
\text{PRINT 9000.}, \text{COST}(JJ+K), JJ=1,4, \text{COMPRIK), COMPUIK),}
\]

\[
\text{COMPSPK), COMPSEIK),}
\]

GO TO (280, 280, 281, 280, 282, 283, 284)

C ** THERMAL CER SUB-TOTALING

\[
\text{SUBUE}(5) = \text{SUBUE}(5) \times \text{COST}(1+K)
\]

\[
\text{SUBT}(5) = \text{SUBT}(5) \times \text{COST}(2+K)
\]

\[
\text{SUBA}(5) = \text{SUBA}(5) \times \text{COMPRIK)
\]

\[
\text{SUBUE}(5) = \text{SUBUE}(5) \times \text{COST}(4+K)
\]

\[
\text{SUBUP}(5) = \text{SUBUP}(5) \times \text{COST}(3+K)
\]

\[
\text{SUBU}(5) = \text{SUBU}(5) \times \text{COMPUIK)
\]

\[
\text{SUBSE}(5) = \text{SUBSE}(5) \times \text{COMPSEIK)
\]

\[
\text{SUBSP}(5) = \text{SUBSP}(5) \times \text{COMPSPK)
\]

GO TO 300

CONT. ON PG 20

PG 197F 25

10-66
SUBEI(7) = SUBEI(7) + COST(1*K)
SUBT(7) = SUBT(7) + COST(2*K)
SUBR(7) = SUBR(7) + COMPiK)
SUBUE(7) = SUBUE(7) + COST(4*K)
SUBUP(7) = SUBUP(7) + COST(3*K)
SUBU(7) = SUBU(7) + COMPiK)
SUBSE(7) = SUBSE(7) + COMPSiK)
SUBSP(7) = SUBSP(7) + COMPSiK)
GO TO 300

C

== AUX PROP CER SUB-TOTAL

SUBEI(2) = SUBEI(2) + COST(1*K)
SUBT(2) = SUBT(2) + COST(2*K)
SUBR(2) = SUBR(2) + COMPiK)
SUBUE(2) = SUBUE(2) + COST(4*K)
SUBUP(2) = SUBUP(2) + COST(3*K)
SUBU(2) = SUBU(2) + COMPiK)
SUBSE(2) = SUBSE(2) + COMPSiK)
SUBSP(2) = SUBSP(2) + COMPSiK)
GO TO 300

C

== STRUCTURE CER SUB-TOTAL

SUBEI(6) = SUBEI(6) + COST(1*K)
SUBT(6) = SUBT(6) + COST(2*K)
SUBR(6) = SUBR(6) + COMPiK)
SUBUE(6) = SUBUE(6) + COST(4*K)
SUBUP(6) = SUBUP(6) + COST(3*K)
SUBU(6) = SUBU(6) + COMPiK)
SUBSE(6) = SUBSE(6) + COMPSiK)
SUBSP(6) = SUBSP(6) + COMPSiK)

CONT. ON PG 21

SUM SUB-TOTALS BY SUBSYSTEMS OF CATALOG ITEMS

C

CONT. ON PG 21

REPPLICABILITY OF THE ORIGINAL PAGE IS POSSIBLE
IF (I,J.NE.1)  
  CONTINUE

CONT. ON PG 22

PG 2 OF 25
C SET FACTOR FOR GENERAL

**E3**

FDSE = 1.0

**CONT. ON PG 25**
IF (IPRINT .EQ. 0) THEN
  RETURN
ENDIF (NACCEP.GT.1) THEN
  GO TO 100
ENDIF (NACCEP.GT.1) THEN
  PRINT 90000
  FORMAT(1HI)
  PRINT 90001
  FORMAT(1X)
  PRINT 90002
  FORMAT(38X,41HNASA SYSTEMS COST/PERFORMANCE STUDY )
  PRINT 9001
  PRINT 9010
END IF
C  PRINT STABILIZATION AND CONTROL INFORMATION

PRINT 104.

FORMAT(12X,25HSTABILIZATION AND CONTROL)

ICONF = NCONF+1.

GO TO (106,108,110,112,114),ICONF

C  DUAL SPIN

PRINT 107.

FORMAT(14X,27HSTABILIZATION AND CONTROL)

GO TO 116

C  YAW SPIN

PRINT 109.

FORMAT(14X,26HSTABILIZATION AND CONTROL)

CONT. ON PG 6
C 3-AXIS MASS EXPULSION

A2

PRINT II1

FORMAT(14X,4HCONFIGURATION -- THREE AXIS MASS EXPULSION)

GO TO I16

C MASS EXPULSION WITH CONTROL MOMENT GYROS (CMG)

A3

PRINT II3

FORMAT(14X.58HCONFIGURATION -- MASS EXPULSION WITH CONTROL MOMENT GYROS)

GO TO I16

C MASS EXPULSION WITH PITCH MOMENTUM WHEEL

A4

PRINT II5

FORMAT(14X.58HCONFIGURATION -- MASS EXPULSION WITH PITCH MOMENTUM WHEEL)

CONT. ON PG 7

PG 60F 38
FORMATTING

PRINT 117, ACCURACY

PRINT AUXILIARY PROPULSION INFORMATION

PRINT 119

FORMAT(12X,20Hauxiliary propulsion)

CONF = NCONF(2)

GO TO (120, 122, 124), ICONF

CONT. ON PG 8
C BIPROPELLANT

A6

PRINT 125

FORMAT(1X,3DIOMPFGURATION = - BIPROPELLANT)

A7

PRINT 127+1

FORMAT(16X,18TOTAL IMPULSE = *F1.0* BM(LB-SEC))

C PRINT DATA PROCESSING INFORMATION

PRINT 129

FORMAT(12X,38DATA PROCESSING AND INSTRUMENTATION)

ICONF = MCONF(3)

GO TO (130+132)*ICONF

C GENERAL PURPOSE PROCESSOR

PRINT 131
PRINT 162

162 FORMAT(12X,16HELECTRICAL POWER).

ICONF = NCONF(S)

GO TO (164,166,168,170,172,174).ICNF

C SHUNT - PADDLE MOUNTED

PRINT 165

165 FORMAT(14X,5H_CONFIGURATION - - SHUNT - PADDLE MOUNTED SOLAR ARRAY)

GO TO 176

C SHUNT - BODY MOUNTED

PRINT 167

167 FORMAT(14X,5H_CONFIGURATION - - SHUNT - BODY MOUNTED SOLAR ARRAY)

GO TO 176

C SHUNT AND DISCHARGE - PADDLE MOUNTED

CONT. ON PG 14
FORMAT(14X,26HCONFIGURATION -- CYLINDER)
GO TO 190
C BOX

FORMAT(14X,21HCONFIGURATION -- BOX)
GO TO 190
C SPHERE

FORMAT(14X,24HCONFIGURATION -- SPHERE)
C PRINT VEHICLE WEIGHT AND LAUNCH WEIGHT

PRINT 191. SATWT. SATWT

F11.2+5H(LBS))
C PRINT EQUIPMENT BAY DIMENSIONS

CONT. ON PG 17
Go to (202, 204), ICONF

C SINGLE SYSTEM

202 PRINT 203

203 FORMAT(14X, 'CONFIGURATION -- SINGLE SYSTEM')

GO TO 206

C DUAL SYSTEM

204 PRINT 205

205 FORMAT(14X, 'CONFIGURATION -- DUAL SYSTEM')

C PRINT REL STATS

206 PRINT 207, MADDLD

207 FORMAT(16X, 'MEAN MISSION DURATION', 7X, F11.4)

PRINT 209, ROLDITRUNC

209 FORMAT(16X, 'RELIABILITY', 7X, F11.3)

CONT. ON PG 19
PRINT 260, SKTAU(3)

FORMAT(16X,26HSUBSYSTEM DEVELOPMENT TIME, 20X, F5.1, 8H(MONTHS))

PRINT SUBSYSTEM QUALIFICATION TIME

PRINT 262, SKTAU(4)

FORMAT(16X,26HSUBSYSTEM QUALIFICATION TIME, 18X, F5.1, 8H(MONTHS))

PRINT SYSTEM DEVELOPMENT AND FLIGHT READINESS TIME

PRINT 264, SKTAU(5)

FORMAT(16X, 46H SYSTEM DEVELOPMENT AND FLIGHT READINESS TIME, F5.1, 8H(MONTHS))

PRINT SCHEDULE DURATION TO LAUNCH

PRINT 266, SKTAU(6)

FORMAT(16X, 29H SCHEDULE DURATION TO LAUNCH, 17X, F5.1, 8H(MONTHS))

PRINT 9000

IF (IPRINT.LE.1) THEN
   CONT. ON PG 24
ELSE
   PG 230F 38

END OF SYSTEM PRINT-OUT --- BEGIN SUBSYSTEM IF REQUIRED

CONT. ON PG 24
C2
RETURN

C3
PRINT 9999, ITITLE
PRINT 1000, NACCEP

1000
PRINT 104
(CONF = NCONF1)

GO TO 1004, 1006, 1008, (1010, 1012), ICONF

C DUAL SPIN

1004
PRINT 107

GO TO 1014

C YAW SPIN

1006
PRINT 109

GO TO 1014

C 3-AXIS MASS EXPULSION

1008
PRINT 111

GO TO 1014

CONT. ON PG 25
PRINT S+C EQUIPMENT IDENTIFIERS.

IK = NEQUIP(I)
PRINT 1020.*(ICHOSE(I)*I=1*IK)

FORMAT(14X,25HEQUIPMENT CODE IDENTIFIER.*14(I*4))
PRINT 1022.*(ICHOSE(I)*I=1*IK)

FORMAT(14X,25HEQUIPMENT QUANTITIES.*14(I*4))
PRINT 1024.*WEIGHT(I)*VOLUME(I)*POWER(I))

FORMAT(24X,7HEIGHT.*F11.2,13HLB), VOLUME.*F11.2,27HFT**3), POWER REQUIREMENT.*F11.1,6H(WATT))

PRINT S+C COSTS

CONT. ON PG 26
```
PRINT 1020. ICHOOSE(I, I=J+K)
PRINT 1022. INCHOOSE(I, I=J+K)
PRINT 1024. WEIGHT(4), VOLUME(4), POWER(4)
PRINT 1026. SUBE(4), SUBT(4), SUBUP(4), SUBUE(4)
PRINT 1028. SSREL(4)
PRINT 1030. TRU(J, J=1+5)

C PRINT E.P. 5/5 INFO.
PRINT 162
ICONF = NCONF(5)
GO TO 11060, 1064, 1062, 1066, 1068, 1070, ICONF

C SHMT / PADDLE

1060
PRINT 165
GO TO 1072

C SHMT / BODY

1062
PRINT 167
GO TO 1072

C SHMT + DSCHG / PADDLE

1064
PRINT 169
GO TO 1072

C SHMT + DSCHG / BODY

1066
PRINT 171
CONT. ON PG 31
```
FORMAT1: 14X, 27HEQUIPMENT BAY STRUCTURE WT., 3X, F11.16H (LBS)/.
14X, 30H SOLAR ARRAY BOOM AND DRIVE WT., F11.16H (LBS)/.
14X, 14ADAPTOR WEIGHT, 16X, F11.16H (LBS))
PRINT 1026, SUB16, SUB16, SUB16, SUB16, SUB16
C ******** END OF SUBSYSTEM PRINT - BEGIN ASSEMBLY LEVEL (IF REQUIRED) ********
PRINT 1032, IERR14
C *
PRINT 9000
IF (PRINT .GE. 2)
T
RETURN
PRINT 9999, TITLE
PRINT 2000, NACCEP
FORMAT1: 50H = ASSEMBLY DESCRIPTIONS - DESIGN NUMBER .13.8H
PRINT 104
PRINT 2010
FORMAT1: 39X, 4HUNIT, 3X, 4HUNIT, 3X, 4HUNIT, 29X, 7H VEHICLE, 7X, 7H VEHICLE/.
+ 1X, 96H (IDENT TYPE NO. WEIGHT VOLUME POWER D.E. COST
T.E. COST, PROD. COST, ENG COST).
[I = 1
[IK = NEQUIP1])
CONT. ON PG 35
IK = MEQUIP(4) + IK
PRINT 9000
PRINT 146
PRINT 2010
DO 2400 I = J * IK

PRINT 2110 * ICHOOSE(I) * NAME(J, I) * J = 1, 3) * MCHOOSE(I) * REL(I, I) * VQL(I),
   PWR(I), (COST(J, I) * J = 1, 4)
C PRINT E.P.

IJ = [K + 1]
[K = MEQUIP(K) + IK
PRINT 9000
PRINT 162
PRINT 2010
DO 2500 I = J * IK

ICHS = ICHOOSE(I) / 100
IF (ICHS EQ 2)
   T
   REL(I, I) = WB
   F
   IF (ICHS EQ 2)
      T
      VQL(I) = VB
   F

CONT. ON PG 37
SUBROUTINE FILTER(INCONF, ICOD)  
  
FILTER CHECKS FOR COMPATIBLE COMBINATIONS OF CONFIGURATIONS  
A MINUS I IS RETURNED FOR UNACCEPTABLE COMBINATIONS  
INCONF IS ARRAY OF CONFIGURATIONS  
ICOD IS CODE RETURNED  
  
DIMENSION INCONF(6)  
  
COMMON /USER1/ ALPHA, AZ, AX, EANT, ALPH, EA, EEANT, EPI, K, MANY,  
OMGDR, POOTRX, POOTRY, POOTZ, POOTX, POOTY, POOTZ, POOTX,  
PHIFOV, PHIRX, PHIRY, PHIRZ, TACCEL,  
THETMX, THOLD, TL, TPRMIN, TSMALL,  
XN, XNY, XNN, XNX, ZN, YN  
  
COMMON /USER3/ARRAY(11,3), BTRMX, NMSEQ, OPSMS, SCSFL  
  
COMMON /USER4/ BWIDTH, FREO1, FREOR, IOPTCM(3), LINK  
  
COMMON /USER1/ APGE, COMRAT, DIA MAX, EEQMT(9), EPHME,  
EOM1W, EOM1XL, EOM1YL, EOM2WL, EOM2W1, EOM2XL, EOM2YL, FE, IAGNY,  
IDEBUG, ISATOR, MB12SH, OPTEMP, ORBINC, PERIGE,  
MICR, RELME, SPEC1, SPEC1, T, XCGSAR  
  
ICOD = 0  
  
C CHECK S AND C  
  
IF (POOTRX LT 0.01 AND NCONF 1 EQ 1)  
  
ICOD = -1  
  
CONT. ON PG 2  
PG 1 OF 5  

10-111
AR
A

DO TO 45

IF (NCONF14) .LE. 3)

GO TO 45

DO 43 I=1,11

DO 43 J=1,3

IF (ARRAY(I+J) .GT. 0)

GO TO 45

CONTINUE

ICODE=-1

CONTINUE

IF (NCONF5) .EQ. 1 .AND. NCONF11 .LT. 3)

CONT. ON PG 5
IzcOD- = I
ICOOE -1
ICONFIS) ESND.NWFI LT3
PG 5 FI
MAL
10-115
SUBROUTINE READDB(IENDDB)
C THIS READS THE DATABASE FOR ONE SUBSYSTEM AT A TIME
C IDB IS SET AS THE DATABASE IS READ BY SCANNING EQUIP NUMBERS

DIMENSION STORE(55)
COMMON /DBCOM/DATAB(55,100),I0B(30)
DATA STORE//55=0./

IF (IENDDB .LE. 1) F
T
GO TO 2

IF (STORE(1) .EQ. 0.) F
T
GO TO 2

DO 1 J=1,55

DATAB(J,1)=STORE(J)

1=2
I0D=STORE(1)/100.
I0B=1
GO TO 3

CONT. ON PG 2
IDOLD = 0
IDB = 1

READ (1, 100) (DATAB(I, I), J = 1, 55)

IF (EDF(I)) 90.110

100

FORMAT (ES.0, A2, 3A6, SE10.0, /, 5(E10.0, /), SE10.0)

110

IF (IDOLD .EQ. 0)

IDOLD = DATAB(I, I)/100.

ID = DATAB(I, I)/100.

C TEST FOR END OF SUBSYSTEM

IF (ID .LT. IDOLD)

GO TO 80

CONT. ON PG 3

PG 2 OF 4

10-117

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR
DO 6: J = 1 + 55

STORE(J) = 0.

I(OBJ(I)) = I - 1
REWIND I
E(NDOB) = I - 1

RETURN

END
DO 7 KKK = 1, 3

NAME (KKK, NOWAT) = (DATA (KKK + 2, J))

A1

CONTINUE

NOWAT = NOWAT + 1

A2

CONTINUE

RETURN.

END

PG 3 FINAL
SUBROUTINE THRM1L USES A METHODOLOGY FOR SIZING THE THERMAL SUB-SYSTEM FOR A VARIETY OF SPACECRAFT. THIS METHODOLOGY DETERMINES SIZE AND PERFORMANCE OF THE THERMAL SUB-SYSTEM.

A GLOSSARY OF VARIABLES FOLLOWS -

CODE IS AS FOLLOWS -

U = USER INPUT
D = DATA BASE
I = INTERNAL
O = OUTPUT

VAR. NAME  CODE  UNITS (DEFAULT)  DESCRIPTION

CONT. ON PG

DATA SIGMA/0.1714E-08/0.05/442.0/.EMISS/60.0/.ALBDO/15S.0/.CONST/1.5/.PIE/3.1415926535/
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALBEDO</td>
<td>1.55 BTU/(HR*FT^2)</td>
</tr>
<tr>
<td>ALPHA</td>
<td>0.30 (DIMENSIONLESS)</td>
</tr>
<tr>
<td>ALT</td>
<td>N.M. ALITUDE</td>
</tr>
<tr>
<td>BV</td>
<td>1.1 VDC MAX BATT. VOLT.</td>
</tr>
<tr>
<td>CA</td>
<td>0.5 Amps BATT TRICKLE</td>
</tr>
<tr>
<td>CONST</td>
<td>1.6 K CONSTANT</td>
</tr>
<tr>
<td>EMISS</td>
<td>60 BTU/(HR*FT^2) EARTH EMISSION</td>
</tr>
<tr>
<td>EPSLON</td>
<td>0.75 (DIMENSIONLESS)</td>
</tr>
<tr>
<td>ETAT</td>
<td>0.73 (DIMENSIONLESS)</td>
</tr>
<tr>
<td>HPS</td>
<td>(BTU/HR) TOTAL HEATER POWER</td>
</tr>
<tr>
<td>MPIPE</td>
<td>(BTU/HR) HEAT DUE TO H.P.</td>
</tr>
<tr>
<td>HTPT</td>
<td>(BTU/HR) TOTAL HEAT PIPE</td>
</tr>
<tr>
<td>HTRPRB</td>
<td>(BTU/HR) BATT. HEATER POWER</td>
</tr>
<tr>
<td>HTRPHR</td>
<td>(BTU/HR) OTHER HEATER POWER</td>
</tr>
<tr>
<td>i</td>
<td>INDEX</td>
</tr>
<tr>
<td>IBTLOC</td>
<td>BATTERY LOCATION</td>
</tr>
<tr>
<td>ICONF</td>
<td>TYPE OF CONFIG.</td>
</tr>
<tr>
<td>ISATOR</td>
<td>1 EARTH ORIENTED</td>
</tr>
<tr>
<td>2 SUN ORIENTED</td>
<td></td>
</tr>
<tr>
<td>3 INERTIALLY ORI.</td>
<td></td>
</tr>
<tr>
<td>NC</td>
<td>NUMBER BATT CEL</td>
</tr>
<tr>
<td>NCONF(1)</td>
<td>S/C MACRO INDEX</td>
</tr>
<tr>
<td>NCONF(6)</td>
<td>VS MACRO INDEX</td>
</tr>
<tr>
<td>ORBINC</td>
<td>DEGREES ORBIT INCLINAT.</td>
</tr>
<tr>
<td>PCM</td>
<td>KG PHASE CHANGE MASS</td>
</tr>
<tr>
<td>PIE</td>
<td>3.14159265 CONSTANT</td>
</tr>
</tbody>
</table>

CONT. ON PG 3
**CC**

**H E H H R E E R R E T**

**R A P P L I P P A T I O N A**

**R A P S E P P O D E E**

**B E S R E E R R A**

**I E R R = X X X X X X X X X**

WHERE 0 MEANS NO SUCH REQUIREMENT, OR 1 MEANS WE HAVE THIS REQUIREMENT.

**INITIALIZATION FOLLOWS**

```
RAOA=0.
RAOB=0.
RAT=0.
HTPWP=0.
HTPRB=0.
HPT=0.
HTPIPE=0.
VCHP=0.
HTPI=0.
TMAX=1.0E+20
TMIN=-1.0E20
PMA=0.
PMIN=0.0
EAT=1.0
I=0
SARA=0.5*0
```

```
10 A1 5
1 = 1 + 1
```

**SEARCH FOR MIN (MAX TEMP) AND MAX (MIN TEMP), AND ACCUMULATE THE POWER (EXCLUDING XMRS AND BATTERIES)**
IF ( ICHOOSE(I) .LE. 0) THEN
  GO TO 50
END IF

IF ( I .EQ. IBTLOC ) THEN
  GO TO 20
END IF

TMAX = MIN ( TMAX, THRMOB(3, I) )

TMIN = MAX ( TMIN, THRMOB(4, I) )
PMIN = THRMOB(2, I) + PMIN
PMAX = THRMOB(1, I) + PMAX

GO TO 10

C == HERE IF WE HAVE THE BATTERY

20

TMINB = THRMOB(4, I) + 460

TMAXB = THRMOB(3, I) + 460

GO TO 10

50

CONTINUE

DMIN = PMIN = 3.41

CONT. ON PG 6
(F(PMAX=5.0, PMIN).

T

PMAX=PVAR=5

QMAX=PVAR*3.41

TMAX=MAX=460.

THIN=THIN+460.

ICONF=0

GO TO (G0+70+80)+ICONF

C ** SATELLITE LENGTH IN INCHES (MUST CONVERT TO CM) (FROM VS)
(CYLINDER)

60

LENGTH=SATLG=2.54*0.75

GO TO 90

C SATELLITE LENGTH IN INCHES (MUST CONVERT TO CM) (FROM VS)
(BOX)

70

LENGTH=SATLG=2.54*0.75

GO TO 90

C SATELLITE LENGTH IN INCHES (MUST CONVERT TO CM) (FROM VS)
(SPHERE)

CONT. ON PG 7
ORBITS GT 500 BUT LT 19000 AND
SOLAR ORIENTED AND
3-AXIS STABILIZED

(EQUATION 3.3.1.1)
C ** ORBITS GT 500 BUT LESS THAN 19000 AND, **
C ** EARTH ORIENTED AND, **
C ** SPIN STABILIZED (EQUATION 3.4.1.2) **

140
ALPHA=0.08
EPSILON=0.73

C = DETERMINE RADIATOR AREA
RADAR=OMAX/((SIGMA*EPSILON*TMAX**4)-(QS*ALPHA))

C = DETERMINE HEATER POWER
HTPWR=1.25*(((SIGMA*EPSILON*RADA=TMIN**4)~-IQMIN))

C = DETERMINE PCM
PCM=(0.26*ALPHA*QS=RADA=CONST)/40.

C = DETERMINE HEAT PIPE
HTPIPE=QMAX*LENGTH)/3.41
IERR=1001101
GO TO 390

C ** ORBITS GT 500 BUT LESS THAN 19000 AND, **
C ** EARTH ORIENTED AND, **
C ** 3-AXIS STABILIZED (EQUATION 3.4.1.1) **

150
ALPHA=0.08
EPSILON=0.73
C ** ORBIT LT 500, ORBITAL INCLINATION LE 30 DEGREES AND. **
C ** EARTH ORIENTED AND. **
C ** SPIN STABILIZED (EQUATION 2.1.2.2) **

\[
\text{GO TO 380} \rightarrow 19 \rightarrow C3
\]

CONT. ON PG 12

PG 1 DF 20
C = DETERMINE ISOThERALIZER HEAT PIPE

HTPIPE = (QMAX\times LNGTH)/3.41
IERR = 10011

GO TO 380

18 B1

ICONF = MCONF(1)

GO TO 1210, 210, 220, 220, 220, 220, 1CONF

C ** ORBIT LT 500, ORBITAL INCLINATION LE 30 DEGREES AND.
C ** SUN ORIENTED AND.
C ** SPIN STABILIZED (EQUATION 2.1.1.2)

210

ALPHA = 0.08

EPSLON = 0.73

C = DETERMINE RADIATOR AREA

RADA = QMAX/((SIGMA\times EPSLON\times TMAX\times 4)-(EMISS\times EPSLON)-0.5\times ALBTT\times ALPHA))

C = DETERMINE HEATER POWER

HTPWR = 1.25/((SIGMA\times EPSLON\times RADA\times TMIN\times 4)-QMIN)

C = DETERMINE HEAT PIPES

HTPIPE = (QMAX\times LNGTH)/3.41
IERR = 10011

GO TO 380

C ** ORBIT LT 500, ORBITAL INCLINATION LE 30 DEGREES AND.
C ** SUN ORIENTED AND.
C ** 3-AXIS STABILIZED (EQUATION 2.1.1.1)

CONT. ON PG. 13

PG 120F 20

10-134
\[ \text{ALPA} = 0.08 \]
\[ \text{EPSLON} = 0.73 \]

\[ \text{C} = \text{DETERMINE RADIATOR AREA} \]
\[ \text{RADA} = \text{QMAX}/(\text{SIGMA} \times \text{EPSLON} \times \text{TMAX}^2) \times (1 - \text{EMISS} \times \text{EPSLON}) \times (1 - \text{ALBOO} \times \text{ALPHA}) \]

\[ \text{C} = \text{DETERMINE HEATER POWER} \]
\[ \text{HTPWR} = 1.25 \times \left( \frac{(\text{SIGMA} \times \text{EPSLON} \times \text{RADA} \times \text{TMIN}^2)}{\text{OMIN}} \right) \]

\[ \text{C} = \text{DETERMINE HEAT PIPES} \]
\[ \text{HTPIPE} = \frac{(\text{QMAX} \times \text{LENGTH})}{3.4 \times 1} \]
\[ \text{IERR} = 10011 \]

\[ \text{GO TO 380} \]

\[ \text{IF ORBITGT 30} \]

\[ \text{C} \]

\[ \text{GO TO (240, 270, 270), ISATOR} \]

\[ \text{ICONE} = \text{NCONF(1)} \]

\[ \text{GO TO (1250, 250, 260, 260, 260), ICONF} \]

\[ \text{C = ORBIT LT 500, ORBITAL INCLINATION GT 30 DEGREES AND, EARTH ORIENTED AND, SPIN STABILIZED (EQUATION 2.2.3.2)} \]

\[ \text{ALPHA} = 0.08 \]
\[ \text{EPSLON} = 0.73 \]

CONT. ON PG 14
C = DETERMINE RADIATOR AREA

\[ \text{RADA} = \frac{\text{QMAX}}{((\text{SIGMA} \cdot \text{EPSLON} \cdot (TMAX)^4)} - (\text{EMISS} \cdot \text{EPSLON} \cdot (TMAX) - (\text{QOS} \cdot \text{ALB0D}) \cdot \text{A (LPHA)/PIE}) \]

C = DETERMINE HEATER POWER

\[ \text{HTRPWR} = 1.25 \times ((\text{SIGMA} \cdot \text{EPSLON} \cdot \text{RADA} \cdot (TMIN)^4) - (\text{QMIN} - (\text{EMISS} \cdot \text{EPSLON} \cdot (TMIN))) \]

[ERR = 1011]

GO TO 380

C = DETERMINE PCM MASS

\[ \text{PCM} = (0.25 \times \text{QOS} \times \text{ALPHA} \times \text{RADA} \times \text{CONST}) / 40 \]

C = DETERMINE ISOTHERMALIZER HEAT PIPE

\[ \text{HTPIPE} = \frac{(\text{QMAX} \times \text{LNGTH})}{3.41} \]

[ERR = 11011011]

GO TO 380

CONT. ON PG 15

PG 10F 20
C = DETERMINE VARIABLE CONDUCTANCE HEAT PIPE

VCHP = OMAX8 * LMTCH / 3.41
IERR = IERR + 1100000000
HTPIPE = HTPIPE / 2.54
VCHP = VCHP / 2.54
RAT = RAD + RADR
HPT = HTRPWR + HTRPRB
HTPT = HTPIPE + VCHP

RETURN

END
SUBROUTINE COMM (IPIC, IERR, ITER, MCONF, MCHOSE, MCHOSE)
INTEGER RESET, SEO, SSS, GRP
REAL LMARG, NF, MODLOS, IBER

DIMENSION IPIC(9), ICHOOSE(11), MCHOSE(11), KPIC(9), MCONF(6),
KCHOOSE(11), PW(2), ISPFLD(2), J2SAVE(2)

DIMENSION SIGNO(2), LMARG(2), TCLOSE(2), GT(2), MODX(2)
DIMENSION BER(14), IBER(14), BESSJ(2), L(MPIC(9))

COMMON /USER4/BWIDTH(2), FREOQ(2), FREQR, OPTCH(3), LINK
NADIR

COMMON /USER1/ APDOEE, CMRRAT, DEMAAX, EEQW(9), EPME,
EOMWT, EDMLXL, EMYIL, EMZIL, EDM2WT,
EDM2XL, EDM2YL, EDM2ZL, FE, IAG, IC
DEBUG, SEQ, MB12SH, OPTEMP, ORBINE, PERID,
MICIO, RELME, SPEC(1, SPEC, XDOMI, XCOSAI

COMMON /BTMN/ ACSNM, ACSWP, ALT, AREA, BATCAP,
BITRAT(2), CLIFE, CONWWI, D, DT,
DQ, FD, HRWWI, IDT, IDTIP,
HPP, HTRPGB, HTRPRB, HRTHPR,
LABBO, NC, OMEOS, PASST, PJ,
PL, PLMIN, POCWNW, RAD, RADAB,
RAT, RJ, SABOLG, SATL, SATTW

COMMON /OBCOM/ OIBI(5), 100, 108130

EQUIVALENCE (J1, KPI(1)), (J7, KPI(6)), (J4, KPI(7)), (J5, KPI(8)),
(J6, KPI(9))

DATA SIGNO /10.10.10., LMARG /6.6./, SLANT /1.10.10.10.,
GTOT /1.10.10., GR /1.10.10/., T /1.10.10.10., NF /1.10.10.10.,
TCLOSE /0.00., POLOSS /0.0., GAMMA /1.0, BETA /1.0/.,
GT /1.10.10.10.10., MODX /0.0., ANOM LOS /0.0/,
COVER /0.0., GRP /0.0/

CBR IS BIT ERROR RATE DEGRADATION DUE TO HARDWARE
CBR IS ARRAY OF DATA RATES

DATA IBER/ .25.50., 1.0, 2.0, 4.0, 8.0, 16, 32, 64, 128, 256, 512, 768 .
1024. /

DATA BER/8.4.4.6.5.5.5.8.2.4.2.4.2.5.4.3.3.10.4.0.3.9. /

CONT. ON PG 2

PG 1 OF 54

10-143
INITIALIZATION OF P1IC AND INCOME

IF (IC=0) THEN

IF (IC=10) THEN

IF (INCOME=4 OR INCOME=5) THEN

CONT. ON PG 4

IF (INCOME=1) THEN

IF (INCOME=1) THEN

BITRAI! = .001 * BITRAIII

MEPRODUCnBrITy OF THP
IF (ITER.EQ.0 .AND. PIC(I).GT.0)

T

ICC = 1

DO 10 I = ICC + 9

10 KCHOS(I) = 0

DO 20 I = 1 + 9

KPIC(I) = PIC(I)

20

IF (ITER.EQ.0 .AND. PIC(I).EQ.0)

T

ENTER AF

KPIC(I) = RESET(I)

F1 = 0.

IF (INCONF.EQ.2 .OR. NCONF.EQ.4)

F

T

F1 = 1.

CONT. ON PG 5
IF (IC .NE. 0)

T
GO TO 700

LIMPIC(1) = IDB(1)
LIMPIC(2) = IDB(2)
LIMPIC(3) = IDB(3)
LIMPIC(4) = IDB(4)
LIMPIC(5) = IDB(5)
LIMPIC(6) = IDB(6)
LIMPIC(7) = IDB(7)
LIMPIC(8) = IDB(8)
LIMPIC(9) = IDB(9)

CALL BESS (BETA, BESSJ, I)
CONTINUE

IF (MCONF .GE. 4 OR BITRAT(2) .EQ. 0)

T
GO TO 40

BITRAT(1) = (BITRAT(1) + BITRAT(2)) / 1.3

CONTINUE

RATE1 = BETA(1)
RATE2 = 0

CONT. ON PG 7
DO 50 I=1,13

IF (BITRAT(1) .GT. IBER(1)) THEN
  RATE1 = IBER(1+1)
ELSE IF (BITRAT(2) .GT. IBER(1)) THEN
  RATE2 = IBER(1+1)
  CONTINUE
END IF

BITRAT(1) = RATE1
BITRAT(2) = RATE2

IF (INCONF.EQ.1) THEN
  GO TO 90
ELSE
  IERR = 1
END IF

IF (LINK.EQ.0) THEN
  8
ELSE
  8
END IF

CONT. ON PG 8
DO TO 890
GO TO 770

IF (DATA1(B2,J1).EQ.1.7)

GO TO 660

J1 = J1 + 1

IF (J1.GT.(DBL(1)))

GO TO 760

GO TO 60

C END OF BBRV SELECTION

B2 90
IC = 2
KXMT = 1
GO TO 110

B3 100
IC = 3

CONT. ON PG 11

PG 106F 54
Go to 230

C HELIX (F3)

IF (DATAFILE1 J2 EQ 31)

GO TO 299

J2 = J2 + 1

IF (J2 GT IDB(2))

GO TO 760

GO TO 220

230

IERR = 30

CONT. ON PG 17

PG 105F 54

10-158
IF (ALT.GT. 20000.)

T

GO TO 770

F

IERR = 0

C PARABOLA (B HIGH_GAIN)

240

IF (DATA(A1,J2).EQ. 1.)

T

GO TO 299

F

J2 = J2 + 1

IF (J2.GT.IDB(2))

T

GO TO 760

F

GO TO 240

C STEERABLE PARABOLA OPTION WILL BE INCLUDED NEXT SEMESTER

CONT. ON PG. 18
IF (SLANT.EQ.-1.E+10)

SLANT=SQRT(APOGEE*(APOGEE+6880))

SLOSS=37.8+20*ALOG10(FREQ(KMTR)*SLANT)

GO TO 320

IF (GTOT.NE.-1.E+10)

IF (CR.NE.-1.E+10.AND.T.NE.-1.E+10)

GO TO 310

IF (NF.NE.-1.E+10.AND.CR.NE.-1.E+10)

GO TO 300

CONT. ON PG 21

PG 20DF 54

10-162
CONT. ON PG 23
IF (BITRAT (XMTX) .EQ. IBER(I))

T

HOLOSS = IBER(I+3)

DO 360 CONTINUE

GO TO 380

360

IERR=10

IF (LINK .EQ. 0)

F

GO TO 370

370

IERR=0

IK=2

IF (DATA(B2,J1) .EQ. 1.024)

F

IK=1

DO 370 I=1,14

CONT. ON PG 25

PG 24
IF (BITRAT(I) .EQ. (BER(I)))

IF (BITRAT(I) .EQ. (BER(I)))

MDLOSS = BER(I*K)

CONTINUE

C BANDWIDTH IN DB

IF (BWIDTH(KXMT) .EQ. -1.E+10)

BWIDTH(KXMT) = BITRAT(KXMT) + 1000

B = 10 * ALOG10 (BWIDTH(KXMT))

C CALCULATION OF ERP

ERP = SIGNAL(KXMT) * SLOSS + B - GTOL + LMARG(KXMT) + TLOSS(KXMT) + POLOS + A

NTLOS + MODLOS + MDLOSS - 228.6

PW(KXMT) = 10. ** ((ERP - GT(KXMT))/10.)

PRINT 385, KXMT, PW(KXMT)

385

FORMAT (4H PW(I,4H) = *E11.4)

C TRANSMITTER SELECTION

CONT. ON PG 26
F
90
0
O-o4
.1J3
108131)
760
GO TO 760
26
390
GO TO 390
480
GO TO 490
J3=J3+1
GO TO 760
53
37
GO TO 390
26
490
IF (LINK.EQ.1) F
CONT. ON PG 33
PG 326F 54
C MODULATION PHASE=1 * FREQUENCY=2 * AMPLITUDE=3
IF (J7.GT.108(2))

IF (DATA1[IJ7].EQ.0)

IF (DATA1[IJ7].EQ.51)

J7=J7+1

CONT. ON PG 37
IF (DATAB1(0D1+J8).EQ.1)  
  
J8 = J8 + 1  
IF (J8.GT.I08(6))  
  
C DIPLEXER POWER  
JL = KPI(4)  
IF (DATAB1(D2+J6).GE.DATAB1(I9+JL))  
  
CONT. ON PG 44
KONT = 0
IN = KPEC(I)

IF (IN .NE. 0), F
T
KCHOSE(I) = DATAB(I .* IN)
WRITE (6,2000) IN, KCHOSE(I) .* IC

FORMAT (4H 690.3(I0)
S 16 16 40
700 CONTINUE

IF (IC .EQ. 9 .AND. ITER .NE. 0), F
T
GO TO 740

IF (IC .EQ. 9 .AND. IPIC(4) .EQ. 0), F
T
GO TO 740
DO 757 I=1,11
DO 756 J=1,1087

IF (DATAB(I,J),NE., ICHOOSE(I))
   T
   GO TO 756

IF (I .GE. ICWMT)
   T
   CWMVT=CWMVT+DATAB(23,J)*NCHOOSE(I)

MT = MT + DATAB(23,J)*NCHOOSE(I)

VOL = VOL + DATAB(24,J)*NCHOOSE(I)
PL = PL + DATAB(16,J)*NCHOOSE(I)
PLMIN = PLMIN + DATAB(18,J)*NCHOOSE(I)

GO TO 757

CONTINUE

CONTINUE

CONT. ON PG 53
4000  
FORMAT('4H 770*11(10)')

ICHOSE(1)=-1

RETURN

END
SUBROUTINE BESS (X, BESJ, NMAX)
DIMENSION BESJ(N), TJ(200)
EULER = 0.577215664901533
PI = 2.0*3.141592653589793
NU2 = 20

IF (1.0 - X) 10, 10, 20

10

HATHN = (1.05)*X + 25.
GO TO 30

20

HATHN = 35./LOG(X))

30

NU = HATHN

TJ(NU+2) = 0.0
TJ(NU+1) = 0.000001

DO 40 J = 1, NU

K = NU - 1 - J
FK = K + K

TJ(K) = FK*TJ(K+1)/X - TJ(K+2)

SUM = 0.0

DO 50 J = 3, NU + 2

SUM = SUM + TJ(J)

CONT. ON PG 2
SUM = SUM + SUM
TK = 1 / (TJ1 + SUM)
N = ABS(NMAX) + 1

DO 60 J = 1, N

BESJ(J) = TK * TJ1(J)

RETURN

END
INTEGER FUNCTION RESET(K)
COMMON /OBCOM/DATA(155,100),ID(30)

IF (K.EQ.1) F
  IF (K.EQ.2 .OR. K.EQ.3 .OR. K.EQ.6) T
    RESET = IDB(1) + 1
  ELSE F
    IF (K.EQ.4 .OR. K.EQ.5) T
      RESET = IDB(2) + 1
    ELSE F
      IF (K.EQ.7 .OR. K.EQ.8 .OR. K.EQ.9) T
        RESET = IDB(K-4) + 1
      ELSE F
        RETURN

CONT. ON PG 2
SUBROUTINE SKEDONEIP, NCONF)
COMMON /USERS/SKOME(7,3)

COMMON /CHOOSE/ COST(5:60), OPI(11:60), ICHOOSE(60), NCHOOSE(60), REL(6:60), OBSKED(17:60), THM(4:60)

COMMON /PRICOM/ ACCRCY, AM, AN, BF, BS, COPIT(7:2), CISTAR, CTOF, ODTE, OGE, ORIN, EOBSTR, FEINV, FEOPS, FEER, OSE, IREL, ITTRUNC, MMDLDD, NAME(19:60), OPS, PAYINV, PAYOL, PAYR, PE, PMR, PNR, POWER(6), PU, PWR(60), QCP, QCR, ROLD(60), SABMT, SATADP, SATINV, SATR, SEIP, SEIR, TSAVE(6).

DIMENSION CONF(22:5), TSUB(6), ICI(5), NEDQIP(5), NCONF(6)
DATA ICI/0.5,6,10,15/

DATA CONF/1.5,1,2,15,31,2,121,2, 6.9,5,12,9,3,5,6,3.8,4.8,2,6,4.2, 22,7,22,2,8,7,5,0.001,3,0.002,13,0.0007,0.002/
C=0
NUM=0

DO 2 J=START,END

TCD=OBSKED(3+J)+OBSKED(4+J)

IF (TCD .GT. TSUB(1))

TSUB(1)=TCD

TCD=OBSKED(5+J)+OBSKED(6+J)

IF (TCD .GT. TSUB(2))

TSUB(2)=TCD

C=C+(1.335*OBSKED(1+J)+1.41*OBSKED(2+J))/1000.

NUM=NUM+NCHOSE(J)

XNUM=NUM
REDUH=XNUM/NEQUIP(IS)
IC=(CIT(IS)+NCONF(IS))
R=REDUH***.125

TSUB(3)=CONF(IC+2)+CONF(IC+5)*R*CONF(IC+1)*C***.6667
TSUB(4)=CONF(IC+3)+CONF(IC+4)*R*C

TSDQ=TSUB(2)

CONT. ON PG 4
A1

IF (TSAVE(I) .LE. TSAVE(J))

T

GO TO 4

DO 3 J = 1, 6

TSAVE(J) = TSAVE(J)

ISSAVE = IS

CONTINUE

CONT. ON PG 6

CONT. ON PG 6
R2
IF (JJ .EQ. 0)
T
GO TO 20
T
TSUB(1) = 0.
T
TSUB(2) = 0.
C = 0.
GO TO J = 1 + JJ
IF (TCO .GT. TSUB(1))
T
TSUB(1) = TCO
T
TCO = SKOME(5 + J) + SKOME(6 + J)
IF (TCQ .GT. TSUB(2))
T
TSUB(2) = TCQ
CONT. ON PG 7
PG 6 OF 8
10-206
SUBROUTINE SANOCIIPIC IERR ITER NCONF ICHOOSE NCHOOSE

ICHOOSE(I) IS SELECTED EQUIP AS FOUR DIGIT EQUIP = MANF.
NCONF IS CONFIGURATION NUMBER. ITER IS NUMBER OF THIS ITERATION.
IERR IS A MULTIPLE MESSAGE ERROR FLAG. IPIC IS THE LAST SET OF SUBSCRIPTS CHOSEN.
COMMON USER LISTS USER INPUT PARAMETERS.
COMMON BTWN LISTS NECESSARY COMMUNICATION BETWEEN SUBROUTINES.
COMMON CORTA HAS LAST SUBSCRIPT FOR EACH PIECE OF EQUIP AND THE NECESSARY PIECE OF THE DATA BASE.

DIMENSION ICHOOSE(9), IPIC(3), ES(6), C(5), DMA(2), G(3), F(9), NCHOOSE(9)
DIMENSION NCONF(6)

COMMON /USER1/ ALPHA, AX, AY, AZ, DPHI, EA, EANT, EPI, K, MA, MV,
OMEG, POOR, POORX, POORY, POOTZ, POOTZT, POTO, POTR, POTOZ, POTR, PHI, PHIY, PHIR, PHIZ, TACEL,
THMIX, THOLD, TL, TP, TPRM, TSM, TSMALL,
XN, XNN, XNWW, XN, YN,
ZN

COMMON /USER1/ APOGEE, COMRAT, DIAMAX, EEDWT(9), EPME,
EMIXL, EOM, EOMLY, EQMLZ, EOMZW, EOMZL,
IDBUG, ISAT, MB2SH, OPT, ORBINC, PERIG,
MICRO, RELME, SPEC(6), SPEC1, T, XCGSAi

COMMON /BTWN/ ACSSN, ACSMW, ALT, ARE, BATCAP,
BITR, CLIDE, CONYWT, D, DT,
DX, DZ, EQL, EQBL, EQBSID,
FC, FF, HBAWT, HPM, HTP,
HTPT, HBPRB, HTPHR, IRTLOC,
LMDO, NC, OMES, PASTR, P,
PL, PLIN, POCW, RADA, RADD,
RAT, RJ, SABLG, SATLG, SATN,
SATW, SATCG, SATTJ, SATZG, SAIX,
SAIY, SAIZ, SIDE, SYSLB, THCMWT,
THRUST, TL, TMKWT, TPTRM, V8,
VCHP, VOL, WAPE, W8, WBT,
WT, XJ, XNZERO, YJ, ZJ

COMMON /OBCOM/DATA(155, 100), IDB(30)
DATA XMD, YM0, ZMD.01, XMDZ, YMD2, ZMD2/30.0003...003.
ACSSN=2.

IF (NCONF(1) .EQ. 1) CONT. ON PG 2
PG 1 OF 50
\[ YCG = 1./12. \times SATYCG \]
\[ ZCG = 1./12. \times SATZCG \]
\[ A = 0.5 \times 12. \times \text{RE} \times \text{APOGEE} \times \text{PERIGEE} \]
\[ EQBLG = EQBLD / 12. \]

\[ IF (ICONF(6) \ EQ. 2) \]
\[ T \]
\[ EQBSID = EQBSID/12. \]
\[ SAIXL = SAIXL / 12. \]
\[ SAIYL = SAIYL / 12. \]
\[ SAIZL = SAIZL / 12. \]
\[ ICONF = ICONF(6) \]

\[ GO TO (111 \times 12 \times 13) \times ICONF \]

11
\[ SIDE = 0 \]
\[ XLNTH = EQBLD \]
\[ GO TO 14 \]

12
\[ SIDE = EQBSID \]
\[ XLNTH = EQBLD \]
\[ GO TO 14 \]

13
\[ SIDE = 0 \]

CONT. ON PG 4
17
INOSE = 1

TEMPIN = XJ
XJ = YJ
YJ = TEMPIN

GO TO 20

18
INOSE = 2

TEMPIN = XJ
XJ = ZJ
ZJ = TEMPIN

GO TO 20

19
INOSE = 3

C NO TRANSPOSITION REQUIRED

20
CONTINUE

TX = XK - (ZJ-YJ) = PHIRX / 57.3
TY = XK - (ZJ-XJ) = PHIRY / 57.3
TZ = 0.
CONF = NCONF(S)

GO TO (21,26,21,26,21,26),ICNF
IF (PERIGE .LT. 65.)
  T
  GO TO 28

IF (PERIGE .GT. 500.)
  T
  GO TO 29

TSX = 0.
TSY = 0.
TSZ = 0.
TAUXS = 0.0
TAUY = 0.0
TAUZ = 0.0
GO TO 40

28
IEAR = -1
ICHOOSE(I) = -1
RETURN

29
TAX = 0.
TAY = 0.
TAZ = 0.
TAUXA = 0.0
TAUYA = 0.0
TAUZA = 0.0
AP = (SIDE*XLNTH)*EQMIXL*EQMIYL/144. + (EQM2XL*EQM2YL/144.)
XCP = EQMIXL*EQM2XL/24.
YCP = 0.0

SIDE12 = SIDE12
ZCP = AMAX(SIDE12, EQM12L, EQM22L) / 124.
XLX = XCG - XCP
XLY = YCG - YCP
XLZ = ZCO - ZCP
R = RE/A
S = 1.02 * ASIN(R)
TS = 2.0 * (3.14159 - S) / OMEG0

GO TO 130131321INSEE

30
TEMPX = XLX

XLX = XLY
XLY = -TEMPX
TSX = HD + AP * (1.0 + EPSLN) * XLY
TSY = HO + AP * (1.0 + EPSLN) * (-XLZ)
TSZ = 0.

GO TO 38

31
TEMPX = XLX

XLX = -XLZ
XLY = TEMPX
TSX = 0.
TSY = HD + AP * (1.0 + EPSLN) * XLZ
TSZ = HO + AP * (1.0 + EPSLN) * (-XLY)

GO TO 38

32
TSX = HD + (AP + 1.0 + EPSLN) * XLY + SREFSP * (1.0 + EPSLN) * DELYSP

CONT. ON PG 9
\[ T \Sigma Y = \text{HO} \times (\text{AP} \times (1 + \text{EPSLN}) \times XLX \times \text{SREFSP} \times (1 + \text{EPSA}) \times \text{DELXSP}) \]
\[ T \Sigma Z = 0. \]

IF \[ \text{PERIGE} \geq 300 \]
\[ \text{GO TO 42} \]
\[ \text{IF} \] \[ \text{PERIGE} \geq 100 \]
\[ \text{GO TO 41} \]

\[ \text{COMPUTE SOLAR TORQUES} \]

\[ \text{SININC} = \sin(\text{ORBITINC}) \]
\[ \text{BETA} = \arctan(\text{WEDNO} \times \text{SININC}) + \text{DB} \]
\[ \text{ALFA} = \text{DB} \]
\[ \text{ALFBAR} = \arccos(\cos(\text{BETA})/(\sqrt{1 - \cos(\text{BETA})^2}) = \arctan(\text{ALFA}^2)) \]
\[ \text{CA} = 2.012 \times \cos(\text{ALFBAR})^2 \]
\[ \text{CY} = -\sin(2 \times \text{ALFA}) \]
\[ \text{CN} = \sin(2 \times \text{ALFA}) \]

CONT. ON PG 10
\[ \text{RHO} = 1.5E-9 \times .02 \times \frac{1}{(\text{PERIGE-65.})/35.)} \]

GO TO 43

\[ \text{RHO} = 3.3E-11 \times 4.333E-4 \times \frac{1}{(\text{PERIGE-100.})/200.)} \]

GO TO 43

\[ \text{RHO} = 1.3E-14 \times 1.538E-2 \times \frac{1}{(\text{PERIGE-300.})/200.)} \]

GO TO 43

\[ \Omega = 3.6E10 \times \frac{\text{RHO}/A}{144} \]

GO TO 144, 146, 148, 100

\[ \text{SREF} = \text{SIDE} \times \text{LNTH} + \text{EQMIXL} \times \text{EQMIXL} / 144. + \text{EQM2XL} \times \text{EQM2XL} / 144. \]

\[ \text{XREF} = \frac{(\text{EQM1XL} - \text{EQM2XL})}{24}. \]

\[ \text{SIDE2} = \text{SIDE} = 12. \]

\[ \text{YREF} = \frac{\text{AXAI} + \text{SIDE2} \times \text{EQM1YL} \times \text{EQM2YL}}{24}. \]

\[ \text{ZREF} = 0. \]

\[ \text{DELTX} = \text{XCG} - \text{XREF} \]

\[ \text{DELTY} = \text{YCG} - \text{YREF} \]

\[ \text{DELTZ} = \text{ZCG} - \text{ZREF} \]

\[ \text{TEMPX} = \text{DELTX} \]

\[ \text{DELTX} = \text{DELTY} - \text{TEMPX} \]

GO TO 50

\[ \text{SREF} = \text{SIDE} \times \text{LNTH} + \text{EQMIXL} \times \text{EQMIXL} / 144. + \text{EQM2XL} \times \text{EQM2YL} / 144. \]

\[ \text{XREF} = \frac{(\text{EQM1XL} - \text{EQM2XL})}{24}. \]

CONT. ON PG 11

PG 106F 50
XMD = TAX + TOX + TSX
YMD = TAY + TGY + TSY
ZMD = TAZ + TOZ + TSZ
PRINT 9002, TAX, TAY, TAZ, TAUXA, TAUYA, TAUZA

9002

FORMAT IX. SHTAX = E11.4, SHTAY = E11.4, SHTAZ = E11.4, 7HTAUXA = E11.4,
7HTAUYA = E11.4, 7HTAUZA = E11.4)
PRINT 9003, TSX, TSY, TSZ, TAUXS, TAUSY, TAUZS

9003

FORMAT IX. SHTSX = E11.4, SHTSY = E11.4, SHTSZ = E11.4, 7HTAUXS = E11.4,
7HTAUYS = E11.4, 7HTAUZS = E11.4)
PRINT 9004, TGY, TGY, TGZ

9004

FORMAT IX. SHTGY = E11.4, SHTGZ = E11.4, SHTOZ = E11.4)

IF (NCONF1) .EQ. 2)
F
T
GO TO 200

IF (NCONF1) .EQ. 3)
F
T
GO TO 300

CONT. ON PG 13
PG 120
C LAST ONE CHECKED AND NONE FOUND

C ACCEPTABLE DEVICE SELECTED

C YAW SPIN CONFIG

C INITIALIZE SKIPPING SOME IF ITERATING

CONT. ON PG 20
P~dHIF

f

,dS40.wTDNI/l DX.F

J

IC 5E6U EARTH SENSOR

WITH

PNIX': P1 ZIA X

I

so 2o04..

CONT. 04 PO 23

P0 2MF

10-230

C  SELECT EARTH SENSOR WITH PHIX<=PHIX

IF (IPC(1) .GT. 0)

T

GO TO 204

23

102

JI=IDO(I)+1

GO TO 206

23

103

CONT. ON PG 23

PG 229F 50

10-230
IF ( J2 .GE. J2E )

T

GO TO 214

IPIC(2)=0

GO TO 211

C ACCEPTABLE COMBINATION FOUND

08

210

IPIC(2)=J2

ICHOSE(5)=DATA(1:J2)

MT=MT*DATA(23,J2)*NCHOSE(5)*DATA(23,J1)*NCHOSE(4)

VOL=VOL*DATA(24,J2)*NCHOSE(5)*DATA(24,J1)*NCHOSE(4)

PL=PL*DATA(18,J2)*NCHOSE(5)*DATA(18,J1)*NCHOSE(4)

RETURN
```plaintext
C CALCULATE F VALUES

F(1) = DPHI/DT / 57.3 * 0.04 * DFE/DY
F(2) = DPHI/DT / 57.3 * 0.04 * DFE/DZ
F(3) = 2.0 * 0.04 * DPHI / 57.3 * DFE/DX
F(4) = XMD/DX
F(5) = YMD/DY
F(6) = ZMD/DZ
FMIN = MAX(F(1), F(2), F(3), F(4), F(5), F(6))
F(7) = DOTR*MC/J / D1/DX

C IERR IX % MAX RATE ERROR TOO SMALL

IF (IFMAX .LT. 2.0 * FMIN)
  F
  T
  IERR = IERR + 10
  FF = 2.0 * FMIN
  IF (IFMAX .LT. FF)
    F
    T
    CONT. ON PG 29
```

CONT. ON PG 29
E1

FF:FR1W

E2

DBX:.4 PHI RX

DBY:.4 PHI RY

DBZ:.4 PHI RZ

IF (DBX .LT. .05)

DBX:.05

IF (DBY .LT. .05)

DBY:.05

IF (DBZ .LT. .05)

DBZ:.05

DBDX=1=DBX

DBDY=1=DBY

DBDZ=1=DBZ

R1=2*POOTRX

CONT. ON PG 30
R2 = 2 * P00TRY
R3 = 2 * P00TRZ
R = MAX(R1, R2, R3)

SELECT 3 GYRS

IF (IPI(1) .GT. 0)

GO TO 304

J1 = I0B(16) + 1

GO TO 306

304
J1 = IPI(1)

IF (IPI(2) .GE. I0B(18) .AND. ITER .EQ. 0)

T

GO TO 308

306
J1 = I0B(17)

GTEST = ORIAB(6 + J1)

IF (GTEST .GT. R)

T

CONT. ON PG 31

PG 305 50

10-238
SELECT EARTH SENSOR

310
J2 := PIC(2)

[F: J2 GE (DB118) .AND. ITER .EQ. 0]

1PIC(2) = 0

[F: (PIC(2) .EQ. 0)]

J2 := (DB117) + 1

[F: (ITER .EQ. 0) .AND. (PIC(2) .NE. 0)]

J2 := J2 + 1

J2 := (DB118)

PPH(N := DB816, J2) := DB8(6, J2) / DB8(6, 11) / DB8(13, J2) :=
ATAN(DB8(13, J2) / DB8(6, 11))

CONT. ON PG 33
IF \( \frac{\text{TREQ}}{\text{MREQ}} \geq 0.02 \) AND \( \text{PDOTS} \lt \text{PDOTA} \)

GO TO 403

403

IF \( \frac{\text{TREQ}}{\text{MREQ}} \lt 1.0 \)

\( \text{IERR} = 100 \)

\( \text{IERR} \% \text{ IXX} \) MEANS DOUBLE GIMBAL CMYS ACCEPTABLE.

IF \( \text{TACCEL} \lt 20.0 \)

\( \text{IERR} = \text{IERR} + 1000 \)

\( \text{CONT. ON PG 36} \)

\( \text{PG 35DF 50} \)

10-243
GO TO 440

C RETURNS HERE TO TEST NEW CMD

GO TO 414

IF ( Sooth .LE. Datab(7, Ji) .AND. (Max .LE. Datab(8, Ji)))

GO TO 417

CONT. ON PG 39
IF (JI .LE. IOB120)

T

GO TO 410

ICHOOSE(4) = -1

ICHOOSE(5) = 0

ICHOOSE(6) = 0

RETURN

C

CMG SELECTED

ICHOOSE(4) = DATAB(1, J1)

CONT. ON PG 40

CONT. ON PG 40
ACCEPTABLE COMBINATION FOUND

\[ IP(1) = J1 \]
\[ IP(2) = J2 \]
\[ IP(3) = J3 \]

WT = WT + XNNN + CHOOSE(5) * DATA(23, J2) + CHOOSE(6) * DATA(23, J3)

VOL = VOL + XNNN + CHOOSE(5) * DATA(24, J2) + CHOOSE(6) * DATA(24, J3)

PL = PL + PNNN + CHOOSE(5) * DATA(16, J2) + CHOOSE(6) * DATA(16, J3)

PLMIN = PLMIN + DATA(18, J1) + XNNN + CHOOSE(5) * DATA(18, J2) + CHOOSE(6) * DATA(18, J3)

NOW THRUST AND IMPULSE

\[ F(1) = 10PHI[01/57.34] \cdot FE/DO \]
\[ F(2) = 10PHI[02/57.34] \cdot FE/DZ \]
\[ F(3) = 2.592E6 \cdot PHI[03] \cdot DX \]

\[ FF = MAX(F(1), F(2), F(3)) \]

\[ Ti = (F(1) + F(2) + F(3)) + TSMALL \]
\[ Xx = IX / DX \]
\[ Yj = IY / DY \]
\[ Zj = IZ / DZ \]

\[ TIA = TAUX \cdot 2.592E6 \cdot TGM / DX \]
\[ TAY = TAUY \cdot 2.592E6 \cdot TGY / DY \]
\[ TAZ = TAUZ \cdot 2.592E6 \cdot TGZ / DZ \]

FC = FF * DX / (XJ + 4PHI) * 14.1E-9

CLIFE = 37.66 + TPRIM + FF * DX + DI / (XJ + 4PHI)

RETURN

CONFIGURATION & SELECT FIXED EQUIPMENT

\[ IP(1) = J1 \]
\[ IP(2) = J2 \]

\[ 10815 + 1 \]

\[ 2 * (IP(2)) + 1 \]
\[ CHOOSE(1) = DATA(11, IP(1)) \]
\[ CHOOSE(2) = DATA(11, IP(2)) \]

CONT. ON PG 45

PG 40F 50
EMS = (EAM - EAM - PH(RZ)) * 0.5

IF (EMS < 0)
  T
  EMS = 0

(EHS = SORT(EHS))

IF (IPIC(1) > 0)
  T
  J1 = IPIC(1)

IF (IPIC(1) = 0)
  T
  J1 = 100(17) + 1

IF (IPIC(2) = 100(13) AND IER = 0)
  F
  GO TO 510

CONT. ON PG 47
ICHECK = 2 MEANS THAT SOLAR ARRAYS ARE PADDLES AND MOUNTED AT CENTER OF VEHICLE. ICHECK = 1 MEANS OTHERWISE.

SOLAR ARRAYS ARE PADDLES.
APPLIED LOAD (ONLY BENDING MOMENT)
COMPUTE WEIGHT PER PADDLE (ASSUME 2 PADDLES)

WE = .5*SORRTT

XMA = 1.25*SABOLG*WE*SORTIC*CA*CE + CE*CE

R = (SABOLG*4*XMA/(PI*5*WE))**.1428

TW = 2.*SORTI*XMA/(PI*E*R))

CONT. ON PG 3
C CHECK FOR APPLICABILITY OF EULER COLUMN STABILITY

\[ \text{FACI} = \frac{\pi^2 \omega_2 \omega_n^2}{(18.0 \times \text{SABOLG} \times \omega_n^2)} \approx 0.333 \]

\[ \text{IF (SIGY - FACI GE 0.)} \]

T

GO TO 1

C EULER COLUMN STABILITY NOT APPLICABLE

\[ \text{TH} = \frac{16 \times \text{SIGY} \times \omega_n^2}{(18.0 \times \text{SABOLG} \times \omega_n^2)} \approx 0.333 \]

\[ \text{R} = \frac{\text{TH} \times \omega_n^2}{(4.0 \times \text{SIGY})} \]

C VOLUME OF SOLAR BOOM

\[ \text{VARY} = \text{R} - \text{TH} \times \text{SABOLG} \]

C SIZING OF EQUIPMENT DAY STRUCTURE

\[ A3 \]

CONTINUE

\[ P = \omega_n \times \text{SATMT} \]

C BENDING MOMENT

\[ \text{XM} = \frac{75 \times \omega_n \times \text{EGBOLG} \times \text{SATMT}}{13} \]

\[ \text{IF (CHECK. EQ. 1)} \]

CONT. ON PG 4

PG 3 OF 8
n4

CE0dLOESATHT-
IC EOU VALENT AXIAL LOAD

F

IF(INCONF10).NE.1)

GOTO 5

C EQUIVALENT AXIAL LOAD

RR = .5*SDAMD

HG = P/12.*PI*RR) + XR/(PI*RR*RR)

C SIZING OF EQUIVALENT MONOCOQUE CYLINDER

TM = .672*(XNXU*XR*XARXL/E)*.3333
FAC2 = XL*2*SORT(*XNXU)/(*RR*TM)

IF(FAC2.LE.31)

T

GO TO 3

CONT.

C EQUIVALENT THICKNESS OF STIFFENED CYLINDER

TBAR = .267*TM

CONT. ON PG 5

PG 4 OF 8

10-262
\begin{align*}
\text{YEO} &= 2.0 \times \text{VED} + 0.219 \times \text{TB} + \text{RR} \\
\text{TOTAL STRUCTURE WEIGHT} &= 4 \\
\text{RETURN} \\
\text{CONTINUE} \\
\text{IF (INCONFIG).NE.2) } &= F \\
\text{RETURN} \\
\text{IF (XKL/W.LE.5) } &= F \\
\text{MM} &= 1.068 \times \text{IXNU} \times \text{XN} \times \text{KL} / \text{E} = .3333 \\
\text{CONT. ON PG } &7 \quad \text{PG. 6 OF 8}
\end{align*}
IBEAB: Z6?PI
SKIN STRINGER ASSEMBLY
C

T = .44 * TBAR
T8 = 1.9 * T
BS = .64 * TS * SQRT(E * TBAR / (XXNU * XM))
B = 1.49 * BS
N = 1. + W/B
N = 4 * N
AN = N
B = W/AN

C

ALPHA = .745/XXNU = .75

C SIZING OF FRAMES

A = E * ALPHA = .2 * TBAR = .2 / XM
RHOF = .40S = (W = .2 / A) * (XXNU = ALPha = .2 / (E * A)) = .25
AF = .041 * XN = N / (E * RHOF = .2 / A)
BF = 3.464 * RHOF
IF = AF / BF
M = 1. + XL / A
AM = M
A = XL / AM

C SIZING OF ENO COVERS

TC = .303 = SORT(A = SATWT / SIGY)
TA = TC
XLD = .5 * W

C VOLUME OF EQUIPMENT BAY STRUCTURE

VEQ = 2. * (1 + TS * BS / B) + (TF * BF / A))

CONT. ON PG 8

PG 7 OF 8

10-265

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR
C MID-SECTION BULKHEAD IS REQUIRED

\[ \text{WL} = 0.455 \times \text{C} \times \text{SATHT} / \text{XLD}^{0.52} \]
\[ \text{TB} = 0.859 \times \text{XLD} \times \text{SQRT} \left( \text{WL} / \text{S1G1Y} \right) \]
\[ \text{VEO} = 2.0 \times \text{VEO} + 0.219 \times \text{WL} \times \text{TB} \]

6 CONTINUE

C TOTAL STRUCTURAL WEIGHT

\[ \text{VEO} = 2.0 \times \text{EQLDG} \times \text{VEO} + 2.0 \times \text{TC} \times \text{WL} \]
\[ \text{STRM} = 0.1 \times \text{VEO} + 0.1 \times \text{VAR} \]

RETURN

END
Determine equipment bay equipment weight and volume

\[ \text{EQWT} = 1.025 \times \text{STINWT} \]

\[ \text{EQVOL} = 1.025 \times \text{SATVOL} \]

\[ \text{TNCWWT} = 0.025 \times \text{STINWT} \]

The thermal control subsystem is accounted for by the 1.025 factor.
Note that volumes are in ft\(^3\).
Account for packing factor:

\[ \text{EQBVL} = 1728 \times \text{EQVOL} \times \text{EOPF} \]

Determine equipment bay length

CONT. ON PG 3

PG 2 OF 29
1. A1
   SATDAM = DIAMAX
   EOBLD = 2. * EOBVOL / SATDAM**2
   GO TO 4

2. A2
   SATDAM = (EOBVOL / .524)**.333
   EOBLD = SATDAM
   IF (SATDAM <= DIAMAX)
   T
   GO TO 4
   F

3. A3
   ICHOOSE = -1
   C
   THAT IS, THIS IS NOT AN ACCEPTABLE MACRO CONFIGURATION
   RETURN

C
Determine Solar Array Dimensions

1ERR=0
SAAREA = 144 * SAAREA
GO TO (11+14+17)*ISHAPE
GO TO (12 + 20) IEQTYP

IF (SPIN.EQ.0)

GO TO 13

IF (SURF .GE. SAREA)

GO TO 20

(SATDAM = SQRT(SAREA/1.88496))

EQBLC = 0.6 * SATDAM
EQBVL = 0.785 * SATDAM**2 = EQBLC

IF (SATDAM .LE. DIAMAX)
**Diagram Description:**

1. **GO TO 10:**
   - IF (ISPIN.EQ.1)
     - T
     - GO TO 20
2. **SURF = SATDAM * EOBLC / 1.4142**
3. **IF (SURF .GE. SAAREA)**
   - T
   - GO TO 20
4. **SATDAM = SQRT(SAAREA) * 1.4142**
5. **EOBLC = SATDAM**
6. **EOBYOL = 0.500 * SATDAM**
7. **EOBLC = EOBYOL**
8. **IF (SATDAM .LE. DIAMAX)**
   - T
   - GO TO 20

**Continuation on Page 8**
A8
SATDAM = DIAMAX
EQLBC = 1.4142 * SAAREA / SATDAM
EQBYOL = 0.500 * SATDAM\^2 = EQLBC
GO TO 20

A9
GO TO (10.20).IEQTYPE

10
19

18
IF (ISPIN.EQ.0)
F
T
GO TO 19

IF (SURF .GE. SAAREA)
F
T
GO TO 20

SATDAM = SQRT(SAAREA/3.14159)
EQBYOL = 0.524 * SATDAM\^3
CONT. ON PG 9

PG 8 OF 29

10-274
IF (SATDAM .LE. DIAMAX)  

T

GO TO 20  

ICHOOSE = -1

C THAT IS WE CANNOT LENGTHEN A SPHERE

RETURN

SURF = 1.5708 * SATDAM**2

IF (SURF .GE. SAAREA)

T

GO TO 20

SATDAM = SQRT(2.*SAAREA/3.14159)

EOBVOL = 0.524 * SATDAM**3

IF (SATDAM .LE. DIAMAX)  

T

GO TO 20

CONT. ON PG 10  

PG 9 OF 29

10-275
ICHOOSE = -1

C THAT IS WE CANNOT LENGTHEN A SPHERE

RETURN

CONTINUE

C DETERMINE SATELLITE LENGTH

SATLG = EQBLG + EQMIXL + EQM2XL

IF (IEQTYPE .EQ. 2)

T

GO TO 54

R7

IF (ISHAPE .EQ. 2) S1, S2, S3

S1

SA3XL = SAREA / SATOM

IF (SA3XL .LE. EQBLG)

T

GO TO 55

R4

CONT. ON PG 11

PG 100F 29

10-276
C DETERMINE EQUIPMENT BAY STRUCTURAL WEIGHT

If (IEQTYP.EQ.2)

EOBSTR = .59*EOBSTR

EOBSTR = EOBSTR - 0.10*EQWT

C DETERMINE STRUCTURAL THERMAL PROTECTION WEIGHT

STTPS = 0.025 * EOBSTR
EOBSTR = EOBSTR + STTPS
EOBWT = EQWT + EOBSTR

C DETERMINE MISSION EQUIPMENT BAY TOTAL VOLUME

If (MB12SH.EQ.2)

GO TO 150

EOMLYL = .785*EQMLYL + 2*EOMLXL

CONT. ON PG 13

PG 126 29
IF (SPIN.EQ.1)
   T
   GO TO 23.
   
   SABOLG = 24.
   SABOOM = 15.2
   
   IF (ISBOFG.EQ.0)
       F
       
       T
       GO TO 23
       
       SADRY = .166*(SA1MT + SA2MT)
       
       CONTINUE
       
       SABMWT = SABOOM + SADRY
       
       CALCULATE HARNESS AND STRUCTURAL TPS WEIGHT
       FIRST NEED MISSION EQUIPMENT WEIGHT AND EXTERNAL EQUIPMENT WEIGHT
       AND VOLUME
       
       EDMWT = EDM1WT + EDM2WT
       EEQMT = 0;
       EEQVOL = 0;
       
       IF (NUMEEO.EQ.0)
           F
           15
           
           T
           15
           
           CONT. ON PG  15

PG  1 OF  20

10-280
CONTINUE

**GO TO 232**

**GO TO 231**

I: = I + 1

**CONTINUE**

**CONTINUE**

**XHEWT = E0QWT + EEQHT**

**XNEYL = E0QYVL + EEQYLV1**

**HARMWT = 0.013(0.5I(NWT - MBT + RCSWP + 0.5E0QWT + 0.5E0QWY) + 0.31E0QYVL/1728)**

**C**

DETERMINE SATELLITE DRY WEIGHT LESS AUXILIARY PROPULSION DRY WEIGHT

**C**

FIRST DETERMINE SOLAR ARRAY WEIGHT

**SA2RT = 0.0**

IF(I(EQTYP.EQ.2))

**T**

**GO TO 233**

**GO TO 234**

**CONT. ON PG 16**

**PG 156F 29**

10-281
CONT. ON PG 20

PG 152F 29

10-285
CONT. ON PG 21
CONTINUE

NEXT DETERMINE CONTRIBUTION OF EXTERNAL EQUIPMENT

EEX = 0.
EEY = 0.
EEZ = 0.

IF (NUMEQD.EQ.0)

CONT. ON PG 22
C8

DO 252 I = 1, NUMEEQ

EEX = EEX + EEDY(I) * EEXCG(I)
EEY = EEY + EEDY(I) * EEYCG(I)
EEZ = EEZ + EEDY(I) * EEZCG(I)

CONTINUE

C9

CONTINUE

SATXCG = IEQBST + STRXCG + EQU + EBXCG + (EOMIST + EOMINT) * EMIXCG + 
(EOM2ST + EOM2WT) * EM2XCG + SAZ + 
(HARMWT + STPTS) * EBXCG / SATM

SATYCG = IEQBST + STRYCG + EQU + EBYCG + (EOMIST + EOMINT) * EMICYG + 
(EOM2ST + EOM2WT) * EM2YCG + SAY + 
(HARMWT + STPTS) * EBYCG / SATM

SATZCG = IEQBST + STZCG + EQU + EBLZCG + (EOMIST + EOMINT) * EMIZCG + 
(EOM2ST + EOM2WT) * EM2ZCG + SAZ + 
(HARMWT + STPTS) * EBLZCG / SATM

C CALCULATE MOMENTS OF INERTIA

C FIRST DETERMINE EQUIPMENT BAY STRUCTURE AND EQUIPMENT BAY

C EQUIPMENT INCREMENTAL INERTIA

SATRAD = 5 * SATOM

IF (ISHAPE - 2) .GT. 67.

F6

STRINF = EQBST + SATRAD**2

STRINW = .5 * EQBST * (SATRAD**2 + .167 * EOBLC**2)
STRINZ = STRINW
EOINX = .5 * EQU + SATRAD**2
EOINY = .0833 * EQU * (3.0 * SATRAD**2 + EOBLC**2)
EOINZ = EOINY
SIDE = SATOM

CONT. ON PG 23

PG 229F 29

10-288.
EXTERNAL EQUIPMENT INCREMENTAL INERTIA
SOLAR ARRAY INERTIAL CALCULATIONS

IF(IEQTYP.EQ.2)
T
GO TO 37

SA3INX = SA3WT * SATRAD + 2
SA3INY = 5 * SA3WT * (SATRAD + 2 + 1.67 * SA3XL + 2)
SA3INZ = SA3INY
GO TO 38

SA1INX = 0.0833 * SAIWT * (SA1YL + 2 + SA1ZL + 2)
SA1INY = 0.0833 * SAIWT * (SA1XL + 2 + SAIYL + 2)
SA1INZ = 0.0833 * SAIWT * (SA1YL + 2 + SA1ZL + 2)
SA2INX = 0.0833 * SA2WT * (SA2YL + 2 + SA2ZL + 2)
SA2INY = 0.0833 * SA2WT * (SA2XL + 2 + SA2ZL + 2)
SA2INZ = 0.0833 * SA2WT * (SA2YL + 2 + SA2ZL + 2)

CONTINUE

CONT. ON PG 25

PG 2 OF 29
DO 43 I = 1, MUREEQ

EEIX = EEIX + EEINX(I) + EEOWT(I) + (SATCG - EEYCG(I))**2 + (SATZCG - EEZCG(I))**2

EEIY = EEIY + EEINY(I) + EEOWT(I) + (SATCG - EEZCG(I))**2 + (SATZCG - EEZCG(I))**2

EEIZ = EEIZ + EEINZ(I) + EEOWT(I) + (SATCG - EEYCG(I))**2 + (SATZCG - EEZCG(I))**2

CONTINUE

CONTINUE

CONTINUE

SATINX = STRINX + EQBSTR*(SATCG - STRYCG)**2 + (SATZCG - STRZCG)**2
* EMINX + EQMT0*(SATCG - EM1YCG)**2 + (SATCG - EM1ZCG)**2
* EM2INX + EQM2T0*(SATCG - EM2YCG)**2 + (SATCG - EM2ZCG)**2
* EQINX + EQNT*(SATCG**2 + SATZCG**2) + SAIY + EEIX

SATINY = STRINY + EQBSTR*(SATZCG - STRZCG)**2 + (SATCG - STRXCG)**2
* EMINY + EQMT0*(SATCG - EM1YCG)**2 + (SATCG - EM1ZCG)**2
* EM2INY + EQM2T0*(SATCG - EM2YCG)**2 + (SATCG - EM2ZCG)**2
* EQINY + EQNT*(SATCG**2 + SATZCG**2) + SAIZ + EEIY

SATINZ = STRINZ + EQBSTR*(SATCG - STRYCG)**2 + (SATZCG - STRZCG)**2
* EMINZ + EQMT0*(SATCG - EM1YCG)**2 + (SATCG - EM1ZCG)**2
* EM2INZ + EQM2T0*(SATCG - EM2YCG)**2 + (SATCG - EM2ZCG)**2
* E0INZ + EQNT*(SATCG**2 + (SATCG - STRXCG)**2) + SAIZ + EEIZ

C COMPUTE DISTANCE FROM C.G. TO MAIN ENGINE(OT), GAS JET LEVER ARMS
C ON ROLL, PITCH, AND YAW AXES, RESPECTIVELY, (DX, DY, DZ). THE
C CONVERSION TO UNITS OF FT IS DONE IN SUBROUTINE SAWDC

CONT. ON PG 28
IF SHAPE-2) 45, 48, 46

DT: SATXGG - 500

DX: .5*EDBLG
DY: DX
DZ: .5*SATDAM

GO TO 47

DT: SATXGG - 500

DX: .5*SATDAM
DY: DX
DZ: DX

GO TO 47

DT: .5*EDBLG

DX: .5*EDBS1D
DY: DT
DZ: DT

PJ: EMIINX

RJ: SATINX-PJ

WRITE (6,1000) EDBLG, ACSWP, THCMT, EDBSTR, SDRMT, SARMT, HARNMT, SATHT

CONT. ON PG 29
SUBROUTINE EP (PIC,ERR,TER,NCONF,ICHOSE,NCONF2)

C SUBROUTINE EP

C WILL SELECT AND SIZE THE ELECTRICAL SUBSYSTEM WHICH WILL BE

C THESE CONFIGURATIONS AS FOLLOWS -

C NCONF (1) = 1 IS DUAL SPIN
C NCONF (1) = 2 IS YAW SPIN
C NCONF (1) = 3 IS MASS EXPULSION
C NCONF (1) = 4 IS MASS EXPULSION (MOMENTUM BIAS)

C NCONF (5) = 5 IS PITCH MOMENTUM BIAS
C NCONF (5) = 6 IS SHUNT - PADDLE
C NCONF (5) = 7 IS SHUNT - BODY
C NCONF (5) = 8 IS S + D - PADDLE
C NCONF (5) = 9 IS S + D - BODY
C NCONF (5) = 10 IS SERIES PADDLE
C NCONF (5) = 11 IS SERIES BODY
C NCONF (6) = 1 IS CYLINDER

C NCONF (6) = 2 IS BOX
C NCONF (6) = 3 IS SPHERE

C A LIST OF THE VARIABLES FOLLOWS -

C VARIABLE HOW USED FROM TO DEFAULT DESCRIPTION
C A INT EP EPS EP FT HE + HP
C A32 INT EPS EPS FT**2 ARRAY AREA
C ALT INT USER EP EPR M ALTITUDE
C AREA INT EPS EPS FT**2 ARRAY AREA
C CA INT EPS EPS EPS R-MIN NEED CP
C CAPMAX INT EPS EPS EPS R-MIN NEED CP
C CCELL INT EPS EPS EPS R-MIN NEED CP
C CHMINT INT EPS EPS EPS 2.0 HRS MIN CHO TR
C CI INT EPS EPS EPS R-MIN INSTR CP
C CISTAR INT EPS EPS EPS R-MIN NEED CEL
C CR INT EPS EPS EPS W-MIN NEED CP
C DATAB INT EPS EPS EPS DATABASE
C DELF INT EPS EPS EPS .03 W-MIN INST CP
C DEL Int EPS EPS EPS .02 W-MIN INST CP
C DELM INT EPS EPS EPS .01 W-MIN INST CP
C DELR INT EPS EPS EPS .06 OR .3 W-MIN INST CP
C DETF INT EPS EPS EPS TABLE TEMP CORR
C ETAC INT EPS EPS EPS 1.0 EFF CHG
C ETAD INT EPS EPS EPS 0.85 EFF DISCH
C ETAE INT EPS EPS EPS 0.85 EFF BATT
C ETAE INT EPS EPS EPS 0.105 SOLAR CL EF
C ETALR INT EPS EPS EPS 0.90 EFF LD REG
C ETAR INT EPS EPS EPS 1.0 PWR DIST LS
C FS INT EPS EPS EPS SIZE FACT

CONT. ON PG 2
<table>
<thead>
<tr>
<th>Column 1</th>
<th>Column 2</th>
<th>Column 3</th>
<th>Column 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>FW</td>
<td>INT</td>
<td>EPS</td>
<td>EPS</td>
</tr>
<tr>
<td>HEDA</td>
<td>INT</td>
<td>EP</td>
<td>EP</td>
</tr>
<tr>
<td>NP</td>
<td>INT</td>
<td>EP</td>
<td>EP</td>
</tr>
<tr>
<td>I</td>
<td>INT</td>
<td>EP</td>
<td>EP</td>
</tr>
<tr>
<td>ICCU</td>
<td>INT</td>
<td>EPR</td>
<td>EPR</td>
</tr>
<tr>
<td>ICELL</td>
<td>INT</td>
<td>EPS</td>
<td>EPS</td>
</tr>
<tr>
<td>ICELLE</td>
<td>INT</td>
<td>EPS</td>
<td>EPS</td>
</tr>
<tr>
<td>ICH</td>
<td>INT</td>
<td>EPR</td>
<td>EPR</td>
</tr>
<tr>
<td>ICHGR</td>
<td>INT</td>
<td>EPR</td>
<td>EPR</td>
</tr>
<tr>
<td>ICHMHD</td>
<td>INT</td>
<td>EPS</td>
<td>EPS</td>
</tr>
<tr>
<td>ICONF</td>
<td>INT</td>
<td>EPR,EPS</td>
<td>EPR,EPS</td>
</tr>
<tr>
<td>IDB</td>
<td>INT</td>
<td>MAIN</td>
<td>EPR,EPS</td>
</tr>
<tr>
<td>IDR</td>
<td>INT</td>
<td>EPR</td>
<td>EPR</td>
</tr>
<tr>
<td>IDRE</td>
<td>INT</td>
<td>EPR</td>
<td>EPR</td>
</tr>
<tr>
<td>IERR</td>
<td>INT</td>
<td>EPR</td>
<td>MAIN</td>
</tr>
<tr>
<td>IJL</td>
<td>INT</td>
<td>EPR</td>
<td>EPR</td>
</tr>
<tr>
<td>IJLRE</td>
<td>INT</td>
<td>EPR</td>
<td>EPR</td>
</tr>
<tr>
<td>IPCU</td>
<td>INT</td>
<td>EPR</td>
<td>EPR</td>
</tr>
<tr>
<td>IPCUE</td>
<td>INT</td>
<td>EPR</td>
<td>EPR</td>
</tr>
<tr>
<td>IDP</td>
<td>INT</td>
<td>EPR</td>
<td>EPR</td>
</tr>
<tr>
<td>IDPE</td>
<td>INT</td>
<td>EPR</td>
<td>EPR</td>
</tr>
<tr>
<td>IPC</td>
<td>INT</td>
<td>EPR,EPS</td>
<td>MAIN</td>
</tr>
<tr>
<td>ISPO</td>
<td>INT</td>
<td>EPR</td>
<td>EPR</td>
</tr>
<tr>
<td>ISPDE</td>
<td>INT</td>
<td>EPR</td>
<td>EPR</td>
</tr>
<tr>
<td>ISR1</td>
<td>INT</td>
<td>EPR</td>
<td>EPR</td>
</tr>
<tr>
<td>ISR1E</td>
<td>INT</td>
<td>EPR</td>
<td>EPR</td>
</tr>
<tr>
<td>ISR2</td>
<td>INT</td>
<td>EPR</td>
<td>EPR</td>
</tr>
<tr>
<td>ISR2E</td>
<td>INT</td>
<td>EPR</td>
<td>EPR</td>
</tr>
<tr>
<td>KI</td>
<td>INT</td>
<td>EPS</td>
<td>EPS</td>
</tr>
<tr>
<td>K2</td>
<td>INT</td>
<td>EPS</td>
<td>EPS</td>
</tr>
<tr>
<td>LMBDO</td>
<td>INT .O</td>
<td>EPR</td>
<td>EPR</td>
</tr>
<tr>
<td>LMBDG</td>
<td>INT</td>
<td>EPS</td>
<td>EPS</td>
</tr>
<tr>
<td>LMBDP</td>
<td>INT</td>
<td>EPS</td>
<td>EPS</td>
</tr>
<tr>
<td>MU</td>
<td>INT</td>
<td>EP</td>
<td>EP</td>
</tr>
<tr>
<td>N</td>
<td>INT</td>
<td>EP</td>
<td>EP</td>
</tr>
<tr>
<td>NB</td>
<td>INT</td>
<td>EPS</td>
<td>EPS</td>
</tr>
<tr>
<td>NC</td>
<td>INT</td>
<td>EPS</td>
<td>EPS</td>
</tr>
<tr>
<td>NCCU</td>
<td>INT</td>
<td>EPR</td>
<td>EPR</td>
</tr>
<tr>
<td>NCH</td>
<td>INT</td>
<td>EPS</td>
<td>EPS</td>
</tr>
<tr>
<td>NCHSOE</td>
<td>INT</td>
<td>EPR,EPS</td>
<td>MAIN</td>
</tr>
<tr>
<td>NCONF1</td>
<td>INT .EPS</td>
<td>MAIN</td>
<td>EPS.MAIN</td>
</tr>
<tr>
<td>NCONF2</td>
<td>INT .EPS</td>
<td>MAIN</td>
<td>EPS.MAIN</td>
</tr>
<tr>
<td>NCONF3</td>
<td>INT .EPS</td>
<td>MAIN</td>
<td>EPS.MAIN</td>
</tr>
<tr>
<td>NPD</td>
<td>INT</td>
<td>EPR</td>
<td>EPR</td>
</tr>
<tr>
<td>NPCU</td>
<td>INT</td>
<td>EPR</td>
<td>EPR</td>
</tr>
<tr>
<td>NSPD</td>
<td>INT</td>
<td>EPR</td>
<td>EPR</td>
</tr>
</tbody>
</table>

CONT. ON PG 3
<table>
<thead>
<tr>
<th>C</th>
<th>WSR</th>
<th>INT</th>
<th>EPS</th>
<th>EPR</th>
<th>EPR</th>
<th>NO SHNT REG#</th>
</tr>
</thead>
<tbody>
<tr>
<td>C</td>
<td>OPTMP</td>
<td>USER</td>
<td>EP</td>
<td>EPR</td>
<td>EPR</td>
<td>15... DEG. C BAT TEMP#</td>
</tr>
<tr>
<td>C</td>
<td>PBOL</td>
<td>INT</td>
<td>EPS</td>
<td>EPS</td>
<td>EPS</td>
<td>WATTPWR B.O.</td>
</tr>
<tr>
<td>C</td>
<td>PD</td>
<td>INT</td>
<td>EPS</td>
<td>EPS</td>
<td>EPS</td>
<td>WATTPWR P.O.-REG#</td>
</tr>
<tr>
<td>C</td>
<td>PEXCES</td>
<td>INT</td>
<td>EPS</td>
<td>EPS</td>
<td>EPS</td>
<td>WATTPHO 2O DISP#</td>
</tr>
<tr>
<td>C</td>
<td>PIC</td>
<td>EPS</td>
<td>EPS</td>
<td>EPS</td>
<td>EPS</td>
<td>WATTAV P.W. O#</td>
</tr>
<tr>
<td>C</td>
<td>PMIN</td>
<td>EPS</td>
<td>EPS</td>
<td>EPS</td>
<td>EPS</td>
<td>WATT MIN P.W. O#</td>
</tr>
<tr>
<td>C</td>
<td>PRED</td>
<td>EPS</td>
<td>EPS</td>
<td>EPS</td>
<td>EPS</td>
<td>WATTTOT P.W. LRE#</td>
</tr>
<tr>
<td>C</td>
<td>PS</td>
<td>EPS</td>
<td>EPS</td>
<td>EPS</td>
<td>EPS</td>
<td>WATTPWR DISP LRE#</td>
</tr>
<tr>
<td>C</td>
<td>AFD</td>
<td>EPS</td>
<td>EPS</td>
<td>EPS</td>
<td>EPS</td>
<td>WATTEOL SOL OUT#</td>
</tr>
<tr>
<td>C</td>
<td>S</td>
<td>EPS</td>
<td>EPS</td>
<td>EPS</td>
<td>EPS</td>
<td>TERP DEC FC#</td>
</tr>
<tr>
<td>C</td>
<td>SOL</td>
<td>EPS</td>
<td>EPS</td>
<td>EPS</td>
<td>EPS</td>
<td>USED IN CALC OF TE#</td>
</tr>
</tbody>
</table>

C | TE   | INT  | EPS  | EPR  | EPR  | ECPS TIME# |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>C</td>
<td>TOTTS</td>
<td>EPS</td>
<td>EPS</td>
<td>EPS</td>
<td>EPS</td>
<td>FIEL3 UNIT BATVOL#</td>
</tr>
<tr>
<td>C</td>
<td>VB</td>
<td>EPS</td>
<td>EPS</td>
<td>EPS</td>
<td>EPS</td>
<td>VOMAIN BAT VLT#</td>
</tr>
<tr>
<td>C</td>
<td>VBM</td>
<td>EPS</td>
<td>EPS</td>
<td>EPS</td>
<td>EPS</td>
<td>VOMAIN CELL U#</td>
</tr>
<tr>
<td>C</td>
<td>VCE</td>
<td>EPS</td>
<td>EPS</td>
<td>EPS</td>
<td>EPS</td>
<td>VOMAIN CELL V#</td>
</tr>
<tr>
<td>C</td>
<td>VCEL</td>
<td>EPS</td>
<td>EPS</td>
<td>EPS</td>
<td>EPS</td>
<td>VOMAIN CELL W#</td>
</tr>
</tbody>
</table>

C | VOL  | O    | MAIN | EPS  | EPS  | KG ARRAY MT# |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>C</td>
<td>WATE</td>
<td>EPS</td>
<td>EPS</td>
<td>EPS</td>
<td>EPS</td>
<td>KGUNIT BAT MT#</td>
</tr>
<tr>
<td>C</td>
<td>WB</td>
<td>EPS</td>
<td>EPS</td>
<td>EPS</td>
<td>EPS</td>
<td>KGUNIT BAT MT#</td>
</tr>
<tr>
<td>C</td>
<td>WB7</td>
<td>EPS</td>
<td>EPS</td>
<td>EPS</td>
<td>EPS</td>
<td>LB CELL WDT#</td>
</tr>
<tr>
<td>C</td>
<td>WCELL</td>
<td>EPS</td>
<td>EPS</td>
<td>EPS</td>
<td>EPS</td>
<td>LB5 EP MT#</td>
</tr>
</tbody>
</table>

COMMON /USER/
- APODEE: COMMAT: DIMALO: EEQWHIT: EPM:
- AMDT: ECDI: EDEM: EGC2MT: EOM:
- IDEBUG: ESSA: MB2: OPTMP: OFFLINE: PIC:
- MIC: RSEC: VER161: SPEC: TCS:

COMMON /BTW/
- ACSN: ACSWP: ALT: AREA: BATCAP:
- BITRATI: CLIFE: CONN: DTF: BIV:
- DX: DZ: EBOLO: EOB10: BIV:
- FC: FF: HRPW: HRPW:
- HPT: HTPR: HTPR:
- LMB: N: OREGS: PASS: J:
- PL: PMLN: POCMNT: RARA: RARA:
- RAT: R: SABOLC: SAIL: SATT:

CONT. ON PG 4
PG 3 OF 39

10-298
A32 = A**1.5
MEDA = ME/A
S = 1.02 * ASIN(MEDA)
N = SQRT(NU)/A32

IETS = S/(PIE*S)
TE = 2*S/N
RFD = 0.1*OPTEMP + 1.0

DO 10 I = 1,5
10 (CHOOSE(I) = 0, IE = 0)

C = LMB60D MUST GO TO REL

MB = 2

DO 20 I = 1,5
20 (CHOOSE(I) = 0, NL = 2
urate = 0.0
ARE = 0.0

C = SET UP DELTA-R AND DELTA-T (RADIATION DEGRADATION AND TEMPERATURE CORRECTION FACTORS)

DELR = .05

IF (ALT.GT.400) THEN
  E = 6
  M1
  DELR = 3

CONT. ON PG 6
NOW WE WILL BE DOING THE EPR MACRO SELECTION IS SAMDO.SLR

C == SHUNT REGULATION DESIGN ==

IF (IPIC1) NE 0

CONT. ON PG 9
(PCU=108(11)+1
NPCU=1
ETAE=0.65
ETAC=1.0
ETAD=1.0
ETAR=1.0

GO TO 210

IF ITER.EQ.0

IF ISR1.LE.ISR1E

GO TO 210

ISR1=IP(C11)
ICELL=IP(C12)
ICHGR=IP(C13)

GO TO 210

ISR1=IP(C11)+1
ICELL=IP(C12)
ICHGR=IP(C13)

CONT. ON PG 10

PG 9 OF 39
ICHOSE1 = 1  
ICHOSE2 = 1  
RETURN

**COMPUTE SELECTION PARAMETERS FOR SHUNT REGULATION DESIGN**

**THIS IS FOR SHUNT REGULATOR, BATTERY AND BATTERY CHARGER**

**DETERMINE NUMBER OF SHUNT REGULATORS REQUIRED**

**DETERMINE EXCESS ARRAY POWER FOR REGULATION**

```
WSR = 1

PS = (PL/ETAR) * (PEDT5 * (ETAD * ETAC * ETAE))

PBOL = PS / (1 - DELR) * (1 - DELF) * (1 - DELT) * (1 - DELI) * (1 - DELM)
PEXCES = PBOL - PLMIN
CAPMAX = 0.0

IF (PEXCES .GE. 720.)
  T
  CAPMAX = 120.
  F

IF (PEXCES .GE. 1440.)
  T
  CAPMAX = 240.
  F
```

CONT. ON PG 12

PG 10-306
[Flowchart Diagram]

IF (PMIN >= PEXCES)
    T
    GO TO 230

WSR = PEXCES / DATA16(ISRI) - 0.9

IF (WSR <= 0)
    T
    WSR = 1

CONTINUE

C ** NOTE -- ADD SPECIAL EQUIPMENT (AS NECESSARY)
C ** SET VOLTAGES FOR THIS DESIGN

VDB = 27.  
VBM = 26.

C ** BATTERY ALGORITHM
C DETERMINE REQUIRED CAPACITIES

(CR = (PLATE/3600) / (LM80D*ETAD))
(CR = CR/VDB)

C DETERMINE MINIMUM INSTALLED CAPACITY

CI = CR*RF

C ** DETERMINE NUMBER OF CELLS IN SERIES TO BE SUPPLIED TO REL

NC = VBM/VC

C DETERMINE SELECTION PARAMETERS ON CELLS

CONT. ON PG 13

PG 12F 33
IF (ITER .GE. 1)

MB = NCHOOSE(2)

CISTAR = C / MB

**Determine charge current rating required for the battery charger.**

CELL = CISTAR
ICH = CELL / CHMINT

**Compare the hardware parameter to the selection parameter.**

IF (DATAB16(ISRI).GE.CAPMAX .AND. DATAB16(CELL).GE.CISTAR .AND. DATAB16(CHOR).GE.ICH)

GO TO 270

IF (ISRI .GE. ISRILE)

GO TO 240

ISRI = ISRI + 1

CONT. ON PG 14
\begin{align*}
&\text{VCELL: DATAB} (24 \cdot \text{ICELL}) \\
&\text{MC} (\text{LL} = \text{GAS} (23 \cdot \text{ICELL}) \\
&\text{WB} = \text{NC} \cdot \text{VCELL} \cdot k_2 \\
&\text{YB} = \text{NC} \cdot \text{VCELL} \cdot k_1 \\
&W_{\text{BT}} = \text{WB} \cdot \text{WB} \\
&W_{\text{BT}} = \text{WB} \cdot \text{WB} \\
&B_{\text{ATCA}} = \text{WB} \cdot \text{DATAB} (6 \cdot \text{ICELL}) \\
&\text{ETAC} = \text{DATAB} (7 \cdot \text{ICHR}) \\
&W_{\text{CM}} = \text{WB} \\
&\text{IF} \ (\text{INCHOSE} (1) \geq \text{NSR}) \\
&T \\
&\text{GO TO 271} \\
&W_{\text{CMCHOSE} (1) = \text{NSR}} \\
&271 \\
&\text{IF} \ (\text{INCHOSE} (2) \geq \text{WB}) \\
&T \\
&\text{GO TO 272} \\
&\text{CONT. ON PG 16} \\
&\text{PG 15F 39} \\
&10-310
\end{align*}
IF (ITER.EQ.0)
  T
  GO TO 300
  F
  IDR=IPIC(1)
  IAR2=IPIC(2)
  ICELL=IPIC(3)
  ICNAR=IPIC(4)
  ICUS=IPIC(5)
  GO TO 360
300
IF (IDR.GE.IDRE)
  T
  GO TO 310
  F
  IDR=IPIC(1)+1
  IAR2=IPIC(2)
  ICELL=IPIC(3)
  ICNAR=IPIC(4)
  ICUS=IPIC(5)
  GO TO 360
CONT. ON PG 19
C ** COMPUTE SELECTION PARAMETERS FOR SHUNT AND DISCHARGE REGULATION -
C ** THIS IS FOR DISCHARGE REGULATOR.BATTERY.BATTERY
C ** CHARGER AND SIZING THE CENTRAL CONTROL UNIT --
C ** DETERMINE NUMBER OF DISCHARGE REGULATORS REQUIRED

C ** DETERMINE EXCESS ARRAY POWER FOR REGULATION

PS=(PL/ETAR)*(1.+TEDS*(1./ETAD+ETAC+ETAE))
PBOL:PS/(1.-DELT)*(1.-DELF)*(1.-DELT)*(1.-DELM))
PEXCES:PBOL-PLMIN

CONT. ON PG 21

PG 20F 39

10-315
\[ F(I_{\text{REF}} \geq 1 \text{ AND } N_{\text{CHOICE}1} \geq N_{\text{DO}}) \]

1. \( N_{\text{DO}} = N_{\text{CHOICE}1} \)
2. \( P_{\text{D}} = P_{\text{L}}/N_{\text{DO}} \text{ (INDENETED)} \)

**C** // DETERMINE NUMBER OF SHUNT REGULATORS REQUIRED

3. \( C3 \)
4. \( 24 \)
5. \( 4 \)
6. \( 24 \)

7. \( \text{CAPMAX} = 0.0 \)

8. \( \text{IF } P_{\text{EXCES}} \geq 720.0 \) \( F \)
9. \( \text{IF } P_{\text{EXCES}} \geq 1440.0 \) \( F \)

10. \( \text{CONT. ON PG 22} \)

11. \( P_{\text{G}} \text{ OF } 39 \)

12. \( \text{NSR}: 1 \)

13. \( \text{CONT. ON PG 22} \)
IF (PLMIN GE PEXCES)

T

GO TO 380

NSR : PEXCES / DATAB16.ISR2) + 0.9

IF (NSR LE 0)

T

NSR = 1

380

CONTINUE

C == SET VOLTAGES FOR SHUNT AND DISCHARGE DESIGN

VDD = 21
VBM = 19

C == SET UP BATTERY SELECTION PARAMETER
DETERMINE REQUIRED CAPACITIES

CR = (IPL + E/3600) / (LMBOD + E)AD
CA = CR / VDB

C
DETERMINE MINIMUM INSTALLED CAPACITY

CI = CA + AFD

C
DETERMINE NUMBER OF CELLS IN SERIES TO BE SUPPLIED TO REL

WNC = VBAR / VC

C
CISTAR IS SELECTION PARAMETER ON CELLS

CONT. ON PG 23
I

IF (ITER .GE. 1)
T
MB = MCLOSE(3)

CISTAR = CI/MB

C ** CHARGER SELECTION
C ** DETERMINE CHARGE CURRENT RATING REQUIRED FOR THE BATTERY CHARGER

ICELL = CISTAR
ICH = ICCELL/CMINT

C ** COMPARTE THE HARDWARE PARAMETER TO THE SELECTION PARAMETER

IF (IAB16.IDR).GE.PQ.AND.(IAB16.ISR2).GE.CAPMAX.AND.(IAB16.IE
1).GE.CISTAR.AND.(IAB16.ICHR).GE.ICH)
T
GO TO 440

IF (IDR .GE. IDRE)
T
GO TO 390

IDR = IDR + 1

CONT. ON PG 24

10-318
**C** == COMPUTE SELECTION PARAMETERS FOR SERIES LOAD REGULATION

**C** == THIS IS FOR THE LOAD REGULATOR, BATTERY, BATTERY CHARGER AND 
SIZING THE SOLAR POWER DISTRIBUTOR AND POWER DISTRIBUTOR

**NLR** IS THE NUMBER OF LOAD REGULATORS REQUIRED

![Flowchart Diagram]

- **C** == DETERMINE EXCESS ARRAY POWER FOR REGULATION
  
  - PS = \((PL/ETAR)*[1 - ROTS \times (1/ETAD)]\)
  
  - POOL = PS * (1 + DELR) * (1 + DELF) * (1 - DEL1) * (1 - DELM)
  
  - PEXCES = POOL - PMIN

- **C** == DETERMINE SELECTION PARAMETERS FOR LOAD REGULATORS

  ![Flowchart Diagram]

  - IF (ITER . GE. 1 . AND. NCLOSE(I) . GE. NLR)
  - NLR = NCLOSE(I)

  - PLR = PL / (ETAR * NLR)

  ![Flowchart Diagram]

  - SET VOLTAGES FOR THIS DESIGN
    - VDB = 27
    - VBM = 23

  ![Flowchart Diagram]

  - SET UP BATTERY SELECTION PARAMETERS
    - DETERMINE REQUIRED CAPACITIES
      - CR = \((PL/TE) / (1 + MBD) ETAD\)
      - CA = CR / VDB

  ![Flowchart Diagram]

  - DETERMINE MINIMUM INSTALLED CAPACITY
    - CI = CA * RFD

*CONT. ON PG 32*
Determine number of cells in series to be supplied to rel.

NC = VBM/VC

CISTAR is selection parameters on cells

IF (ITER .GE. 1)
  T
  NB = NCHOOSE(2)
  CISTAR = CI/NB

Charger selection parameter

C = Determine charge current rating required for the battery charger

CCELL = CISTAR
ICH = CCELL/CHMIN

Compare the hardware parameter to the selection parameter

IF (DATA(6,1) .GE. PLR.AND.DATA(6,CELL).GE.CISTAR.AND.DATA(6,ICH).GE.ICH)
  T
  GO TO 580

CONT. ON PG 33

PG 326 38
C == BODY MOUNTED SPHERE NON-SPINNING

E8

FM=3.4

LMBD=5

C COMPUTE ENERGY BALANCE EQUATION

E9

PS=PL/ETAR=11.1/ETAC=ETAE(1)=

C COMPUTE SIZING FACTOR

FS=LMBD0=LMBD0=11.-DELR=11.-DERL=11.-DELI=11.-DELM)

C COMPUTE ARRAY AREA

AI=PS/FS=SOL=ETAI()

C COMPUTE ARRAY WEIGHT

WATE=AI*FM

C CONVERT TO ENGLISH FROM METRIC

AREA=AI*10.7642285
WATE=WATE*2.20462

RETURN

END
C  INITIALIZE ICHOOSE, NCHOOSE, IERR AND SELECT HARDWARE NOT SIZED
I.E., THE FILL AND VENT VALVE AND RELIEF VALVE

C

A1

DO 2 I=1,14

ICHOOSE(I)=0

2

NCHOOSE(I)=0

2

IERR=0
NCHOOSE(1)=6
NCHOOSE(2)=2
NCHOOSE(3)=4
NCHOOSE(4)=9

DO 299 I=5,8

299

NCHOOSE(I)=1

I=I08(I) + 1
J1=I0B(I) + 1
ICHOOSE(I7) = DATAB(I,I)
ICHOOSE(I0) = DATAB(I0,J1)

A2

CONTINUE

C  THRUSTER SELECTION
C  FIRST CHECK TO SEE IF THERE IS AN ACCEPTABLE THRUSTER IN THE DATA
C  BASE

J1E=I0B(1)
J1=1

A3

THUST=DATAB(6,J1)

CONT. ON PG 4

PG 3 OF 63

10-337
IF (THRUSt < DE.FMAX)

T

GO TO 12

IF (J1.EQ.J1E)

T

GO TO 11

J1 = J1 + 1

GO TO 10

C NO ACCEPTABLE THRUSTERS

11

[CHOOSE(J1) = -1

RETURN

C AT LEAST ONE ACCEPTABLE THRUSTER

12

CONTINUE

C SELECT PNEUMATIC ATTITUDE AND CONTROL THRUSTERS
C FIRST DETERMINE SET OF ALL THRUSTERS WHICH SATISFY THE INEQUALITY
C THRUST GE ACTHST

CONT. ON PG 5

PG 4 OF 63

10-338
**C** CHOOSE THAT THRUSTER FROM THE ABOVE SET WHICH MINIMIZES THE
QUANTITY, ABS(THRUST - ACTHST).

\[ I = 1 \]
\[ J = 1 \]
\[ \text{THRUST} = \text{DATA}(6 + J1) \]
\[ \text{DIFOLD} = \text{ABS}(	ext{THRUST} - \text{ACTHST}) \]

\[ \text{CONT. ON PG 7} \]

\[ \text{PG 6 OF 63} \]

10-340
Select pneumatic translational thrusters using above procedure.
IF (J2.EQ.J1E) THEN
    GO TO 24
    J2 = J2 + 1
    GO TO 21
B2
IACCP1(I) = J2
I = I + 1
GO TO 22

CONTINUE

[MAX = I - 1
I = 1
J2 = IACCP1(I)
THRUST = DATA8(I,J2)
DIFOLD = ABSITHRUST - TTMST)

B3
ICHOSE(2) = DATA8(I,J2)
JSAVE = J2

IF (I.EQ.[MAX) THEN
    GO TO 8
B5
CONT. ON PG 9
J2 = IACCP(1)
THrust = DATAB(6, J2)
DIFNEW = ABS(THRust - THUST)

IF (DIFNEW <= DIFOLD)
T

GO TO 27

IF (F < MAX)
T

GO TO 26

GO TO 28

DIFOLD = DIFNEW

GO TO 25

J2 = SAVE

T(1) = DATAB(6, J2)

CONT. ON PG 10
THRUSTERS HAVE BEEN SELECTED
SET NUMBER OF EACH TYPE OF THRUSTER
CHECK TO SEE IF CYCLE LIFE REQUIREMENT IS SATISFIED

[iErr: 0]

IF((DAB[1:7,1])LT.CLIFE):
T
[iErr: 1]

IF((DAB[1:7,2])LT.CLIFE):
T
[iErr: iErr + 10]

iErr: 1 implies that the cycle life of the attitude and control thrusters is too short. iErr: 10 implies that the cycle life of the translational thrusters is too short. iErr: 11 implies that the cycle lives of both thrusters are too short.

PRELIMINARY CALCULATIONS FOR SELECTION OF PNEUMATIC ISOLATION VALVES AND FILTERS

PTI = DAB[1:8,1]

RHO = 1.02 - 7*PTI
WDOPTPR = (3.*ACTHST + 2.*TTHTST)/65.
CDAISO = WDOPTPR/SQRT(200.*RHO/1.29E-3)
RNAX = 200./WDOPTPR^2

SET LAST EQUIPMENT INDICES

J3E = (0B12)
J4E = (0B13)
J5E = (0B14)
J6E = (0B15)

DETERMINE HARDWARE INDICES

CONT. ON PG 11

PG 10F 63

10-344
J3 = [DB(1) + 1]  
J4 = [DB(2) + 1]  
J5 = [DB(3) + 1]  
J6 = [PIC(4) + 1]  
J7 = [100SF]  
J8 = [109(2)]  
J9 = [10813]  
J10 = [PIC(14) + 1]  

CONTINUE

C THE HARDWARE INDICES ARE SET

32

(F\(D\)B[7, J3].LT. CDAISO OR \(D\)B[7, J4].GT. RMAX)

T

GO TO 33

C ISOLATION VALVE AND FILTER ARE ACCEPTABLE

DELPI = (1.29E-3/RHO) \(\times\) (WDOTPR/\(D\)B[7, J3])\(^2\)

DELPI = \(D\)B[7, J4] \(\times\) WDOTPR\(^2\)

\(I\)CHOOSE(3) = \(D\)B[1, J3]

\(I\)CHOOSE(4) = \(D\)B[1, J4]

C PRELIMINARY CALCULATIONS FOR SELECTION OF REGULATOR AND TANK

PREG = PTI + 2 \times DDELPI + DELPI

CDAREG = WDOTPR/SQRT(5600 \times PREG/1.27E4)

WPR = 1.1 \times TOTIMP/65.

ACSWP = WPR

VPRT = 3.4E3 \times WPR/28.

IF PREG.LT.DAB[8, J5]. OR PREG.GT.DAB[9, J5]. OR DAB[7, J5].LT.
CDAREG. OR DAB[6, J5].LT. VPRT)

T

GO TO 33

CONT. ON PG 15

PG 1 OF 63

10-348
REGULATOR AND TANK ARE ACCEPTABLE

IF (CHOOSE(5) = DATA(I+J5))

IF (CHOOSE(6) = DATA(I+J6))

INHT = DATA(23+J6)

SIZE PLUMBING AND CONNECTORS

PCWATE = 0.2 * DATA(23+J6) + CHOOSE(6)

STORE LAST INDICES ACCEPTABLE

IPIC(1) = J3
IPIC(2) = J4
IPIC(3) = J5
IPIC(4) = J6
N(7) = II
N(8) = JJ
N(9) = J1
N(10) = J2
N(11) = J3
N(12) = J4
N(13) = J5
N(14) = J6

DO 322 I = 1, 8

J = N(I)
WT = WT + CHOOSE(I) * DATA(23+J)
VOL = VOL + CHOOSE(I) * DATA(24+J)
PL = PL + CHOOSE(I) * DATA(16+J)
PLMIN = PLMIN + CHOOSE(I) * DATA(18+J)

CONTINUE

CONTINUE

DOMIT = WT + PCWATE - WEIGHT(I)
WT = WT + ACSWP + PCWATE

RETURN

CONT. ON PG 16
PG 195F 63
HARDWARE SELECTION NOT ACCEPTABLE - INCREMENT HARDWARE INDICES

IF (I3 < J3) THEN
   GO TO 34
END IF

IF (I4 < J4) THEN
   GO TO 35
END IF

IF (I5 < J5) THEN
   GO TO 36
END IF

IF (I6 < J6) THEN
   GO TO 37
END IF
C  NO ACCEPTABLE HARDWARE COMBINATION

16 C9

ICHOOSE(1) = -1

RETURN

16 D1

J3 = J3 + 1

GO TO 32

16 D2

J3 = IDB(1) + 1

J4 = J4 + 1

GO TO 32

16 D3

J3 = IDB(1) + 1

J4 = IDB(2) + 1

J5 = J5 + 1

GO TO 32

16 D4

J3 = IDB(1) + 1

J4 = IDB(2) + 1

J5 = IDB(3) + 1

J6 = J6 + 1

GO TO 32

CONT. ON PG 18

PG 176 63

10-351
IF(NCONF(2).EQ.3)

GO TO 82

C THIS IS MONOPROPELLANT CONFIGURATION
Determine maximum thrust from SANDC

FMAX = AMAX1(THMST,ITMST)

IF(FMAX.LT.1000.AND.TOTIMP.LT.200000.AND.TOTIMP.GE.10000.)

GO TO 39

C THIS IS NOT AN ACCEPTABLE CONFIGURATION

ICHUSE(1) = -1
RETURN

CONT. ON PG 19
IF (ITER.NE.0) THEN
  GO TO 42
END IF

C INITIALIZE ICHOSE,MCHOSE,IERR AND SELECT HARDWARE NOT SIZED
C I.E., THE RELIEF VALVE, FILL AND VENT VALVE AND FILL AND DRAIN

CALL 40 I = 1, 1
I MCHOSE(I) = 0

CALL 40 I = 2, 1
I MCHOSE(I) = 0
I IERR = 0
I MCHOSE(1) = 5
I MCHOSE(2) = 2
I MCHOSE(3) = 4
I MCHOSE(4) = 9

CALL 41 I = 5, 1
I MCHOSE(I) = 1
I I = IDB(5) + 1
I JJ = IDB(6) + 1
I KK = IDB(11) + 1
I ICHOSE(9) = DATAB(I, JJ)
I ICHOSE(10) = DATAB(I, KK)
I ICHOSE(11) = DATAB(I, I)

CALL 42 CONTINUE

C THRUSTER SELECTION

CONT. ON PG .20
FIRST CHECK TO SEE IF THERE IS AN ACCEPTABLE THRUSTER IN THE DATA BASE

\[ \text{JIE} = [\text{DB}(\text{B})] \]
\[ \text{JI} = [\text{DB}(\text{L})] + 1 \]

100

\[ \text{THRUST} = \text{DATABASE}(\text{JI}) \]

\[ \text{IF(THRUST.CO.FMAX) F} \]
\[ \text{T} \]
\[ \text{GO TO 120} \]

\[ \text{IF(J1.EQ.JIE) F} \]
\[ \text{T} \]
\[ \text{GO TO 110} \]

\[ \text{J1 = J1 + 1} \]
\[ \text{GO TO 100} \]

C NO ACCEPTABLE THRUSTERS

110

\[ \text{CHOOSE}(-1) \]
\[ \text{RETURN} \]

CONT. ON PG 21

PG 200F 83

10-354
AT LEAST ONE ACCEPTABLE THRUSTER

20 D6
120 CONTINUE

SELECT PNEUMATIC ATTITUDE AND CONTROL THRUSTERS
FIRST DETERMINE SET OF ALL THRUSTERS WHICH SATISFY THE INEQUALITY
THRUST ≥ ACST

I = 1
J1 = [DB(7)] + 1

130 THRUST = DATAB(6+J1)

IF (THRUST ≥ ACST)

T

GO TO 150

140

D7

I F (I1 = J1)

F

IF (I1 = J1)

T

GO TO 160

J1 = J1 + 1

GO TO 130

150 [ACCEPT(I) = J1]

CONT. ON PG 22

PG 2 DF 63
CHOOSE THAT THRUSTER FROM THE ABOVE SET WHICH MINIMIZES THE
QUANTITY, ABS(THRUST - ACTHST)

\[
\begin{align*}
&I = 1 \\
&J_{I} = \text{ACCEPT}(I) \\
&\text{THRUST} = \text{DATA}(I + 1) \\
&\text{DIFOLD} = \text{ABS}(\text{THRUST} - \text{ACTHST}) \\
&\text{D9}
\end{align*}
\]

\[
\begin{align*}
&(D9) \\
&170 \\
&(J_{SAVE} = J_{I}) \\
&\text{IF}(I \text{ EQ. I}_{MAX}) \\
&\text{F} \\
&\text{GO TO 200} \\
&\text{T} \\
&\text{GO TO 200}
\end{align*}
\]

\[
\begin{align*}
&I = I + 1 \\
&J_{I} = \text{ACCEPT}(I) \\
&\text{THRUST} = \text{DATA}(I + 1) \\
&\text{DIFNEW} = \text{ABS}(\text{THRUST} - \text{ACTHST}) \\
&\text{IF(DIFNEW LE DIFOLD)} \\
&\text{F} \\
&\text{E3} \\
&\text{T} \\
&\text{E2}
\end{align*}
\]

CONT. ON PG 23

PG 220F 63
SELECT PNEUMATIC TRANSLATIONAL THRUSTERS USING ABOVE PROCEDURE
THRUSTERS HAVE BEEN SELECTED
SET NUMBER OF EACH TYPE OF THRUSTER
CHECK TO SEE IF CYCLE LIFE REQUIREMENT IS SATISFIED

ERRZ 10 ERRZ 0 T T
IF(DATAB1+J1) LT CLIFE)

ERR = 1

ERR = ERR + 10

ERR = 1 IMPLIES THAT THE CYCLE LIFE OF THE ATTITUDE AND CONTROL
THRUSTERS IS TOO SHORT. ERR = 10 IMPLIES THAT THE CYCLE LIFE OF
THE TRANSLATIONAL THRUSTERS IS TOO SHORT. ERR = 11 IMPLIES THAT
THE CYCLE LIVES OF BOTH THRUSTERS ARE TOO SHORT
PRELIMINARY CALCULATIONS FOR SELECTION OF MONOPROPELLANT ISOLATION
VALVES AND FILTERS

WDOTF = (Rdot + 2 * TTHST)/200.
C NO ACCEPTABLE COMBINATIONS

ICHOOSE(I) = -1
RETURN

CONT. ON PG 30

PG 29DF 63
C THE HARDWARE INDICES ARE SET

IF (DATAB(7,J3).LT.CDAISO .OR. DATAB(7,J4).GT.RMAX)

T

GO TO 55

F

C FUEL CIRCUIT ISOLATION VALVES AND FILTERS ARE ACCEPTABLE

DELPIS = (1.29E-3/RHOF)*(WDOTF/DATAB(7,J3))**2

DELPFI = (DATAB(7,J4)*WDOTF)**2

ICHOOSE(3) = DATAB(1,J3)

ICHOOSE(4) = DATAB(1,J4)

IPIC(1) = J3

IPIC(2) = J4

C PRELIMINARY CALCULATIONS FOR SELECTION OF PNEUMATIC REGULATOR

PTI = DATAB(I,J1)

PFT = PTI + 2*DELPIS + 2*DELPFI

PREG = PFT + 2*DELPIS

WDOTPR = 29.1*0.02E-7*WREG/WDOTF/RHOF

COAREG = WDOTPR/SORT15000.*WREG/1.27E4)

IF (PREG.LT.DATAB(8,J5).OR.PREG.GT.DATAB(9,J5).OR.DATAB(7,J5).LT.

COAREG)

T

GO TO 55

F

C REGULATOR IS ACCEPTABLE

ICHOOSE(5) = DATAB(1,J5)

CONT. ON PG 93
Preliminary Calculations for Selection of Pneumatic Isolation Valve

\[ \text{RHOPR} = 3000 \times 1.02 \times 10^{-7} \]
\[ \text{CDAISO} = \frac{\text{WHTPR}}{\text{SORT}} \times \left( \frac{200 \times \text{RHOPR}}{1.29 \times 10^{-3}} \right) \]

\[ \text{IF} (\text{DATAB}(100) \times 10^6 < \text{CDAISO}) \]

\[ \text{GO TO 55} \]

Pneumatic Isolation Valve is Acceptable

\[ \text{IF} (\text{DATAB}(100) \times 10^6 > \text{CDAISO}) \]

\[ \text{GO TO 55} \]

Preliminary Calculations for Selection of Fuel Tank and Pneumatic Tank

\[ \text{WF} = \frac{\text{TOTIMP}}{200} \]
\[ \text{VF} = \text{WF} \times 0.036 \]
\[ \text{VFT} = \text{WF} \times 0.036 \]
\[ \text{VPRT} = \text{PFT} \times \text{VFT} \times (1000 - 2 \times \text{PFT}) \]
\[ \text{WPRT} = \text{WDBS} \times \text{VPRT} \]
\[ \text{ACSMP} = \text{WF} \times \text{WPRT} \]

Select Fuel Tank

\[ \text{IF} (\text{DATAB}(100) \times 10^6 < \text{PFT}) \]

\[ \text{GO TO 55} \]

CONT. ON PG 34
• PRESSURE WHICH SATISFIES DATA BASE TANK ONE FUEL IN OR PRESSURE CRITERIA.

[Flowchart diagram]

C AT LEAST ONE FUEL TANK IN DATA BASE WHICH SATISFIES PRESSURE DESIGN CRITERIA.

J7SAVE = J7

VFTMAX = FTAB(6, J7)

CHECK TO SEE IF THERE IS AT LEAST ONE TANK IN DATA BASE WHICH SATISFIES BOTH PRESSURE AND VOLUME DESIGN CRITERIA.

J7 = 10810 + 1

CONT. ON PG 35
NO TANK IN DATA BASE WHICH SATISFIES BOTH THE PRESSURE AND VOLUME DESIGN CRITERIA. SELECT TANK WITH LARGEST VOLUME WHICH SATISFIES PRESSURE DESIGN CRITERIA.

\[ JJ7 = ICB(10) + 1 \]

\[ S003 \]

\[ IF (DATAB7.JJ7) .LT. PFT OR. DATAB6.JJ7) .LT. VFTMAX \]

\[ JJ7 = JJ7 + 1 \]

CONT. ON PG 36

PG 360F 63
Ic SlZE
-PLUMING AND CONNECTORS

PCWATE = .2*(DATAB(23+J7)+NCHOSE(7)+DATAB(23+J8)+NCHOSE(8))
N(9) = JJ
N(10) = KK
N(11) = JJ
N(12) = J1
N(13) = J2
N(14) = J3
N(15) = J5
N(16) = J6
N(17) = J7
N(18) = J8

DO 542 I=1,11

J = N(I)
WT = WT + NCHOSE(I)*DATAB(23+J)
VOL = VOL + NCHOSE(I)*DATAB(24+J)
PL = PL + NCHOSE(I)*DATAB(16+J)
PLMIN = PLMIN + NCHOSE(I)*DATAB(19+J)

CONTINUE

542 CONTINUE

PRINT = WT + PCWATE - WEIGHT(I)
WT = WT + ACSPW + PCWATE
RETURN

IF (I3.LT.J3E) CONT. ON PG 38

C HARDWARE SELECTION NOT ACCEPTABLE - INCREMENT HARDWARE INDICES
THIS IS NOT AN ACCEPTABLE CONFIGURATION

CONT. ON PG 41
$\text{I} = \text{CHOSE}(1) = -1$  
\text{RETURN}  

$\text{I} = \text{CHOSE}(15)$  
\text{CONTINUE}  

$\text{F}$  
\text{IF (ITER .NE. 0))}  
\text{T}  
\text{GO TO 65}  

\text{INITIALIZE ICHOSE, NCHOSE, IERR AND SELECT HARDWARE NOT SIZED}  
\text{I.E. FILL AND DRAIN VALVES, FILL AND VENT VALVE AND RELIEF VALVE}  

\text{DO 64 I = 1, 14}  
$\text{I} = \text{CHOSE}(1) = 0$  
$\text{NCHOSE}(1) = 0$  
$\text{IERR} = 0$  
$\text{NCHOSE}(1) = 6$  
$\text{NCHOSE}(2) = 2$  
$\text{NCHOSE}(3) = 3$  
$\text{NCHOSE}(4) = 3$  
$\text{NCHOSE}(5) = 4$  
$\text{NCHOSE}(6) = 4$  
\text{DO 649 I = 7, 11}  
$\text{NCHOSE}(1) = 1$  
$\text{NCHOSE}(12) = 2$  
$\text{NCHOSE}(13) = 1$  
\text{CONT. ON PG 42}  

\text{PG 4 OF 63}  

10-375
THRUSTER SELECTION

FIRST CHECK TO SEE IF THERE IS AN ACCEPTABLE THRUSTER IN THE DATA BASE

\[
\text{FMAX} = \max(\text{AMAX1}, \text{ACTHST}, \text{ITHST})
\]

\[
\text{JIE} = \text{DB}(3)
\]

\[
\text{J1} = \text{DB}(12) + 1
\]

IF \(\text{THrust} \geq \text{FMAX}\)

GO TO 121

IF \(\text{J1} = \text{JIE}\)

GO TO 111

\[
\text{J1} = \text{J1} + 1
\]

CONT. ON PG 43

PG 42F 63
NO ACCEPTABLE THRUSTERS

11
ICHOOSE(1') = -1
RETURN

AT LEAST ONE ACCEPTABLE THRUSTER

121
CONTINUE

SELECT PNEUMATIC ATTITUDE AND CONTROL THRUSTERS
FIRST DETERMINE SET OF ALL THRUSTERS WHICH SATISFY THE INEQUALITY
THRUST GE ACTHST

1 = 1
J1 = IDB(12) * 1

THRUST = DATAB(6, J1)

IF (THRUST GE ACTHST)

GO TO 151

J2

IF (J1 EQ J1E)

CONT. ON PG 44
CONTINUE

\[ J_4 \]
\[ J_1 = J_1 + 1 \]
\[ \text{GO TO } 131 \]

\[ J_5 \]
\[ [\text{ACCEPT}(I) = J_1 \]
\[ I = I + 1 \]
\[ \text{GO TO } 141 \]

\[ 161 \]
\[ \text{CONTINUE} \]
\[ JMAX = I - 1 \]

\[ C \text{ CHOOSE THAT THRUSTER FROM THE ABOVE SET WHICH MINIMIZES THE} \]
\[ \text{C QUANTITY: } |\text{ABS(THRUST - ACTHST)}| \]

\[ J_6 \]
\[ J_1 = [\text{ACCEPT}(I) \]
\[ \text{THRUST} = \text{DATAB}(I+J_1) \]
\[ \text{DIFOLO} = |\text{ABS(THRUST - ACTHST)}| \]

\[ 171 \]
\[ [\text{CHOSE}(I) = \text{DATAB}(I+J_1) \]

\[ JSAVE = J_1 \]

\[ I IF I.I.EQ.(JMAX) \]
\[ T \]
\[ F \]

\[ \text{CONT. ON PG } 45 \]
DO 201
IRCCPT(JI)

IFNEO BSLIFHRUS -CTHST)

TO 0

COIFN.G 4
IFOE = 0 IFNE

I=JAVE 46

CNAT. ON PG
PG 45DF 83

CONT. ON PC 46

J7
GO TO 201

J8
I = I + 1

JI = IACCPT(JI)
THRUSTR: DATAB(JI)
DIFNEW: ABS(THRUST - ACTHST)

IF(DIFNEW .LE. DIFOLD)

IF(I .LT. IMAX)

GO TO 181

GO TO 191

GO TO 191

GO TO 201

19)
DIFOLD = DIFNEW

GO TO 171

201
J1 = JSAVE

T(I) = DATAB(I, J1)

10-379
SELECT PNEUMATIC TRANSLATIONAL THRUSTERS USING ABOVE PROCEDURE

1 = 1
J2 = 10B(12) + 1

211
THrust = DATA(6,J2).

IF (THrust. GE. TThST)
T

GO TO 231

221

IF (J2 .EQ. JIE)
F

GO TO 241

I = I + 1
GO TO 221

231
IACCPT(I) = J2

CONT. ON PG 47
J9

CONTINUE

MAX = I - 1
I = I + 1
J2 = IAMCPT(I)
THRU = DAB(6,J2)
DIFOLD = ABS(THRU - TTHST)

K1

ICHOSE(I2 = DAB(1, J2)

JSAVE = J2

IF(I .EQ. IMAX)

T

GO TO 281.

K2

I = I + 1

J2 = IAMCPT(I)
THRU = DAB(6,J2)
DIFNEW = ABS(THRU - TTHST)

IF(DIFNEW .LE. DIFOLD)

T

GO TO 271

F

IF(I .LT. IMAX)

T

CONT. ON PG 48
THRUSTERS HAVE BEEN SELECTED
SET NUMBER OF EACH TYPE OF THRUSTER
CHECK TO SEE IF CYCLE LIFE REQUIREMENT IS SATISFIED

IF(DATAB(7, J1) .LT. CLIFE)  
F  
T  
IF(DATAB(7, J2) .LT. CLIFE)  
F  
T  
IERR = IERR + 10
CONT. ON PG 49  
PG 49 DF 63

10-382
IERR = 1 IMPLIES THAT THE CYCLE LIFE OF THE ATTITUDE AND CONTROL THRUSTERS IS TOO SHORT. IERR = 10 IMPLIES THAT THE CYCLE LIFE OF THE TRANSLATIONAL THRUSTERS IS TOO SHORT. IERR = 11 IMPLIES THAT THE CYCLE LIVES OF BOTH THRUSTERS ARE TOO SHORT.

PRELIMINARY CALCULATIONS FOR SELECTION OF BIPROPELLANT ISOLATION VALVES AND FILTERS

\[ \text{RHOF} = 0.032 \]

\[ \text{RHOO} = 0.054 \]

\[ \text{WDOOT} = \frac{3 \cdot \text{ACTHST} + 2 \cdot \text{THST}}{1260 \cdot (1 + \text{XMR})} \]

\[ \text{WDOOT} = \frac{\text{WDOOT} \cdot \text{XMR}}{1260 \cdot (1 + \text{XMR})} \]

\[ \text{CDAISO} = \frac{\text{WDOOT} \cdot \text{SORT} \cdot \text{ISO} \cdot \text{RHOF} \cdot 1.29E-3}{1260 \cdot (1 + \text{XMR})} \]

\[ \text{RMAXF} = 50 \cdot \text{WDOOT} \]

\[ \text{RMAXO} = 50 \cdot \text{WDOOT} \]

C SET LAST EQUIPMENT INDICES

\[ J3E = 1DB114 \]

\[ J4E = 1DB114 \]

\[ J5E = 1DB115 \]

\[ J6E = 1DB115 \]

\[ J7E = 1DB14 \]

\[ J8E = 1DB2 \]

\[ J9E = 1DB116 \]

\[ J10E = 1DB116 \]

\[ J11E = 1DB15 \]

C DETERMINE HARDWARE INDICES

DO 66 (E = 1, 9)

IF (PIC1(I) .NE. 0) THEN

GO TO 67

CONT. ON PG 50

CONT. ON PG 63

10-383
CONT. ON PG 53
FUEL CIRCUIT AND OXIDIZER CIRCUIT ISOLATION VALVES AND FILTERS ARE ACCEPTABLE

\[ \text{PRELIMINARY CALCULATIONS FOR SELECTION OF PNEUMATIC REGULATOR} \]

CONT. ON PG 57
\[
\text{PTI} = \text{DATA}(\theta, J1) \\
\text{PFT} = \text{PTI} + 2.\times\text{OLPISF} + 2.\times\text{OLPIS}F \\
\text{POT} = \text{PTI} + 2.\times\text{OLPISO} + 2.\times\text{OLPIS}O \\
\text{PRED} = \text{MAX}(\text{PFT}, \text{POT}) \\
\text{WDOTPR} = 1.05\times1.02E-7\times20.\times\text{PREG}\times(\text{WDOTF/RHO}F \times \text{WDO}TO/RHOO) \\
\text{COAREG} = \text{WDO}TPR/\text{SQRT}(5600.\times\text{PREG}/1.27E4) \\
\]

\[
\text{IF} (\text{PREG.LT.DAT}A(8, J7) \text{OR.PREG.GT.DAT}A(9, J7) \text{OR.DAT}A(7, J7)).LT.\text{COAREG}) \\
\]

\[
\text{GO TO 81} \\
\text{C REGULATOR IS ACCEPTABLE} \\
\]

\[
\text{IPIC(7)} = \text{DAT}A(1, J7) \\
\text{IPIC(5)} = J7 \\
\]

\[
\text{PRELIMINARY CALCULATIONS FOR SELECTION OF PNEUMATIC ISOLATION VALVE} \\
\text{RHOPR} = 1.02E-7\times3000. \\
\text{COASSO} = \text{WDO}TPR/\text{SQRT}(200.\times\text{RHOPR}/1.29E-3) \\
\]

\[
\text{IF(DAT}A(7, J8).LT.\text{COASSO}) \\
\]

\[
\text{GO TO 81} \\
\text{C PNEUMATIC ISOLATION VALVE IS ACCEPTABLE} \\
\]

\[
\text{IPIC(8)} = \text{DAT}A(1, J8) \\
\text{IPIC(6)} = J8 \\
\]

\[
\text{PRELIMINARY CALCULATIONS FOR SELECTION OF FUEL TANK.OXIDIZER} \\
\]

CONT. ON PG 58
C TANK AND PNEUMATIC TANK

WP = I.1*TOTIMP/280.
WF = WP/I.1*XMR
VF = WF/RHOF
VFT = 1.1*VF
WO = WF*XMR
ACSHF = WF + WO
VO = WO/RHOO
VOT = 1.1*VO

VPRT = PFT*(VFT + VOT)/13000. - 2.*PFT

  T
  GO TO 81
  E8
  F

C FUEL TANK, OXIDIZER TANK AND PNEUMATIC TANK ARE ACCEPTABLE

ICHOOSE(9) = DATAB11,J9

ICHOOSE(10) = DATAB11,J10
ICHOOSE(11) = DATAB11,J11

TSIZE1 = DATAB23,J9 + DATAB23,J10 + DATAB23,J11

C SIZE PLUMBING AND CONNECTORS

PCWATE = 2*(DATAB23,J9)*NCHOOSE(9)+DATAB23,J10*NCHOOSE(10) + DATAB23,J11*NCHOOSE(11))

[I PIC(7) = J9
 I PIC(8) = J10
 I PIC(9) = J11
 M(12) = KK
 M(13) = LL
 M(14) = JJ
 M(15) = JJ
 M(16) = JJ

M(3) = J3
 M(4) = J4
 M(5) = J5
 M(6) = J6

CONT. ON PG. 59.
FUNCTION GAM(X)

IF (X.GT.1)

GO TO 2

Z=X

IF (Z.GT.0.)

T

GO TO 3

Z=Z+1.

GO TO 1

Z=X-1.

T1=Z+.5

TZ0=Z+5.
T1=TI0**2
T1=EXP(-TZ0)*T1=2.50662927465

GAMZ=T1*(1.+.76.18009173/(Z+1.)-86.50532033/(Z+2.)+24.01409822/(Z+3.)-1.231739516/(Z+4.))*1.20858003E-2/(Z+5.)-536382E-5/(Z+6.))

CONT. ON PG 2

PG 1 OF 2
FUNCTION CERF(Y)
DIMENSION B(28), A(26), AA(17), BB(19)

IF (Y > 4.0)
  t
GO TO 2

DATA AZERO / 3.88730365/

DATA A(1) / -1.38163142/
DATA A(2) / .647316404/
DATA A(3) / -3.05931024/
DATA A(4) / 1.386797472/
DATA A(5) / -0.35924745/
DATA A(6) / .023691751/
DATA A(7) / -0.0884736263/
DATA A(8) / .00308566171/

DATA A(9) / -0.01006386351/
DATA A(10) / .000307546328643/
DATA A(11) / -0.88261983E-04/
DATA A(12) / .23845096E-04/
DATA A(13) / -0.50791002E-05/
DATA A(14) / .146597217E-05/
DATA A(15) / -0.033515993E-05/
DATA A(16) / .007280579E-05/

DATA A(17) / -0.01505791E-05/
DATA A(18) / -0.297094742E-08/
DATA A(19) / -0.56021739E-09/
DATA A(20) / -0.10131623E-09/
DATA A(21) / -0.17506504E-10/
DATA A(22) / -0.029103813E-10/
DATA A(23) / -0.4653264E-12/
DATA A(24) / -0.7164815E-13/

DATA A(25) / -0.1063749E-13/
DATA A(26) / -0.125467E-14/
DATA B(27) / .0/
DATA B(28) / .0/
DATA AAZERO / 1.970705272/
DATA AA(1) / -0.14339740271775/
DATA AA(2) / -0.0029736189220261/
DATA AA(3) / -0.98035180E-05/

CONT. ON PG 2
```
C
DELMU=2190.
C
EXPENDABLES STD. DEV. INCR.

C
DELSIG=365.
C

*** SYS PARAM SPECIFICATION ***

C
R-SHIFT NCHOSE AND COLUMNS OF
DATA BY I BEGINNING WITH THE
THIRD SUB-SYS

C
JMIN=NI(2)+1
JMAX=NIMSS)

C
INITIALIZE

DO 130 I=1,6

130
Z(I)=DATAB(I,JMIN)

C
SHIFT NCHOSE

WZ=NCHOSE(JMIN)

DO 140 J=JMIN,JMAX

C
SHIFT DATAB

WY=NCHOSE(J+1)
NCHOSE(J+1)=NZ
NZ=WY

DO 140 I=1,6

C
CONTINUE

C
INSERT EXPENDABLES PARAMETERS

CONT. ON PG  6
```

10-407
DATA1: JMIN = 3. * (ACSWP + TNKWT) / TPRIM
DATA2: JMIN = 4.
DATA3: JMIN = EU
DATA4: JMIN = ESIG
DATA5: JMIN = DELRU
DATA6: JMIN = DELSIG

C SET MAX NUM OF REDUNDANT ELE.

MAX(JMIN) = MAXEXP
SAVMAX(JMIN) = MAX(JMIN)

C RESET N(K)

DO 150 K = 2, NSS

150 N(K) = N(K) + 1
JMAX = JMAX + 1

C SWEEP DATAB AND COMPUTE MODEL PARAMETERS

DO 160 J = 1, JMAX

MODL = INT(DATAB(2, J) + 1)

C FOR MODEL TYPE 4

IF (MODL .EQ. 4) THEN
T

F

GO TO 160.

C MAX NUM OF REDUNDANCIES

NMX(J) = DATAB(6, J) + 1

NMX(J) = NMX(J) - NCHOSE(J)
SAVMX(J) = NMX(J)

CONT. ON PG 7
DATA(3,J)=DATA(3,J)*FC  
DATA(4,J)=LAMS*1.0E-09

CONTINUE

C TIME STEP IMCR.

DELH=TRUNC/FLOAT(TRAUC-1)

CALCULATE RELIABILITY MATRIX

LIM=NIMSS)

IF (IDEBUG .EQ. 0)

GO TO 165

WRITE(6,3000)IMAX(J),J=1,LIM)

WRITE(6,3000)ICHOOSE(J),J=1,LIM)

WRITE(6,4000)(1,DATA(I,J),I=1,6),J=1,LIM)

WRITE(6,4000)(1,DATA(I,J),Sch=1,6),J=1,LIM)

FORMAT(61017,/)
INITIALIZATION OF PARAMETERS BEFORE ENTRY TO THE REDUNDANCY ALLOCATION PROCEDURE

CONT. ON PG 16
COMPUTE INITIAL RELIABILITY FOR SINGLE AND DOUBLE STRING SYSTEMS

DO 320 I = 1, ITURNC
    ROLD(I) = R(I) + 1
    IF (RFNL .GE. 0.99) THEN
        DO TO 305
    ELSE
        ROLD(I) = ROLD(I) * EXP(-1.0 * DELH * FLOAT(I-1) / ALPHA)*1.6
    END IF
    DO 310 J = 2, JMAX
        ROLD(I) = ROLD(I) * R(J)
        R(I) = 1.0 - (1.0 - ROLD(I))**2
    END DO 310
    CONTINUE

CALL OSF (DELH, ROLD, ITURNC)
MMOLD = RFIXED * Z(I, ITURNC)

CONT. ON PG 17
CALL QSF (DELA, RLZ, ITRUNC)

OSMDO = RFIXED = Z (ITRUNC)

CK MRDOLDER AGAINST USER SPECI
GO TO REDAP
ALSO RETURN POINT FOR REDAP

OLDRHO = -1.0

IF (IOS.EQ.0)

T

GO TO 350

IF (OSMDO .LT. SPECI)

F

T

GO TO 390

GO TO 351

350

IF (MRDOLDER .LT. SPECI)

F

T

GO TO 390

CONT. ON PG 18

PG 17 OF 27
COMPUTE SUBSYS RELIABILITIES

DO 363 K = 1, KSS

JMIN = JMAX + 1
JMAX = N(K)
SSREL(K) = 1.0

DO 362 J = JMIN, JMAX

SSREL(K) = SSREL(K) * R(I(TRAUNC, J))

CONTINUE

MISSION EQUIPMENT RELIABILITY

SSREL(6) = RFNL

COMRESS NCHOOSE AND ADD RED.

DO 360 J = NMIN(2) - 1

DO 370 J = 1, JMIN

NCHOOSE(J) = NCHOOSE(J) + WRI(J)

CONT. ON PG 19
18 - 390
NCHOSE(J) = NCHOSE(J+1) + NRI(J+1)

C EXPENDABLES INFO RETURN

TPR1 = TPR1 + FLOAT(3*WR(JMIN))
PRINT 1000, TPR1

1000 FORMAT (1X,6H TPR1 = E11.4)
PRINT 1100

1100 FORMAT (1X,2HM = E11.4)
ITRUNC = ITRUNC

IF (IDS .EQ. 0)

GO TO 385

MMOD = DSMOD
ROLD(ITRUNC) = R0(ITRUNC)
DO 381 J = 1, JMAX

381 NCHOSE(J) = NCHOSE(J) + 2

385 RETURN

CONT. ON PG 20
MAIN REDUNDANCY ALLOCATION PROCEDURE
(REDAP)

IF MAX NUM RED EXCEEDED, RETURN
OTHERWISE CONTINUE PROCEDURE

IF (NSR .GE. NSMX)

T

GO TO (490,510) [RTN]

SELECT MODULE TO ADD A RED. IF
J .GE. JMAX GO TO SYS UPDATE
PROCEDURE.

DO 440 J = JMIN, JMAX

IF (NRI(J) .GE. NMX(J))

T

GO TO 440

MODL = DATAB(2,J) + 1

(F (MODL .EQ. 3) .AND. (NRI(J+1) .GE. NMX(J+1)))

CONT. ON PG 21

PG 208F 27
SUBROUTINE RIMODIJ.DELH.ITRUNC.NT.IADD.IOPT

COMMON /DBCOM/ R(31),NR(60),R(31),60),X(31),RDI(31),ROUM(31),SAVR(31),SAVR(31),XNew(31),NM0(60),SAVX(60),COST(60),OUM(32),13)

COMMON /CHOSE/ COSTM(5,60),CHSIR(5,60),CCHSIE(60),CCHSIE(60),CCHSIE(60),SYSPAM(60),SKD(7,60),TMH(4,60)

COMMON /PRTCOM/ AM,AN,BF,BS,CDPI(7,2),CISTAR,CTOT,OOTE,DE,DRIT,ECBSTR,FEENS,FEOPS,FEER,OSR,IREL,ITRUNC,MMOSL,NAME1(3,60),DPS,PAYINV,PAYQ,PARY,PAYT,OMBP,OMPA,OMPM,PU,PWR(6),OCP,OCR,ROLD(60),SABMWT,SSRTP,SATINV,SATR,SEIP,SEIR,SKTR(6),SSREL(6),SUBE(7),SUBF(7),SUBU(7),SAI,RAU(5,6),ROUM,TC,IE,TF,TOOLR,TOOLU,TOPS,TRUNC,TS,TIT1,VLUE(6),VOL(6),WEIGHT(6),XLMI,XME,XMEINV,XME,XMEL,XMEYL

REAL LAM,LAMBAR,LAMS

SUBROUTINE RIMOD
PURPOSE
TO COMPUTE THE RELIABILITY FUNCTION FOR MODULE J AFTER REDUNDANCIES ARE ADDED TO THE MODULE.
USAGE
CALL RIMODJR.NR.J.DELH.ITRUNC.NT.IADD.IOPT

DESCRIPTION OF PARAMETERS

J - INPUT MODULE NUM
DEL HI DELTA TIME, THE TIME INCREMENT
ITRUNC - THE NUM OF TIME POINTS
NT - INPUT OPTION PARAMETER
IADD - INPUT OPTION PARAMETER
IOPT - INPUT OPTION PARAMETER

REMARKS
OPTION PARAMETER VALUE ACTION

NT = 1 ONLY COMPUTE RELIABILITY AT TRUNCATION TIME. RETURN VALUE IN R (TRUNC).
ITRUNC COMPUTE RELIABILITY AT EACH TIME RETURN VALUES IN R.
IADD = 0 ADD NO REDUNDANCIES BEFORE COMPUTING THE RELIABILITY FUNCTION.
IADD = 1 ADD REDUNDANCIES BEFORE COMPUTING THE RELIABILITY FUNCTION.

CONT. ON PG 2
INC THE RELIABILITY FUNCTION.

GLOBAL VARIABLES PASSED THOUGH COMMON
R - THE RESULTING RELIABILITY FUNCTION
NR - INPUT VECTOR OF THE NUM OF REDUNDANCIES BY MODULE
NCHOS - INITIAL NUM OF ELEMENTS IN MODULES
SYSPAR - MATRIX OF MODEL PARAMETERS

SYSPAR(2,J)= MODEL ID FOR J-TH MODULE
FOR FURTHER DESCRIPTION SEE COMMENTS PRECEDING THE PARTICULAR MODEL OF INTEREST.

ROOT2:SORT(2,0)
MOD=INT(SYSPAR(2,J)+1)
GO TO (10,90,120,160,10), MOD

MODELS 1 AND 5
VARIABLES SIZE ORIGIN DEFN
LAMS 1 INT SENSE/SWITCH FAILURE RATE
LAM 1 INT FAILURE RATE
Q 1 INT DORMANCY FACTOR
DC 1 INT MODULE DUTY CYCLE
MI 1 INT NUM OF STANDBY ELEMENTS

NI 1 INT NUM OF ACTIVE ELEMENTS
SYSPAR I,J GLOBAL MODEL PARAMETERS FOR J-TH MODULE
I=3 VALUE OF LAM
I=4 VALUE OF LAMS
I=5 VALUE OF Q
I=6 VALUE OF DC

CONT. ON PG 3
C INCR IN STANDBY MODE

A2
20
MI=NODE0
R1=NRED+1ADD

C CALCULATION OF MODEL CONSTANTS

A3
30
(F IHI.NE 0)

0=0=0=LAM5/LAM

DOB=0/10C+(1.-DC)=0

LAMBAR=LAM*(DC*(1.-DC)=0)

C-----------COMPUTATION OF RELIABILITIES-----------

L1=RRED
L12=L1M-1

DO DO (IN0=1,NT)

DO FOR EACH TIME POINT IN Descending ORDER, NT TO 1

I=ITRUNC+1-IND
TIME=DELH=FLOAT(1-1)

SUM ACCUMULATES RELIABILITY

SUM=1.0

C EXPONENTIAL CONSTANT

CONT. ON PG 5
IF (Z.LT.0.0)

T

AN = 1 - AN

COMPUTATION OF CUMULATIVE BINOMIAL PROBABILITIES

Z = AN

AN = AN + AN
SUM = AN

IF (LIMIT.ED.0)

T

GO TO 110

Z = (1 - Z)/Z

L = LIMIT + 1

DO 100 L = 1, L = LIMIT

L = L - 1
AN = AN * (FLOAT(INT(-L)/FLOAT(L+1))) * Z

100 SUM = SUM + AN

CONT. ON PG 10

P 9 OF 14
IF \( |Z| > 4.0 \)

\[
AN = 1.0 - AN
\]

\[
AN = 0.5 \times (1.0 - AN)
\]

IF \( Z \leq 0.0 \)

\[
AN = 1.0 - AN
\]

\[
Z = \exp(-1.0 \times \text{BCYC} \times (\text{DEL} \times (K - 1))/\text{AB}) \times 88)
\]

\[
\text{LLLIM} = \text{LIM} + 3
\]

\[
\text{AAA} = Z \times \text{MC}
\]

\[
\text{SUM} = \text{AAA}
\]

\[
Z = (1.0 - Z)/Z
\]

DO 125 LLL = 1, LLLIM

\[
L = \text{LLL} - 1
\]

\[
\text{AAA} = \text{AAA} \times (\text{FLOAT} (\text{MC} - L) / \text{FLOAT} (L + 1)) \times Z
\]

125

\[
\text{SUM} = \text{SUM} + \text{AAA}
\]

\[
Z = \text{SUM} \times AN
\]

COMPUTATION OF CUMULATIVE BINOMIAL PROBABILITIES

\[
\text{AN} = Z \times \text{MI}
\]

\[
\text{SUM} = \text{AN}
\]

CONT. ON P. 12
IF (Z .LT. 0.0)

T

(R(K) = 1.0 - R(K))

CONTINUE

RETURN

END
SUBROUTINE OSF(H,Y,Z,NDIM)

PURPOSE

TO COMPUTE THE VECTOR OF INTEGRAL VALUES FOR A GIVEN
EQUIDISTANT TABLE OF FUNCTION VALUES.

USAGE

CALL OSF(H,Y,Z,NDIM)

DESCRIPTION OF PARAMETERS.

H - THE INCREMENT OF ARGUMENT VALUES.

Y - THE INPUT VECTOR OF FUNCTION VALUES.

Z - THE RESULTING VECTOR OF INTEGRAL VALUES. Z MAY BE
IDENTICAL WITH Y.

NDIM - THE DIMENSION OF VECTORS Y AND Z.

REMARKS

NO ACTION IN CASE NDIM LESS THAN 3.
SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED

NONE

METHOD

BEGINNING WITH Z(1)=0, EVALUATION OF VECTOR Z IS DONE BY
MEANS OF SIMPSONS RULE TOGETHER WITH NEWTONS 3/8 RULE OR A
COMBINATION OF THESE TWO RULES. TRUNCATION ERROR IS OF
ORDER H^5 (I.E. FOURTH ORDER METHOD). ONLY IN CASE NDIM=3
TRUNCATION ERROR OF Z(2) IS OF ORDER H^4.

FOR REFERENCE, SEE

(1) F. B. HILDEBRAND, INTRODUCTION TO NUMERICAL ANALYSIS,
McGRAW-HILL, NEW YORK/TORONTO/LONDON, 1956, PP. 71-76.

(2) R. ZURMUEHL, PRAKTISCHE MATHEMATIK FUR INGENIEURE UND
PHYSIKER, SPRINGER, BERLIN/GOTTINGEN/HEIDELBERG, 1963,
PP. 214-221.

DIMENSION Y(1),Z(1)

IF(NDIM-5)7.8:1

C NDIM IS GREATER THAN 5. PREPARATIONS OF INTEGRATION LOOP

SUM1=Z(1)+Y(1)

SUM1=SUM1+SUM1

CONT. ON PG 2
SUM1=HT=(Y(1)+SUM1+Y(3))
AUX1=Y(4)+Y(4)
AUX1=AUX1+AUX1
AUX1=SUM1+HT=(Y(3)+AUX1+Y(5))
AUX2=HT=(Y(1)+3.875*(Y(2)+Y(5))+2.625*(Y(3)+Y(4))+Y(6))
SUM2=Y(5)+Y(5)
SUM2=SUM2+SUM2

SUM2=AUX2-HT=(Y(4)+SUM2+Y(6))
Z(1)=0
AUX=Y(3)+Y(3)
AUX=AUX+AUX
Z(3)=SUM2-HT=(Y(2)+AUX+Y(4))
Z(4)=SUM2
Z(4)=SUM2

IF(IN=5.5=2
C INTEGRATION LOOP

DO 4 I=1,NOIM=2
SUM1=AUX1
SUM2=AUX2
AUX1=Y(I-1)+Y(I-1)
AUX1=AUX1+AUX1
AUX1=SUM1+HT=(Y(I-2)+AUX1+Y(I))
Z(I-2)=SUM1

IF(I-NOIM)=3=6=6

AUX2=Y(I)+Y(I)

AUX2=AUX2+AUX2
AUX2=SUM2+HT=(Y(I-1)+AUX2+Y(I+1))
C-\( I-I \)\( 1.25 \)\( I-I \)\( ZI-I \)\( ZI \)\( I-I \)\( AUX1 \)\( ZI-I \)\( ZI-I \)\( AUX2 \)\( RETURN \)

C END OF INTEGRATION LOOP.

C END OF INTEGRATION LOOP.

C END OF INTEGRATION LOOP.

C END OF INTEGRATION LOOP.

C NDIM IS EQUAL TO 4 OR 5

SUM2=1.125*HT=(Y1+Y(2)+Y(2)+Y(2)+Y(3)+Y(3)+Y(3)+Y(4))

SUM1-Y(2)+Y(2)
SUM1=SUM1+SUM1
SUM1=HT=(Y(1)+SUM1+Y(3))
Z(1)=0.
AUX1=Y(3)+Y(3)
AUX1=AUX1+AUX1
Z(2)=SUM2-HT=(Y(2)+AUX1+Y(4))

CONT. ON PG 4
IF(IND(M-5) <= 0.9)

\( \text{AUX1} = Y(4) \times Y(4) \)

\( Z(5) = \text{SUM1} + \text{HT} \times (Y(3) + \text{AUX1} \times Y(5)) \)

IF(NOM IS EQUAL TO 3)

\( \text{SUM1} = \text{HT} \times (1.25 \times Y(1) + Y(2) + Y(2) - 25 \times Y(3)) \)

\( \text{SUM2} = Y(2) + Y(2) \)

\( \text{SUM2} = \text{SUM2} + \text{SUM2} \)

\( Z(3) = \text{HT} \times (Y(1) + \text{SUM2} + Y(1)) \)

\( Z(1) = 0 \)

\( Z(2) = \text{SUM1} \)

RETURN

RETURN

END
''IF (ARRAY(K+11*I).EQ.0.)''

''GO TO 160''

''NTABL=NTABL+1''

''XSRT(NTABL)=ARRAY(K+11*I)''

''GRANL(NTABL)=ARRAY(K+12*I)''

''TLPTL(NTABL)= (ARRAY(K+9*I)+ARRAY(K+10*I)).*NCHOSG(I)''

''IF (ARRAY(K+10*I).NE.0.)''

''MUX=1''

''ANOLH=ANOLH+ARRAY(K+5*I).*NCHOSG(I)''

''IF (NEWFL.EQ.0.)''

CONT. ON PG 6
C  WE NEED NTABH

5  A5
   NTABH = NASEQ

C  WE NEED NTABH

K = -1
   DO 230 I = 1, NTABH

TTCFM = TTCFM + ARRAY(K+2, I) + ARRAY(K+3, I) + ARRAY(K+4, I)
       TTCFL = TTCFL + ARRAY(K+4, I)

IF (ARRAY(K+7, I) .EQ. 0.) T
   GO TO 210
F

NTABH = NTABH + 1

NSRT(NTABH) = ARRAY(K+7, I)
GRAND(NTABH) = ARRAY(K+8, I)
TLPT(NTABH) = ARRAY(K+9, I) + ARRAY(K+8, I)

210

IF (ARRAY(K+6, I) .NE. 0.) F
   NTABH = NTABH + 1
T
   AUX = 1

CONT. ON PG 7

PG 6 OF 22
IF (ARRAY(I+I1), EQ. 0.)

GO TO 220

IF (ARRAY(I+10), NE. 0.)

F

T

MUX = 1

ANOLH = ANOLH - ARRAY(I+5)

230

ANOLL = ANOLL - ARRAY(I+9)

6

240

CONTINUE

C ANOLH - NO OF ANOL PTS IN HIGH TAB

C ANOLL - NO OF ANOL PTS IN LOW TAB

CONT. ON PG 8

PG 2 OF 22
COPY(2*1) = TLAND
COPY(3*1) = SSR
COPY(4*1) = WOLMAX
BITRATE = WOLMAX * TLAND * SSR
PRINT 1000, BITRATE

1000
FORMAT (13H BITRATE (1) = \*E11.4)

GO 370 AM = 1, 10

N = RR - 1
TT = 2 * M * 7.0125

IF TT .GE. BITRATE

GO 10 380

CONTINUE

IERROR = IERROR
IERROR = IERROR + IERROR

C IERROR = 100 BIT RATE TOO LARGE

RETURN

380
BITRATE = TT

BITRAT(1) = BITRATE
JL = 0

C ORDER LOW SAMPLE RATE

CONT. ON PG. 14
IF (IN'TABL.EQ.1)  
T
GO TO 420

300
CONTINUE

CALL ORDER (IN'TABL,X5AT,EPTL,GRNL,X2,MEIDAN)

JL=JL+1

IF (JL.EQ.2)  
T
GO TO 420

DO 410 I=1,MEIDAN

400
IF (X5RT(I).LE.XR2)
F

X5RT(I):=X5RT(I)/2.
TLPTL(I):=TLPTL(I)**2.

CONT. ON PG 15

PG 10F 22
SFR = HIGHEST RATE IN LOW RATE TABLE

SFL = SFR / SFR

SFL = SUB FRAME LENGTH

IF (SFL ≤ 2^N)

T

GO TO 440

F

IF (SFL ≥ 2^N)

T

GO TO 440

F

GO TO 440

DO 430 N = 5, 7

CONT. ON PG 16

PG 135F 22
IF (SCSFL .NE. 0.)

T

IER4 = 1000

T

IER5 = IERR + IERR4

TOTCM = TOTCM + 1.5

GO 450  NN = 1.100

N = NN

IF (TOTCM .LE. 2. =M) F

GO TO 470

470

CONTINUE

460

CONTINUE

TOTCM = 2. =M

COP1(1,1) = TOTCM
CWDLN = N
N = NCONF(3)

GO TO (480, 500), N

CONT. ON PG 18

PG 128  22

10-464
IF (ARRAY[K+7:1] .EQ. 0.) T
GO TO 40
NTABH = NTABH + 1
KSR(TABH) = ARRAY(K+7:1)
ORAMH(TABH) = ARRAY(K+8:1)
TLPTH(TABH) = ARRAY(K+5:1) + ARRAY(K+6:1)
40
IF (ARRAY(K+6:1) .NE. 0.) T
MUX = 1
IF (ARRAY(K+11:1) .EQ. 0.) T
GO TO 50
CONT. ON PG 4
IF (INMAX.NE.1.024E0)

T

STRFL = 1

C ORDER TELEM POINTS BY SAMPLE RATE - HIGH

IF (INTABH.EQ.1)

T

GO TO 100

A2

70 CONTINUE

CALL ORDER (INTABH, HSRT, TLPTH, GRANH, XM2, MEDIAN)

JL = JL + 1

IF (JL.EQ.2)

T

GO TO 100

A2

DO 90 I = 1, MEDIAN

CONT. ON PG 6

PG 5 OF 14

10-477
IF (HSRT(I) .LE. XM2)  
T  
GO TO 90  

MSRT(I) = MSRT(I)/2. 
TLPTH(I) = TLPTH(I)/2. 
GO TO 90  

CONTINUE 
GO TO 70 

A3  
100  
SSR = HSRT(I)  
C  
SSR = MAIN FRAME RATE  

IF (NTABM .EQ. 1)  
F  

JL = 0  

CONT. ON PG 7  

PG 6 OF 14
CALL ORDER (INTABH, GRANM, TLPTH, HSRT, XM2, MEDIAN)

JL = JL + 1

IF (JL .EQ. 2)

GO TO 140

GO 130 I = 1, MEDIAN

IF (GRANHI(I) .LE. XM2)

GO TO 130

GRANHI(I) = GRANHI(I) / 2

TLPTH(I) = 2 * TLPTH(I)

GO TO 120

130 CONTINUE

GO TO 110
A9

SFR = XSRT(1)

SFR = HIGHEST RATE IN LOW RATE TABLE

SFL = SFR / SFR

SFL = SUB FRAME LENGTH

N = 5

IF (SFL.LE.2.N) T

GO TO 260

IF (SFL.GE.2.N) F

GO TO 260

DO 250 N = 5, 7

NPI = N + 1

IF (SFL.GE.2.N AND SFL.LE.2.NONPI) F

CONT. ON PG 13

PG 120F 14
END
# Systems Cost/Performance Analysis (Study 2, 3) Final Report
### Report Title
**Systems Cost/Performance Analysis (Study 2, 3) Final Report**  

### Report No.
ATR-75(7363)-3, Vol. III

### Publication Date
31 March 1975

### Security Classification
Unclassified

## Military and Government Offices

<table>
<thead>
<tr>
<th>Office</th>
<th>Address</th>
<th>Attn:</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>NASA Scientific and Technical Information Facility</td>
<td>P. O. Box 33, College Park, Maryland 20740</td>
<td>Mr. F. J. Cepollina</td>
<td>(3)</td>
</tr>
<tr>
<td>NASA Headquarters</td>
<td>Washington, D. C. 20546</td>
<td>Mr. R. D. Kramer, PD13</td>
<td>(1)</td>
</tr>
<tr>
<td>NASA, MSFC</td>
<td>Huntsville, Alabama 35812</td>
<td>Mr. R. D. Kramer, PD13</td>
<td>(25)</td>
</tr>
<tr>
<td>NASA, JSC</td>
<td>Houston, Texas 77058</td>
<td>Mr. J. P. Loftus, Jr., Code AT</td>
<td>(1)</td>
</tr>
<tr>
<td>NASA, KSC</td>
<td>Florida 32899</td>
<td>Mr. H. E. McCoy, Code FP-B</td>
<td>(1)</td>
</tr>
</tbody>
</table>

### AFR 80-45 Distribution Statement

☐ NO DISTRIBUTION STATEMENT  
(Classified documents only)

☐ A. APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED  

Approved by: [Signature]

Date: 17 April 1975

(for the Aerospace Corporation)

(if list comprises two or more sheets, complete this signature block on last sheet only)

Sheet 1 of 1