D82A

[redacted]



THE AEROSPACE CORPORATION

(NASA-CR-140027) OPERATIONS ANALYSIS (STUDY 2.1). PROGRAM SEPSIM (SOLAR ELECTRIC PROPULSION STAGE SIMULATION) (Aerospace Corp., El Segundo, Calif.) 38 p HC $5.00 CSCL 22A

N74-33260

Unclas

G3/30 17143

Report No. ATR-74(7341)-2

# OPERATIONS ANALYSIS ( STUDY 2.1)

# PROGRAM SEPSIM (SOLAR ELECTRIC PROPULSION STAGE SIMULATION)

Prepared by T. J. Lang

Mission Analysis Department
Guidance and Flight Dynamics Subdivision
Guidance and Control Division
Engineering Science Operations

28 June 1974

Contract No. NASW-2575

Systems Engineering Operations
THE AEROSPACE CORPORATION
El Segundo, California

Report No.
ATR-74(7341)-2

OPERATIONS ANALYSIS (STUDY 2.1)
PROGRAM SEPSIM (SOLAR ELECTRIC
PROPULSION STAGE SIMULATION)

Prepared by

T. J. Lang

T. J. Lang
Mission Analysis Department

Approved by

H. K. Karrenberg

H. K. Karrenberg, Head
Mission Analysis Department
Guidance and Flight Dynamics
Subdivision
Guidance and Control Division
Engineering Science Operations

Robert R. Wolfe

Robert R. Wolfe
NASA Study 2.1 Director
Advanced Mission Analysis Directorate
Advanced Orbital Systems Division
Systems Engineering Operations

# ABSTRACT

Program SEPSIM is a FORTRAN program which performs deployment, servicing and retrieval missions to synchronous equatorial orbit using a Space Tug with a continuous low thrust upper stage known as a Solar Electric Propulsion Stage (SEPS). The SEPS ferries payloads back and forth between an intermediate orbit and synchronous orbit, and performs the necessary servicing maneuvers in synchronous orbit. The Tug carries payloads between the Orbiter and the intermediate orbit, deploys fully fueled SEPS vehicles, and retrieves exhausted SEPS vehicles when and if required.

The program is presently contained in subroutine form in the Logistical On-orbit VEhicle Servicing (LOVES) Program, but can also be run independently with the addition of a simple driver program.

## CONTENTS

# 1. INTRODUCTION

SEPSIM (Solar Electric Propulsion Stage Simulation) is the name given to a group of FORTRAN subroutines contained in the Logistical On-orbit VEhicle Servicing (LOVES) Program. The function of these subroutines is to determine the performance capability of a specified Space Tug/Solar Electric Propulsion Stage (SEPS) combination for any number of deployment/servicing/retrieval missions to synchronous equatorial orbit.

The high thrust Space Tug operates in the low altitude regime while the low thrust SEPS operates at high altitudes (above 8000 n mi) only. The payload to be deployed is carried by the Tug from the Orbiter to an intermediate orbit. The payload to be retrieved is carried from synchronous orbit by the SEPS to the same intermediate orbit, where the two vehicles rendezvous and exchange payloads. The SEPS delivers its new payload to synchronous equatorial orbit, while the Tug returns its new payload to the Orbiter. The constituents of the deployed payload weight are distributed as required in synchronous orbit by phasing maneuvers performed by the SEPS. The SEPS is assumed to operate in the "ground-based mode"; that is, the SEPS is initially launched with a specified amount of fuel, and when that fuel is exhausted (or nearly so) the SEPS is either returned to Earth for refurbishment or abandoned in space.

In normal usage a call is made to the executive subroutine of SEPSIM containing information on the mission to be flown (e.g., weight to be deployed, weight to be retrieved, servicing maneuvers to be performed in synchronous orbit). The program determines the optimal (minimum SEPS fuel) intermediate orbit at which changeover should occur based upon the characteristics of the Tug and SEPS vehicles which are stored in a COMMON area. Program outputs include intermediate orbit altitude and inclination, SEPS ascent and descent times, and SEPS fuel remaining at the end of the mission.

SEPSIM is designed to be called for a number of missions in series. This is the manner in which the routines are used in the LOVES Program and, if it is desired to run SEPSIM separately from LOVES, a simple driver can be coded for this purpose. In this usage, the time and fuel remaining on a SEPS vehicle are monitored and Tug flights are automatically initiated to launch new SEPS vehicles as required and return expended ones, if required. Missions which are found to be within the capability of the Tug alone are performed without the aid of SEPS, and payloads which cannot be boosted to a circular orbit of at least 8000 n mi altitude by the Tug are not launched since the SEPS is not allowed to operate in the region of the Van Allen radiation belts (due to rapid solar cell degradation).

The overall method of solution is outlined in Section 2 and program usage is discussed in Section 3. Appendix A contains simplified program flow-charts, while more detailed logic flow can be derived from the program listing in Appendix B. A sample output is presented in Appendix C which displays various options which are available to the user.

## 2. METHOD

The simulation of Space Tug/SEPS operation is performed using the following set of subroutines: (flow charts for the two main subroutines are included in Appendix A and listings for all subroutines are contained in Appendix B)

SEPX: This is the executive routine of SEPSIM. In it the decisions are made as to when a new SEPS must be launched or retrieved, whether the given deploy and retrieve payloads can be handled in a single Tug/SEPS encounter, or whether additional Tug flights are required to support the mission. This logic has been kept separate from the SEPS performance equations to facilitate logic changes to incorporate mission modes other than the "ground based mode" which is presently baselined.

SEPIM: In this routine the performance of the Tug/SEPS combination is computed by calls to more detailed performance subroutines. The delivery, servicing, and retrieval of payloads is accomplished and the fuel and time remaining on the SEPS are decremented accordingly.

TUGCP: This subroutine calls the appropriate Space Tug configuration to determine the Tug capability. Presently only a single stage cryogenic Tug performance model (CRYO1) is available.

CRYO1: In this routine the $\Delta V$ capability of a single stage cryogenic Space Tug carrying the required deploy and retrieve payloads is computed. The Tug is initially either filled with propellant or filled to the gross weight limit of the Shuttle. Impulsive $\Delta V$'s are assumed with a specified performance reserve to allow for finite burn effects and propellant margin.

INTORB: From the $\Delta V$ which can be supplied by the Tug, this routine computes the optimal (minimum SEPS fuel required) changeover orbit altitude and inclination. This is done by a table lookup procedure, since the optimization study for a synchronous equatorial mission orbit has been previously accomplished in a manner which is dependent only upon the $\Delta V$ which can be supplied by the Tug.

SEPDV: Given the optimal changeover orbit calculated by subroutine INTORB, SEPDV determines the $\Delta V$ (and corresponding mass ratio) which must be expended by the SEPS. Edelbaum's* simplified equations for low thrust vehicles are used in calculating the $\Delta V$ required. Again, a performance reserve factor is provided.

PLUPD: With the mass ratio from subroutine SEPDV, this routine decrements both the propellant on board and the SEPS remaining lifetime for each payload carried up or down by the SEPS.

FAZS: This subroutine performs the on-orbit servicing maneuvers using the SEPS and decrements the fuel and lifetime of the SEPS accordingly.

*Edelbaum, T. N., "Propulsion Requirements for Controllable Satellites", ARS Journal, pp. 1079 - 1089, (August, 1961).

## 3. USAGE

### 3.1 GENERAL INPUT:

SEPS configuration:

| | |
|---|---|
| MS | SEPS structural weight (lbs) |
| E | solar electric propulsion efficiency |
| P | solar cell power used for thrust (watts) |
| SISP | SEPS specific impulse (sec) |
| SEPK | = 0. SEPS is expendable when fuel is depleted |
| | = 1. SEPS is reusable |
| SR | SEPS performance reserve factor |
| TSEP | thrusting time available on a fully fueled SEPS (days) |
| MPT* | mass of propellant onboard (lbs) |
| TLEFT* | corresponding thrusting time (days) |

Tug configuration:

| | |
|---|---|
| TYPE | Hollerith description of Tug |
| NSTG | number of stages (present code accepts only NSTG = 1) |
| SPAR (3) | indicates whether each stage is solid (=1) or liquid (=0) |
| | (present code assumes SPAR = 0, 0, 0) |
| WS (3) | structural weight of each stage (lbs) |
| WPA (3) | allowable propellant weight for each stage (lbs) |
| EISP (3) | effective ISP for each stage (sec) |
| REUSE (3) | indicates whether each stage is reusable (=1.) or |
| | expendable (=0.) |
| WGA | maximum gross weight of Tug and payloads |
| | which is allowed by the Orbiter (lbs) |
| TR | Tug performance reserve factor |

*If the SEPS is to be fully fueled initially, then the program calculates these values from the TSEP input.

Servicing information:

| | |
|---|---|
| NSERV | number of on-orbit satellite servicing legs (maximum = 10) |
| DTHETA (10) | angular phasing maneuver performed for each service leg (deg) |
| MPLS (10) | payload weight carried on each servicing leg |

3.2 SPECIFIC INPUT:

| | |
|---|---|
| MPLA | weight of payload to be deployed (lbs) |
| MPLB | weight of payload to be retrieved (lbs) |
| ERFLG | = 0 do not erase previous maneuver |
| | = 1 erase previous maneuver |
| NEXIT | set to zero on DATA card of driver program. |

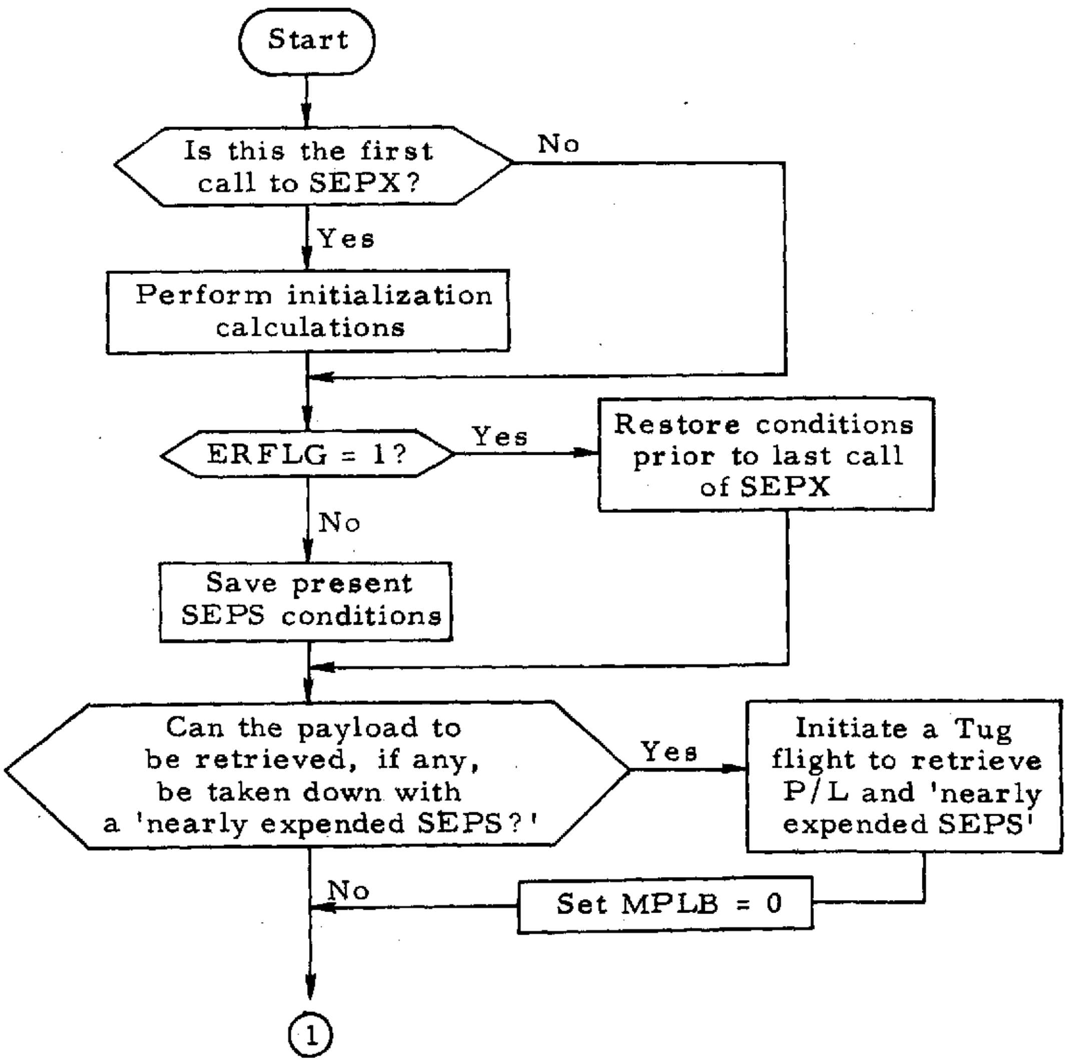
3.3 OUTPUT:

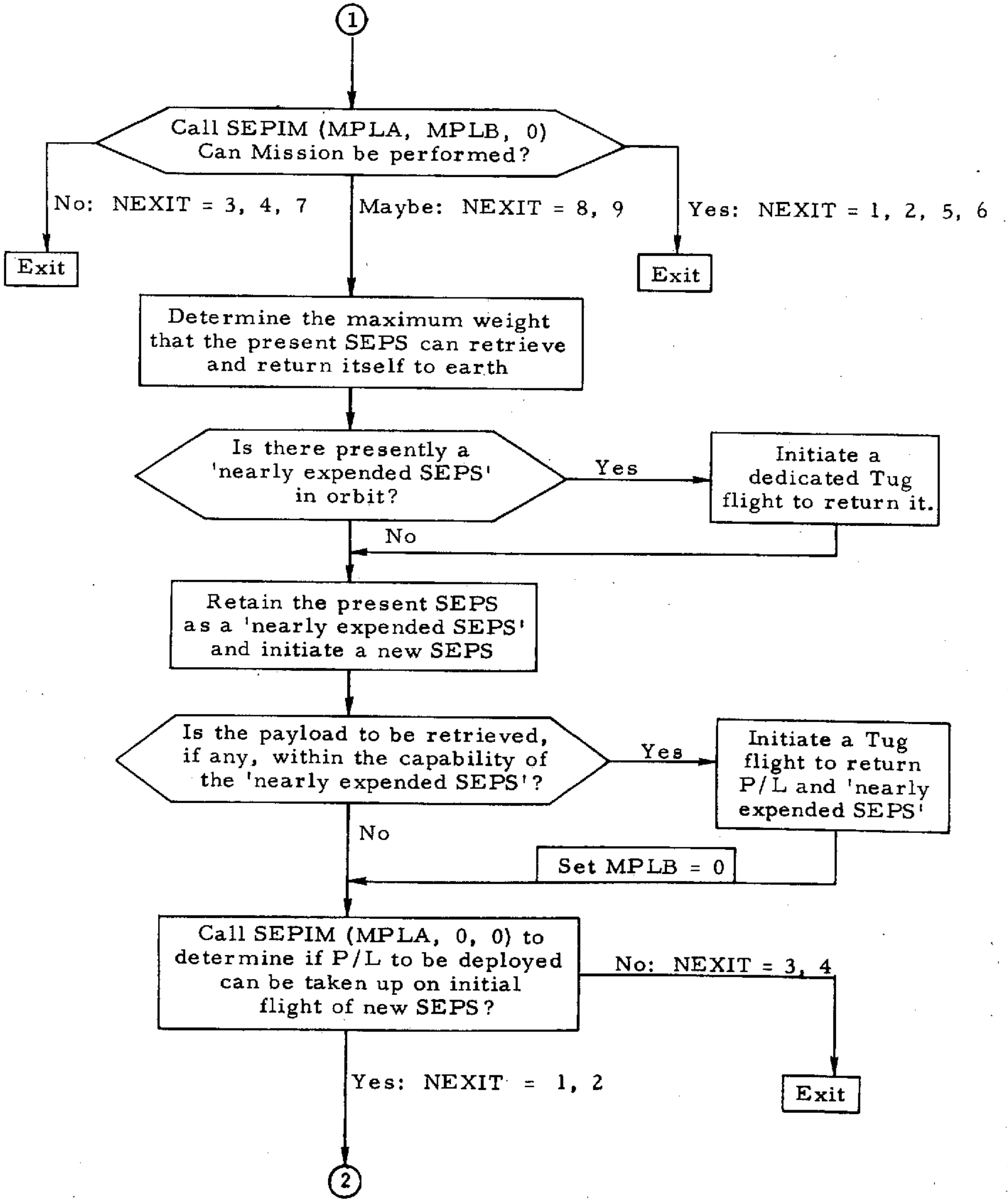
| | |
|---|---|
| NTUGS | number of Tug flights required to do the mission and return the expended SEPS if required |
| TLEFT | SEPS life remaining after accomplishing the mission (days) |
| MPT | SEPS fuel remaining after accomplishing the mission (lbs) |
| HCO | optimal changeover orbit altitude (n mi) |
| ICOS | optimal changeover orbit inclination (deg) |
| TU | time required by SEPS for ascent maneuvers (days) |
| TS | time required by SEPS for on-orbit servicing maneuvers (days) |
| TD | time required by SEPS for descent maneuvers (days) |

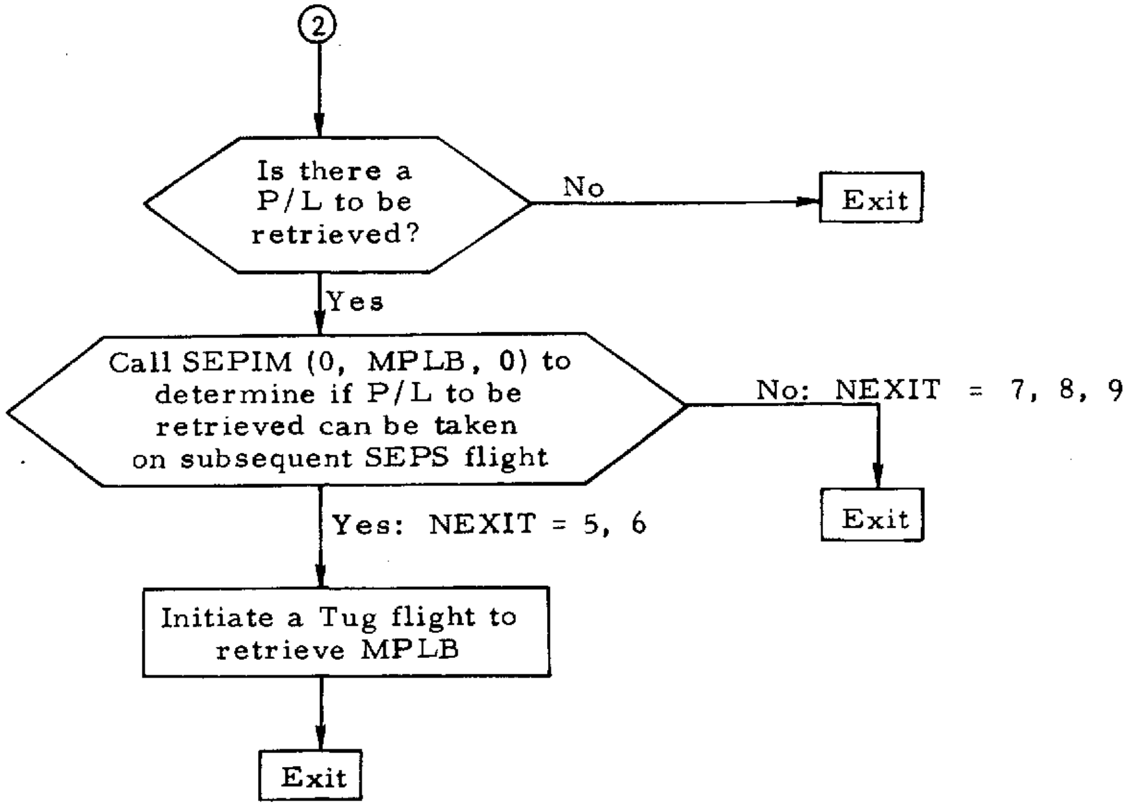
# APPENDIX A

## SEPX General Flow Chart

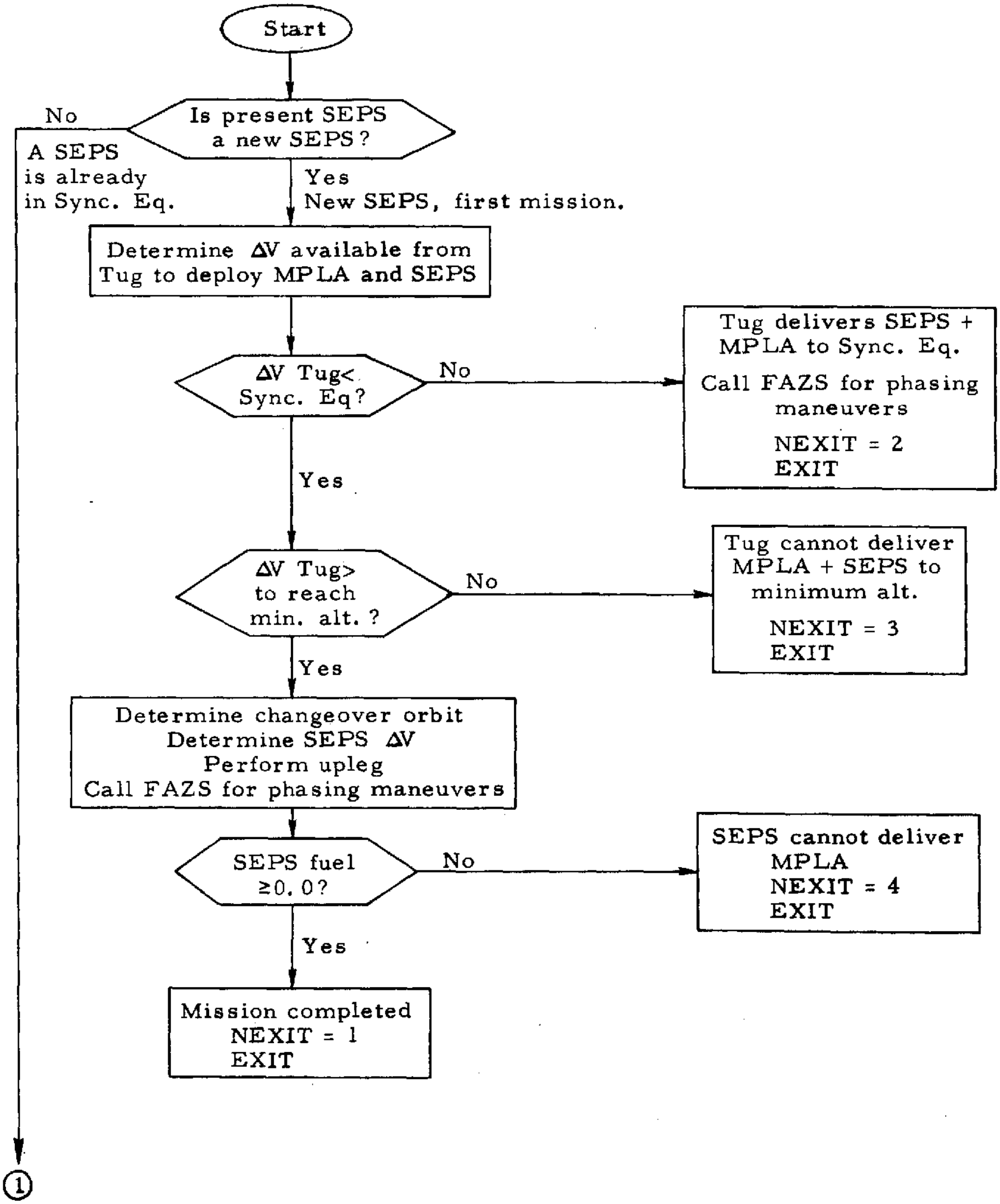
| | | | |
|---|---|---|---|
| Input: | MPLA | - | Deployed P/L |
| | MPLB | - | Retrieved P/L |
| | ERFLG | - | 0 = don't erase previous maneuver |
| | | | 1 = erase previous maneuver |
| Output: | NEXIT | - | Type of exit from the program i.e., mission possible or not. |
| | NTUGS | - | Number of tugs required for mission |
| | TLEFT, MPT | - | Time and fuel left on SEPS vehicle in orbit. |

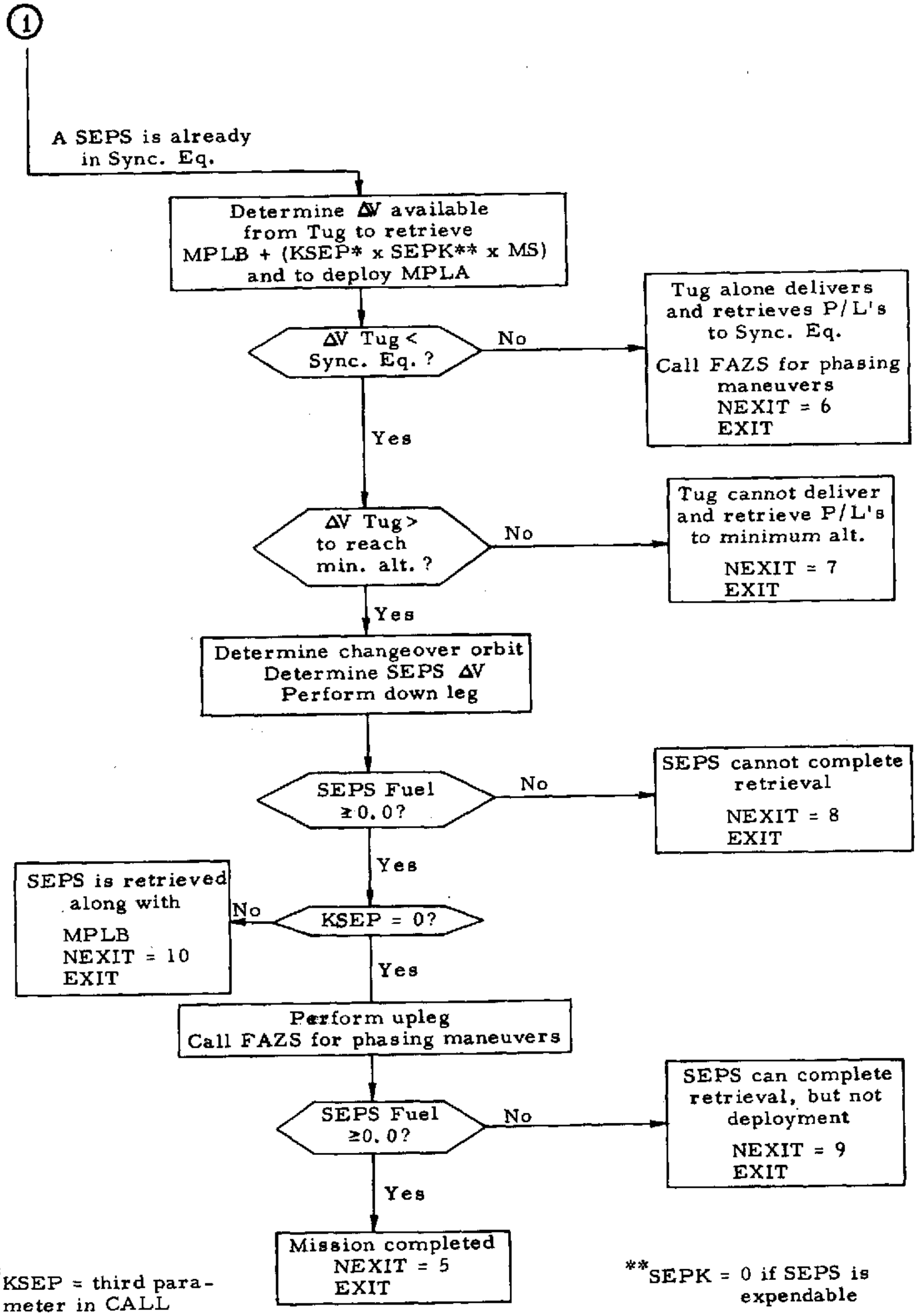



1
Call SEPIM (MPLA, MPLB, 0)
Can Mission be performed?
No: NEXIT = 3, 4, 7
Maybe: NEXIT = 8, 9
Yes: NEXIT = 1, 2, 5, 6
Exit
Exit
Determine the maximum weight that the present SEPS can retrieve and return itself to earth
Is there presently a 'nearly expended SEPS' in orbit?
Yes
Initiate a dedicated Tug flight to return it.
No
Retain the present SEPS as a 'nearly expended SEPS' and initiate a new SEPS
Is the payload to be retrieved, if any, within the capability of the 'nearly expended SEPS'?
Yes
Initiate a Tug flight to return P/L and 'nearly expended SEPS'
No
Set MPLB = 0
Call SEPIM (MPLA, 0, 0) to determine if P/L to be deployed can be taken up on initial flight of new SEPS?
No: NEXIT = 3, 4
Exit
Yes: NEXIT = 1, 2
2


2
Is there a P/L to be retrieved?
No
Exit
Yes
Call SEPIM (0, MPLB, 0) to determine if P/L to be retrieved can be taken on subsequent SEPS flight
No: NEXIT = 7, 8, 9
Exit
Yes: NEXIT = 5, 6
Initiate a Tug flight to retrieve MPLB
Exit

SEPIM General Flow Chart






*KSEP = third parameter in CALL sequence
(= 1 for SEPS final mission,
= 0 otherwise)

**SEPK = 0 if SEPS is expendable
= 1 if SEPS is retrieved

APPENDIX B

```
      SUBROUTINE  SEPX  (MPLA,MPLB,ERFLG,NEXIT )
C
C                 SEPX  THE SEP EXECTIVE ROUTINE IT PERFORMS THE
C                       LOGIC OF UTILIZING OF THE SEPS VEHICLE
C
C                 SPECIFIC INPUT
C                       MPLA  PAYLOAD TO BE DEPLOYED
C                       MPLB  PAYLOAD TO BE RETRIEVED
C                       ERFLG  0 = DO NOT ERASE PREVIOUS MANEUVER
C                              1 = ERASE THE PREVIOUS MANEUVER
C                       NEXIT SET TO 0 ON DATA CARD OF DRIVER
C
C                 COMMON INPUT (SEPVEH)
C                       MS
C                       MPT   AMOUNT OF FUEL REMAINING
C                       TLEFT AMOUNT OF TIME REMAINING
C                       E
C                       P
C                       SISP  SPECIFIC IMPULSE SEPS
C                       MDT
C                       PTCAP
C                       TSEP
C                       PSEP
C                       SG    GRAVITY CONSTANT
C
C                 OUTPUT
C                       NEXIT TYPE OF EXIT FROM SEPIM SUBROUTINE
C                       NTUGS NUMBER OF TUG FLIGHTS REQUIRED TO DO THE
C                             MISSION AND RETURN THE EXPENDED SEPS,IF
C                             ANY.  NTUGS WILL BE BETWEEN 1 AND 3.
C                       TLEFT TIME AND FUEL REMAINING ON SEPS VEHEHICLE
C                       MPT         IN ORBIT
C                 NOTE: CARDS MARKED [REDO3[ CAUSE PROGRAM TO RECYCLE
C                  FOR NEXIT=3. THESE CARDS SHOULD BE REMOVED FOR
C                   NON BATCH USAGE , SUCH AS LOVES PROGRAM.
      COMMON/TUGVEH/TYPE,NSTG,SPAR(3),WS(3),WPA(3),EISP(3)
     X             ,REUSE(3),WGA,TR
      COMMON/MISC/MU,G,FEAS(2)
      COMMON/SEPVEH/SEPS,MS,E,P,SISP,SEPK,SR,TSEP
      COMMON/OUTP/  TO,TU,HCO,ICOS,MPT,TLEFT,MDT,NTUGS
```

```
      COMMON/C1/MISSN
      COMMON/C2/TS
      COMMON/SERVIS/NSERV,DTHETA(10),MPLS(10),RSERV,VSERV
      REAL MPLS
      REAL         MDT,MPT,MPTSV,MPLB,MPTSV1,ISPS,MPLA
      REAL MU
      INTEGER      ERFLG
C
C                    SEARCH INPUT PARAMETERS FOR MISTAKES
C
      IERR=0
      IF ( REUSE(1) .LT. 0.5 .AND. SEPK .GT. 0.5 ) GO TO 160
      IF ( REUSE(1) .LT. 0.5 .AND. MPLB .GT. 0.0 ) IERR=15
      IF ( SEPK .LT. 0.5 .AND. MPLB .GT. 0.0 ) IERR=16
      IF ( IERR .EQ. 0 ) GO TO 5
      MPLB=0.0
      PRINT 1002,IERR
    5 CONTINUE
      RTINC=500.0
      HCO=160.
      TU=0.0
      TS=0.0
      TD=0.0
      ICOS=28.5
      TG   = G
      SG  = G
C
      IF (NEXIT .GE. 1 )  GO TO 10
C
C                   INITIALIZATION CALCULATATIONS
      C = SISP * 9.80621
      MDT = ( F* P * 4.409246 ) / ( C * C )
      TLEFT = TSEP
      MPT =  86400.0 * MDT * TLEFT
      TSAVE = TLEFT
      MPTSV = MPT
      RTCAP = 0.0
      RTCPSV = 0.0
C                            INITIALIZATION COMPLETE
C
```

```
C
   10 NTUGS = 1
      KTHREE=0                                                          PE003
      IF ( ERFLG .GE. 1 )  GO TO 20
C
C                          NO - SAVE PRESENT CONDITIONS
C
      MPTSV  =  MPT
      TSAVE  =  TLEFT
      RTCPSV = RTCAP
      GO TO 30
C
C                          ERASE -
C
   20 TLEFT  =  TSAVE
      PRINT  1050
      MPT  =  MPTSV
      RTCAP  =  RTCPSV
      ERFLG  =  0
C
C                          OK - NOW SEE IF PAYLOAD TO BE RETRIEVED, IF
C                               ANY, CAN BE BROUGHT DOWN BY A NEARLY
C                               EXPENDED SEPS.
C
   30 IF  ( MPLB .LT. 5 )  GO TO 40
      IF ( SEPK .LT. 0.5 ) GO TO 40
      IF  ( MPLB .GT. RTCAP ) GO TO 40
C
C                      INITIATE A TUG FLIGHT TO RETRIEVE PAYLOAD
C                      AND THE NEARLY EXPENDED SEPS.
C
      MPLB = 0.0
      RTCAP  =  0.0
      NTUGS  =  2
C
C                          NOW TRY TO PERFORM THE REMAINING MISSION
C                          WITH THE PRESENT SEPS
C
   40 CALL  SEPIM (MPLA,MPLB,0,NEXIT )
C
```

```
C                SEE IF IT CAN BE DONE - 1,2,5,6 OK -  3,4,7  NO
C
      IF  ( NEXIT .LT. 8 )  GO TO 90
C
C                ITS NOT CONCLUSIVE - CONTINUE
C                      DETERMINE THE MAX WEIGHT THAT THE
C                      PRESENT SEPS CAN RETRIEVE AND RETURN ITSELF
C                      TO EARTH.
C
      IF ( KTHREE .EQ. 1 ) GO TO 130                                      RED03
      MPTSV1=MPTSV
      TSAV1=TSAVE
      IF ( RTCAP .LE. 5)   GO TO 50
      NTUGS = 2
   50 RTCAP=0.0
      IF ( SEPK .LT. 0.5 ) GO TO 75
C
C                NOW RETAIN THE PRESENT SEPS AS A NEARLY EXPENDED SEPS
C                AND DETERMINE ITS RETREIVE CAPABILITIES
C
      RTCAP=RTINC+10.0
   60 CALL SEPIM  (0.0,RTCAP,1,NEXIT )
      IF ( NEXIT .LE. 5  .OR. NEXIT .EQ. 9 ) GO TO 100
      IF ( NEXIT .EQ. 7 .OR. NEXIT .EQ.8 ) GO TO 70
      MPT = MPTSV1
      TLEFT = TSAV1
      RTCAP=RTCAP+RTINC
      GO TO 60
C
C                IT CAN NO LONGER DO IT - INITIATE A NEW SEPS
C
   70 RTCAP=RTCAP-RTINC
   75 TLEFT=TSEP
      MPT = TSEP *86400.0 * MDT
C
C                CHECK ON THE NEARLY EXPENDED ONE
C
      IF ( MPLB .LT.  5 ) GO TO 80
      IF ( MPLB .GT. RTCAP ) GO TO 80
C
```

```
C                    THE PAYLOAD TO BE RETRIEVED IS WITHIN THE
C                    CAPABILITY OF THE NEARLY EXPENDED SEPS.
C                    INITIATE A TUG FLIGHT TO RETRIEVE PAYLOAD
C                    AND SEPS.
C
      MPLB = 0.0
      PTCAP = 0.0
      NTUGS = NTUGS + 1
C
C                  CONTINUE - SEE IF THE PAYLOAD TO BE DEPLOYED CAN
C                             BE TAKEN UP ON INITIAL FLIGHT OF NEW SEPS
C
   80 CALL  SEPIM (MPLA,0,0,NEXIT )
      IF ( NEXIT .GT. 4 )  GO TO 110
      IF ( NEXIT .GE. 3 )  GO TO 90
C
C                  CONTINUE - SEE IF THERE IS A PAYLOAD TO BE RETRIEVED.
C
      IF ( MPLB .LE. 5 )  GO TO 90
C
C                  CAN PAYLOAD TO BE RETRIEVED BE TAKEN ON SUBSEQUENT
C                  SEPS FLIGHT
C
      CALL  SEPIM (0,MPLB,0,NEXIT )
      IF  (NEXIT .LE. 4 .OR.  NEXIT .EQ. 10 )  GO TO 120
      IF ( NEXIT .GE. 7 ) GO TO 90
      NTUGS = NTUGS + 1
   90 CONTINUE
      IF ( NEXIT .EQ. 3 ) GO TO 95                                      RED03
      PRINT 1030,MISSN,MPLB,MPLA,TD,TU,TS,NSERV,HCO,ICOS,TLEFT,MPT,NEXIT
     X,NTUGS
      IF ( KTHREE .EQ. 1 ) GO TO 96                                     RED03
      RETURN
   95 IF ( KTHREE .EQ. 1 ) GO TO 130                                    RED03
      KTHREE=1                                                          RED03
      NSAVE=NSERV                                                       RED03
      TSSV=TS                                                           RED03
      NSERV=0                                                           RED03
      TS=0.0                                                            RED03
      W1SV=MPLA                                                         RED03
```

```
      W2SV=MPLB                                                         RED03
      MPLA=0.0                                                          RED03
      MPLB=0.0                                                          RED03
      GO TO 40                                                          RED03
   96 IF ( NEXIT .EQ. 4 .OR. NEXIT .GE. 7 ) GO TO 130                   RED03
      NSERV=NSAVE                                                       RED03
      TS=TSSV                                                           RED03
      MPLA=W1SV                                                         RED03
      MPLB=W2SV                                                         RED03
      KTHREE=0                                                          RED03
      GO TO 40                                                          RED03
C                 ***   ERROR CONDITIONS   ***
C
C
  100 IERR = 1
      GO TO 150
  110 IERR = 2
      GO TO 150
  120 IERR = 3
      GO TO 150
  130 IERR=4                                                            RED03
  150 PRINT 1000 , IERR,NEXIT
      STOP
  160 IERR=10
      PRINT 1001,IERR
      RETURN
 1000 FORMAT (1H0,T30,12H***** ERROR ,I3,45H IN SEP EXEC - INCORRECT LOG
     XIC FLOW.  NEXIT =,I3 )
 1001 FORMAT (20X,*-INPUT ERROR *,I2,* DETECTED IN SEPX, PROCEED TO NEXT
     X DATA CASE*)
 1002 FORMAT (20X,*-INPUT ERROR *,I2,* DETECTED IN SEPX, MPLB HAS BEEN S
     XET TO ZERO*)
 1030 FORMAT (T13,I2,4X,F7.1,2X,F7.1,1X,F6.2,2X,F6.2,1X,F6.2,*(*,I2,*)*,
     X2X,F7.1,2X,F4.1,2X,F6.2,2X,F8.2,4X,I2,4X,I2)
 1050 FORMAT (20X,*-PREVIOUS MANEUVER HAS BEEN ERASED-*)
      END
```

```
      SUBROUTINE  FAZS
C   PERFORMS SEPS PHASING, ASSUMING CONSTANT SEP THRUSTING.
C   INPUTS:  NSERV=NUMBER OF SERVICE LEGS.
C            DTHETA= ANGULAR TRAVEL (DEG) OF EACH SERVICE LEG.
C            MPLS= PAYLOAD (LBS) ON EACH SERVICE LEG.
C            PSERV,VSERV= PERIOD (SEC) AND VELOCITY (MPS) OF SERVICE ORBIT.
C   OUTPUTS:  MPT= FEUL REMAINIG AFTER PHASING (LBS).
C             TLEFT= TIME REMAINING ON SEPS AFTER PHASING (DAYS).
      COMMON/SEPVEH/SEPS,MS,E,P,SISP,SEPK,SR,TSEP
      COMMON/MISC/MU,SG,FEAS(2)
      COMMON/OUTP/  TO,TU,HCO,ICOS,MPT,TLEFT,MDT,NTUGS
      COMMON/SERVIS/NSERV,DTHETA(10),MPLS(10),PSERV,VSERV
      COMMON/C2/TS
      REAL MS,MPT,MDT,MPLS,MKG
      REAL  MU
      F=(MDT*9.80621*SISP)/2.204623
      CONST1=(3.0*F*PSERV)/(4.0*VSERV)
      TS=TLEFT
      DO 100 N=1,NSERV
      MKG=(MS+MPT+MPLS(N))/2.204623
      REV=SQRT((MKG*DTHETA(N))/(360.*CONST1))
      TLEFT=TLEFT-((REV*PSERV)/86400.)
      MPT=MPT-(MDT*REV*PSERV)
  100 CONTINUE
      TS=TS-TLEFT
      RETURN
      END
```

```
      SUBROUTINE  SEPIM (MPLA,MPLB,KSEP,NEXIT)
C
C              SEPIM THIS SUBROUTINE COMPUTES THE PERFORMANCE
C                    OF THE SEPS ON A DEPLOY MISSION.
C
C              SPECIFIC INPUT
C                    MPLA  PAYLOAD TO BE DEPLOYED
C                    MPLB  PAYLOAD TO BE RETRIEVED
C                    KSEP  ERASE FLAG
C                                0 = DONT ERASE PRIEVIOUS MANEUVER
C                                1 = ERASE PRIEVIOUS MANEUVER
C                    NEXIT SET TO 0 PRIOR TO ENTRY
C
C              OUTPUT
C                    NEXIT TYPE OF EXIT FROM SEPS IF MISSION POSSIBLE
C                    NTUGS NUMBER OF TUG FLIGHTS REQUIRED TO
C                          DO THE MISSION AND RETURN EXPENDED SEPS,
C                          IF ANY.  NTUGS WILL BE BETWEEN 1 AND 3
C                    TLFT  TIME AND FUEL REMAINING ON SEPS IN ORBIT
C                    MPT
      COMMON/SEPVEH/SEPS,MS,E,P,SISP,SEPK,SR,TSEP
      COMMON/OUTP/  TD,TU,HCO,ICOS,MPT,TLEFT,MDT,NTUGS
      COMMON/MISC/ MU,SG,FEAS(2)
      COMMON/TABLE/TUGDV(20)
      COMMON/TUGVEH/TYPE,NSTG,SPAR(3),WS(3),WPA(3),EISP(3)
     X             ,REUSE(3),WGA,TR
      COMMON/SERVIS/NSERV,DTHETA(10),MPLS(10),PSERV,VSERV
      REAL        MPLA,MPLB,MS,MPT,MRTUG,HCO,ICOS
      REAL         MPLS
      REAL MDT
      REAL MU
      TS=0.0
      TU=0.0
      TD=0.0
      HCO=160.
      ICOS=28.5
C
C              FIRST TEST IF THERE IS A SEPS AVAILABLE
C
      IF ( TLEFT .LT. TSEP -.001 )  GO TO 20
```

```
C
C                 NO - ITS A NEW SEPS
      TLEFT = TSEP
      WPLA = MPLA + MS + MPT
      WPLB =  0.0
C
C                CALL TUGCP TO DETERMINE TUG CAPABILITY
C
      CALL TUGCP  (WPLA,WPLB,MRTUG,DVTUG)
      IF (DVTUG .LT. TUGDV(13) )  GO TO 10
C
C                      TUG DELIVERS SEPS AND MPLA TO SYNC EQ:
C
      TLEFT = TLEFT -.005
      TU = 0.0
      TD = 0.0
      HCO=19323.
      ICOS=0.0
      IF (NSERV.GT.0) CALL FAZS
      NEXIT = 2
      RETURN
C
C                      NEXT CHECK IF ITS CAPABLE
C
   10 NEXIT = 3
      IF ( DVTUG .LT. TUGDV(1) )   RETURN
C
C                 ITS OK - CONTINUE DETERMINE CHANGEOVER ORBIT
C
      CALL  INTORB (DVTUG,HCO,ICOS)
C
C                 DETERMINE THE SEPS DELTA V
C
      CALL  SEPDV (HCO,ICOS,DVSEP,MRSEP)
C
C                 PERFORM UP LEG AND PHASING
C
      CALL  PLUPD (MPLA,MRSEP,TU )
      IF (NSERV.GT.0) CALL FAZS
C
```

```
C                   SET NEXIT AND TEST IF THERE IS FUEL REMAINING
C
      NEXIT = 1
      IF ( MPT .GE. 0.0 ) RETURN
C
C                   SEPS CANNOT DELIVER THE PAYLOAD - SET FLAG AND ABORT
C
      NEXIT = 4
      RETURN
C
C                   THIS ENTRY POINT FOR SEPS AVAILABLE
C                   IN SYNC EQ. ORBIT
C
   20 WPLB= MPLB+(KSEP*SEPK*MS)
      WPLA = MPLA
C
C                         DETERMINE THE TUG CAPABILITY
C
      CALL  TUGCP (WPLA,WPLB,MRTUG,DVTUG )
      IF ( DVTUG .LT. TUGDV(13) )  GO TO 30
C
C                         NO - TUG ALONE CAN DELIVER AND RETRIEVE
C                              PAYLOADS TO SYNC EQ ORBIT
C
      TU = 0.0
      TD = 0.0
      HCO=19323.
      ICOS=0.0
      IF (NSERV.GT.0) CALL FAZS
      NEXIT = 6
      RETURN
C
C                         TUG ALONE CAN NOT DO IT- CHECK IF ALL OK
C
   30 NEXIT = 7
      IF ( DVTUG .LT. TUGDV(1) ) RETURN
C
C                   ITS OK - CONTINUE
C                         DETERMINE CHANGEOVER ORBIT
      CALL  INTORB (DVTUG,HCO,ICOS )
```

```
      CALL   SEPDV (HCO,ICOS,DVSEP,MRSEP)
      CALL   PLUPD (MPLB,MRSEP,TD )
C
C                             SET UP AND CHECK CONSTRAINTS
C
      NEXIT = 8
      IF ( MPT .LT. 0.0 )    RETURN
      IF ( KSEP .EQ. 0 )     GO TO 40
C
C                             SEPS RETRIEVED ALONG WITH PAYLOAD
C
      TU = 0.0
      NEXIT = 10
      RETURN
C
C                             CONTINUE PROCESS
C
   40 CALL   PLUPD (MPLA,MRSEP,TU)
      IF (NSERV.GT.0) CALL FAZS
      NEXIT = 9
      IF ( MPT .LT. 0.0 )  RETURN
C
C                             MISSION COMPLETE
C
      NEXIT = 5
      RETURN
      END
```

```
      SUBROUTINE  TUGCP (WPLA,WPLB,MRTUG,DVTUG )
C
C                 TUGCP - CALLS THE APPROPRIATE TUG EQUATIONS.
C                       ( AT PRESENT - ONLY OPTION IS SINGLE
C                         STAGE CRYOGENIC TUG. )
C
      CALL  CRYO1 (WPLA,WPLB,MRTUG,DVTUG)
      RETURN
      END
```

```
      SUBROUTINE  CRYO1 (WPLA,WPLB,MRTUG,DVTUG)
C
C                 CRYO1- FINDS THE DELTA V CAPABILITY OF A
C                        SINGLE STAGE TUG WITH PAYLOADS WPLA AND WPLB.
C
C
      COMMON/TUGVEH/TYPE,NSTG,SPAP(3),WS(3),WPA(3),EISP(3)
     X                  ,REUSE(3),WGA,TR
      COMMON/MISC/MU,TG,FEAS(2)
      REAL MU
      REAL        MRTUG
      WP=WPA(1)
      IF ((WS(1)+WPA(1)+WPLA) .GT.WGA) WP=WGA-(WS(1)+WPLA)
      MRTUG=(WP+WS(1)+WPLA)/(WS(1)+WPLA)
      IF (REUSE(1).LT.0.5) GO TO 100
      BZ=WS(1)+WS(1)+WPLA+WPLB
      CZ=-WP*(WPLB+WS(1))
      WP1 = ( -BZ + ((( BZ*BZ) -( 4.0*CZ)) **0.5) )/ 2.0
      MRTUG=(WP1+WPLB+WS(1))/(WPLB+WS(1))
  100 ALMR=ALOG(MRTUG)
      DVTUG=TG*EISP(1)*ALMR/(TR+1)
      RETURN
      END
```

```
      SUBROUTINE  PLUPD (MPLU,MRSEP,T )
C
C                 PLUP - CARRIES SEPS PAYLOAD UP
C
      COMMON/SEPVEH/SEPS,MS,E,P,SISP,SEPK,SP,TSEP
      COMMON/OUTP/  TD,TU,HCO,ICOS,MPT,TLEFT,MDT,NTUGS
      REAL          MS,MPT,MDT,MPLU,MRSEP,MPT1
      MPT1 = (( MPT + MS + MPLU ) / MRSEP) - (MS+MPLU)
      T  =  (MPT - MPT1 ) / (86400.0 * MDT)
      TLEFT  = TLEFT - T
      MPT = MPT1
      RETURN
      END
```

```
      SUBROUTINE  SEPDV (HCO,ICOS,DVSEP,MRSEP )
C
C                 SEPDV - CALCULATES THE REQUIRED SEP DELTA VELOCITY
C                         NEEDED FOR SYNC EQ. AND THE CORRESPONDING
C                         MASS RATIO.
C
C                 INPUT
C                       HCO   ORBIT ALTITUDE
C                       ICOS  INCLINATION
C
C                 OUTPUT
C                       DVSEP THE SEP DELTA V
C                       MRSEP THE MASS RATIO
C
      COMMON/SEPVEH/SEPS,MS,E,P,SISP,SEPK,SP,TSEP
      COMMON/MISC/MU,SG,FEAS(2)
      REAL        HCO,ICOS,MU,HS,ICO,MRSEP,MDT,MPT,MS
      HS = 19323.0
      MU = 1.40765388 E 16
      RE =  3443.9308
C
C                 SET UP CONSTANTS
C
      RCO = (HCO +RE ) * 6076.1155
      VCO = (MU / RCO) **0.5
      RS  = (HS + RE ) * 6076.1155
      VS  = (MU / RS ) **0.5
      ICO =(ICOS / 57.295779513)*1.570796326794
      CICO =  COS (ICO )
      DVSEP = (( VCO*VCO) +(VS*VS) -((VS +VS)* VCO* CICO) )**.5
      EXP1=   DVSEP/ (SG*SISP)
      MRSEP =  EXP(EXP1)
      RETURN
      END
```

```
      SUBROUTINE  INTORB (DVTUG,HCO,ICOS )
C
C                INTORB - AN INTERPOLATION SCHEME TO DETERMINE
C                         THE OPTIMUM CHANGEOVER ORBIT ALTITUDE
C                         AND INCLINATION.
C
C                INPUT
C                      DVTUG - TUG DELTA V
C
C                OUTPUT
C                      HCO   ALTITUDE OF CHANGEOVER ORBIT
C                      ICOS  INCLINATION OF CHANGEOVER ORBIT
C
C
      COMMON/TABLE/TUGDV(20)
      DIMENSION              ALT(20),INC(20)
      REAL        ICOS,INC,HCO
      DATA        TUGDV/ 10295.74,10600.0,10900.0,11200.0,11500.0,
     X                   11800.0,12100.0,12400.0,12700.0,13000.0,
     X                   13300.0,13600.0,13835.17, 7* 0.0/
      DATA        ALT  / 8000.0,8000.0,8000.0,8000.0,8000.0,8500.0,
     X                   9500.0,10500.0,11500.0,13000.0,14500.0,
     X                   17000.0,18000.0, 7*0.0/
      DATA        INC  / 28.5,19.6,15.8,12.8,10.14,8.86,8.52,7.67,
     X                   6.4,5.5, 3.87, 2.45, 8* 0.0 /
C
C                TEST THE UPPER AND LOWER BOUNDS
C
      IF (DVTUG .GE. TUGDV(1))GO TO 10
      DVTUG = TUGDV(1)
      PRINT  100
      GO TO 15
   10 IF (DVTUG .LE. TUGDV(13))  GO TO 15
      DVTUG = TUGDV(13)
      PRINT  110
C
C                FIND THE RANGE OF DELTA V
C
   15 DO  20  NP1 =2,13
      IF (DVTUG .LE. TUGDV(NP1))GO TO 30
```

```
   20 CONTINUE
C
C                    IF LOOP IS EXUSTED ABORT
C
      PRINT 120
      RETURN
C                    FOUND THE RANGE COMPUTE THE ALT AND INC.
   30 NPO = NP1 - 1
      FRAC = ( DVTUG - TUGDV(NPO )   ) / (TUGDV(NP1) - TUGDV(NPO ))
      HCO  =  ALT(NPO )  + FRAC* (ALT(NP1) - ALT(NPO ))
      ICOS =  INC(NPO )  + FRAC* (INC(NP1) - INC(NPO ))
      RETURN
100   FORMAT (T10,68H*****   ERROR - DVTUG HAS BEEN INCREASED TO A MINIM
     XUM VALUE.    *****)
110   FORMAT (T10,68H*****   ERROR - DVTUG HAS BEEN DECREASED TO A MAXIM
     XUM VALUE.    *****)
120   FORMAT (T10,47H*****   FATAL PROBLEM IN INTORB.  ABORT    *****)
      END
```

## APPENDIX C

### Sample Output

For this Appendix the SEPSIM subroutines were attached to a simple driver program and run in the batch processing mode. The basic input parameters are listed at the top of the output sheet. For each mission number (representing separate calls to SEPX) the input values of deployed and retrieved payload weights are printed along with some of the output results for that case. The output includes the time spent by the SEPS performing descent, ascent, and servicing maneuvers (the input value NSERV, the number of servicing legs, is shown in parenthesis), the altitude and inclination of the optimal intermediate orbit, the time and fuel remaining on the SEPS after performing the mission, the exit type (NEXIT), and the number of Tug flights required to support that mission.

In the output sample it is significant to notice that Tug/SEPS capability is sufficient to perform all the missions (EXIT = 1, 5, or 6). In mission 8, the input value ERFLG has been set to 1 so that mission 7 is erased and replaced by mission 8 (which, in this case, is identical to mission 7). On mission 11 a new SEPS is launched (EXIT = 1) since the old SEPS fuel is nearly exhausted. The old SEPS is not immediately retrieved (note that TUGS = 1), but waits in synchronous orbit for a payload small enough for it to retrieve. The input for mission 13 includes ERFLG = 1 causing mission 12 maneuvers to be erased and replaced by mission 13 maneuvers (the only difference is the addition of 2 servicing legs). By mission 16, SEPS #2 has become exhausted, and SEPS #3 is launched (EXIT = 1). SEPS #2 remains in synchronous orbit waiting for a small retrieval task and SEPS #1 is returned to earth in a separate Tug flight (TUGS = 2) for refurbishment. The performance of mission 18 requires three Tug flights (TUGS = 3). In one flight, SEPS #4 and a 6080 lb satellite are launched. SEPS #4 delivers the satellite and performs the five required servicing tasks. In the second flight, SEPS #2 is returned to earth for refurbishing while SEPS #3 remains in synchronous orbit. In the third flight, the Tug ascends unladen to a 10623.7 nmi circular orbit where it receives a 4585 lb satellite retrieved from synchronous orbit by SEPS #4. The Tug returns the satellite to the Orbiter and the SEPS #4 ascends (unladen) to synchronous orbit. Since only one line of output has been allowed for each mission, it is this last Tug/SEPS flight which is reflected in the output (hence EXIT = 5). Similar situations arise in missions 21 and 25.

The above 25 cases were executed (including compilation time) in less than 0.75 sec of CDC 7600 computer time. The amount of fuel remaining on the SEPS has been accurately tracked and Tug flights initiated as required to support the missions.

The driver program used to generate the above cases has been given the acronym of PRFORM and its listing is shown after the sample output cases. Program PRFORM acts as a driver routine for the following Tug stage configuration performance calculation routines:

1. Two liquid stages in a "slingshot" mode of operation (SSHOT)
2. A single stage liquid vehicle (SSLQD)
3. A liquid stage and a solid kick upper stage (TRNKC)
4. Tug operations using a solar electric propulsion upper stage (SEPSIM)

To access the SEPSIM portion of code using the PRFORM driver routine, it is necessary to set the input variables via a NAMELIST input as specified in the USAGE section of this document (Page 5). In addition, the variable "SEPS" must be set equal to 1 to indicate that use of the SEPSIM code is desired. When running the program in the batch mode, the last input data case should be "NSTG = -1." This causes the program to cease operation with no error conditions.

```
SEP VEHICLE:   MS        F       P          ISP      TSEP    SEPK
             2600.0    .670   21000.0    3000.0    400.0     1.00

TUG VEHICLE:  TYPE        WS        WPA        WGA       EISP     REUSE
           28 FT CENT  5676.0   46594.0   65000.0    427.71    1.00

CONSTANTS:      MU                  G         SR       TR
          .140765388E+17    32.1725    0.00       .01
 NEXIT CODE:
   1 = A NEW SEPS IS LAUNCHED AND PERFORMS THE MISSION
   2 = TUG ALONE DELIVERS MPLA AND NEW SEPS TO MISSION ORBIT. (PHASING DONE BY SEPS)
   3 = TUG CANNOT DELIVER MPLA + NEW SEPS TO MIN. ALT.
   4 = NEW SEPS CANNOT DELIVER MPLA.
   5 = SEPS ALREADY IN ORBIT PERFORMS MISSION.
   6 = TUG ALONE PERFORMS DEL AND RETR, PHASING DONE BY SEPS.
   7 = TUG CANNOT DEL AND RETR PAYLOADS TO MIN. ALT.
   8 = SEPS ALREADY IN ORBIT CANNOT COMPLETE RETR.
   9 = SEPS ALREADY IN ORBIT CANNOT COMPLETE DEP.
   10 = SEPS IS RETR ALONG WITH MPLB
```

OUTPUT RESULTS:

| MISSION | SAT. WT.(LBS) | | TIME (DAYS) | | | INT. ORBIT | | RESIDUALS | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| NO. | RETR | DEPLOY | DOWN | UP | SERV | ALT | INC | TIME | FUEL | EXIT | TUGS |
| 1 | 0.0 | 10010.0 | 0.00 | 135.33 | 0.00( 0) | 8000.0 | 11.1 | 264.67 | 1639.17 | 1 | 1 |
| 2 | 0.0 | 9158.0 | 18.23 | 57.17 | 0.00( 0) | 12566.6 | 5.8 | 189.26 | 1172.16 | 5 | 1 |
| 3 | 0.0 | 8572.0 | 14.46 | 46.97 | 0.00( 0) | 13217.7 | 5.3 | 127.83 | 791.71 | 5 | 1 |
| 4 | 0.0 | 8428.0 | 12.57 | 43.50 | 0.00( 0) | 13376.5 | 5.1 | 71.76 | 444.46 | 5 | 1 |
| 5 | 0.0 | 3496.0 | 0.00 | 0.00 | 0.00( 0) | 19323.0 | 0.0 | 71.76 | 444.46 | 6 | 1 |
| 6 | 0.0 | 3974.0 | 0.00 | 0.00 | 0.00( 0) | 19323.0 | 0.0 | 71.76 | 444.46 | 6 | 1 |
| 7 | 0.0 | 4228.0 | 0.00 | 0.00 | 0.00( 0) | 19323.0 | 0.0 | 71.76 | 444.46 | 6 | 1 |
| | -PREVIOUS MANEUVER HAS BEEN ERASED- | | | | | | | | | | |
| 8 | 0.0 | 4228.0 | 0.00 | 0.00 | 0.00( 0) | 19323.0 | 0.0 | 71.76 | 444.46 | 6 | 1 |
| 9 | 800.0 | 3948.0 | 5.72 | 10.35 | 0.00( 0) | 16994.1 | 2.5 | 55.69 | 344.92 | 5 | 1 |
| 10 | 0.0 | 1488.0 | 0.00 | 0.00 | 0.00( 0) | 19323.0 | 0.0 | 55.69 | 344.92 | 6 | 1 |
| 11 | 0.0 | 9950.0 | 0.00 | 112.06 | 0.00( 0) | 8397.5 | 9.1 | 287.94 | 1783.29 | 1 | 1 |
| 12 | 2787.0 | 9500.0 | 61.00 | 114.89 | 0.00( 0) | 8079.3 | 9.9 | 112.05 | 693.95 | 5 | 1 |
| | -PREVIOUS MANEUVER HAS BEEN ERASED- | | | | | | | | | | |
| 13 | 2787.0 | 9500.0 | 61.00 | 114.89, | 10.40( 2) | 8079.3 | 9.9 | 101.65 | 629.53 | 5 | 1 |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 14 | 3570.0 | 3475.0 | 43.95 | 41.48 | 0.00( 0) | 10057.2 | 8.0 | 16.31 | 101.04 | 5 | 1 |
| 15 | 0.0 | 2766.0 | 0.00 | 0.00 | 0.00( 0) | 19323.0 | 0.0 | 16.31 | 101.04 | 6 | 1 |
| 16 | 0.0 | 8352.0 | 0.00 | 99.65 | 0.00( 0) | 8960.6 | 8.7 | 300.35 | 1860.19 | 1 | 2 |
| 17 | 5912.0 | 7478.0 | 117.73 | 127.20 | 0.00( 0) | 8000.0 | 17.5 | 55.46 | 343.47 | 5 | 1 |
| 18 | 4585.0 | 6080.0 | 54.07 | 24.93 | 21.15( 5) | 10623.7 | 7.5 | 213.85 | 1324.43 | 5 | 3 |
| 19 | 1405.0 | 6080.0 | 24.95 | 46.12 | 0.00( 0) | 12018.9 | 6.1 | 142.77 | 884.25 | 5 | 1 |
| 20 | 0.0 | 9400.0 | 15.59 | 57.20 | 0.00( 0) | 12315.6 | 5.9 | 69.98 | 433.43 | 5 | 1 |
| 21 | 3240.0 | 9250.0 | 28.75 | 16.02 | 0.00( 0) | 13346.0 | 5.1 | 257.62 | 1595.51 | 5 | 3 |
| 22 | 772.0 | 10010.0 | 31.00 | 87.44 | 0.00( 0) | 10286.2 | 7.9 | 139.18 | 861.96 | 5 | 1 |
| 23 | 0.0 | 10040.0 | 17.18 | 56.48 | 0.00( 0) | 11630.5 | 6.3 | 55.51 | 343.78 | 5 | 1 |
| 24 | 0.0 | 7637.0 | 8.87 | 31.73 | 0.00( 0) | 14258.0 | 4.1 | 14.90 | 92.29 | 5 | 1 |
| 25 | 1557.0 | 7478.0 | 0.00 | 0.00 | 0.00( 0) | 19323.0 | 0.0 | 316.73 | 1961.62 | 6 | 3 |

```
      PROGRAM PREFORM (INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
      COMMON/TUGVEH/TYPE,NSTG,SPAR(3),WS(3),WPA(3),EISP(3)
     X              ,REUSE(3),WGA,TR
      COMMON/MISC/MU,G,FEAS(2)
      COMMON/SEPVEH/SEPS,MS,E,P,SISP,SEPK,SR,TSEP
      COMMON/SERVIS/NSEPV,DTHETA(10),MPLS(10),PSERV,VSERV
      COMMON/C1/MISSN
      DIMENSION DVLEG(10),PLEG(10)
      INTEGER ERFLG
      INTEGER SEPS,SPAR
      REAL MS,MU,MPLA,MPLB,MPLS
      DATA  ERFLG/0/
      DATA TYPE/10H28 FT CENT/
      DATA NSERV/0/
      DATA PSERV,VSERV/86165.,3074.66/
      NAMELIST/INPUT/TYPE,NSTG,SPAR,WS,WPA,EISP,REUSE,WGA,TR
     X              ,NLEG,DVLEG,PLEG
     X              ,SEPS,MS,E,P,SISP,SEPK,SR,TSEP
     X              ,MPLA,MPLB
     X              ,NSERV,DTHETA,MPLS,PSERV,VSERV
     X              ,ERFLG
      NEXIT = 0
3     MISSN = 0
3     MU=1.40765388 E16
5     G = 32.1725
7   5 READ  (5,INPUT )
12    IF (NSTG .LT. 0)  STOP
   C
   C             PERF - SETS UP AND CHOOSES THE SPECIFIC
   C                    PERFORMANCE SUBROUTINE TO BE EXECUTED
   C
   C             SSHOT - SLINGSHOT - LIQUID UPPERS
   C             SSLQD - SINGLE STAGE LIQUID
   C             TRNKC - TRANS KICK - SOLID UPPERS
   C             SEPSIM- SEPS SIMULATOR
   C
16    IF ( SEPS .NE. 0 )  GO TO 40
17    IF ( NSTG .GT. 1 )  GO TO 10
23    CALL  FA70  (DVLEG,PLEG,NLEG)
25    CALL  SSLQD (DVLEG,PLEG,NLEG)
```

```
 30          GO TO 50
 31       10 DO 20  I = 2,NSTG
 37          IF (SPAR(I) .NE. 0 ) GO TO 30
 40       20 CONTINUE
 42          CALL  FAZO  (DVLEG,PLEG,NLEG)
 44          CALL  SSHOT (DVLEG,PLEG,NLEG)
 47          GO TO 50
 50       30 CALL  TRNKC (DVLEG,PLEG,NLEG )
 53          GO TO 50
 54       40 IF  ( MISSN .GT. 0 )   GO TO 45
 57          PRINT 1010 ,  MS,E,P,SISP,TSEP,SEPK,TYPE,WS(1),WPA(1),WGA
            X             ,EISP(1),REUSE(1),MU,G,SR,TR
126          PRINT 1030
132          PRINT  1020
136       45 MISSN =  MISSN + 1
140          CALL SEPX  (MPLA, MPLB,ERFLG,NEXIT )
143          GO TO 5
144       50 CONTINUE
144          WRITE  (6,1000 )    NSTG, (WS(I),I=1,3    ), (WPA(J),J=1,3   ),
            X                    (EISP(K),K=1,3)     ,(REUSE(L),L=1,3)    ,WGA,
            X                    NLEG, (DVLEG(I),I=1,10)   ,(PLEG(J),J =1,10)   ,
            X                    SEPS,(SPAR(I),I =1,3)    , FEAS(1),FEAS(2)
204          GO TO 5
      1000  FORMAT  (1H1,/// T20,*NSTG =*,I3,/  T30,*WS =*, T40,3F13.6 ,/
            X          T30,*WPA =*,T40,3F13.6,/  T30,*EISP =*,T40,3F13.6 ,/
            X          T30,*REUSE =*,T40,3(F13.6)/T30,*WGA =*,T40,F13.6 ///
            X          T20,*NLEG =*,I3,/ T30,*DVLEG =*,T40,5F13.6,/T40,5(F13.6)/
            X          T30,*PLEG =*,T40,5F13.6/T40,5 F13.6 ///T20,*SEPS =*,I3,/
            *          T20,*SPAR =* 3I10, /// T20,*OUTPUT =* 2F10.6 )
      1010  FORMAT (1H1,/// T20,*SEP VEHICLE:    MS       E       P        ISP
            X  TSEP   SEPK*,/ T34,F6.1,2X,F5.3,2X,F7.1,2X,F7.1,2X,F5.1,4X,F4.2,
            X// T20,*TUG VEHICLE:  TYPE        WS         WPA      WGA       EISP
            X REUSE*/
            X T31,A10,2X,F6.1,2X,F7.1,2X,F7.1,2X,F7.2,3X,F4.2//
            X                                              T20,*CONSTANTS:      MU
            X                G        SR       TR*/ T30,E15.9,2X,F8.4,2X,F5.2,4X,
            X F4.2)
      1020  FORMAT (1H0,/ T30,*OUTPUT RESULTS:*//T10,*MISSION   SAT. WT.(LBS)
            X           TIME (DAYS)       INT. ORBIT      RESIDUALS*/ T12,*NO.
            X   RETR    DEPLOY      DOWN     UP     SERV        ALT     INC     TIME
```

```
      Y   FUEL      EXIT  TUGS*)
 1030 FORMAT(22X,*NEXIT CODE:*,/22X,*1 = A NEW SEPS IS LAUNCHED AND PERF
     XORMS THE MISSION*,/22X,*2 = TUG ALONE DELIVERS MPLA AND NEW SEPS T
     XO MISSION ORBIT. (PHASING DONE BY SEPS)*,/22X,*3 = TUG CANNOT DELI
     XVER MPLA + NEW SEPS TO MIN. ALT.*,/22X,*4 = NEW SEPS CANNOT DELIVE
     XR MPLA.*,/22X,*5 = SEPS ALREADY IN ORBIT PERFORMS MISSION.*,/22X,*
     X6 = TUG ALONE PERFORMS DEL AND RETR, PHASING DONE BY SEPS.*,/22X,*
     X7 = TUG CANNOT DEL AND RETR PAYLOADS TO MIN. ALT.*,/22X,*8 = SEPS
     XALREADY IN ORBIT CANNOT COMPLETE RETR.*,/22X,*9 = SEPS ALREADY IN
     XORBIT CANNOT COMPLETE DEP.*,/22X,*10 = SEPS IS RETR ALONG WITH MPL
     XA*)
      END
```

# AEROSPACE CORPORATION

## INTERNAL DISTRIBUTION LIST

(REFERENCE: COMPANY PRACTICE 7-21-1)

REPORT TITLE

PROGRAM SEPSIM (SOLAR ELECTRIC PROPULSION STAGE SIMULATION)

| REPORT NO. | PUBLICATION DATE | SECURITY CLASSIFICATION |
|---|---|---|
| ATR-74(7341)-2 | 28 June 1974 | Unclassified |

(NOTE: FOR OFF-SITE PERSONNEL, SHOW LOCATION SYMBOL, e.g. JOHN Q. PUBLIC/VAFB)

J. R. Allder
J. M. Baker
H. K. Karrenberg
L. J. Kulakowski
T. J. Lang (4)
J. L. LeMay
L. R. Sitney
L. T. Stricker
G. W. Timpson
M. G. Wolfe
R. R. Wolfe
S. T. Wray

APPROVED BY Hans Karrenberg DATE 6/7/74

SHEET ____ OF ____

AEROSPACE CORPORATION

**EXTERNAL DISTRIBUTION LIST**

(REFERENCE: COMPANY PRACTICE 7-21-1)

| REPORT TITLE | | |
|---|---|---|
| PROGRAM SEPSIM (SOLAR ELECTRIC PROPULSION STAGE SIMULATION) | | |
| REPORT NO. ATR-74(7341)-2 | PUBLICATION DATE 28 June 1974 | SECURITY CLASSIFICATION Unclassified |

(NOTE: FOR OFF-SITE PERSONNEL, SHOW LOCATION SYMBOL, e.g. JOHN Q. PUBLIC/VAFB)

NASA Scientific & Technical
Information Facility (3)
P.O. Box 33
College Park, Maryland 20740

NASA - Headquarters
Washington, D.C. 20546

Mr. V. N. Huff, Code MTE (50)

R. R. Carley, Code MTE (2)

Dr. J. W. Wild, Code MTE (1)

NASA, Mr. Duncan Collins (1)
P. O. Box 92960
Worldway Postal Center
Los Angeles, Ca. 90009
Building 120, Room 1406B

APPROVED BY Robert R Wolf DATE 6/28/74

SHEET____ OF ____

AEROSPACE FORM 2394 REV 7-66