

Consiglio Nazionale delle Ricerche

**THE SIRIO - 2**

**FLIGHT DYNAMICS SYSTEM**

**General Description and Operating Guide**

A. Cardillo - A. Foni - A. Santoro

184

**CNUCE**

A cura di : A. Cardillo  
A. Foni  
A. Santoro

Copyright - Novembre 1981

by - CNUCE - Pisa

Istituto del Consiglio Nazionale delle Ricerche

**THE SIRIO-2 FLIGHT DYNAMICS SYSTEM**  
**General Description and Operating Guide**

**By : A. Cardillo**  
**A. Foni**  
**A. Santoro**

## List of contents.

1. Introduction.
  
2. FDS Overall Design.
  - 2.1 FDS Control Procedures.
  - 2.2 FDS File System.
  - 2.3 FDS Load Modules.
  - 2.4 VMOSIP Procedure.
  
3. Operating FDS.
  
4. The FDS exec procedures.
  - 4.1 FDS exec procedure.
  - 4.2 FDSPRT        ''
  - 4.3 ADP           ''
  - 4.4 ADPDEF       ''
  - 4.5 ASP           ''
  - 4.6 ASPDEF       ''
  - 4.7 OMP           ''
  - 4.8 OMPDEF       ''

## 1. Introduction.

The SIRIO-2 Flight Dynamics System (FDS) is an interactive system implemented under VM/370 - CMS, by means of which it is possible to operate a set of modules for SIRIO-2 spacecraft flight control. It has been obtained as modification of SIRIO FDS (ref. 1) of which maintains the functional capabilities. Being the SIRIO-2 FDS requested to control SIRIO-2 mission only during the geostationary phase, a simplification to the former system has been made in order to ensure the same SIRIO FDS capabilities with a simpler structure.

The reader of this document is supposed to be familiar with either the CMS-VM/370 Operating System, either with SIRIO FDS. For the System functional description see refs. (2), (3), (4), (5).

## 2. FDS Overall Design.

FDS has been designed so that each of its functions can be logically seen as a module composed of several EXEC procedures and one load module. Four modules can be distinguished in the FDS with associated the following functions:

- 1) determination of the spin axis attitude;
- 2) simulation of the time channels interval data;
- 3) computation of attitude/orbit maneuvers;
- 4) Eclipse/Interferences prediction.

Each module is a subsystem of the FDS, with an associated identification name as follows:

- 1) ADP - Attitude Determination Program
- 2) ASP - Attitude Simulation Program
- 3) OMP - Orbit/Attitude Maneuvers Program
- 4) LUNA - Eclipse/Interferences Program

Each one of the subsystems is completely independent from the others and can be executed under control of supervisor EXEC procedures.

A file system provides each subsystem with work areas and with the capability of exchanging data with other subsystems.

An overview of FDS organization is given in fig. 1.

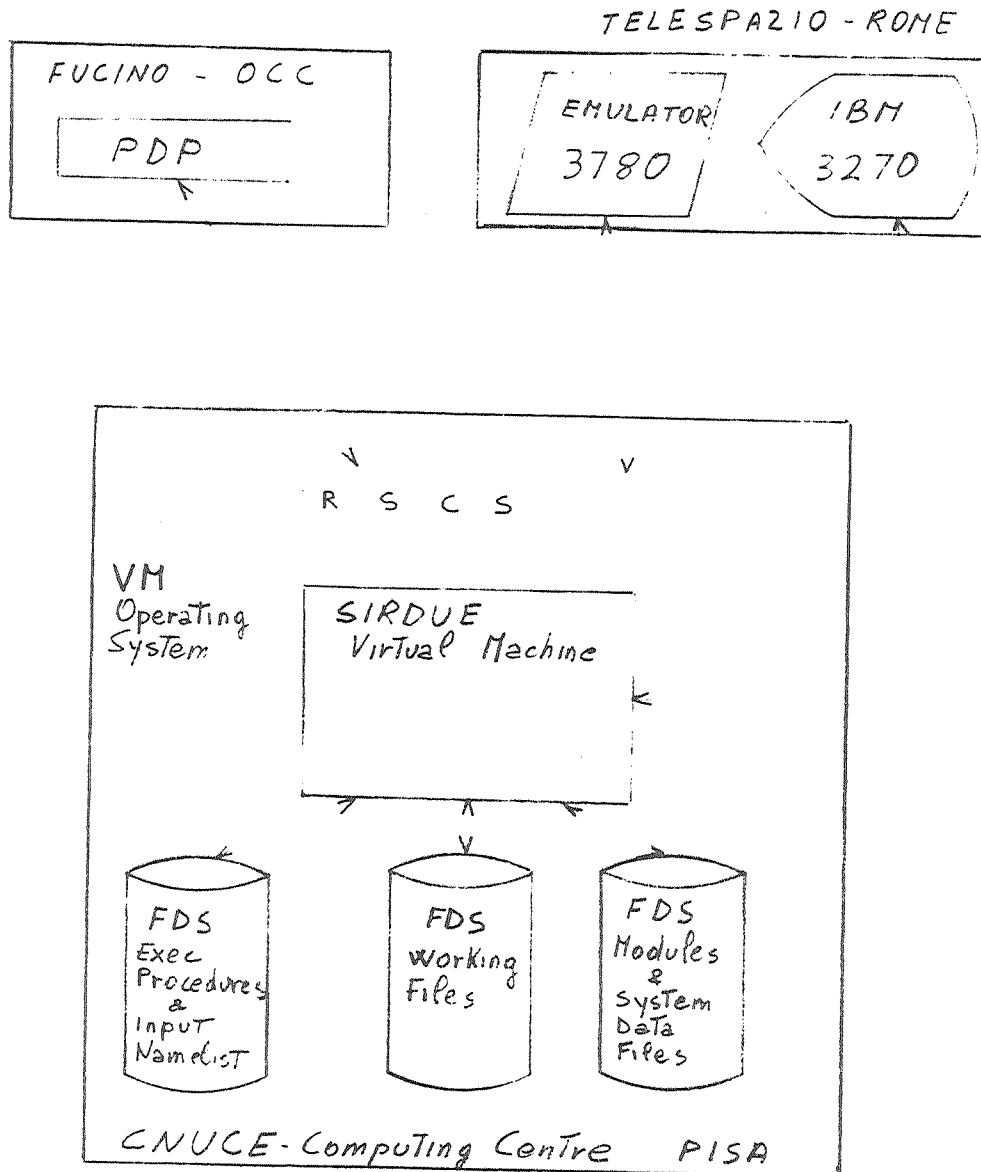


Fig. 1

## 2.1 FDS Control Procedures

The CMS command `'' FDS ''` given from SIRDUE virtual machine, enables the user to run any of the FDS subsystems. This command executes the FDS EXEC procedure, residing on the A-disk (191) of the SIRDUE virtual machine, in order to define all the hardware resources (i.e. disks and printers) needed for a correct execution of the system functions.

This allows the user to access FDS and to avoid the possibility of errors in defining the mentioned resources.

Once the virtual machine configuration has been set up (fig.2), the FDSXEC EXEC procedure, is automatically executed.

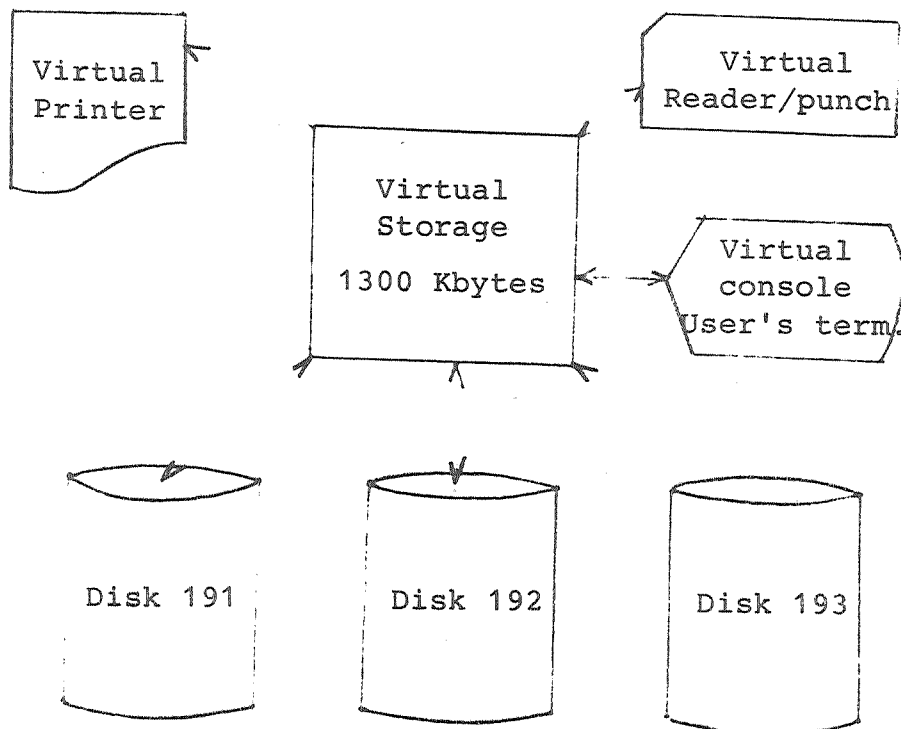


Fig. 2 - SIRDUE Virtual Machine



FDSXEC is a conversational procedure able to interpret and execute any FDS command. The FDS command accepted by the interpreter are either valid subsystem identification names followed by an optional parameter, or the VMOSIP or the END commands.

The operation flow inside each subsystem is the same for all the subsystems of the FDS.

When typing at the terminal keyboard the identification name of a subsystem an EXEC procedure of the same name is executed which can be accompanied by one of the following optional parameters:

ALL (default value)

EDIT

FILS

NOPT

1) When the option ALL (or when no option) is specified, the user is provided with the capability of editing a disk file containing the namelist parameters for the subsystem currently in execution. The file will be read at Load Module execution time and the namelist parameters will be initialized with these values.

The default definition of the User Defined Files is displayed and the user is made able to modify the default values from the terminal keyboard. The final file system for the subsystem is displayed with information on the file status. The user can then continue execution, restart the files definition procedure or exit from the subsystem

environment.

2) When the option EDIT is specified, the user is allowed to perform only the edit phase after which the execution continue automatically.

3) When the option FILS is specified the edit phase is bypassed and only file definition is allowed before execution of the Load Module.

4) When the option NOPT is specified, the execution of the Load Module starts automatically without user's intervention.

Upon return from the Load Module, the FDSPRT EXEC procedure takes over control in order to spool to the proper printing device the results of the operations performed.

## 2.2 FDS File System.

The system provides the user with a file system in order to be able to exchange information between the various subsystems of the FDS, to input data to a program by means of disk files, and to manage the working areas needed for the program execution.

A program indentifies a file by means of the virtual adress of a FORTRAN unit which has been made correspondent to the file through the FILEDEF command.

### 2.3 FDS Load Modules.

Four Load Modules exist which make up the system in order to perform the functions described in the previous paragraphs.

The Attitude Determination Program (ADP) has been developed to satisfy the attitude determination requirements of the SIRIO-2 spacecraft. The program module is composed of two main tasks; the first of these makes it possible to perform a deterministic computation of the attitude, the second gives an attitude estimate and sensor biases on the basis of a statistical model. The input data to the program are retrieved from the time channels information inside the telemetry frames acquired during a complete orbit. These data are the measurements of the spin period, the Sun colatitude, the Earth horizon crossing times and the dihedral angle.

The Attitude Simulator Program (ASP) is designed to work in conjunction with the ADP for testing purposes, prelaunch analyses and training mission support personnel. The main purpose of the ASP is to generate simulated telemetry data in order to enable the user to run FDS in simulation mode.

The Orbit Maneuver Program (OMP) has been designed to provide the SIRIO-2 mission support personnel with the possibility of computing and achieving desired on-station orbital maneuvers. Specifically, the OMP is able to compute the detailed maneuver scenarios and jet firing tables necessary to perform out-of-plane corrections, East-West

station keeping maneuvers, and simultaneous in-out-of-plane corrections. In addition, it is possible to reconstruct any maneuver given the jet firing sequence. This allows the calibration of maneuvers already performed or the fine tuning of proposed maneuvers.

For functional description of the ADP, ASP, OMP modules, see refs. (6), (7), (8).

LUNA program provides the user, for a given orbit and over a selected time period, with detailed tables concerning the Earth-Moon shadow occurrences during the satellite motion and Sun interferences on the ground station antenna.

#### 2.4 VMOSIP procedure.

VMOSIP is an FDS command which allows the user to make available to the FDS interactive programs a tape ephemeris file produced by the Orbit Determination Program running under OS/VS2 in batch mode.

In the following an example of VMOSIP command execution is given :

vmosip

FILE 'FILE PT03P001 A1 ' NOT POUND.  
TAPES DO NOT EXIST  
TAPE TO BE MOUNTED ? (YES, NO OR EXIT)  
yes  
TAPE ID:  
c129  
SCRIVI LA PASSWORD DEL NASTRO:  
TAPE 181 ATTACHED  
NASTRO C129 PRONTO TRA DUE MINUTI (TEMPO RESIDUO 120 MIN)  
CONTINUE (YES OR NO)  
yes  
FILE NUMBER:  
6  
ENTER FILE NAME FILE TYPE (DEFAULT IS 'ORBIT FILE')  
REPLACE ? (YES OR NO)  
yes

- ADP	- ATTITUDE DETERMINATION PROGRAM
- ASP	- ATTITUDE SIMULATOR PROGRAM
- LUNA	- ECLIPSE/INTERFERENCE PREDICTION PROGRAM
- OMP	- ORBIT MANEUVER PROGRAM
- VMOSIP	- COPY ORBIT FILE FROM TAPE
- END	- EXIT FROM PDS AND ENTER CMS ENVIRONMENT

end

## 3. Operating FDS.

To run SIRIO-2 FDS, the '' FDS '' command must be sent, after SIRDUE virtual machine has been activated. A typical displays sequence is shown in fig. 3 where OMP execution is invoked.

fds

```

*****
*           *
*   S I R I O - 2   *
*           *
*   F L I G H T     *
*           *
*   D Y N A M I C S *
*           *
*   S Y S T E M     *
*           *
*****

```

```

- ADP      - ATTITUDE DETERMINATION PROGRAM
- ASP      - ATTITUDE SIMULATOR PROGRAM
- LUNA     - ECLIPSE/INTERFERENCE PREDICTION PROGRAM
- OMP      - ORBIT MANEUVER PROGRAM
- VHOSIP   - COPY ORBIT FILE FROM TAPE
- END      - EXIT FROM FDS AND ENTER CMS ENVIRONMENT

```

omp

Fig. 3

## OMP USER DEFINED FILES

FILEDEF 20 ORBIT FILE B4  
 FILEDEF 52 SCEND FILE D1  
 FILEDEF 53 INTEPHEM FILE D4  
 FILEDEF 88 SCDATA FILE D1

ANY CHANGE? (YES OR NO)

no

## OMP FINAL FILE SYSTEM

FT01F001	TERMINAL			
FT02F001	DISK	OMP	NAMLST	A1
FT05F001	TERMINAL			
FT06F001	DISK	FILE	FT06F001	A1
FT08F001	DISK	FILE	FT08F001	A1
FT14F001	DISK	SLP	FILE	D1
FT20F001	DISK	ORBIT	FILE	B4
FT25F001	DISK	STATUS	FILE	D1
FT48F001	DISK	TINCOP	FILE	D1
FT52F001	DISK	SCEND	FILE	D1
FT53F001	DISK	INTEPHEM	FILE	D4
FT70F001	DISK	MESSAGES	FILE	D1
FT88F001	DISK	SCDATA	FILE	D1
FT93F001	DISK	WTABLE	FILE	D4
FT94F001	DISK	STATION	FILE	D1
FT95F001	DISK	WCMDSH	FILE	D1

FILE 'STATUS FILE D1' NOT FOUND.  
 FILE STATUS FILE D1 NOT FOUND  
 FILE SCEND FILE D1 ALREADY EXISTS ON DISK  
 FILE 'INTEPHEM FILE D4' NOT FOUND.

(CONT, REST OR EXIT)

cont

```

*****
*
*   ORBIT   MANEUVER   PROGRAM
*
*           NOW   READY
*
*****

```

Fig. 3

When no questions or lists are displayed (usually masks or warning messages are presented), the Carriage Return (or Enter) keys, must be stroked to continue the FDS execution.

In case the user gives a wrong or not allowed answer to the actual display, the same is presented again for a new user's intervention.

In the following the complete display sequences related to the request of ADP and ASP executions, are presented.

It must be noted that VMOSIP and LUNA execution requestes, cause the immediate execution of the respective modules.



## 4. - FDS exec procedures.

## 4.1 - FDS exec procedure.

```
&CONTROL OFF
ACC 192 B
ACC 193 D
SCREEN
&BEGTYPE
```

```
*****
*           *
*   S I R I O - 2   *
*           *
*   F L I G H T   *
*           *
*   D Y N A M I C S   *
*           *
*   S Y S T E M   *
*           *
*****
```

```
&END
&READ ARGS
-IN SCREEN
&BEGTYPE
```

```
- ADP      - ATTITUDE DETERMINATION PROGRAM
- ASP      - ATTITUDE SIMULATOR PROGRAM
- LUNA     - ECLIPSE/INTERFERENCE PREDICTION PROGRAM
- OMP      - ORBIT MANEUVER PROGRAM
- VHOSIP   - COPY ORBIT FILE FROM TAPE
- END      - EXIT FROM FDS AND ENTER CMS ENVIRONMENT
```

```
&END
&READ ARGS
&IF &INDEX EQ 0 &GOTO -IN
&IF &INDEX GT 2 &GOTO -IN
&IN1 = &1
&IN2 = &2
&IF &IN1 EQ VHOSIP &GOTO -VMO
&IF &IN1 EQ ADP &GOTO -ADP
&IF &IN1 EQ ASP &GOTO -ASP
&IF &IN1 EQ LUNA &GOTO -LUNA
```

```
&IF &IN1 EQ OMP &GOTO -OMP
&IF &IN1 EQ END &EXIT
&GOTO -IN
-VMO EXEC VHOSIP
&IF &RETCODE NE 0 &TYPE NOT ON SYSTEM
&READ ARGS
&GOTO -IN
-ADP &GLOBAL2 = 0
EXEC ADP &IN2
&GOTO -IN
-OMP EXEC OMP &IN2
&GOTO -IN
-LUNA EXEC LUNA &IN2
&GOTO -IN
-ASP &GLOBAL2 = 0
EXEC ASP &IN2
&GOTO -IN
```

## 4.2 FDSPRT exec procedure.

```
&CONTROL OFF
&IF &GLOBAL LT 3 &GOTO -ERR
-CONT CP SPOOL E CONT
FILEDEF F6 DISK FILE FT06F001 A1
&IF &1 EQ OMP &GOTO -POMP
FILEDEF F7 DISK FILE FT07F001 A1
FILEDEF F8 DISK FILE FT08F001 A1
FILEDEF F9 DISK FILE FT09F001 A1
&IF &1 EQ ADP &GOTO -PTPR
FILEDEF F10 DISK FILE FT10F001 B1
&GOTO -PTPR
-POMP FILEDEF F8 DISK FILE FT08F001 A1
-PTPR FILEDEF PT PR (RECFM VBA LRECL 137 BLKSIZE 141
MOVE F6 PT
&IF &1 EQ OMP &GOTO -POTTO
MOVE F7 PT
-POTTO MOVE F8 PT
&IF &1 EQ OMP &GOTO -END
MOVE F9 PT
&IF &1 EQ ADP &GOTO -END
MOVE F10 PT
-END CP SPOOL E NOCONT
CP CLOSE E DIST &1
&EXIT
-ERR EXEC UNENTRY FDSPRT
```

## 4.3 ADP exec procedure.

```

&CONTROL OFF
&IF &GLOBAL LT 2 &GOTO -ERR
ERASE FILE FT06F001
ERASE FILE FT07F001 B1
ERASE FILE FT08F001 B1
ERASE FILE FT09F001 B1
&GLOBAL1 = 1
&IF &INDEX EQ 0 &GOTO -ALL
&IF &INDEX NE 1 &GOTO -ERP
&IN = &1
&IF &IN EQ ALL &GOTO -ALL
&IF &IN EQ EDIT &GOTO -EDT
&IF &IN EQ FILS &GOTO -FLS
&IF &IN EQ NOPT &GOTO -FLD
&GOTO -ERP
-ALL &IN = ALL
-EDT SCREEN
TED ADP NAMLST A1
&IF &IN NE ALL &GOTO -FLD
-FLS EXEC ADPDEF
&IF &GLOBAL1 EQ 0 &GOTO -EX
&GOTO -GO
-FLD FILEDEF 1 TERM
FILEDEF 2 DISK ADP NAMLST A1
FILEDEF 5 TERM
FILEDEF 6 DISK FILE FT06F001 A1 (RECFM VBA LRECL 137 BLKSIZE 141)
FILEDEF 7 DISK FILE FT07F001 A1 (RECFM VBA LRECL 137 BLKSIZE 141)
FILEDEF 8 DISK FILE FT08F001 A1 (RECFM VBA LRECL 137 BLKSIZE 141)
FILEDEF 9 DISK FILE FT09F001 A1 (RECFM VBA LRECL 137 BLKSIZE 141)
FILEDEF 14 DISK TLMTRY FILE B1 (RECFM FB LRECL 312 BLKSIZE 312)
FILEDEF 15 DISK OABIAS FILE B1 (RECFM FB LRECL 50 BLKSIZE 50)
FILEDEF 16 DISK ORBIT FILE B4 (RECFM VS LRECL 1096 BLKSIZE 1100)
FILEDEF 88 DISK SLP FILE D1 (DSORG DA XTENT 400 LRECL 2264 BLKSIZE 2264)
FILEDEF 99 DISK FILE FT99F001 A1 (RECFM F LRECL 128 BLKSIZE 128)
-GO ADPLOAD
-PRT SCREEN
&TYPE OUTPUT ON PRINTER? (YES OR NO)
&READ ARGS
&IF &INDEX NE 1 &GOTO -PRT
&IF &1 EQ NO &GOTO -EX
&IF &1 NE YES &GOTO -PRT
EXEC FDSPRT ADP
-EX &IF &GLOBAL2 EQ 0 &GOTO -EX1
CP SPOOL D SIRIO2
PUN FILE FT99F001
CP SPOOL D *
-EX1 &EXIT
-ERP &TYPE INVALID PARM(S) &1 &2
&READ ARGS
&GOTO -EX
-ERR EXEC UNENTRY ADP

```

## 4.4 ADPDEF exec procedure.

```

&CONTROL OFF
&IF &GLOBAL LT 3 &GOTO -ERR1
&N1 = TLMTRY
&T1 = FILE
&M1 = B1
&N2 = OABIAS
&T2 = FILE
&M2 = B1
&N3 = ORBIT
&T3 = FILE
&M3 = B4
-FILDES FILEDEF * CLEAR
SCREEN
&BEGTYPE
      ADP USER DEFINED FILES
&END
&TYPE FILEDEF 14 &N1 &T1 &M1
&TYPE FILEDEF 15 &N2 &T2 &M2
&TYPE FILEDEF 16 &N3 &T3 &M3
-CH? &BEGTYPE
      ANY CHANGE? (YES OR NO)
&END
&READ ARGS
&IF &INDEX NE 1 &GOTO -CH?
&IF &1 EQ NO &GOTO -DEF
&IF &1 NE YES &GOTO -CH?
SCREEN
-CHA &TYPE ENTER NO., FN, FT, FM OR END
&READ ARGS
&IF &INDEX EQ 0 &GOTO -ERR
&IF &INDEX GT 4 &GOTO -ERR
&IF &1 EQ END &GOTO -DEF
&IF &INDEX EQ 1 &GOTO -ERR
&IF &1 EQ 14 &GOTO -C14
&IF &1 EQ 15 &GOTO -C15
&IF &1 EQ 16 &GOTO -C16
-ERR &TYPE NOT ALLOWED
&GOTO -CHA
-C14 &N1 = &2
&IF &INDEX EQ 2 &GOTO -CHA
&T1 = &3
&IF &INDEX EQ 3 &GOTO -CHA
&M1 = &4
&GOTO -CHA
-C15 &N2 = &2
&IF &INDEX EQ 2 &GOTO -CHA
&T2 = &3
&IF &INDEX EQ 3 &GOTO -CHA
&M2 = &4
&GOTO -CHA
-C16 &N3 = &2
&IF &INDEX EQ 2 &GOTO -CHA
&T3 = &3
&IF &INDEX EQ 3 &GOTO -CHA
&M3 = &4

```

```
&GOTO -CHA
-DEF SCREEN
FILEDEF 1 TERM
FILEDEF 2 DISK ADP NAMLIST A1
FILEDEF 5 TERM
FILEDEF 6 DISK FILE FT06F001 A1 (RECFM VBA LRECL 137 BLKSIZE 141
FILEDEF 7 DISK FILE FT07F001 A1 (RECFM VBA LRECL 137 BLKSIZE 141
FILEDEF 8 DISK FILE FT08F001 A1 (RECFM VBA LRECL 137 BLKSIZE 141
FILEDEF 9 DISK FILE FT09F001 A1 (RECFM VBA LRECL 137 BLKSIZE 141
FILEDEF 14 DISK &N1 &T1 &M1 (RECFM FB LRECL 312 BLKSIZE 312
FILEDEF 15 DISK &N2 &T2 &M2 (RECFM FB LRECL 50 BLKSIZE 50
FILEDEF 16 DISK &N3 &T3 &M3 (RECFM VS LRECL 1096 BLKSIZE 1100
FILEDEF 88 DISK SLP FILE D1 (DSORG DA XTENT 400 LRECL 2264 BLKSIZE 2264
FILEDEF 99 DISK FILE FT99F001 A1 (RECFM P LRECL 128 BLKSIZE 128
&BEGTYPE
      ADP FINAL FILE SYSTEM
&END
FILEDEF
&TYPE
STATE &N1 &T1 &M1
&IF &RETCODE NE 0 &TYPE FILE &N1 &T1 &M1 NOT FOUND
STATE &N2 &T2 &M2
&IF &RETCODE NE 0 &TYPE FILE &N2 &T2 &M2 NOT FOUND
STATE &N3 &T3 &M3
&IF &RETCODE NE 0 &TYPE FILE &N3 &T3 &M3 NOT FOUND
STATE SLP FILE D1
&IF &RETCODE NE 0 &TYPE FILE SLP FILE D1 NOT FOUND
-TDC &BEGTYPE
      (CONT, REST OR EXIT)
&END
&READ ARGS
&IF &INDEX NE 1 &GOTO -TDC
&IF &1 EQ EXIT &GOTO -EX
&IF &1 EQ REST &GOTO -FILDES
&IF &1 NE CONT &GOTO -TDC
&EXIT
-EX &GLOBAL1 = 0
FILEDEF * CLEAR
&EXIT
-ERR1 EXEC UNENTRY ADPDEF
```

## 4.5 ASP exec procedure.

```

&CONTROL OFF
&IF &GLOBAL LT 2 &GOTO -ERR
ERASE FILE PT06P001
ERASE FILE PT07P001
ERASE FILE PT08P001
&GLOBAL1 = 1
&IF &INDEX EQ 0 &GOTO -ALL
&IF &INDEX NE 1 &GOTO -ERP
&IN = &1
&IF &IN EQ ALL &GOTO -ALL
&IF &IN EQ EDIT &GOTO -EDT
&IF &IN EQ PLS &GOTO -PLS
&IF &IN EQ NOPT &GOTO -PLD
&GOTO -ERP
-ALL &IN = ALL
-EDT SCREEN
TED ASP NAMLST A1
&IF &IN NE ALL &GOTO -PLD
-PLS EXEC ASPDEF
&IF &GLOBAL1 EQ 0 &GOTO -EX
&GOTO -GO
-PLD FILEDEF 1 TERM
FILEDEF 2 DISK ASP NAMLST A1
FILEDEF 5 TERM
FILEDEF 6 DISK FILE PT06P001 A1 (RECFM VBA LRECL 137 BLKSIZE 141
FILEDEF 7 DISK FILE PT07P001 A1 (RECFM VBA LRECL 137 BLKSIZE 141
FILEDEF 8 DISK FILE PT08P001 A1 (RECFM VBA LRECL 137 BLKSIZE 141
FILEDEF 14 DISK TLMTRY FILE B1 (RECFM FB LRECL 312 BLKSIZE 312
FILEDEF 15 DISK OABIAS FILE B1 (RECFM FB LRECL 50 BLKSIZE 50
FILEDEF 16 DISK ORBIT FILE B4 (RECFM VS LRECL 1096 BLKSIZE 1100
FILEDEF 88 DISK SLP FILE D1 (DSORG DA XTENT 400 LRECL 2264 BLKSIZE 2264
-GO ASPLOAD
-PRT SCREEN
&TYPE OUTPUT ON PRINTER? (YES OR NO)
&READ ARGS
&IF &INDEX NE 1 &GOTO -PRT
&IF &1 EQ NO &GOTO -EX
&IF &1 NE YES &GOTO -PRT
EXEC FDSPRT ASP
-EX &EXIT
-ERP &TYPE INVALID PARM(S) &1 &2
&READ ARGS
&GOTO -EX
-ERR EXEC UNENTRY ASP

```

## 4.6 ASPDEF exec procedure.

```

&CONTROL OFF
&IF &GLOBAL LT 3 &GOTO -ERR1
&N2 = TLMTRY
&T2 = FILE
&M2 = B1
&N4 = OABIAS
&T4 = FILE
&M4 = B1
&N3 = ORBIT
&T3 = FILE
&M3 = B4
-FILDES FILEDEF * CLEAR
SCREEN
&BEGTYPE
    ASP USER DEFINED FILES
&END
&TYPE FILEDEF 14 &N2 &T2 &M2
&TYPE FILEDEF 15 &N4 &T4 &M4
&TYPE FILEDEF 16 &N3 &T3 &M3
-CH? &BEGTYPE
    ANY CHANGE? (YES OR NO)
&END
&READ ARGS
&IF &INDEX NE 1 &GOTO -CH?
&IF &1 EQ NO &GOTO -DEF
&IF &1 NE YES &GOTO -CH?
SCREEN
-CHA &TYPE ENTER NO., FN, FT, PH OR END
&READ ARGS
&IF &INDEX EQ 0 &GOTO -ERR
&IF &INDEX GT 4 &GOTO -ERR
&IF &1 EQ END &GOTO -DEF
&IF &INDEX EQ 1 &GOTO -ERR
&IF &1 EQ 14 &GOTO -C14
&IF &1 EQ 15 &GOTO -C15
&IF &1 EQ 16 &GOTO -C16
-ERR &TYPE NOT ALLOWED
&GOTO -CHA
-C14 &N2 = &2
&IF &INDEX EQ 2 &GOTO -CHA
&T2 = &3
&IF &INDEX EQ 3 &GOTO -CHA
&M2 = &4
&GOTO -CHA
-C15 &N4 = &2
&IF &INDEX EQ 2 &GOTO -CHA
&T4 = &3
&IF &INDEX EQ 3 &GOTO -CHA
&M4 = &4
&GOTO -CHA
-C16 &N3 = &2
&IF &INDEX EQ 2 &GOTO -CHA
&T3 = &3
&IF &INDEX EQ 3 &GOTO -CHA
&M3 = &4

```



```

&GOTO -CHA
-DEF SCREEN
FILEDEF 1 TERM
FILEDEF 2 DISK ASP NAMLIST A1
FILEDEF 5 TERM
FILEDEF 6 DISK FILE FT06F001 A1 (RECFM VBA LRECL 137 BLKSIZE 141
FILEDEF 7 DISK FILE FT07F001 A1 (RECFM VBA LRECL 137 BLKSIZE 141
FILEDEF 8 DISK FILE FT08F001 A1 (RECFM VBA LRECL 137 BLKSIZE 141
FILEDEF 14 DISK &N2 &T2 &M2 (RECFM FB LRECL 312 BLKSIZE 312
FILEDEF 15 DISK &N4 &T4 &M4 (RECFM FB LRECL 50 BLKSIZE 50
FILEDEF 16 DISK &N3 &T3 &M3 (RECFM VS LRECL 1096 BLKSIZE 1100
FILEDEF 88 DISK SLP FILE D1 (DSORG DA XTENT 400 LRECL 2264 BLKSIZE 2264
&BEGTYPE
    ASP FINAL FILE SYSTEM
&END
FILEDEF
&TYPE
STATE &N2 &T2 &M2
&IF &RETCODE EQ 0 &TYPE FILE &N2 &T2 &M2 ALREADY EXISTS ON DISK
STATE &N4 &T4 &M4
&IF &RETCODE EQ 0 &TYPE FILE &N4 &T4 &M4 ALREADY EXISTS ON DISK
STATE &N3 &T3 &M3
&IF &RETCODE NE 0 &TYPE FILE &N3 &T3 &M3 NOT FOUND
STATE SLP FILE D1
&IF &RETCODE NE 0 &TYPE FILE SLP FILE D1 NOT FOUND
-TDC &BEGTYPE
    (CONT, REST OR EXIT)
&END
&READ ARGS
&IF &INDEX NE 1 &GOTO -TDC
&IF &1 EQ EXIT &GOTO -EX
&IF &1 EQ REST &GOTO -FILDES
&IF &1 NE CONT &GOTO -TDC
&EXIT
-EX &GLOBAL1 = 0
FILEDEF * CLEAR
&EXIT
-ERR1 EXEC UNENTRY ASPDEF

```

## 4.7 OMP exec procedure.

```

&CONTROL OFF
&IF &GLOBAL LT 2 &GOTO -ERR
&IF &INDEX EQ 0 &GOTO -ALL
&IF &INDEX NE 1 &GOTO -ERP
&IN = &1
&IF &IN EQ ALL &GOTO -ALL
&IF &IN EQ EDIT &GOTO -EDT
&IF &IN EQ FILS &GOTO -FLS
&IF &IN EQ NOPT &GOTO -FLD
&GOTO -ERP
-ALL &IN = ALL
-EDT SCREEN
TED OMP NAMLIST A1
&IF &IN NE ALL &GOTO -FLD
-FLS EXEC OMPDEF
&IF &GLOBAL1 EQ 0 &GOTO -EX
&GOTO -GO
-FLD FILEDEF 1 TERM
FILEDEF 2 DISK OMP NAMLIST A1 (LRECL 80 BLKSIZE 80 RECFM FB
FILEDEF 5 TERM
FILEDEF 6 DISK FILE FT06F001 A1 (RECFM VBA LRECL 137 BLKSIZE 141
FILEDEF 8 DISK FILE FT08F001 A1 (RECFM VBA LRECL 137 BLKSIZE 141
FILEDEF 14 DISK SLP FILE D1 (DSORG DA XTENT 400 LRECL 2264 BLKSIZE 2264
FILEDEF 20 DISK ORBIT FILE B4 (RECFM VS LRECL 1096 BLKSIZE 1100
FILEDEF 48 DISK TIMCOF FILE D1 (DSORG DA XTENT 62 LRECL 248 BLKSIZE 248
FILEDEF 52 DISK SCEND FILE D1 (RECFM VBS LRECL 692 BLKSIZE 3414
FILEDEF 53 DISK INTEPHM FILE D4 (RECFM VS LRECL 2804 BLKSIZE 2808
FI 70 DISK MESSAGES FILE D1 (DSORG DA XTENT 100 LRECL 216 BLKSIZE 216
FILEDEF 88 DISK SCDATA FILE D1 (RECFM FB LRECL 80 BLKSIZE 3200
FILEDEF 93 DISK WTABLE FILE D4 (RECFM VS LRECL 804 BLKSIZE 808
FI 94 DISK STATION FILE D1 (DSORG DA XTENT 201 LRECL 4004 BLKSIZE 4004
FILEDEF 95 DISK WCHDSH FILE D1 (RECFM VBS LRECL 712 BLKSIZE 3568
-GO OMPLOAD
-PRT SCREEN
&TYPE OUTPUT ON PRINTER? (YES OR NO)
&READ ARGS
&IF &INDEX NE 1 &GOTO -PRT
&IF &1 EQ NO &EXIT
&IF &1 NE YES &GOTO -PRT
EXEC FDSPRT OMP
-EX &EXIT
-ERP &TYPE INVALID PARAM(S) &1 &2
&READ ARGS
&EXIT
-ERR EXEC UNENTRY OMP

```

## 4.8 OMPDEF exec procedure.

```

&CONTROL OFF
&N2 = SCEND
&T2 = FILE
&M2 = D1
&N4 = INTEPHEN
&T4 = FILE
&M4 = D4
&N5 = SCDATA
&T5 = FILE
&M5 = D1
&N3 = ORBIT
&T3 = FILE
&M3 = B4
-FILDES FILEDEF * CLEAR
SCREEN
&BEGTYPE

```

## OMP USER DEFINED FILES

```

&END
&TYPE FILEDEF 20 &N3 &T3 &M3
&TYPE FILEDEF 52 &N2 &T2 &M2
&TYPE FILEDEF 53 &N4 &T4 &M4
&TYPE FILEDEF 88 &N5 &T5 &M5
-CH? &BEGTYPE

```

## ANY CHANGE? (YES OR NO)

```

&END
&READ ARGS
&IF &INDEX NE 1 &GOTO -CH?
&IF &1 EQ NO &GOTO -DEF
&IF &1 NE YES &GOTO -CH?
SCREEN
-CHA &TYPE ENTER NO., FN, PT, FM OR END
&READ ARGS
&IF &INDEX EQ 0 &GOTO -ERR
&IF &INDEX GT 4 &GOTO -ERR
&IF &1 EQ END &GOTO -DEF
&IF &INDEX EQ 1 &GOTO -ERR
&IF &1 EQ 52 &GOTO -C52
&IF &1 EQ 53 &GOTO -C53
&IF &1 EQ 20 &GOTO -C20
&IF &1 EQ 88 &GOTO -C88
-ERR &TYPE NOT ALLOWED
&GOTO -CHA
-D45 &N1 = &2
-C52 &N2 = &2
&IF &INDEX EQ 2 &GOTO -CHA
&T2 = &3
&IF &INDEX EQ 3 &GOTO -CHA
&M2 = &4
&GOTO -CHA
-C53 &N4 = &2

```

```

&IF &INDEX=EQ 2 &GOTO -CHA
&T4 = &3
&IF &INDEX EQ 3 &GOTO -CHA
&M4 = &4
&GOTO -CHA
-C20 &N3 = &2
&IF &INDEX EQ 2 &GOTO -CHA
&T3 = &3
&IF &INDEX EQ 3 &GOTO -CHA
&M3 = &4
&GOTO -CHA
-C88 &N5 = &2
&IF &INDEX EQ 2 &GOTO -CHA
&T5 = &3
&IF &INDEX EQ 3 &GOTO -CHA
&M5 = &4
&GOTO -CHA
-DEF SCREEN
-PLD FILEDEF 1 TERM
FILEDEF 2 DISK OMP NAHLST A1 (LRECL 80 BLKSIZE 80 RECFM FB
FILEDEF 5 TERM
FILEDEF 6 DISK FILE PT06F001 A1 (RECFM VBA LRECL 137 BLKSIZE 141
FILEDEF 8 DISK FILE PT08F001 A1 (RECFM VBA LRECL 137 BLKSIZE 141
FILEDEF 14 DISK SLP FILE D1 (DSORG DA XTENT 400 LRECL 2264 BLKSIZE 2264
FILEDEF 20 DISK &N3 &T3 &M3 (RECFM VS LRECL 1096 BLKSIZE 1100
FILEDEF 25 DISK STATUS FILE D1 (XTENT 110 DSORG DA LRECL 204 BLKSIZE 20
*FILEDEF 45 DISK &N1 &T1 &M1 (RECFM FB LRECL 80 BLKSIZE 800
FILEDEF 48 DISK TINCOP FILE D1 (DSORG DA XTENT 62 LRECL 248 BLKSIZE 248
FILEDEF 52 DISK &N2 &T2 &M2 (RECFM VBS LRECL 692 BLKSIZE 3414
FILEDEF 53 DISK &N4 &T4 &M4 (RECFM VS LRECL 2804 BLKSIZE 2808
PI 70 DISK MESSAGES FILE D1 (DSORG DA XTENT 100 LRECL 216 BLKSIZE 216
FILEDEF 88 DISK &N5 &T5 &M5 (RECFM PB LRECL 80 BLKSIZE 3200
FILEDEF 93 DISK WTABLE FILE D4 (RECFM VS LRECL 804 BLKSIZE 808
PI 94 DISK STATION FILE D1 (DSORG DA XTENT 201 LRECL 4004 BLKSIZE 4004
FILEDEF 95 DISK WCHDSH FILE D1 (RECFM VBS LRECL 712 BLKSIZE 3568
&BEGTYPE

```

## OMP FINAL FILE SYSTEM

```

&END
FILEDEF
&TYPE
STATE SLP FILE D1
&IF &RETCODE NE 0 &TYPE FILE SLP FILE D1 NOT FOUND
STATE &N3 &T3 &M3
&IF &RETCODE NE 0 &TYPE FILE &N3 &T3 &M3 NOT FOUND
STATE STATUS FILE D1
&IF &RETCODE NE 0 &TYPE FILE STATUS FILE D1 NOT FOUND
*STATE &N1 &T1 &M1
STATE TINCOP FILE D1
&IF &RETCODE NE 0 &TYPE FILE TINCOP FILE D1 NOT FOUND
STATE &N2 &T2 &M2
&IF &RETCODE EQ 0 &TYPE FILE &N2 &T2 &M2 ALREADY EXISTS ON DISK
STATE &N4 &T4 &M4
&IF &RETCODE EQ 0 &TYPE FILE &N4 &T4 &M4 ALREADY EXISTS
STATE MESSAGES FILE D1

```

```
&IF &RETCODE NE 0 &TYPE FILE MESSAGES FILE D1 NOT FOUND  
STATE &N5 &T5 &M5  
&IF &RETCODE NE 0 &TYPE FILE SCDATA FILE D1 NOT FOUND  
STATE STATION FILE D1  
&IF &RETCODE NE 0 &TYPE FILE STATION FILE D1 NOT FOUND  
-TDC &BEGTYPE
```

(CONT, REST OR EXIT)

```
&END  
&READ ARGS  
&IF &INDEX NE 1 &GOTO -TDC  
&IF &1 EQ EXIT &GOTO -EX  
&IF &1 EQ REST &GOTO -FILDES  
&IF &1 NE CONT &GOTO -TDC  
&EXIT  
-EX &GLOBAL1 = 0  
FILEDEF * CLEAR  
&EXIT  
-ERR1 EXEC UMENTRY OMPDEF
```

## References.

- (1) - The Flight Dynamics System for the control of the SIRIO Spacecraft during its operational life. G. Faconti, S. Trumpy. CNUCE N. 122.
- (2) - IBM Virtual Machine Facility/370 : Introduction, File N. S370-20, Order N. GC20-1800-2
- (3) - IBM System/360 Operating System Introduction, File N. S360-20, Order N. GC28-6534-3.
- (4) - IBM Virtual Machine Facility/370 : Remote Spooling Communication Subsystem (RSCS) User's Guide, File N. S370-30, Order N. GC20-1816-0.
- (5) - Flight Dynamics System Installation Reference Material and Operating Guide. G. Faconti. CNUCE N. 139.
- (6) - SIRIO Attitude Determination Program (ADP) Manuale d'uso. G. Faconti, M. Lucchesi, G. Pasquinelli. CNUCE N. 129.
- (7) - Attitude Simulation Program (ASP) User's Manual. G. Faconti, P. Guidotti. CNUCE N. 136.
- (8) - Orbit Maneuver Program, Manuale d'uso. N. Celandroni, A. Foni. CNUCE N. 140.