5/19/80

Multisatellite Fourier Analysis Program
PIONEER System Documentation

I. Overview

The satellite-dependent (SD) PIONEER routines allow the PIONEER 10 and 11 data to be processed through the Fourier program's analysis and output procedures. The satellite-independent (SI) routines are contained in separate source and load modules from all SD code, and thus, any satellite may be linked via JCL.

This document describes the SD internal code description for PIONEER Fourier analysis. The main document containing the SI description must be reviewed prior to this one.

II. Input Required

1. Satellite independent namelist INPUT

a. The RATES parameter may have the following values:

RATES Parameter	Rate Signified
SR1A	A1 A2 B CI ¬CIII
SR1B	A2 B K1 ¬CIII
SR1C	DI DII ¬F
SR1D	DI DOT E1 ¬F
SR2A	SI5 ¬SIII ¬SIIA ¬SIII
SR2B	SI6 ¬SII ¬SIIA ¬SII
SR2C	SI7 ¬SII ¬SIIA ¬SII
SR2D	SI8 ¬SII ¬SIIA ¬SII
SR2E	¬SI SII5 ¬SIIA ¬SIII
SR2F	¬SI SII6 ¬SIIA ¬SIII
SR2G	¬SI SII6 ¬SIIA ¬SIII
SR2H	¬SI SII7 ¬SIIA ¬SIII

b. The parameter SATID must be either 'PIONEER F' or 'PIONEER G'.

2. PIONEER namelist PIO

&PIO INTAPE,SRCE,ZVOL

This namelist must appear after each namelist set of the SI routines.

Name	Type	Default	Description
INTAPE	I*4	2	1=user-input flux tape 2=search catalog for flux tape with user-specified times

SRCE	A4	blank	The flux catalog source name
ZVOL	A8	blank	Volume-serial name of the user-input flux tape if INTAPE=1

3. Flux Tape

This is a standard label, variable length record tape in the PIONEER flux tape format.[1] It is either user provided or provided by the flux catalog. The tape uses unit 9, with DSN = PIOFLUX.

4. Flux Catalog

The flux catalog[1] is used only if INTAPE=T in the INPUT namelist. The catalog file name must be entered in the JCL for unit 30. Currently, PIONEER catalog is in:

'SBPIO.FLUXCAT.DATA'

III. Output Generated

(See Fourier Plot Program SI Documentation)

IV. Module Documentation

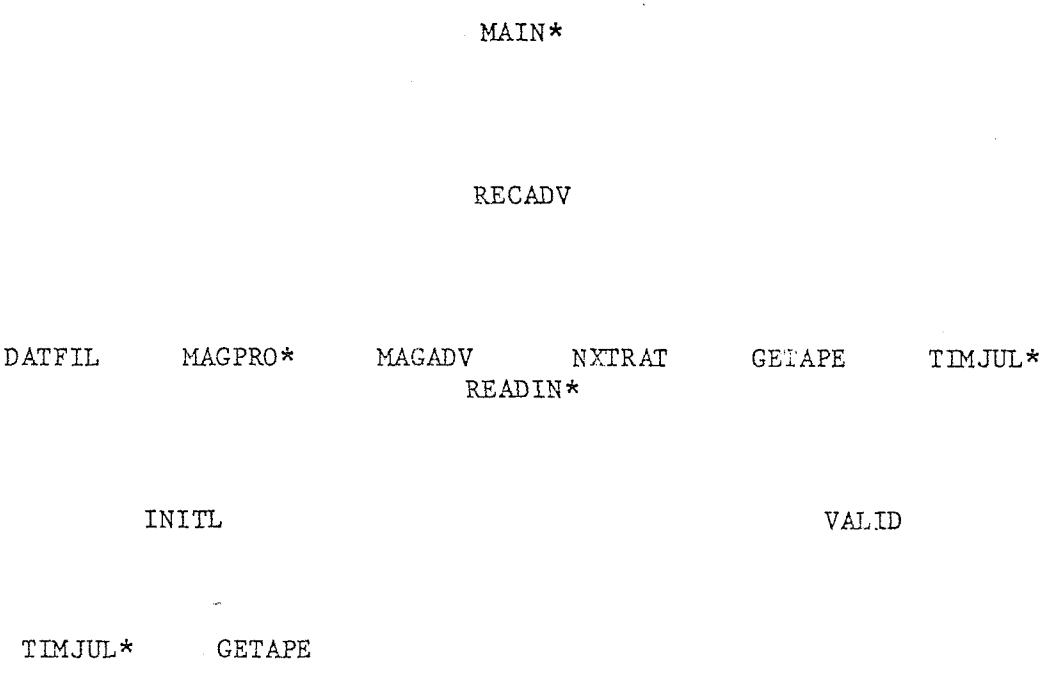
Module	Description
DATFIL	Fills the counts and time arrays, checking for valid data
GETAPE	Finds and mounts a flux tape
INITL	Initialized I/O devices, common blocks, and reads in the SD namelist PIO
NXTRAT	Skips seven records on the flux tape so the next one read is a rates record
MAGADV	Reads an averaging interval of magnetic field data
RECADV	Reads in one rate and magnetic field data average into the RATDAT and MAGFLD commons
VALID	Validates the input namelist data for the SI READIN module

[1] "PIONEER/HELIOS, Flux Data Base Generator (FLXDBG) Maintenance Programmer's Introduction", CSC Document, 1978.

V. Program Structure

A. Block Diagram

*=Satellite independent module



B. Algorithm

The SI module READIN reads the SI namelist and calls VALID to validate them. Then it calls INITL to read in the SD namelist PIO and initialize the I/O devices. Control is returned to MAIN which succesively calls RECADV to accummulate one average point of sector counts and if desired, magnetic field data. RECADV reads in a flux record, stores it (DATFIL), and collects magnetic field data if desired (MAGPRO, MAGADV). Before reading the next rates record, six non-rates records are skipped. (NXTRAT).

C. Error Handling

The following return codes and messages may be printed:

Return Code	Description
-	'I/O ERROR WHILE SPACING TO NEXT RECORD' This flags a tape read error on the flux tape.
2	'CAN NOT FIND THE SPACECRAFT ID IN CATALOG'

Check the spelling of the S/C ID and make sure the catalog is the correct one for PIONEER.

3 'DATA SET COUNT EXCEEDS 4 IN SEARCH FOR SOURCE: Satellite ID SRCE=name user entered'

Check the SRCE parameter in the INPUT namelist.

4 The user-defined average interval does not match the catalog source name's interval.

- 'NO TAPES IN CURRENT TIME RANGE'

The flux catalog does not contain the full time range of flux tapes as required by the user.

5 'ERROR FROM DREAD IN CATALOG: STOP.

An I/O error was discovered in the flux catalog. Rerun the job, and if persistent, the catalog must be replaced.

- 'INPUT TAPE READ ERROR,SKIP THIS RATE'

A tape read error on the flux tape caused a volume to be skipped.

VI. Common Block Definitions

There is only one SD common block, used to read in one flux tape rates record in RECADV:

```
Common: /PSUMRC/HTIME(6),H11,HMDAYC,MSECC,MSTIME,ISR1(9,4),ISR2(9,8),IUR(49),
        SSCOM1(4),ISCOM2(6),TSR7(9,4),TSR2(9,8),TUR(49),HSCOM(10),
        HROLL,HFLG,SANG
```

SEE "PIONEER/HELIOS Flux Data Base Generator (FLXDBG) Maintenance Programmer's Introduction", CSC, 1978, page 7 for the definition of the rates record values.

VII. Individual Module Documentation

All modules were designed, coded, and tested by Jenny S. Jacques, Code 664, 1979-1980.

A. 1. Module: DATFIL - Fills the data arrays with counts and time.

2. Calling sequence:

```
Subroutine DATFIL(K,NUM,NOFF,ISR,TSR,IREJEC,
                  QUSED,QRATS)
```

Name	Type	I/O	Description
K	I*4	I,O	Pointer to the counts array, increments with each rate stored
NUM	I*4	I	Number of rates to process
NOFF	I*4	I	NOFF + NUM is the rate number in the input arrays to process
ISR(9,NUM)	I*4	I	Input counts array
TSR(9,NUM)	R*4	I	Input time array
IREJEC(6)	I*4	O	Number of averages rejected for each rate due to differing sector times
QUSED(6)	L*1	O	T =data has been included in the arrays for the rate
QRATS(20)	L*1	I	T =this rate is being used

3. Called by: RECADV
Calls: none

4. Common usage:

Common	Variables	I,O
RATDAT	Counts,Accum	O

5. Local variables: none

6. Logic:

A main loop is set up that loops through the input data rates. If one is being processed this run, a check to ensure all sectors are collected over the same time is done. If they are, the counts and time are summed into the output arrays and QUSED (rate #) = T. Otherwise, nothing is summed for that rate.

B. 1. Module: GETAPE – Finds and mounts a flux tape onto unit 9

2. Calling sequence: Subroutine GETAPE(SRCE,ZSLOT,INTAPE,QDONE)

Name	Type	I/O	Description
SRCE	A4	I	Flux catalog source name
ZSLOT	A8	I,O	Volume - serial name of the flux tape, input if user-provided output if obtained from the catalog
INTAPE	I*4	I	1 = User-provided flux tape 2 = Fetch flux tape from the catalog T = No flux tape in the time range was found in the catalog

3. Called by: INITL
Calls: none

4. Common usage:

Common	Variables	I,O
NAMES	ZSAR,ASAT	I
TIMES	IBTIME,IETIME,INTSEC	I

5. Local variables:

Name	Type	Description
WORD(272)	R*4 I*4 I*4 I*2	These arrays are all equivalenced and are used to read in the flux catalog records. They are different modes to interpret various byte fields according to the number type in the field.
W(2) ZTEST	R*4 R*8	These variables are equivalenced in order to extract the satellite ID's first 8 characters and compare with ZSAT.
IVOL(2) ZVOLUM	I*4 R*8	These variables are equivalenced in order to extract the volume name from the catalog and call MOUNT with it.
OSRCE OZSLOT	A4 A8	Records the catalog source name, volume name, and flux tape option GETAPE from RECADV or INITL.

ZVOLD	A8	Records the previous flux tape volume name to compare with the present one. If they are the same, the tape is rewound. If not the same, the old one is unloaded and the new one mounted.
NDFILE	I*4	Record number of the catalog where desired source catalog begins.
NEXTRA	I*4	Word number in the catalog of the source names. Used to locate the correct source.
NFIRST	I*4	Word number in the catalog of the spacecraft ID's, used to locate the correct spacecraft ID.
NUMTPS	I*4	The number of used flux tapes for a particular Satellite and source.
QSTART	L*1	T = firts to,e through GETAPE for the run.

6. Logic:

If the user provides a tape, it is simply mounted and GETAPE returns (INTAPE = 1). If the flux catalog is used (INTAPE = 2), it must be searched for the correct satellite, then the correct source for tht satellite, then the correct time range to process. A tape volume name is thus fetched with the correct Satellite ID, source, and times, and it is mounted. Then GETAPE returns.

In both cases of INTAPE, there may have been a flux tape previously used in this run with another namelist set. If so, the tape is not unloaded and mounted again, but is simply rewounded to the beginning.

C. 1. Module: INITL - Initializes the I/O devices, common blocks, and reads in the PIO namelist.

(See the "Fourier Plot Program Satellite Independent System Documentation" for a basic description of .INITL. The differences of additions/deletions are described below.)

Differences or Additions/Deletions

1. There are 12 possible rates
2. A namelist PIO is read in
3. GETAPE is called to fetch and mount the flux tape

D. 1. Module: NXTRAT - Advances flux tape 7 records

2. Calling sequence: Subroutine NXTRAT(*)

* is the return if an end of file is read

3. Called by: RECADV
Calls: none

4. Common usage: none

5. Local variables: none

6. Logic:

A loop to read seven records is done. If an I/O error occurs, the record is skipped. If an end of file occurs, the routine returns to a statement number in REDADV. (The FREAD statements use a negative unit number to prevent the input buffer from being transferred twice. See the IBM FTIO booklet.)

E. 1. Module: MAGADV - Magentic field tape advance - This routine collects the magnetic field data, within the time range passed, from the fourier magnetic field data base tape.

2. Calling Sequence:

SUBROUTINE MAGADV (INTSEC,INTRVL,QNEW)

Name	Type	I,O	Description
INTSEC	I*4	I	Averaging interval in seconds of the input data tape
INTRVL(2)	I*2	I	Time range to collect the data over, in modified Julian time

3. Module Cross Reference:

Called by: RECADV
Calls: TIMJL2,JULTIM

4. Common Usage:

Common	Variables	I,O
MAFGLD	BMAG,QPSECT,QTSECT, COSIN,BSQR,MAGCNT,IZFILE	I,O
MAGIN	all	O

5. Significant Local Variables:

Name	Type	Description
MTIME	I*4	Modified Julian time (MJT) from magnetic field tape
QWAIT	L*1	T = Interval on tape is later than current time range
QEOF	L*1	T = And end of file mark was detected on the magnetic field tape
IEND	L*1	Ending of time range (MJT) to process

6. Logic:

Check to see if the last time left a record not used yet in the buffer (QWAIT=T). If so, skip around the FREAD. Otherwise, read in a record from the magnetic field tape. Loop, summing as many records as necessary to complete the time range. If an EOF occurs, continue to the next file. If an EOV occurs, set QOFF to true and end the magnetic field tape processing. (Further calls to MAGADV simply return)

F. 1. Module: RECADV - Reads in one average of sectored counts data and, if desired, magnetic field data.

(See the "Fourier Plot Program Satellite Independent System Documentation" for a basic description or additions/deletions are described below.)

Differences or Additions/Deletions

1. PSUMRC is used to contain the flux tape rates data records.
2. DATFIL is called to sum the flux tape rates data into the counts and times arrays.
3. The length of the record determines whether a solar correction angle must be added to PHIO array.
4. GETAPE is called to fetch a new tape if the current one ends with time still left to process.

G. 1. Module: VALID - Validates the input satellite independent namelist values.

(See the "Fourier Plot Program Satellite Independent System Documentation" for a basic description of VALID. The differences/additions/deletions are listed below.)

Differences on Additions/Deletions

1. There are 12 possible rate ID's.
2. The rate ID's to validate are unique to PIONEER 10 and 11.

VIII. Program Assumptions and Restrictions

1. The flux tape requires 32K core if BUFNO = 1 in the DCB is specified.
2. The flux tape must be of the standard format for flux tapes for PIONEER 10 and 11.
3. The catalog must be the flux tape catalog named in Section II. 3.

7/11/80

Multisatellite Fourier Analysis Program
User's Guide

I. Overview

The Fourier Program is divided into two groups:

1. Satellite independent code (SI) which performs the analysis.
2. Satellite dependent code (SD) which reads in alluser and satellite data and prepares it for the SI code.

The SI load modules are contained in 'SEJSS.FOURIER.SI.LOAD' and the SD load modules are located according to their name as follows:

Satellite	SI Load Module
PIONEER F,G (sectored)	'SEJSS.FOURIER.PIONEER.LOAD'
PIONEER F,G (PHA)	'SEJSS.FOURIER.PIOPHA.LOAD'
ISEE 3	'SEJSS.FOURIER.ISEE3.LOAD'
IMP8	'SEJSS.FOURIER.IMP.LOAD'
HELIOS A,B	'SEJSS.FOURIER.HELIOS.LOAD'

Both the SI and SD load modules are specified in the JCL as described later in this document. Documentation for user input, output, error handling, and JCL in the SD routines is documented separately for each satellite.

II. Input Required

A. Namelist parameters for five namelists as follows:

1. Namelist: INPUT

Name	Type	Default	Description
FROM(6)	I4	Ø	Beginning time for analysis in year (1978=78), month, day, hour, minute, second.
TO(6)	I4	Ø	Ending time for analysis in year (1978=78), month, day, hour, minute, second.
NUMAVG	I4	1	The number of input intervals (volumes) to average into one point.
RATES(6)	A8	blanks	Names for the rates to be processed.
FPARMS(9)	A8	blanks	The run parameters which specify those Fourier parameters to output on plots or tapes.

Name	Type	Default	Description
			Choices are: `AØ'=flux `A1',`A2',`A3'=anisotropy harmonics 1-3 `PHI1',`PHI2',`PHI3'=Angle PHI for harmonics 1-3 `FLOW'=flow parameter `MAG'=magnetic field data
			**The plots are determined by AØ,A1,A2, and A3, AØ=do flux plot, A1,A2,A3=anisotropy plots for the first, second, and third harmonics. All other FPARMS parameters are used for the tape option.
SATID	A16	blanks	EBCDIC satellite name.
INTSEC	I4	900	Number of seconds per input interval.
QPRINT	L1	F	T=print FOURIER results.
QPLOT	L1	F	T=there will be plots made.
QTAPES	L1	F	T=there will be tapes created.
QMAGNT	L1	F	T=process magnetic field data.

2. Namelist: PRINTR

Name	Type	Default	Description
IPRINT	I4	1	1=print only Fourier results 2=print counts and accumulation times in addition to Fourier results.

3. Namelist: PLOTS

Name	Type	Default	Description
DEVICE	I4	1	1=Create SD4060 plot tape. 2=Create CalComp plot tape.
PLTDEN	I4	4	Plot point density in points/sm. The plot is 24 cm long.
QRATPL	L1	F	Create a rate (flux) plot.
QANIPL	L1	F	Create an anisotropy and angle plot.

Name	Type	Default	Description
QPOLPL	L1	F	Create polar (cm) plots.
QBARR	L1	F	Include error bars on the flux or anisotropy plots if they are being created.
IHARMS	I4	0	0 = Do no anisotropy arrows on cam plots 1 = do first anisotropy dash arrow only 2 = do second anisotropy solid line only 3 = do both 1 and 2
FLMIN	R4	data adjusted	Flux plot minimum if automatic scaling not desired.
FLMAX	R4	data adjusted	Flux plot maximum if automatic scaling is not desired.

4. Namelist: TAPES

This must be used if and only if QTAPES=T in the INPUT namelist.

Name	Type	Default	Description
QRTAPE	L1	F	Create a tape of Fourier parameters.
IRFILE	I4	1	Start file number of Fourier tape.
ZRVOL	A8	blank	Volume-serial name of the tape to be used for Fourier output.
QSTAPE	L1	F	Create a tape of counts and accumulation time.
ISFILE	I4	1	Start file number of counts tape.
ZSVOL	A8	blank	Volume-serial names of the tape to be used for counts output.

5. Namelist: MAGNT

This must be used if and only if QMAGNT=T in the INPUT namelist.

Name	Type	Default	Description
IHISTS	I4	0	0 = do no magnetic field histograms 1 = do the phi histogram only 2 = do the theta histogram only 3 = do both phi and theta histograms
ZMVOL	A8	blank	Magnetic field data base tape name.
IZFILE	I4	I	Start file number on magnetic field data base tape.

----NOTE----

The INPUT determines which of the other four namelists are to be used. These other four namelist must appear in the order listed. The set of namelist describing the characteristics of the job run may be repeated any number of times with varying parameters. This allows several plots with different rate combinations. All parameters except SATID, RATES, and FPARMS default to the last value used in the previous namelist set.

B. Tape input is required as follows:

1. Plot Tape

This is a 7-track 556 BPI tape to be used for SD4060 of Calcomp plots if desired.

2. Rates Input Data

This is satellite dependent (SD).

3. Fourier Tape

This is a tape used for Fourier parameter output if QTAPES=T in the INPUT namelist and QRTAPE=T in the TAPES namelist, device unit 10 is used for this tape.

4. Counts Tape

This is a tape for the counts and accumulation time output, only when QTAPES=T in the INPUT namelist and QSTAPE=T in the TAPES namelists. Device until 11 is used for this tape.

5. Magnetic Field Data Base Tape

This is a tape created for the Fourier program which contains the magnetic field data as described in documentation on the Fourier MAGDBG documentation.

C. Subroutine Substitution

There are two positions one may insert any subroutine by JCL methods:

1. SUB1 - This subroutine is called before the Fourier routine so that the data may be altered before Fourier analysis.

2. SUB2 - This subroutine is called after Fourier analysis so that the data may be altered before being output in some form.

If a SYSLIB card pointing to an object module with SUB1 and/or SUB2 as members are inserted before the SYSLIB card pointing to the Fourier program object modules, then the program will accept the new SUB1, SUB2 routines. If this card is not inserted, then the dummy SUB1, SUB2 routines, which merely return control, are used.

Ex:

No Substitution:

```
//SYSLIB  DD  DSN=SEJSS.FOURIER.SI.LOAD,DISP=SHR
//        DD  DSN-SYS1.FORTLIB,DISP=SHR
//        DD  DSN=SYS2.FORTLIB.DISP=SHR
//        DD  DSN-SYS2.WOLFPLOT,DISP=SHR
```

Substitution: (USRID.PROG.LOAD has member SUB1)

```
//SYSLIB  DD  DSN=USRID.PROG.LOAD,DISP=SHR
//        DD  DSN=SEJSS.FOURIER.SI.LOAD,DISP=SHR
//        DD  DSN=SYS1.FORTLIB,DISP=SHR
//        DD  DSN=SYS2,FORTLIB,DISP=SHR
//        DD  DSN=SYS2.WOLFPLOT,DISP=SHR
```

III. Output Generated

Automatic Output

1. The input namelist data is printed.

2. A summary of how many plot tape files were created is printed.

User Option Output

Output	Namelist: option flag
1. Fourier Parameter Printout	INPUT: QPRINT=T PRINT: IPRINT=1
2. Fourier Parameter and Counts Printout	INPUT: QPRINT=T PRINT: IPRINT=2
3. Flux or Rates	INPUT: QPLOTS=T PLOTS: QRATPL=T
4. Anisotropy Plots	INPUT: QPLOTS=T PLOTS: QANIPL=T
5. Polar or Cam Plots	INPUT: QPLOTS=T PLOTS: QPOLPL=T
6. Tape of Fourier Parameters	INPUT: QTAPES=T TAPES: QRTAPE=T
7. Tape of Counts and Accumulation Time	INPUT: QTAPES=T TAPES: QSTAPE=T

IV. Running the Program

The following steps must be done to submit a run.

Step 1

Hang all required tapes in the slots and determine the namelist parameters to be used.

Step 2

Edit the TSO file which contains the JCL and namelists to run the job. Change:

1. The JOBCARD
2. Plot tapes VOL=SER names
3. Namelist parameters to suit the desired input and output.
4. All SD unit DD card specifications as required by the particular satellite.

Step 3

Submit the job using:

1. `SUB^*' if still in edit mode.
2. `SUB^name' if the file has been renamed and saved under a new file name.

---If still in edit and the file is not renamed, end the edit session with END^N command.---

V. Error Handling

Return Code

1 The namelist parameters are checked to ensure typing errors were not introduced. The program stop will return code of 1 if any parameters are not valid.

- If ther is an I/O error while creating the anisotropy plots, the message:

  'RECORD # OF HARMONIC # SKIPPED, I/O DISK ERR.'

7 If there is a timing problem the message:

'JULTIM HAS BAD TIMES'

is issued and the program stops with a return code of 7. Consult person who maintains the program.

- When an I/O error is encountered while creating polar plots, the message:

  'DISK READ ERROR IN POLAR ROUTINE'

  is used and the program continues. Some data will be missing.

- When an I/O error is encountered while creating flux plots, the message:

  'RECORD # SKIPPED DUE TO I/O ERROR FROM DISK'

  is issued, and the program continues. Some data will be missing on the flux plot.

VI. JCL Required

The JCL is contained in the file 'SEJSS.MULTISAT.FOURIER(JCL)'. This file contains only that JCL required for the SI routines. Comment cards indicate where SD JCL is to be included.

4/18/80

Multisatellite Fourier Analysis Program
PIONEER PHA User's Guide

I. Overview

The satellite-dependent (SD) PIONEER PHA routines allow the PIONEER 10 and 11 PHA data to be processed through the Fourier program's analysis and output procedures. The satellite-independent (SI) routines are contained in separate source and load modules from all SD code, and thus, any satellite may be linked via JCL. The PHA data are considered a separate satellite from the other PIONEER sectored rates data.

This document describes the SD user input required for PIONEER Fourier analysis. The main document containing the SI user input must be reviewed prior to this one.

II. Input Required

1. Satellite independent namelist INPUT'S RATES parameter and SATID are left blank.

2. Sectored PHA Data Base

   This data base resides on disk. It is created specifically for this program by someone desiring a particular set of energies to be analyzed. The data set name must be obtained from this person.

III. Error Handling

The following return codes and messages may be printed:

Return Code	Description
-	'INPUT DISK READ ERROR,SKIP THIS RATE'
-	A tape read error on the flux tape caused a volume to be skipped.

IV. JCL Required

1. Load module to link with SI routines:
   'SEJSS.FOURIER.PIOPHA.LOAD'

2. Unit 49 defined as the input Sectored PHA disk file name.

4/24/80

The Fourier Program
Pioneer Differential Rates

Overview

The Fourier Plot Program has an option to change the data before it goes into Fourier analysis. This is done via a subroutine called SUB1 (See documentation on the Fouier Program, SUB1) in this case subtracting one set of sectored counts from the next and storing this differential as a new set of counts. These counts then undergo the same analysis as before. SUB1 also changes the labels appropriately.

To involk this process, SUB1 for differential rates is compiled into a load module and linked as the first SYSLIB data set. This causes the new SUB1 to override the dummy (simply returns) SUB1 in the Fourier load module, and thus be used to create differential sectored rates.

JCL Required

1. SYSLIB DD DSN='SEJSS.FOURIER.DIFRNTL.LOAD',DISP=SHR as the first SYSLIB data set.

2. Same JCL as for PIONEER sectored rates.

~00100 THE MULTISATELLITE FOURIER ANALYSIS PROGRAM CONTAINS A CALL TO
00200 A SUBROUTINE CALLED SUB2 AFTER THE DATA HAS BEEN FOURIER ANALYZED.
00400 THE DATA MAY THUS BE MANIPULATED AFTER ANALYSIS BY LINKING IN TO A
00500 USER WRITTEN SUB2 LOAD MODULE IN THE SYSLIB JCL LIST OF DATA SETS.
00600 A DEFAULT SUB2 IS LOCATED IN THE SI LOAD LIBRARY (SATELLITE INDEPEN-
00700 DENT LIBRARY) WHICH SIMPLY RETURNS. IF THE USER WANTS TO ALTER THE
00800 DATA AFTER ANALYSIS, HE MUST WRITE A SUBROUTINE CALLED SUB2 AND
00900 CREATE A LOAD MODULE FROM IT VIA ADDTOLIB. THIS LOAD DATA SET IS
01000 THEN LISTED AS THE FIRST ONE IN THE SYSLIB CARD WHEN SUBMITTING
01100 THE PROGRAM. THE FOLLOWING IS A LIST OF THE SUB2 MODULES CREATED:
01200
01300
01400 1. SOURCE FILE:'SEJSS.MULTISAT.SUB2(SIG2CHK)'
01500 LOAD FILE :'SEJSS.FOURIER.SIG2CHK.LOAD'
01600 DESCRIPTION: THIS ROUTINE NEGATES ANY ANISOTROPY VALUE WHICH
01610 IS LESS THAN TWICE ITS DEVIATION SUCH THAT IT IS
01620 NOT PLOTTED.
01700

```
00100 ********************************************************************************
00200 *
00300 *      THE FOLLOWING DOCUMENTATION BRIEFLY DESCRIBES THE JCL LINKAGE
00400 *      FOR ALL SATELLITES USING THE FOURIER PACKAGE.
00500 *
00600 *      THE FOURIER PROGRAM IS DIVIDED INTO TWO PARTS:
00700 *      1. THE SATELLITE INDEPENDENT PART (SIP)
00800 *      2. THE SATELLITE DEPENDENT PART (SDP)
00900 *
01000 *      THE SIP CONSISTS OF THE LOAD MODULE 'SEJSS.FOURIER.SI.LOAD'
01100 *      AND THE NAMELISTS &INPUT,&PRINT,&PLOTS,&TAPES, AND &MAGNT.
01200 *
01300 *      THE SDP CONSISTS OF THE LOAD MODULE AND NAMELISTS AS DESCRIBED
01400 *      FOR EACH SATELLITE.  THE PIONEER AND PIUNEER PHA ANALYSIS ARE
01500 *      ARE TREATED AS TWO SEPARATE SATELLITES, AND THEIR LOAD MODULES
01600 *      AND NAMELISTS ARE DEFINED AS FOLLOWS:
01700 *
01800 *
01900 *      SATELLITE          LOAD MODULE                NAMELIST(S)
02000 *      ---------          -----------                -----------
02100 *
02200 *  PIONEER F,G      'SEJSS.FOURIER.PIONEER.LOAD'     &PIU SRCE,INTAPE,ZVOL
02300 *  PIO-PHA          'SEJSS.FOURIER.PIOPHA.LOAD'      NO EXTRA NAMELISTS

02500 *
02600 *
02700 *      THE SYSLIB CARD SHOULD THUS READ :
02800 *      DSN=SEJSS.FOURIER.PIONEER.LOAD    FOR PIUNEER ANALYSIS
02900 *      DSN=SEJSS.FOURIER.PIUPHA.LOAD     FOR PIONEER PHA ANALYSIS
03000 *
03100 *      THE SYSLIN CARD SHOULD THUS READ:
03200 *      DSN=SEJSS.FOURIER.PIONEER.LOAD(RATDAT)  FOR PIONEER ANALYSIS
03300 *      DSN=SEJSS.FOURIER.PIOPHA.LOAD(RATDAT)   FOR PIONEER PHA ANALYSIS
03400 *
03500 *      THE FT30 CARD IS NEEDED FOR PIONEER ANALYSIS, NUT FOR PIUNEER PHA
03600 *      THE FT49 CARD IS NEEDED FOR PIONEER PHA ANLAYSIS, NOT FOR PIONEER
03700 *
03800 *      THE NAMELISTS MAY BE REPEATED ANY NUMBER OF TIMES IN ONE RUN. THIS
03900 *      IS USEFUL IF DIFFERENT RATES,OUTPUT OPTIONS, ETC ARE DESIRED.
04000 *      EACH NAMELIST SET MUST CONSIST OF THE &INPUT NAMELIST FOLLOWED
04100 *      BY A NAMELIST FOR EACH OPTION REQUESTED BY THE &INPUT NAMELIST
04200 *      OPTION PARAMETERS:
04300 *      QPRINT=T ----- &PRINT NAMELIST MUST BE SPECIFIED
04400 *      QPLOTS=T ----- &PLOTS NAMELIST MUST BE SPECIFIED
04500 *      QTAPES=T ----- &TAPES NAMELIST MUST BE SPECIFIED
04502 *      QMAGNT   ----- &MAGNT NAMELIST MUST BE SPECIFIED

04506 *      AFTER EACH SIP SET OF NAMELISTS ARE SPECIFIED,ANY SDP NAMELISTS
04508 *      MUST BE SPECIFIED, AFTER WHICH ANOTHER SET OF SIP-SDP NAMELISTS
04510 *      MAY BE SPECIFIED.   BUT THE ORDER OF SIP NAMELISTS MUST BE:
04512 *      &INPUT, &PRINT,&PLOTS,&TAPES,&MAGNT,&(SDP NAMELISTS).  ONE OR
04514 *      MORE OF THE SIP NAMELISTS MAY BE OMITTED IF THE FLAG FOR IT
04516 *      WAS .FALSE. IN THE &INPUT NAMELIST.
04600 *
04700 *      THERE SHOULD ALWAYS BE BE THE NAMELIST &PIO AFTER THE LAST OF
04800 *      THE SIP NAMELISTS (FOR EACH SET!) TO SPECIFY THE SRCE,ETC., BUT
04900 *      NO NAMELIST IS USED FOR PIONEER PHA DATA ANALYSIS.
05000 *
05100 ********************************************************************************
05200 *
05300 *      DEFINITION OF NAMELIST PARAMETERS
05400 *      ---------------------------------
```

```
05600 *
05700 * &INPUT:
05800 *      NAME        TYPE         DESCRIPTION
05900 *  ____________  ____________  ____________________________________________
06000 *
06100 *  FROM(6)        INTEGER    YR,MON,DAY,HR,MN,SC OF START OF RUN
06200 *  TO(6)          INTEGER    YR,MON,DAY,HR,MN,SC OF END OF RUN

06400 *  INTSEC         INTEGER    AVERAGE INTERVAL OF INPUT VOLUMES, SECONDS
06500 *  RATES(6)   8 CHAR. EA      RATE ID'S TO PROCESS FOR THE PARTICULAR SAT.
06600 *  FPARMS(9)  8 CHAR. EA      FOURIER PARAMETERS TO PROCESS WHEN MAKING
06700 *                             ANISOTROPY VS ANGLE PLOTS OR OUTPUT TO A TAPE
06800 *  SATID     16 CHAR.         SATELLITE ID IN EBCDIC
06900 *  QPRINT     LOGICAL         T=GET FOURIER LISTING (DEFAULT=F)
07000 *  QPLOTS     LOGICAL         T=CREATE PLOTS    (DEFAULT=F)
07100 *  QTAPES     LOGICAL         T=CREATE FOURIER OUTPUT TAPES
07200 *  QMAGNT     LOGICAL         T=INCLUDE MAGNETIC FIELD PROCESSING
07300 *
07400 *&PRINT:
07500 *      NAME        TYPE         DESCRIPTION
07600 *  ____________  ____________  ____________________________________________
07700 *
07800 *     IPRINT      INTEGER       1=LIST FOURIER PARMS ONLY, 2=LIST COUNTS TOO
07900 *
08000 *&PLOTS:
08100 *      NAME        TYPE         DESCRIPTION
08200 *  ____________  ____________  ____________________________________________
08300 *
08400 *     DEVICE      INTEGER        PLOT DEVICE: 1=SD4060, 2=CALCOMP, 3=PRINTER
08500 *                                             (DEFAULT=1)

              DLTDEN   INTEGER   PLOT PLTS/CM, FRAME=24 CM long, Max=4
08700 *                                             (DEFAULT=4)
08800 *     QRATPL      LOGICAL        T=CREATE FLUX PLOTS  (DEFAULT=F)
08900 *     QANIPL      LOGICAL        T=CREATE ANISOTROPY VS ANGLE PLOTS (DEF=F)
09000 *     APOLPL      LOGICAL        T=CREATE POLAR, OR CAM PLOTS   (DEF=F)
09100 *     QHARM2      LOGICAL        T=INCLUDE SECOND HARM ON CAM PLOTS (DEF=F)
09200 *     FLMIN       REAL           FLUX PLOT MINIMUM   (DEF=MIN OF DATA)
09300 *     FLMAX       REAL           FLUX PLOT MAXIMUM    (DEF=MAX OF DATA)
09400 *
09500 *&TAPES:
09600 *      NAME        TYPE         DESCRIPTION
09700 *  ____________  ____________  ____________________________________________
09800 *
09900 *     QRTAPE      LOGICAL          T=CREATE TAPE OF FOURIER PARAMETER (DEF=F)
10000 *     QSTAPE      LOGICAL          T=CREATE TAPE OF COUNTS   (DEF=F)
10100 *     IRFILE      INTEGER          FILE NUMBER OF FOURIER PARM. TAPE
10200 *     ISFILE      INTEGER          FILE NUMBER OF COUNTS TAPE
10300 *     ZRVOL       6 CHAR.          VOLUME NAME OF FOURIER PARM. TAPE
10400 *     ZSVOL       6 CHAR.          VOLUME NAME OF COUNTS TAPE
10500 *
10600 *&MAGNT:
10700 *      NAME        TYPE         DESCRIPTION
10800 *  ____________  ____________  ____________________________________________

10900 *
11000 *     QHIST       LOGICAL          T=PLOT PHI,THETA CAM HISTOGRAMS ON CAM
11100 *                                     PLOT   (DEF=F)
11200 *     ZMVOL       6 CHAR           VOLUME NAME OF MAGNETIC FIELD INPUT TAPE
11300 *     IZFILE      INTEGER          FILE NUMBER OF MAGNETIC FIELD INPUT TAPE
11400 *
11500 ************************************************************************
11600 *
11700 *      INPUT NAMELIST FOR PIONEER F,G ANALYSIS
11800 *
```

```
12000 *      NAME          TYPE          DESCRIPTION
12100 *  ____________  ____________  ____________________________________________
12200 *
12300 *      SRCE          4 CHAR          SOURCE NAME OF FLUX CATALOG SOURCE TO USE
12400 *      INTAPE        INTEGER         1=USER PROVIDED FLUX TAPE,2=SEARCH FLUX
12500 *                                      CATALOG FOR TAPE OF TIME RANGE (DEF=2)
12600 *      ZVOL          6 CHAR          VOLUME NAME OF USER PROVIDED FLUX TAPE
12700 *
12800 ************************************************************************
END OF DATA
```

```
08400 *      DEVICE        INTEGER         PLOT DEVICE: 1=SD4060,2=CALCOMP,3=PRINTER
```

IOCUMENT FOR BIT2ON

PREPARED BY Laura F. Casswell
COMPUTER SCIENCES CORPORATION, SYSTEMS SCIENCES

May 1981

JCL
SB#PR.LIB.CNTL
(BIT2ON)

INDEX

PIONEER-F BIT2ON PROGRAM

A. PROBLEM:

There is a problem with a number of bits in a Pioneer-F detecter; these bits (usually 2 and 6) intermittently fail to register. Thus, incorrect values are obtained for the rates. Since these values often differ greatly from the general trend of the data, they can cause significant errors in the fluxes.

B. SOLUTION

To correct this problem, the program BIT2ON was written. This program looks for data deviating significantly from the general trend and then tests the effect of turning on these faulty bits. If the resulting value is closer to the trend, the rate is corrected to this value. The program QUICKBIT is a version of BIT2ON which processes only the event type rates (for the specific rates, see the program description in the Programmer's Guide).

A. PROGRAM DESCRIPTION

1. BIT2ON

When the Pioneer EDR tapes are processed by PIODRP, the PIoneer rates tapes are generated, containing the values of the sectored and unsectored rates (converted to decimal form). For detailed information on the rates tape format, see Appendix B. Because of the bit failures in the satellite (usually 2 and 6), some of the rates on the EDR tapes are incorrect. Thus, there are incorrect rates written to the rates tapes by PIODRP in some cases. In order to deal with this, the BIT2ON program was developed to reprocess the rates tapes, and generate the "BIT2ON corrected" rates tapes.

The initial rates tapes are read by BIT2ON (in PFRBIT). At this point, there are two options to the approach to the reprocessing: TIMESKIP, and TIMECOPY. In the TIMESKIP option, the user can skip forward to a requested time on the input tape, and begin reprocessing from this point directly to the output tape. With the TIMECOPY option, the user may first copy part of a previously reprocessed tape to the output tape, then skip to the desired time on the input tape and start processing to the output tape following the copied data.

For the reprocessing, the program must first identify the format (A or B), and then whether the rate is sectored or unsectored. This is done in BIT2ON. The subroutine BIT2ON will then call BITUS for unsectored rates, or BITSS for sectored rates. These routines will first compare the incoming rate to the known exceptions and zero values. If a match occurs the rate is reset and the routine returns to BIT2ON for the next rate. If the rate does not match an exception or a zero rate, the routine will then move on to the bit turnon.

Before the bit turnon, the rate must be converted from decimal back to the spacecraft log form. This is accomplished with DECLOG. For a complete description of the spacecraft log - decimal conversion, see the HELDRP manual. Once the rate is in log form, the routine uses GETPUT to turn on bit 2. The rate is then converted back to decimal form by LOGDEC. At this point it must be compared to the table of permitted rates (set up in RATTAB). If it is an allowed rate, the new rate is then compared to the trend. Then, if the new rate is closer to the trend than the old rate, it replaces the previous value of the rate. This same procedure is followed for the bit 6 turnon. After the bit turnons, the trend is reset to the current rate value.

2. QUICKBIT

The QUICKBIT version of BIT2ON follows the same procedures as described above, with two exceptions: the bit 6 turn-on is not done, and only events type rates are processed. This means that the following rates are processed: SR1(ABCD), R1, R2(AB), R3A, R4B, R5B, R9(ABCD), R10(ABCDEFGH), R11(AB), R12(AB), R14(ABCD), and R15(AB).

This makes QUICKBIT about twice as fast as BIT2ON.

B. CALLING ORDER OF ROUTINES

```
PFRBIT
   RATTAE
   BIT2ON
      BITSS
         DECLOG
         GETPUT  (BTMNP)
         LOGDEC
      BITUS
         DECLOG
         GETPUT  (BTMNE)
         LOGDEC
```

C. DESCRIPTION OF ROUTINES

PFRBIT : found in SBPIO.RATELIST.SOURCE

This is the BIT2ON control routine. It mounts the tapes and skips to the requested day on the input tape, or copies up to a requested day form a previous BIT2ON tape and then skips to the same day on the input tape.

BIT2ON : found in SBPIO.RATELIST.SOURCE

This subroutine handles both formats A and B. It seperates the rates into sectored and unsectored and and calls the appropriate subroutines (BITSS or BITUS).

BITSS : found in SBPIO.RATELIST.SOURCE

This routine checks for sectored rates exceptions,and zeros, trend checks the data, and calls the bit mihipulating routines. It will replace the sectored rate with the calculated (bit turned-on) rate if this new rate is closer to the trend, and reset the trend to the most recent rate.

BITUS : found in SBPIC.RATELIST.SOURCE

This is similar tc BITSS, except that it is used to process the unsectored rates.

DECLOG : found in SBPIO.PFBDISP.SOURCE

This routine converts the rates decimal values back into the satellite log value.

GETPUT (BTMNP) : found in SDHEL.HELDRP1.SOURCE

This routine turns on the requested bit.

LOGDEC : found in SEPIO.PFRDISP.SOURCE

This routine converts the log back to decimal form.

USER'S GUIDE

A. DESCRIPTION.

The BIT2ON and QUICKBIT programs are designed to do a bit correction and trend check of Pioneer-F rates data. It can process all or part of an input tape; this processing may take place after copying a portion of a previously processed tape.

Only one input tape may be processed per run. The program has two options: TIMESKIP and TIMECOPY. The TIMESKIP option processes an input tape directly to the beginning of an output tape. The input tape may be skipped forward to a requested start time, and will continue to a requested end time. The TIMECOPY option allows a portion of a previously processed tape to be copied to the output tape before new processing begins. In both options an input start and end time determine the records to be processed. If zeros are entered for the start time, processing will begin at the first record of the tape. If zeros are entered for the end time, processing will run to the end of the tape.

If either the input or the output tape is not specified, the request will be ignored. If neither TIMESKIP nor TIMECOPY is entered in the input card, the request will be ignored. An error message is also written if the requested time interval is not found on the specified tapes.

The BIT2ON program processes all of the sectored and unsectored rates, turning on bit 2 and bit 6 in a 12 bit word. Certain rates values which are known to be exceptions (734, 778, 2296, 10208, 12256, 8421376, 16482304, 14254080, 14385152) are changed to pad (-20000000), and values which would be set to zero by turning on bits 2, 6, and 10 (14516224, 14647296) are automatically set to zero. These exceptions and zero values are read into BIT2ON as data, in a namelist (see sample JCL... &EXCEPS...). The sectored and unsectored exceptions are read in seperately (IEXCS and IEXCU). All rates coming in are re-trend checked and failed ony if they are greater than 16 times the last value.

The QUICKBIT program is a version of BIT2ON which has been modified to make it two times as fast. QUICKBIT processes only the event type rates: SR1(ABCD), R1, R2(AB), R3A, R4B, R5B, R9(ABCD), R10(ABCDEFGH), R11(AB), R12(AB), R14(ABCD), and R15(AB). The bit 6 turn-on has also been removed.

B. JCL FOR RATES

The program requires 160K of main storage.

```
//SBPIOBT2 JOB (SB0012356F,T,SA0001,001001),BF3,MSGLEVEL=1
```

```
//* BIT2ON THISDATE
//GO EXEC PGM=BIT2ON,REGION=200K
//STEPLIB   DD   DSN=SBPIO.LIB.LOAD,DISP=SHR
//GO.FT05F001 DD DDNAME=DATA5
//GO.FT06F001 DD SYSOUT=A,DCB=(RECFM=VBA,LRECL=137,BLKSIZE=7265)
//GO.FT20F001 DD DUMMY
//GO.FT30F001 DD DUMMY
//GO.FT08F001 DD SYSOUT=A,DCB=(RECFM=VBA,LRECL=137,BLKSIZE=7265)
//GO.FT09F001 DD DSN=BIORAT,UNIT=(6250,,DEFER),DISP=SHR,
//  VOL=SER=DUMRAT,DCB=DEN=3
//GO.FT10F001 DD DSN=BIORAT,UNIT=(6250,,DEFER),DISP=SHR,
//  DCB=(RECFM=VBS,LRECL=1740,BLKSIZE=8704,BUFNO=1,DEN=3),
//  VOL=SER=DUMOUT,LABEL=(,SL,,OUT)
//GO.SYSUDUMP DD SYSOUT=A
//*DATACARD  DTYPE(1-8),ID(9-10),DTAPE(13-20),DTPOUT(21-28),
//*          DTPCPY(29-36),HTIME: START:YR(37-38),MN(39-40),DAY(41-42)
//*                               END:  YR(49-50),MN(51-52),DAY(53-54)
//*
//* DTYPE=TIMESKIP     DTAPE IS MOVED FORWARD TO THE REQUESTED
//*                    START DAY AND PROCESSING IS STARTED AT
//*                    THE BEGINNING OF DTPCPY.
//* DTYPE=TIMECPY      DTPOUT IS COPIED TO DTPCPY UP TO THE
//*                    REQUESTED START TIME, THEN DTAPE IS MOVED
//*                    TO THE REQUESTED TIME AND PROCESSING IS STARTED
//*                    ONTO DTPCPY, UP TO THE REQUESTED END TIME.
//*DATACARD
//*DTAPE ID DTAPE    DTPOUT   DTPCPY   YYMMDDOYYMMDDO
//GO.DATA5  DD *
 &EXCEPS NEXCS=0,IEXCS=100*999999,NEXCU=0,IEXCU=100*999999,NZERO=0,
  IZERO=100*999999,&END
 &EXCEPS NEXCS=09,IEXCS(1)=734,IEXCS(2)=8421376,IEXCS(3)=16482304,
  IEXCS(4)=14254080,IEXCS(5)=14385152,IEXCS(6)=778,IEXCS(7)=2296,
  IEXCS(8)=10208,IEXCS(9)=12256,NEXCU=9,IEXCU(1)=734,
  IEXCU(2)=8421376,IEXCU(3)=16482304,IEXCU(4)=10208,
  IEXCU(5)=12256,IEXCU(6)=14254080,IEXCU(7)=14385152,
  IEXCU(8)=778,IEXCU(9)=2296,
  NZERO=2,IZERO(1)=14516224,IZERO(2)=14647296 &END
 &TRENDS LASTUS=128*0,LASTSS=128*0, &END
 &TRENDS LASTUS(9,1)=0174,LASTUS(4,2)=0186,LASTUS(5,2)=369,
 LASTUS(9,2)=0281,LASTUS(9,3)=0281,LASTUS(14,3)=514,
 LASTUS(14,4)=0192,LASTUS(14,6)=642,LASTUS(14,7)=102,
 LASTUS(14,8)=634,LASTUS(9,4)=0399,&END
TIMESKIP F   E00344   DSD04    DSD04    80090408009070
// EXEC NOTIFYTS
```

QUICKBIT : 0.35 CPU, 0.30 IO for approximately 1500 records (on the IBM 360/91)

BIT2ON : 0.70 CPU, 0.30 IO for approximately 1500 records (on the IBM 360/91)

JCL to run these may be found in SBPIO.LIB.CNTL under the member names : BIT2ON, and BIT2QK.

C. DATA CARDS

The trends, exceptions and zeros are read in through namelist inputs.

trends:

```
&TRENDS LASTSS(I,J)=N,LASTUS(I,J)=N ,&END
```

LASTSS = the table of sectored rates trends
LASTUS = the table of unsectored rates trends

exceptions and zeros:

```
&EXCEPS NEXCS=N1,IEXCS(1)=?,....IEXCS(N1)=?,
 NEXCU=N2,IEXCU(1)=?,.......IEXCU(N2)=?,
 NZERO=N3,IZERO(1)=?,.......IZERO(N3)=?,&END
```

NEXCS = the number of sectored exceptions
IEXCS = the value of the sectored exception
NEXCU = the number of unsectored exceptions
IEXCU = the value of the unsectored exception
NZERO = the number of zero exceptions
IZERO = the value of the zero exception

For an example of typical values see the JCL FOR RATES section.

Input data for tapes and processing:

card columns	description
1-8	option, must contain word TIMESKIP or TIMECOPY
9-10	HID: should be blank F
11-12	blank
13-20	input tape; data to be processed
21-28	previously processed tape to be copied
29-36	tape to which data will be copied and processed
37-38 ; 49-50	2 digit start year ; 2 digit end year
39-40 ; 51-52	start month ; end month
41-42 ; 53-54	start days ; end days
43-44 ; 55-56	start hour ; end hour
45-46 ; 57-58	start minutes ; end minutes
47-48 ; 59-60	start seconds ; end seconds

D. OUTPUT

Program output for a successful run will list: requested start and end dates, start and end date in modified julian day, the number of records processed, and the tapes involved. The program should produce one output tape of processed data.

E. ABENDS AND ERROR MESSAGES.

The following is a list of program error and information messages with appropriate user response.

1. ***CHECK TAPES FOR PROBLEM: THE NUMBER OF RECORDS COPIED (XXXXX) IS NOT THE SAME AS THE NUMBER OF RECORDS SKIPPED ON THE INPUT TAPE (XXXXX).

cause : in using the TIMECOPY option, the number of records copied from the previously processed tape is not the same as the number of records skipped on the input tape.

user response : the user should check tapes to insure that the proper tapes are being used, since in the TIMECOPY option, the files of the input and the final output tape should correspond.

2. ** UNEXPECTED END OF FILE REACHED, END OF PROCESSING FOOR THIS REQUEST ***

cause : the end of the tape was reached before the requested start time was found.

user response : check input tape and start time entered.

3. *** END OF OUTPUT RECORDS REACHED BEFORE THE REQUESTED TIME INTERVAL WAS FOUND. END PROCESSING FOR THIS REQUEST ***

cause : in the TIMECOPY option, the program copied the entire previously processed tape without reaching the requested start time.

user response : check tape requested to be copied, and the requested start time.

4. *** ERROR DTYPE (XXXXXXXX) NOT THE SAME AS EITHER DTIMES (TIMECOPY) OR DTIMIC (TIMECOPY) SO REQUEST WILL BE IGNORED ***

cause : the option entered did not correspond to either TIMESKIP or TIMECOPY.

user response: check the first eight columns of data card, to make sure the requested option is valid.

5. *** DTAPE OR DTPCPY IS NOT SPECIFIED ***

cause : a tape was not specified for either DTAPE (the input tape) or DTPCPY (the output tape).

user response: check data card (columns 13-20, and 29-36) to make sure that each tape is specified properly, starting

in columns 13 and 29.

APPENDIX A: PROLOGUES

```
CH1   ROUTINE PFRBIT
CH
CH2   MAKES BIT2ON CORRECTIONS AFTER SKIPPING TO THE REQUESTED DAY
CH2   ON THE INPUT TAPE, OR AFTER COPYING UP TO THE REQUESTED DAY
CH2   FROM A PREVIOUS BIT2ON TAPE AND SKIPPING TO THE SAME DAY ON
CH2   THE INPUT TAPE.
CH
CH4   CALLS: BIT2ON, DBMJD
CH
CH5   VARIABLES:
CH5   DTYPE          R*8          TIMESKIP: TO SKIP ON THE INPUT TAPE
CH5                                         UP TO THE REQUESTED DAY
CH5                               TIMECOPY: TO COPY FROM DTPOUT TO DTPCPY
CH5                                         AND THEN SKIP FORWARD ON DTAPE
CH5   HID            I*2          PIONEER ID, F OR G
CH5   DTAPE          R*8          INPUT TAPE
CH5   DTPOUT         R*8          OLD OUTPUT TAPE TO BE COPIED
CH5   DTPCPY         R*8          NEW OUTPUT TAPE FOR BIT2ON
CH5   HTIME(12)      I*2          HTIME(1-3):START TIME; YEAR, MONTH, DAY
CH5                               HTIME(7-9):END TIME; YEAR, MONTH, DAY
CH5                               IF HTIME(1) IS 0, THE PROGRAM WILL START
CH5                                  AT THE BEGINNING OF THE TAPE.
CH5                               IF HTIME(7) IS 0, THE PROGRAM WILL PROCESS
CH5                                  TO THE END OF THE TAPE.
CH
CH7   L CASSWELL FEBRUARY 1980  (MODIFIED FROM PFRBIT OF ED RONISH 1978)
CH
CH9 PFRBIT ************************************************************
CH
```

```
CH1   SUBROUTINE BIT2ON
CH
CH2   THIS SUBROUTINE CYCLES THROUGH THE SECTORED AND UNSECTORED
CH2   RATES, CALLING BITSS AND BITUS TO TEST FOR EXCEPTIONS AND DO
CH2   THE BIT 2 AND 6 TURNON.  THERE ARE SECTIONS FOR BOTH A AND B
CH2   FORMAT.
CH
CH3   CALLED BY : PFFFIT
CH
CH4   CALLS: BITUS, BITSS
CH
CH7   ED RONISH    1978
CH
CH9 BIT2ON ************************************************************
CH
```

```
CH1   SUBROUTINE BITUS       *** FOR UNSECTORED RATES ***
CH
CH2   THIS SUBROUTINE TESTS FOR EXCEPTIONS, AND THEN ATTEMPTS A BIT 2
CH2   OR A BIT 6 TURNON (IF NOT RATES EXCEPTION).  EXCEPTIONS ARE PAD-
CH2   DED, WHILE OTHER RATES ARE COMPARED TO THE TREND (AFTER BIT TURN-
CH2   ON) TO DETERMINE WHETHER OR NOT BIT TURNONS BRING THE RATES CLOSER
CH2   TO THE TREND.  THE RATES MUST BE CONVERTED BACK TO LOGS BEFORE THE
CH2   BIT TURNON, CONVERTED BACK TO DECIMAL FORM, AND THEN TESTED TO ELIM
CH2   INATE ANY FORBIDDEN LOGS.  THE TREND IS RESET TO THE MOST RECENT
CH2   RATE AT THE END OF THE ROUTINE.
CH
CH3   CALLED BY: BIT2ON
CH
CH4   CALLS: LOGDEC, DECLOG, GETPUT(BTMNP,IGET)
CH
CH5   INPUT VARIABLES:
CH5    IUS - RATE ID SUBSCRIPT
CH5    IRATE - RATE
CH5    L -
CH5    HBTRT - BITRATE
CH5    HFMT - FORMAT
CH
CH7    ED RONISH     1978
CH
CH9BITUS ************************************************************
CH
```

```
CH1   SUBROUTINE BITSS       *** FOR SECTORED RATES ***
CH
CH    THIS SUBROUTINE TESTS FOR EXCEPTIONS, AND THEN ATTEMPTS A BIT
CH2   2 OR A BIT 6 TURNON (IF NOT RATES EXCEPTION).  EXCEPTIONS ARE
CH2   PADDED, WHILE OTHER RATES ARE COMPARED TO THE TREND (AFTER BIT
CH2   TURN-ON) TO DETERMINE WHETHER OR NOT BIT TURNONS BRING THE RATES
CH2   CLOSER TO THE TREND.  THE RATES MUST BE CONVERTED BACK TO LOGS
CH2   BEFORE THE BIT TURNON, CONVERTED BACK TO DECIMAL FORM, AND THEN
CH2   TESTED TO ELIMINATE ANY FORBIDDEN LOGS.  THE TREND IS RESET TO
CH2   THE MOST RECENT RATE AT THE END OF THE ROUTINE.
CH
CH3   CALLED BY: BIT2ON
CH
CH4   CALLS: LOGDEC, DECLOG, GETPUT(BTMNP,IGET)
CH
CH5   INPUT VARIABLES:
CH5    M -
CH5    IUS - RATE ID SUBSCRIPT
CH5    IRATE - RATE
CH5    L -
CH5    HBTRT - BITRATE
CH5    HFMT - FORMAT
CH
CH7   E BONISH   1978
CH
CH9 BITSS  ************************************************************
```

APPENDIX B: RATES TAPE FORMAT (TAKEN FROM PIODRP MANUAL)

A DESCRIPTION

The rates tapes are 7-track, 800 bpi tapes with standard OS/360 labels written in the binary mode and odd parity with conversion. They contain variable length, blocked records with a maximum buffer length (BLKSIZE) of 8704 bytes and a maximum logical record length (LRECL) of 1740 bytes. These tapes contain the time-ordered Pioneer GSFC/CRT events per seconds (rates) data and related spacecraft information. Each logical record contains selected spacecraft information and all the rates data for one or more pages (each page represents one fourth of an experiment cycle). All rates which fail the trend check will be indicated by a negative rate value. Whenever a rate with the value of zero fails the trend check, it will be indicated by a negative one (-1). Padded rates data will be indicated by the value -20000000.

B LOGICAL RECORD FORMAT

Mnemonic	Description
MSPAG1	Time of day (milliseconds) for first page contained in record
MSNEXT	Time of day (milliseconds) for page which is expected to immediately follow last page in record
RMJDP1	Day (relative modified Julian day) for first page contained in record
RMJDEX	Day (relative modified Julian day) for page which is expected to immediately follow last page in record
ABFILE	absolute file number
TCFLAG	Time correction flag = 0, no correction = 7, suspect time or corrected time
NPAGES	Number of pages (one-quarter experiment cycle) included in record (maximum of six for format A and five for format B)
BITRAT	Bit rate (1-16, 2-32, 3-64, 4-128, 5-256, 6-512, 7-1024, 8-2048)
FORMAT	Format (1-A, 2-A/D, 3-B, 4-B/D)
MODE	Mode =0 or 1, real time =2 or 3, memory readout =4 or 5, telemetry store
DSSID	DSS identification

ESCID	Extended frame counter (ESC subcom ID)
RATFLG	RAT flag (roll attitude timer) =0, good value =1, old value =2, missing value =3, corrected value
SPNFLG	ASPNPDC flag (spin period)
SPFFLG	SPF flag (spin flag period)
RIPFIG	RRIPPHEC flag -- pulse/roll index pulse phase error
ROLLAT	Roll attitude timer (RAT)
SPNPDC	Spin period (ASPNPDC)
RIPPEC	Roll pulse/roll index pulse phase error (ARIPPHEC)
SPSGRR	Spin period sector generator (SPSG) roll reference =0, 0 degrees =1, 180 degrees
SPSGMD	Spin period sector generator (SPSG) mode =0, non-spin averaging =1, ACS =2, spin averaging
MSRAT	Roll attitude time (milliseconds of RAT)
DCVOIT	DC bus voltage
DCCURR	DC bus current
SPTEMP	Spacecraft platform temperature
SNR	Signal-to-noise ratio
SPARE1	Spare (currently set to zero)
SPARE2	Spare (currently set to zero)
N1	All subcom data associated with first page of data contained in record. Refer to Tables A-1 and A-2 in the PICDRP manual for a description of formats A and E, respectively. All rates data associated with first page of data contained in record. Each page consists of four sets (two sectored and two unsectored) of 16 rates which are uniquely identified by corresponding rate sequence IDs appearing in associated set of subcom data. Rates data associated with each page appears in 64 consecutive words as follows: 1 - Sectered rate (first set) SR1(1-8) SR2(1-8) 16 - Sectered rate (first set) 17 - Unsectored rate (first set) R1-R8 R9-R16

	32 - Unsectored rate (first set) 33 - Sectered rate (second set) SR1(1-8) SR2(1-8) 48 - Sectered rate (second set) 49 - Unsectored rate (second set) R1-R8 R9-R16 64 - Unsectored rate (second set) Refer to table A-3 in the PIODRP manual to determine rates data associated with each unsectored and sectcred rate sequence ID Note that redundant sectcred rates data occurs whenever corresponding sectored rate sequence ID is not updated from previous value =100, format A =112, format B
N2	All subcom and rates data for second page of data contained in record (see description of first page) =356, format A =368, format B
N3	Third page of data =632, format A =656, format B
N4	Fourth page of data =908, format A =944, format B
N5	Fifth page of data =1184, format A =1232, format B
1460	Sixth page of data (format A only)

Structure of used tape record:
Each Record is 272 words long (FB).
WORDS(1 - 270) characteristics of use d tapes.

Each tape has 6 words long information
so one Record contains maximum of 45 tapes (6, 45).

Description of six words. as follows.

tape
(1, 1).
↓
(1, 6)

WORDS(1) = START AFN
WORDS(2) = END AFN
WORDS(3&4) = TAPE # (A8)
WORDS(5) = Number of blks allocated for that tape
WORDS(6) = " " WRITTEN " ...

(1, 2)
↓
(6, 2)

WORDS(7) = START AFN
WORDS(8) = END AFN
WORDS(9&10) = TAPE # (A8)
WORDS(11) = Number of blks
WORDS(12) = Number of blks written.

upto 45 tapes (6, 45)

WORDS(271) = Pointer to next used tape record
WORDS(272) = number of Flex tapes in this record.

User GUIDE for DIRFIX

FIND the record no. that contains ~~for that~~ flux tapes.
Alloc 'SDHEL.FLUXCAT2.DATA' F(FT10F001)
Call ~~DIRFIX~~ 'SBCID.DIRFIX.LOAD(DIRFIX)'
~~does not~~

DIRFIX works on 2 namelists:
namelist #1 &LOC RECORD = , WORDS = 1, 10, FUNC = 0 or 1 &END

Record = Record number you want work on

WORDS = Dimension in (2) start & end words.
1 = modify the record. You don't need WORDS
FUNC = 0 = list the words from start to end.
always give start and end words
defaults to only one word.
=1 don't need WORDS information
but you should give next namelist- &DATA

2. &DATA FCAT = , CAT = , HCAT = , &END.
FCAT = Real number
CAT : integer *4 or Alphanumeric field 'E012'
HCAT = integer *2

when you use second name list &
at present DIRFIX does not prompt you with any thing
next step is ~~go back and~~ using FLOC you can see the
same words ~~that~~ with FUNC=0.

To change start and end AFN on 'PENC' source
you have to modify Record 1.

START AFN = WORDS (199)

END AFN = WORDS (202)

IF it is any other source you should list ~~and~~ the words
how Record one and search for START & END AFNS

Modification of last AFN in 'PEWC' source

Description - In order to use the latest data added to the 'PEWC' (Post Encounter Corrected) Pioneer-F flux source, it is necessary to modify the last AFN (absolute file number) in this flux source

Procedure -

1). The last absolute file number is the same as the last AFN which appears in the Pioneer-F '6250' flux source. The '6250' flux source is a 4 to 1 compression of the 1600 BPI or 'EDR' source of the Pioneer-F spacecraft. (See section E in this guide for information on creating the 6250 flux source.) Obtain a current flux catalog listing by stabbing the FLIST member DECNTGT in 'SBP10.LIB.CNTL'. Compare the last AFN in the 'EDR' source with that in the '6250' source for Pioneer-F. If they are not equal perform the steps in Section F of this guide for updating the '6250' source. If they are equal make note of this last AFN.

CLIST LSTCAT

2) Execute the following operation on 360/91 using SBP10) BION

a) ATTR DCB RECFM(F) LRECL(1088) BLKSIZE(1088)
b) ALLOC DA(FLUXCAT2.DATA) N T SP(50) U(DCB)
c) COPY FLUXCAT.DATA FLUXCAT2.DATA
d) ALLOC DA(FLUXCAT2.DATA) F(FT10F001)
e) See following page

done in CLIST, following page

* $ cards begin in column # 2

```
READY
call 'SBC10.DIRFIX.LOAD(DIRFIX)'
NOW GIVE LOC NAMELIST WITH EVERYTHING RECORD,WORDS,FUNC=0 OR 1
* &loc record=1,words=199,202,func=0,&end
  199 0000CDBF :::`       52671          0.27113E-79        0 -12865
  200 000000F9 :::9         249          0.12817E-81        0    249
  201 00017386 :::f       95110          0.48958E-79        1  29574
  202 0004799E :::[:     293278          0.15097E-78        4  31134
NOW GIVE LOC NAMELIST WITH EVERYTHING RECORD,WORDS,FUNC=0 OR 1
 &loc func=1,&end
WAITING FOR DATA NAME LIST
 &data cat(202)=297940,&end
WORDS YOU CHANGED IF FUNC=1
  199,0000CDBF :::`       52671          0.27113E-79        0 -12865
  200 000000F9 :::9         249          0.12817E-81        0    249
  201 00017386 :::f       95110          0.48958E-79        1  29574
  202 00048BD4 ::(M      297940          0.15337E-78        4 -29740
NOW GIVE LOC NAMELIST WITH EVERYTHING RECORD,WORDS,FUNC=0 OR 1
 &loc record=249,words=31,35,func=0,&end
   31 0003E1E9 :::Z      254441          0.13097E-78        3  -7703
   32 0004799E :::[:     293278          0.15097E-78        4  31134
   33 D7C5D5C3 PENC -674900541         -0.38267E 28   -10299 -10813
   34 F0F64040 06   -252297152         -0.60381E 58    -3850  16448
   35 00000000 ::::            0          0.0                0      0

NOW GIVE LOC NAMELIST WITH EVERYTHING RECORD,WORDS,FUNC=0 OR 1
 &loc func=1,&end
WAITING FOR DATA NAME LIST
 &data cat(32)=297940,&end
WORDS YOU CHANGED IF FUNC=1
   31 0003E1E9 :::Z      254441          0.13097E-78        3  -7703
   32 00048BD4 ::(M      297940 was now  0.15337E-78        4 -29740
   33 D7C5D5C3 PENC -674900541         -0.38267E 28   -10299 -10813
   34 F0F64040 06   -252297152         -0.60381E 58    -3850  16448
   35 00000000 ::::            0          0.0                0      0
NOW GIVE LOC NAMELIST WITH EVERYTHING RECORD,WORDS,FUNC=0 OR 1
/*
READY
FREE DA(FLUXCAT2.DATA)
READY
COPY 'FLUXCAT2.DATA' FLUXCAT.DATA
STAB FLS
=3LIB(FCATLST)
ENDINPUT
```

2) ~~Show test loadlibcommand~~

Last AFN from '6250' source.

Same word is changed here

This is done in SB#PR.LIB.CLIST (UPD-PENC)

DOES CAT(50) is changed rather than CAT(32) ?

Check the output from this run to be sure the last AFN in the 'PENC' source equals that of the '6250' source. If so run the FLIST member FCATBACK in SBP10.LIB.CNTL to make a backup of the 'SBP10..FLUXCAT.DATA'.

5-24-79
NAL

PIONEER TRAJECTORY TAPE PROGRAM: LIST AND/OR CREATE APL DATA TAPE OF JUPITER DATA

This program uses Pioneer Trajectory Tapes which have been previously transformed from an EBCDIC to a binary tape format. Depending on the input parameters specified below, the program will list and/or generate an output tape with the following variables for the specified tapes and files. The listing contains the date (month/day/year); the time (hour:min:sec); the Jupiter-s/c distance (Rj) -- B1MAGR/71372; the clock angle to the Earth (deg) -- ARCTAN(YP1SFF/XP1SFF)-ARCTAN((YP1SFF-YPGSFF)/(XP1SFF-XPGSFF)); the clock angle to the Sun (deg) -- ARCTAN(YP1SFF/XP1SFF)-ARCTAN((YP1SFF-YPHSFF)/(XP1SFF-XPHSFF)); the ecliptic latitude (deg) -- ARCSIN(ZP1SFF/B1MAGR); Jupiter latitude (deg) -- B1LATP; Jupiter longitude (deg) -- B1LONP; Jupiter-s/c distance (km) -- B1MAGR. The tape contains a header record, with the header from the Trajectory tape, as well as a verbal description of the data format. The data format is 6R*8, 6I*2; the variables are Jupiter-s/c distance (Rj), clock angle to Earth (deg), clock angle to the Sun (deg), ecliptic latitude (deg), Jupiter latitude (deg), Jupiter longitude (deg), month, day, year (actually, year-1900), hour, minute, second. These values are computed by the formulas above.

Input Parameters:

- NFIRST - first file of Trajectory Tape to be processed
- NLAST - last file of Trajectory Tape to be processed
- DTAPIN - Pre-processed Pioneer Trajectory Input Tape
- QLIST - logical variable (=.TRUE. or .FALSE.) indicating whether or not to generate a listing of variables. (Default is .TRUE.)
- DTPOUT - the output tape for the APL data tape
- NOUT - the first file to be written on the output tape
- QWRITE - logical variable indicating whether or not to generate an output tape. (Default is .TRUE.)

(Program is currently stored in file 'SENAL.LIB2.CNTL(JUPTRAJ)' on the IBM 360.)

Jupiter Encounter

SB#PR.JUPTRAJ.FORT

normal/other Trajectory lists through -
SB#PR.LIB.CNTL(TRAJECT).

DETAILED DATA FORMAT DESCRIPTION

Physical Record 1:

	I2	size of this physical record, including the block descriptor word and the segment descriptor words.	block descriptor word
	2X		
logical record 1	I2	size of this logical record (the header plus 4 bytes of segment descriptor word)	segment descriptor word
	2X		
	11A4	Header: literal information	
	I4	Header: data record size, not including the 4 byte segment descriptor word	
logical record 2	I2	size of this logical record (data plus 4 bytes of segment descriptor word)	segment descriptor word
	2X		
	6R*8	data	
	6I*2		
logical record 3	I2	size of this logical record (the data plus 4 bytes of segment descriptor word)	segment descriptor word
	2X		
	6R*8	data	
	6I*2		

.

.

.

Physical Record 2 - n:

	I2	size of this physical record, including the block descriptor word and the segment descriptor words.	block descriptor word
	2X		
logical record 1	I2	size of this logical record (the data plus 4 bytes of segment descriptor word)	segment descriptor word
	2X		
	6R*8	data	
	6I*2		
logical record 2	I2	size of this logical record (the data plus 4 bytes of segment descriptor word)	segment descriptor word
	2X		
	6R*8	data	
	6I*2		

.

.

.

COMPUTATION FOR NUMBER OF DATA RECORDS/BLOCK

Physical Record 1:

$$\frac{((\text{Block\_Size} - 4) - \text{Header\_Logical\_Record\_Size})}{(\text{Data\_Record\_Size (from Header)} + 4)} = 106^{*}$$

Physical Record 2 - (n-1):

$$\frac{(\text{Block\_Size} - 4)}{(\text{Data\_Record\_Size (from Header)} + 4)} = 113^{*}$$

Physical Record n:

$$\frac{(\text{Block\_Size} - 4)}{(\text{Data\_Record\_Size (from Header)} + 4)} = 85^{*}$$

* These numbers have been computed for the Pioneer Trajectory Tape (for APL use) created on May 23, 1979. The original Pioneer Tape was VOL=SER=HGDØ36, file 14. This is the Saturn Encounter Prediction data.

11/25/80

PIONEER PLASMA PARAMETER PLOT PROGRAM
User's Guide

I. Overview

This program reads PIONEER plasma parameters from tapes supplied by AMES. The first datum from each hour is used to create listings and plots of parameter vs. time.

II. Input Required

A. AMES Plasma Parameter Tape as described in Project 1335-2, Technical Note 6.

B. User namelist INPUT:

Variable	Type	Default	Description
FROM(3)	I*4	0	Year-1900, day of year, and milliseconds of day of start of plot run.
TO(3)	I*4	0	Year-1900, day of year, and milliseconds of day of end of run, inclusive.
INTRVL	I*4	1	Number of hours to average together into one point.
IDENS	I*4	20	Number of intervals per plot frame.
AXMIN,AXMAX	R*4	Supplied by data	Ordinate minimum and maximum values. If not specified, the data minimum and maximum are used.
IDIVS	I*4	8	Number of divisions in the abscissa, used for tic marks and labels.
ZTAPE	A6	blank	Input tape name in EBCDIC.
IDEV	I*4	2	Output plotter device: 1=SD4060, 2=Calcomp 12", 3=Printer plot.
IVAR	A4	blank	Parameter name desired as describwed in the AMES Project 1335-2 Technial Note 6. Example: Free proton bulk velocity: 'PV07'.

Variable	Type	Default	Description
QLABL(16)	A16	blanks	Label associated with IVAR. Example: Free proton bulk velocity: 'PLASMA VELOCITY'.
QDEBUG	L*1	F	T=Print debug statements.
QPLOT	L*1	T	T=create plots
QLIST	L*1	T	T=create data listings

C. Plot tapes (If not printer plots): If making an SD4060 plot, use 7 or 9 track. Calcomp must have 7 track 556 BPI only.

III. Output Generated

A. Data Listing:

The plot number is indicated, then a columnar listing of year, day, hour, and parameter value is listed.

B. Plots:

A choice of SD4060, Calcomp 12", or printer plots is offered. The printer plots are of poor quality and are used usually for debug purposes only.

The plots are linear scale with a tic mark grid and appropriate labels.

IV. Error Handling

The STOP N statement which stops the program with return code N is used for fatal errors.

Return Code	Error Description
1	Input namelist value error. A senstence describing which parameter was in error is printed.
2	"ERROR IN POSITION,START,STOP TIME=" The times read from the plasma tape are inconsistent with user-input times.
3	"ERROR IN READING TAPE IN GETDAT" A tape read error was encountered on the AMES tape.
4	"END OF FILE REACHED IN GETPRM" While positioning to the start of run, an end of file was encountered.
5	"ERROR ON TAPE IN POSITN" While positioning to the start of run, an error was encountered on the input tape.

V. Program Performance and JCL

The program will produce 6 plots for the Calcomp plotter on the 360/75 in $.4^5$ CPU, 1.1^5 I/O. Plots on the printer take slightly more I/O time, and the 360/91 would at least half both times.

The program required 220K core to run, using LOADER. The JCL for a run is held in 'SBPIO.LIB.CNTL(SPLASPLT)'. It links into load module 'SBPIO.PLASPLT.LOAD' as follows:

```
// (JOBCARD)
// EXEC LOADER,REGION=225K,PARM='SIZE=215K,EP=MAIN'
//SYSLIB DD DSN=SPBIO.PLASPLT.LOAD,DISP=SHR
//    DD DSN=SYS2,WOLFPLOT,DISP=SHR
//SYSLIN DD DSN=SBPIO.PLASPLT.LOAD(MAIN),DISP=SHR
//       DD DSN=SBPIO.PLASPLT.LOAD(TIMES),DISP=SHR
//FT09F001 DD DCB=(DEN=2,RECFM=VS,LRECL=4112,BLKSIZE=4116),
//      LABEL=(,NL),UNIT=(800,,DEFER),VOL=SER=DUM09
//*-- THIS CARD NOT NEEDED IF NO CALCOMP PLOTS TO GENERATE
//PLOTTAPE DD DCB=(,DEN=1),LABEL=(,BLP,,OUT),UNIT=(7TRACK,,DEFER),
//    DSN=CALCOMP,VOL=SER=TAPENAME
//*-- THIS CARD NOT NEEDED IF NO SD4060 PLOTS TO GENERATE
//WOLF4060 DD LABEL=(,NL,,OUT),UNIT=(1600,,DEFER),
//    DCB=(BUFNO=1,DEN=3),DSN=NAME,VOL=SER=XXXXX
//* -- THE FOLLOWING IS A COMPLETE LIST OF POSSIBLE INPUT PARAMETERS
//*&INPUT FROM=,TO=,INTRVL=,IDENS=,AXMIN=,AXMAX=,ZTAPE=,IDEV=,
//*      IDIVS=,QDEBUG=,QPLOT=,QLIST=,IVAR=,QLABL=,&END
//*-- EXAMPLE
//DATA5 DD *
 &INPUT FROM=78,125,0,TO=78,131,0,INTRVL=1,IDENS=168,ZTAPE='XXXXX',
      IDIVS=7,QPLOT=T,QLIST=T,IDEV=3,IVAR='PV07',QLABL='PLASMA VELOCITY',&END
// EXEC NOTIFYTS
```

This program uses IBM FORTRANH, WOLFPLOT, and FTIO packages. The source is archived under 'SEJSS.PLASPLT.SOURCE(PLOT)'

Also SB#PR.LIB.CNTL (SPLASPLT)

11/25/80

PIONEER PLASMA PARAMETER PLOT PROGRAM
System Documentation

I. Overview

This program reads PIONEER plasma parameters from tapes supplied by AMES. The first datum from each hour is used to create listings and plots of parameter vs. time.

II. Input Required

A. AMES Plasma Parameter Tape as described in Project 1335-2, Technical Note 6.

B. User namelist INPUT:

Variable	Type	Default	Description
FROM(3)	I*4	0	Year-1900, day of year, and milliseconds of day of start of plot run.
TO(3)	I*4	0	Year-1900, day of year, and milliseconds of day of end of run, inclusive.
INTRVL	I*4	1	Number of hours to average together into one point.
IDENS	I*4	20	Number of intervals per plot frame.
AXMIN,AXMAX	R*4	Supplied by data	Ordinate minimum and maximum values. If not specified, the data minimum and maximum are used.
IDIVS	I*4	8	Number of divisions in the abscissa, used for tic marks and labels.
ZTAPE	A6	blank	Input tape name in EBCDIC.
IDEV	I*4	2	Output plotter device: 1=SD4060, 2=Calcomp 12", 3=Printer plot.
IVAR	A4	blank	Parameter name desired as described in the AMES Project 1335-2 Technical Note 6. Example: Free proton bulk velocity: 'PV07'.

Variable	Type	Default	Description
QLABL(16)	16A1	blanks	Label associated with IVAR. Example: Free proton bulk velocity: 'PLASMA VELOCITY'.
QDEBUG	L*1	F	T=Print debug statements.
QPLOT	L*1	T	T=Create plots
QLIST	L*1	T	T=Create data listing

C. Plot tapes (If not printer plots): If making an SD4060 plot, use 9 track, 1600 BPI. Calcomp must have 7 track 556 BPI.

III. Output Generated

A. Data Listing:

The plot number is indicated, then a columnar listing of year, day, hour, and parameter value is listed.

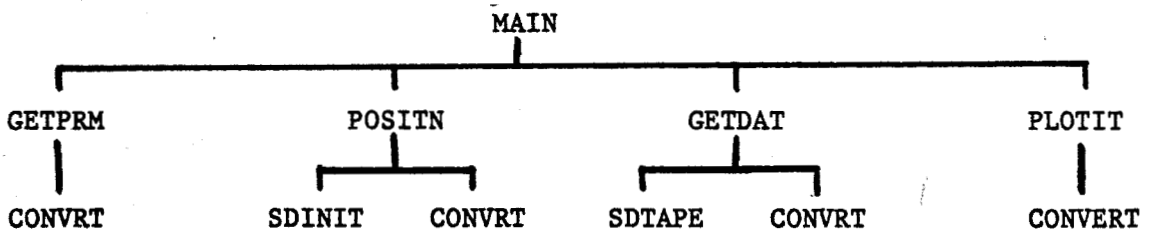
B. Plots:

A choice of SD4060, Calcomp 12", or printer plots is offered. The printer plots are of poor quality and are used usually for debug purposes only.

The plots are linear scale with a tic mark grid and appropriate labels.

IV. Program Structure

A. Block Diagram

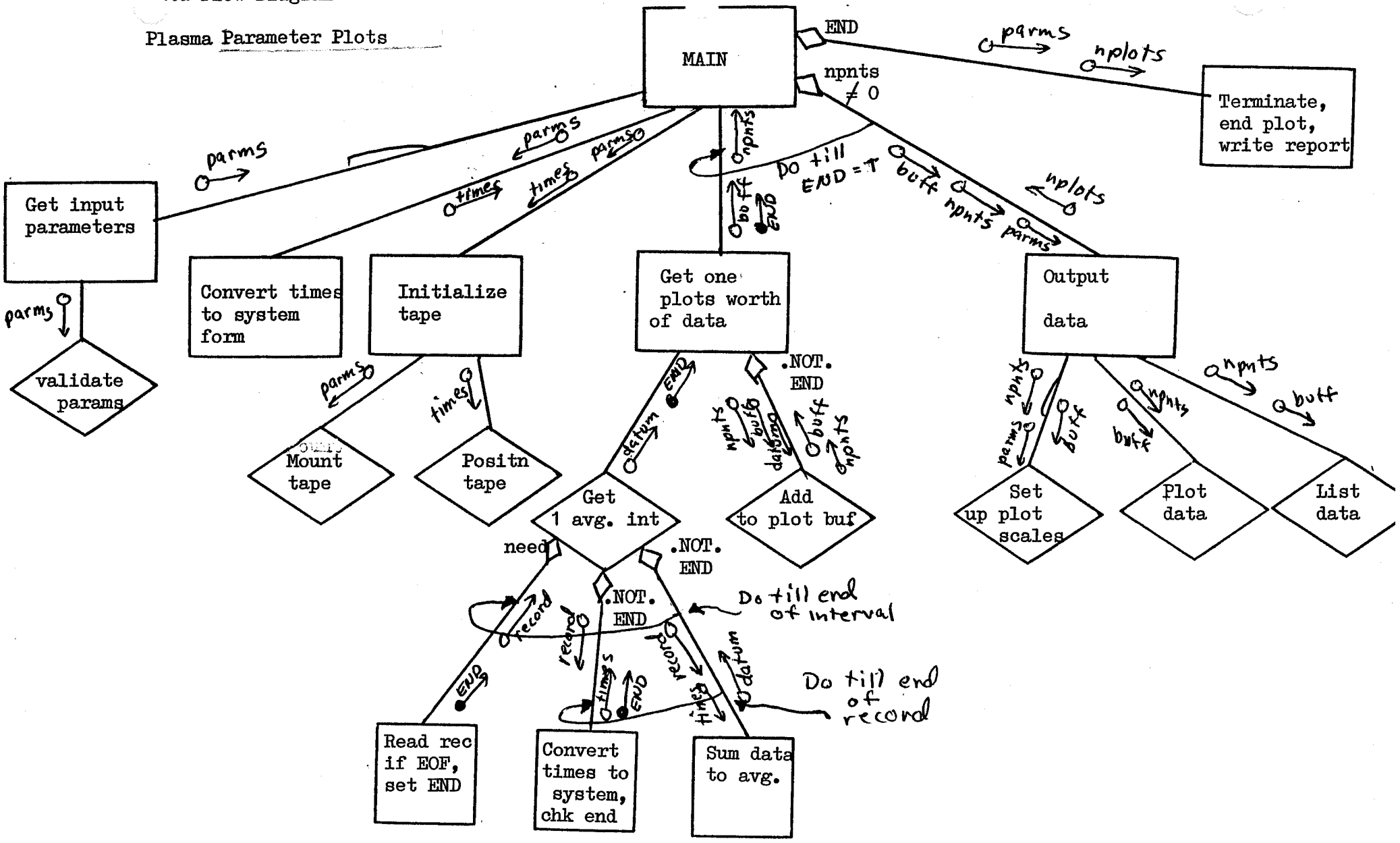


B. Module Definition

1. MAIN - Controls program flow
2. GETPRM - Reads user parameters and validates them
3. POSITN - Positions tape to start time
4. GETDAT - Collects one plot's worth of data
5. PLOTIT - Plots the data
6. CONVRT - Converts to and from system time form

Data Flow Diagram

Plasma Parameter Plots



7. SDTAPE - Supplied by AMES to extract parameter and time from the tape
8. SDINIT - Supplied by AMES to initialize commons for SDTAPE

V. Data Flow Diagram

(See next page)

VI. Error Handling

The STOP N statement which stops the program with return code N is used for fatal errors.

Return Code	Error Description
1	Input namelist value error. A sentence describing which parameter was in error is printed.
2	"ERROR IN POSITION,START,STOP TIME=" The times read from the plasma tape are inconsistent with user-input times.
3	"ERROR IN READING TAPE IN GETDAT" A tape read error was encountered on the AMES tape.
4	"END OF FILE REACHED IN GETPRM" While positioning to the start of run, an end of file was encountered.
5	"ERROR ON TAPE IN POSITN" While positioning to the start of run, an error was encountered on the input tape.

VII. Detail on Coding

A. Timing System: Times are converted to a real number which is intervals (hours) since start of run.

B. Common Blocks:

1. /PARMS/START,STOP,ZTAPE,INTRVL,IDENS,PSTART,PEND,PTOTAL,AXMIN, AXMAX,IDIVS,QDEBUG,QAXIS

   (See Section II.B for details on those variables which are namelist variables. Other are listed below.)

Variable	Type	Description
START,STOP	R*4	System time for start and end of run
PSTART,PEND	R*4	System time for start and end of current plot frame
PTOTAL	R*4	Total intervals per plot
QAXIS	L*1	T=Calculates the ordinate axis minimum and maximum from the plot data

2. See AMES Publications Project 1335-2, Technical Note 7, page 5 for definitions of common variables in SDINFO,SDFLAG,SDATA, SDKEY,SDNEWV.

3. /TIMES/IEPOCH,INTDY,MSINT

Variable	Type	Description
IEPOCH	I*4	Closest leap year-1900 before start time of run
INTDY	I*4	Number of intervals per day
MSINT	I*4	Number of milliseconds per interval

Common Name	Module Occurrence	Input/Output	Function
PARMS	GETDAT	I,O	Holds plotting paramters
	GETPRM	O	
	PLOTIT	I	
	POSITN	I	
TIMES	CONVRT	I	Holds timing information to convert to and from system times
	GETPRM	O	
SDATA,SDKEYS	POSITN	O	Holds AMES tape reading parameters
SDINFO	SDTAPE	I,O	
SDVAR,SDFLAG,	GETDAT	I,O	Holds AMES tape to be returned variables
SDNEWV	POSITN	O	
	SDTAPE	I,O	
VARBLS	GETPRM,POSITN,	I	Holds plasma parameter information.
	PLOTIT	O	

4. /VARBLS/INPT(4),NUMPRM,QLABL(16)

Variable	Type	Description
INPUT	4A4	The first three are the time parameters 'TI01','TI02', and 'TI06'. The fourth is the user input parameter name.
NUMPRM	I*4	Number of parameters, equal to 4.
QLABL	16A1	Label for parameter input by user.

C. Individual Module Documentation

All modules were designed, coded, and tested by Jenny Jacques, Code 664, September 1980.

1.a. MAIN - Controls program flow

b. Calls: Wolfplot routines, POSITN,GETDAT,GETPRM,PLOTIT

c. No commons used

d. Local Variables:

Name	Type	Description
BUFF(500,2)	R*4	Holds up to 500 plot data, BUFF(N,1)=parameter value BUFF(N,2)=time (system form)

Name	Type	Description
QEND	L*1	T=end of run data
NPNTS	I*4	Number of data to plot
IPLDEV	I*4	Wolfplot plotter device specification
NPLOTS	I*4	Number of plots done
QPLOT	L*1	T=create plots
QLIST	L*1	T=create listing of data

e. Logic:

The user parameters are read in (GETPRM) and plotter and tape are initialized (POSITN,PLOTST,CPRIME). Then the plot loop is done,first filling BUFF with one plot's data (GETDAT), then plotting it (PLOTIT), etc. When QEND=T, stop after plotting any data collected in BUFF.

2.a. CONVRT - Convert to and from year, day, ms of day, and system time form of one real number used for plotting.

b. Calling sequence:
CALL CONVRT(ITIME,SYSTEM,ITYPE)

Variable	Type	Description
ITIME(3)	I*4	Year, day, ms of day
SYSTEM	R*4	System time form of ITIME
ITYPE	I*4	1=Convert to system 2=Convert to ITIME

c. Calls: none
Called by: GETPRM,GETDAT,PLOTIT,POSITN

d. Common blocks:

Common	Variables	I,O
TIMES	IEPOCH,INTDY,MSINT	I

e. Local variables: none

f. Logic:

The algorithm used multiplies the years since epoch leap year by 365.25 and truncates. This adds 1 day every four years. Then the number of intervals per day are added and this becomes the system time form. This process is inverted when converting from system time form.

3.a. GETDAT - Gets the data for one plot

b. Calling sequence:
CALL GETDAT(BUFF,NPNTS,QEND)

See MAIN, Section VII.C.1 for variable explanation.

c. Calls: CONVRT
Called by: MAIN

d. Common blocks:

Common	Variables	I,O
PARMS	START,STOP,PSTART,PEND	I,O
	INTRVL,IDENS,PTOTAL,QDEBUG	I
SDVAR	VARVAL	I,O
SDFLAG	LEOF,LERR	I,O
SDNEWV	LVN	I,O

e. Local variables:

Variable	Type	Description
IHR,IOLDHR	I*4	Holds current and previous hour number of datum.
NUM	I*4	Number of data in SUM
SUM	R*4	Sum of parameters for an average
IODAY	I*4	Previous datum's day numbers

f. Logic:

After variables are initialized, there is a loop collecting one averaging interval of data. This interval is stored and the loop continues until the plot end time is encountered.

3.a. GETPRM - Get user parameters from namelist

b. Calling sequence:
CALL GETPRM(IPLDEV,QPLOT,QLIST)

See MAIN, Section VII.C.1 for definition of IPLDEV, QPLOT and QLIST.

c. Calls: SDINIT,CONVRT
Called by: MAIN

d. Common blocks:

Common	Variables	I,O
TIMES	All	O
PARMS	All but PEND	O

e. Local variables:

Variable	Type	Description
IHI(7),ILO(7)	I*4	Maximum and minimum input integer allowable values: (1)-(3) → FROM and to (4) → INTRVL, (5) → IDENS, (6) → IDEV, (7) → IDIVS
Variable	Type	Description
ICODE(3)	I*4	Wolfplot plotter code for each allowable output plotter device: (1)=SD4060, (2)=Calcomp 12", (3)=printer

f. Logic:

Read in the INPUT namelist. Validate the parameters. Initialize timing variables and plotting variables.

4.a. PLOTIT - Plots one frame of data

b. Calling sequence:

```
CALL PLOTIT(BUFF,NPNTS,NPLOTS,QPLOT,QLIST)
```

See Section VII.C.1 MAIN for description of variables.

c. Calls: CONVRT, Wolfplot routines
Called by: MAIN

d. Common blocks:

Common	Variables	I,O
PARMS	PSTART,PEND,IDIVS,QDEBUG	I
	QAXIS,AXMIN,AXMAX	I,O

e. Local variables:

Variable	Type	Description
POS	R*4	Used as abscissa position for each tic mark and label
AMAX,AMIN	R*4	Returned ordinate maximum and minimum if to be obtained from the data
INCS	I*4	Same as IDIVS
QDAY(3)	L*1	Holds EBCDIC day number
RINC	R*4	Time increment between tic marks
IOLDY	I*4	Holds previous day number to check for change
ITIME(3)	I*4	Year, day, ms of day for each datum label
QDATE(8)	L*1	Holds EBCDIC run date
QHOUR(2)	L*1	Holds EBCDIC hour for label
QPLTS(11)	L*1	Holds EBCDIC plot number label
QTIME(21)	L*1	Holds EBCDIC start time label

f. Logic:

After intializing variables, the ordinate extreme values are calculated if QAXIS=T. The plot frame is scaled, labels put on, tic marks on upper and left axis drawn, and the time axis labels drawn. Then the data is listed, then plotted and the plot frame is ended.

5.a. POSITN - Positions input tape to correct time

b. Calling sequence:

```
CALL POSITN
```

c. Calls: SDINIT,SDTAPE,FTIO,CONVRT
Called by: MAIN

d. Common blocks:

Common	Variables	I,O
PARMS	START,STOP,ZTAPE	I
SDVAR	IDVAR,VARVAL	I,O
SDINFO	KPRTUN,KTAPUN	O
SDFLAG	LEOF,LERR	I
SDKEYS	IDKEY,KEYVAL	O

e. Local variables:

Variable	Type	Description
TIME	R*4	System time form of the record

f. Logic:

The common blocks are initialized and the input tape is mounted, then rewound to close the DCB for future Fortran reads. A loop follows to read in each record until the start time is reached. This routine returns the first plot data.

6. SDINIT,SDTAPE - AMES supplied these routines to decode their plasma parameters tape. See the document AMES Project 1335-2, Technical Note 7 for explanation of these subroutines.

VIII. Program Performance and JCL

The program will produce 6 plots for the Calcomp plotter on the 360/75 in $.4^5$ CPU, 1.1^5 I/O. Plots on the printer take slightly more I/O time, and the 360/91 would at least half both times.

The program required 220K core to run, using LOADER. The JCL for a run is held in 'SBPIO.LIB.CNTL(SPLASPLT)'. It links into load module 'SBPIO.PLASPLT.LOAD' as follows:

```
// (JOBCARD)
// EXEC LOADER,REGION=225K,PARM='SIZE=215K,EP=MAIN'
//SYSLIB DD DSN=SPBIO.PLASPLT.LOAD,DISP=SHR
//    DD DSN=SYS2,WOLFPLOT,DISP=SHR
//SYSLIN DD DSN=SBPIO.PLASPLT.LOAD(MAIN),DISP=SHR
//       DD DSN=SBPIO.PLASPLT.LOAD(TIMES),DISP=SHR
```

```
//FT09F001 DD DCB=(DEN=2,RECFM=VS,LRECL=4112,BLKSIZE=4116),
//       LABEL=(,NL),UNIT=(800,,DEFER),VOL=SER=DUM09
//*-- THIS CARD NOT NEEDED IF NO CALCOMP PLOTS TO GENERATE
//PLOTTAPE DD DCB=(,DEN=1),LABEL=(,BLP,,OUT),UNIT=(7TRACK,,DEFER),
//     DSN=CALCOMP,VOL=SER=TAPENAME
//*-- THIS CARD NOT NEEDED IF NO SD4060 PLOTS TO GENERATE
//WOLF4060 DD LABEL=(,NL,,OUT),UNIT=(1600,,DEFER),
//     DCB=(BUFNO=1,DEN=3),DSN=NAME,VOL=SER=XXXXX
//* -- THE FOLLOWING IS A COMPLETE LIST OF POSSIBLE INPUT PARAMETERS
//*&INPUT FROM=,TO=,INTRVL=,IDENS=,AXMIN=,AXMAX=,ZTAPE=,IDEV=,
//*       IDIVS=,QDEBUG=,QPLOT=,QLIST=,IVAR=,QLABL=,&END
//*-- EXAMPLE
//DATA5 DD *
 &INPUT FROM=78,125,0,TO=78,131,0,INTRVL=1,IDENS=168,ZTAPE='XXXXX',
      IDIVS=7,QPLOT=T,QLIST=T,IDEV=3,IVAR='PV07',QLABL='PLASMA VELOCITY',&END
// EXEC NOTIFYTS
```

This program uses IBM FORTRANH, WOLFPLOT, and FTIO packages.
The source is archived under 'SEJSS.PLASPLT.SOURCE(PLOT)'

Plasma Parameters

The following table describes the parameters available on the AMES plasma tapes. They may be used with the plasma parameter plot program.

Name (IVAR)	Description
CS01	Chi-Square
EP01	Error with free proton temperature (° K)
EP07	Error with free proton bulk velocity (KM/SEC)
EP08	Error with free proton bulk azimuthal angle (degrees)
EP09	Error with free proton bulk polar angle (degrees)
EP10	Error with free proton number density (proton/CC)
PT01	Free proton temperature (° K)
PV07	Free proton bulk velocity (KM/SEC)
PA08	Free proton bulk azimuthal angle (degrees)
PA09	Free proton bulk polar angle (degrees)
PN10	Free proton bulk number density (proton/CC)
** TI01	Year
** TI02	Day of year
** TI06	Milliseconds of day of year
TI91	Year data was generated
TI92	Day of year data was generated

** Included in all listings and plots

Plasma Velocity Tapes

P-10

1977	May 8	thru	1978	Dec. 31
1980	Jan. 1	thru	1980	Dec 31

P-11

1978	May 8 07:00	thru	1978	Dec 31	08:09
1979	June 8	thru	1979	Sept. 25	

Dr. McDonald

The above are the periods currently available for Plasma velocities.

JHB
10/20/80

```
      PFPV01 IN SBPIO'S @60631, READ-ONLY, P-10 PLASMA VELOCITY
             USAGE: SLOT-     0 SINCE 75/120;  VOL-     0 SINCE 80/294
      PFPV02 IN SBPIO'S @60632, READ-ONLY, P-10 PLASMA VELOCITY
             USAGE: SLOT-     0 SINCE 75/120;  VOL-     0 SINCE 80/294
      PFPV03 IN SBPIO'S @60633, READ-ONLY, P-10 PLASMA VELOCITY
             USAGE: SLOT-     0 SINCE 75/120;  VOL-     0 SINCE 80/294
      PFPV04 IN SBPIO'S @60634, READ-ONLY, P-10 PLASMA VELOCITY
             USAGE: SLOT-     0 SINCE 75/120;  VOL-     0 SINCE 80/294
      PFPV05 IN SBPIO'S @60635, READ-ONLY, P-10 PLASMA VELOCITY
             USAGE: SLOT-     0 SINCE 75/120;  VOL-     0 SINCE 80/294
      PGPV01 IN SBPIO'S @60636, READ-ONLY, P-11 PLASMA VELOCITY
             USAGE: SLOT-     0 SINCE 75/120;  VOL-     0 SINCE 80/294
      PGPV02 IN SBPIO'S @60637, READ-ONLY, P-11 PLASMA VELOCITY
             USAGE: SLOT-     0 SINCE 75/120;  VOL-     0 SINCE 80/294
      PGPV03 IN SBPIO'S @60638, READ-ONLY, P-11 PLASMA VELOCITY
             USAGE: SLOT-     0 SINCE 75/120;  VOL-     0 SINCE 80/294
      PGPV04 IN SBPIO'S @60639, READ-ONLY, USED BY ZBKMW, P-11 PLASMA VELOCITY
             USAGE: SLOT-     1 SINCE 75/120;  VOL-     0 SINCE 80/294
e{
---END OF TLSUPDTE LISTING          NUMBER OF ERRORS-      0          RETURN CODE- 0
READY
```

National Aeronautics and
Space Administration

Ames Research Center
Moffett Field, California 94035

NASA

Reply to Attn of: SST:245-3

August 27, 1980

Dr. F.B. McDonald, Chief
Laboratory for High Energy Astrophysics
Code 660
NASA-Goddard Space Flight Center
Greenbelt, MD 20771

Dear Dr. McDonald:

In response to your letter of July 25, this letter is sent with three magnetic tapes that contain unedited solar wind plasma parameters from Pioneers 10 and 11. Pioneer 10 plasma data for 1977 beginning with day 128, and Pioneer 11 data both for 1978 beginning with day 125, and for 1979 for days 152 to 268, are the time intervals included. We do not yet have Pioneer 10 plasma data in this form for 1979, or for Pioneer 11 for 1977.

The format of these tapes and a sample program for reading them are described in Informatics, Inc., Technical Notes 6 and 7, Project 1335-2, which are also included.

Unedited plasma parameters occasionally will contain incorrect values. The bulk speed values should contain the fewest incorrect cases, with most of those remaining within 10% of the correct value. The number density and temperature values, respectively, should have increasingly greater percentages of incorrect values.

Sincerely,

John D. Mihalov

John D. Mihalov
Research Scientist
Theoretical and Planetary Studies Branch

Enclosures

Project 1335-2 - Technical Note 6

Originator: Alice Crema

Date: December 28, 1973
April 26, 1974 Revision 1
Oct. 20, 1975 Revision 2

Subject: Plasma Parameter
Data Format
Block II and Block III

TABLE OF CONTENTS

Listed here

The Plasma Parameter Data exists on 9-track magnetic tape and/or disk dataset. The data on a 9-track magnetic tape is written using density of 800 BPI and variable sequential record format. The logical record length equals 4112 and the block size equals 4116.

The Plasma Parameter data contains an infinite number of variables within ID information, proton parameter and alpha parameter categories. New variables may be added at any time. The Plasma Parameter datasets are set up with descriptor records which contain the four letter code names of the variables existing in that type of data record. The descriptor records are at the beginning of the dataset describing the data existing on that dataset. The purpose of the descriptor records is to give flexibility to the data records. If, for example, a new proton variable is to be added to the proton data records, a new descriptor record would be added and no change would be needed to the existing data records. For example, there is a descriptor record for the Block III one-temperature proton parameter model and also a descriptor for the Block II one-temperature alpha parameter model.

The Plasma Parameter Data records are read with routine SDTAPE. Refer to Technical Note 1335-2-7 for the description of routine SDTAPE.

Part 2

PLASMA PARAMETER VARIABLE CODE NAMES AND DESCRIPTIONS

AT01 = FREE ALPHA TEMPERATURE (DEGREES KELVIN)
AT02 = FREE ALPHA TEMPERATURE PERPENDICULAR (DEGREES KELVIN)
A 3 = FREE ALPHA TEMPERATURE PARALLEL (DEGREES KELVIN)
AT04 = SPARE
AT05 = SPARE
AT06 = SPARE
AV07 = FREE ALPHA BULK VELOCITY (KM/SEC)—
AA08 = FREE ALPHA BULK AZIMUTHAL ANGLE (DEGREES)
AA09 = FREE ALPHA BULK POLAR ANGLE (DEGREES)
AA10 = FREE ALPHA NUMBER DENSITY (PROTONS/CC)
AA11 = FREE ALPHA ANISOTROPIC AZIMUTHAL ANGLE (DEGREES)
AA12 = FREE ALPHA ANISOTROPIC POLAR ANGLE (DEGREES)

AF51 = FIXED ALPHA TEMPERATURE (DEGREES KELVIN)
AF52 = FIXED ALPHA TEMPERATURE PERPENDICULAR (DEGREES KELVIN)
AF53 = FIXED ALPHA TEMPERATURE PARALLEL (DEGREES KELVIN)
AF54 = SPARE
AF55 = SPARE
AF56 = SPARE
AF57 = FIXED ALPHA BULK VELOCITY (KM/SEC)—
AF58 = FIXED ALPHA BULK AZIMUTHAL ANGLE (DEGREES)
AF59 = FIXED ALPHA BULK POLAR ANGLE (DEGREES)
AF60 = FIXED ALPHA NUMBER DENSITY (PROTONS/CC)
AF61 = FIXED ALPHA ANISOTROPIC AZIMUTHAL ANGLE (DEGREES)
AF62 = FIXED ALPHA ANISOTROPIC POLAR ANGLE (DEGREES)

Key: △ Data on Pn 8-9 tapes
o Data on Pn 10-11 tapes

∞∞CS01 = Chi-Square

EA = ERROR WITH ALPHA TEMPERATURE (DEGREES IN KELVIN)
EA02 = ERROR WITH ALPHA TEMPERATURE PERPENDICULAR (DEGREES IN KELVI
EA03 = ERROR WITH ALPHA TEMPERATURE PARALLEL (DEGREES IN KELVIN)
EA04 = SPARE
EA05 = SPARE
EA06 = SPARE
EA07 = ERROR WITH ALPHA BULK VELOCITY (KM/SEC)
EA08 = ERROR WITH ALPHA BULK AZIMUTHAL ANGLE (DEGREES)
EA09 = ERROR WITH ALPHA BULK POLAR ANGLE (DEGREES)
EA10 = ERROR WITH ALPHA NUMBER DENSITY (PROTONS/CC)
EA11 = ERROR WITH ALPHA ANISOTROPIC AZIMUTHAL ANGLE (DEGREES)
EA12 = ERROR WITH ALPHA ANISOTROPIC POLAR ANGLE (DEGREES)
EP01 = ERROR WITH FREE PROTON TEMPERATURE (DEGREES KELVIN)
EP02 = ERROR WITH FREE PROTON TEMPERATURE PERPENDICULAR (DEGREES KELVIN)
EP03 = ERROR WITH FREE PROTON TEMPERATURE PARALLEL (DEGREES KELVIN
EP04 = ERROR WITH FREE PROTON TEMPERATURE PERPENDICULAR (DEGREES KELVIN)
EP05 = ERROR WITH FREE PROTON TEMPERATURE INWARD (DEGREES KELVIN)
EP06 = ERROR WITH FREE PROTON TEMPERATURE OUTWARD (DEGREES KELVIN)
EP07 = ERROR WITH FREE PROTON BULK VELOCITY (KM/SEC)
EP 8 = ERROR WITH FREE PROTON BULK AZIMUTHAL ANGLE (DEGREES)
EP09 = ERROR WITH FREE PROTON BULK POLAR ANGLE (DEGREES)
EP10 = ERROR WITH FREE PROTON NUMBER DENSITY (PROTON/CC)
EP11 = ERROR WITH FREE PROTON ANISOTROPIC AZIMUTHAL ANGLE (DEGREES)
EP12 = ERROR WITH FREE PROTON ANISOTROPIC POLAR ANGLE (DEGREES)
EP13 = ERROR WITH FREE PROTON TEMPERATURE PERPENDICULAR FOR COMPONENT 1 (DEGREES KELVIN)
EP14 = ERROR WITH FREE PROTON ANISOTROPIC AZIMUTHAL ANGLE OF 'EP13' (DEGREES)
EP15 = ERROR WITH FREE PROTON ANISOTROPIC POLAR ANGLE OF 'EP13' (DEGREES)
EP16 = ERROR WITH FREE PROTON TEMPERATURE PERPENDICULAR FOR COMPONENT 2 (DEGREES KELVIN)
EP17 = ERROR WITH FREE PROTON ANISOTROPIC AZIMUTHAL ANGLE OF 'EP16' (DEGREES)
EP18 = ERROR WITH FREE PROTON ANISOTROPIC POLAR ANGLE OF 'EP16' (DEGREES)

II = MODE: 0 = UNSUPPRESSED/1 = SUPPRESSED
ID02 = SPACECRAFT ID NUMBER
ID03 = BIT RATE
ID04 = MODE: 0 = FSM/1 = MFM
ID05 = CALIBRATION IDENTIFICATION FOR BLOCK 2
IDA5 = FIRST PART OF CALIBRATION IDENTIFICATION FOR BLOCK 3
IDB5 = SECOND PART OF CALIBRATION IDENTIFICATION FOR BLOCK 3
ID06 = TYPE OF PROTON OR ALPHA MODEL
ID07 = MODEL NUMBER
ID08 = DETECTOR: 0=A/1=B
ID09 = MODE: 1,2,3,4
ID10 = NUMBER OF STEPS/CYCLE
ID11 = FOR DETECTOR A: SECTOR INTEGRATION PERIOD
ID12 = NUMBER OF HALF-REVOLUTIONS PER STEP
ID13 = COLLECTOR NUMBER OF PEAK REPORT
ID14 = ORDINAL NUMBER FOR THIS CYCLE IN COMBINED CYCLE (E.G.,1)
ID15 = NO. OF CYCLES COMBINED WITH AND INCLUDING THIS CYCLE (E.G.,2)
ID16 = SPARE
ID17 = SPARE
ID18 = SPARE
ID19 = SPARE
ID20 = SPARE
ID21 = SPARE
ID22 = SPARE

PF51 = FIXED PROTON TEMPERATURE (DEGREES KELVIN)
PF52 = FIXED PROTON TEMPERATURE PERPENDICULAR (DEGREES KELVIN)
PF53 = FIXED PROTON TEMPERATURE PARALLEL (DEGREES KELVIN)
PF54 = FIXED PROTON TEMPERATURE PERPENDICULAR (DEGREES KELVIN)
PF55 = FIXED PROTON TEMPERATURE INWARD (DEGREES KELVIN)
PF56 = FIXED PROTON TEMPERATURE OUTWARD (DEGREES KELVIN)
PF57 = FIXED PROTON BULK VELOCITY (KM/SEC)
PF58 = FIXED PROTON BULK AZIMUTHAL ANGLE (DEGREES)
PF59 = FIXED PROTON BULK POLAR ANGLE (DEGREES)
PF60 = FIXED PROTON NUMBER DENSITY (PROTONS/CC)
PF61 = FIXED PROTON ANISOTROPIC AZIMUTHAL ANGLE (DEGREES)
PF62 = FIXED PROTON ANISOTROPIC POLAR ANGLE (DEGREES)
PF63 = FIXED PROTON TEMPERATURE PERPENDICULAR FOR COMPONENT 1 (DEGREES KELVIN)
PF64 = FIXED PROTON ANISOTROPIC AZIMUTHAL ANGLE FOR COMPONENT 1 (DEGREES)
PF65 = FIXED PROTON ANISOTROPIC POLAR ANGLE FOR COMPONENT 1 (DEGREES)
PF66 = FIXED PROTON TEMPERATURE PERPENDICULAR FOR COMPONENT 2 (DEGREES KELVIN)
PF67 = FIXED PROTON ANISOTROPIC AZIMUTHAL ANGLE FOR COMPONENT 2 (DEGREES)
PF68 = FIXED PROTON ANISOTROPIC POLAR ANGLE FOR COMPONENT 2 (DEGREES)

PT01 = FREE PROTON TEMPERATURE (DEGREES KELVIN)
PT02 = FREE PROTON TEMPERATURE PERPENDICULAR (DEGREES KELVIN)
PT03 = FREE PROTON TEMPERATURE PARALLEL (DEGREES KELVIN)
PT04 = FREE PROTON TEMPERATURE PERPENDICULAR (DEGREES KELVIN)
PT05 = FREE PROTON TEMPERATURE INWARD (DEGREES KELVIN)
PT06 = FREE PROTON TEMPERATURE OUTWARD (DEGREES KELVIN)
PV07 = FREE PROTON BULK VELOCITY (KM/SEC)
PA08 = FREE PROTON BULK AZIMUTHAL ANGLE (DEGREES)
PA09 = FREE PROTON BULK POLAR ANGLE (DEGREES)
PN10 = FREE PROTON NUMBER DENSITY (PROTON/CC)
PA11 = FREE PROTON ANISOTROPIC AZIMUTHAL ANGLE (DEGREES)
PA12 = FREE PROTON ANISOTROPIC POLAR ANGLE (DEGREES)
PT13 = FREE PROTON TEMPERATURE PERPENDICULAR FOR COMPONENT 1 (DEGREES KELVIN)
PA14 = FREE PROTON ANISOTROPIC AZIMUTHAL ANGLE OF 'PT13' (DEGREES)
PA15 = FREE PROTON ANISOTROPIC POLAR ANGLE OF 'PT13' (DEGREES)
PT16 = FREE PROTON TEMPERATURE PERPENDICULAR FOR COMPONENT 2 (DEGREES KELVIN)
PA17 = FREE PROTON ANISOTROPIC AZIMUTHAL ANGLE OF 'PT16' (DEGREES)
PA18 = FREE PROTON ANISOTROPIC POLAR ANGLE OF 'PT16' (DEGREES)

RN01 = NOISE LEVEL FOR COLLECTOR 1 (1.E-14 AMPS)
RN02 = NOISE LEVEL FOR COLLECTOR 2 (1.E-14 AMPS)
RN03 = NOISE LEVEL FOR COLLECTOR 3 (1.E-14 AMPS)
RN04 = NOISE LEVEL FOR COLLECTOR 4 (1.E-14 AMPS)
RN05 = NOISE LEVEL FOR COLLECTOR 5 (1.E-14 AMPS)
RN06 = SPARE
RN07 = SPARE
RN08 = SPARE
RN09 = SPARE
RN10 = SPARE
RN11 = SPARE
RN12 = SPARE
RN13 = SPARE
RN14 = SPARE
RN15 = SPARE
RN16 = SPARE
RN17 = SPARE
RN18 = SPARE
RN19 = SPARE
RN20 = SPARE
RN21 = SPARE
RN22 = SPARE
RN23 = SPARE
RN24 = SPARE
RN25 = SPARE
RN26 = SPARE
RP01 = RATIO OF PEAK REPORT/NOISE (1.E-13 AMPS)
RP02 = PEAK REPORT (1.E-14 AMPS)

TI01 = YEAR
TI02 = DAY OF YEAR
TI06 = MILLISECONDS OF DAY OF YEAR

TI91 = YEAR DATA WAS GENERATED
TI92 = DAY OF YEAR DATA WAS GENERATED

Part 3

PLASMA PARAMETER DATA DESCRIPTOR RECORDS

DESCRIPTION OF DESCRIPTOR RECORDS

Column 1

I02
I06
D01
D02
D03
D04
D05
D06
I91
I92
S01
T01
V07
A08
A09
N10
P01
P07
P08
P09
P10
F51
F57
F58
F59
F60

Column 2

TI02
TI06
ID01
ID02
ID03
ID04
ID05
ID06
TI91
TI92
CS01
PT02
PT03
PV07
PA08
PA09
PN10
PA11
PA12
EP02
EP03
EP07
EP08
EP09
EP10
EP11
EP12
PF52
PF53
PF57
PF58
PF59
PF60
PF61
PF62

Column 3

TI02
TI06
ID01
ID02
ID03
ID04
ID05
ID06
TI91
TI92
CS01
PT04
PT05
PT06
PV07
PA08
PA09
PN10
PA11
PA12
EP04
EP05
EP06
EP07
EP08
EP09
EP10
EP11
EP12
PF54
PF55
PF56
PF57
PF58
PF59
PF60
PF61
PF62

Column 4

TI02
TI06
ID01
ID02
ID03
ID04
IDA5
IDB5
ID07
ID08
ID09
ID10
ID11
ID12
ID13
ID14
ID15
ID16
ID17
ID18
ID19
ID20
ID21
ID22
TI91
TI92
CS01
RP01
RN01
RN02
RN03
RN04
RN05
PT01
PV07
PA08
PA09
PN10
EP01
EP07
EP08
EP09
EP10

Column 5

TI02
TI06
ID01
ID02
ID03
ID04
IDA5
IDB5
ID07
ID08
ID09
ID10
ID11
ID12
ID13
ID14
ID15
ID16
ID17
ID18
ID19
ID20
ID21
ID22
TI91
TI92
CS01
RP02
RN01
RN02
RN03
RN04
RN05
PT02
PT03
PV07
PA08
PA09
PN10
PA11
PA12
PT13
PA14
PA15
PT16
PA17
PA18
EP02
EP03
EP07
EP08
EP09
EP10
EP11
EP12
EP13
EP14
EP15
EP16
EP17
EP18

Column 6

TI02
TI06
ID01
ID02
ID03
ID04
IDA5
IDB5
ID07
ID08
ID09
ID10
ID11
ID12
ID13
ID14
ID15
ID16
ID17
ID18
ID19
ID20
ID21
ID22
TI91
TI92
CS01
RP02
RN01
RN02
RN03
RN04
RN05
PT01
PV07
PA08
PA09
PN10
EP01
EP07
EP08
EP09
EP10

Column 7

TI02
TI06
ID01
ID02
ID03
ID04
IDA5
IDB5
ID07
ID08
ID09
ID10
ID11
ID12
ID13
ID14
ID15
TI91
TI92
PT01
PV07
PA08
PA09
PN10
EP01
EP07
EP08
EP09
EP10
CS01
RP02
RN01
RN02
.
.
.
RN26

Column 8

AS FOR 7

RN01
RN02
.
.
RN05

Column 9

TI02
TI06
ID01
ID02
ID03
ID04
IDA5
IDB5
ID07
ID08
ID09
ID10
ID11
ID12
ID13
ID14
ID15
TI91
TI92
PT02
PT03
PV07
PA08
PA09
PN10
PA11
PA12
EP02
EP03
EP07
EP08
EP09
EP10
EP11
EP12
CS01
RP02
RN01
RN02
.
.
.
RN26

Column 10

AS FOR 9

RN01
RN02
.
.
RN05

27	36	39	44	53					38	NWDS
TI01	TI01	TI01	TI01						TI01	
TI02	TI02	TI02	TI02	FIRST					TI02	
TI06	TI06	TI06	TI06						TI06	
ID01	ID01	ID01	ID01	53					ID01	
ID02	ID02	ID02	ID02						ID02	
ID03	ID03	ID03	ID03	ITEMS					ID03	
ID04	ID04	ID04	ID04						ID04	
ID05	ID05	ID05	IDA5	OF					IDA5	
ID06	ID06	ID06	IDB5						IDB5	
TI91	TI91	TI91	ID07	JTYPE					ID07	
TI92	TI92	TI92	ID08						ID08	
CS01	CS01	CS01	ID09	7					ID09	
AT01	AT02	AT04	ID10						ID10	
AV07	AT03	AT05	ID11	(LAST					ID11	
AA08	AV07	AT06	ID12						ID12	
AA09	AA08	AV07	ID13	5					ID13	
AN10	AA09	AA08	ID14						ID14	
EA01	AN10	AA09	ID15	MISS-					ID15	
EA07	AA11	AN10	ID16						ID16	
EA08	AA12	AA11	ID17	ING)					ID17	
EA09	EA02	AA12	ID18						ID18	
EA10	EA03	EA04	ID19						ID19	
AF51	EA07	EA05	ID20						ID20	
AF57	EA08	EA06	ID21						ID21	
AF58	EA09	EA07	ID22						ID22	
AF59	EA10	EA08	TI91						TI91	
AF60	EA11	EA09	TI92						TI92	
	EA12	EA10	CS01						CS01	
	AF52	EA11	RP02						PT01	
	AF53	EA12	RN01						PV07	
	AF57	AF54	RN02						PA08	
	AF58	AF55	RN03						PA09	
	AF59	AF56	RN04						PN10	
	AF60	AF57	RN05						EP01	
	AF61	AF58	AT01						EP07	
	AF62	AF59	AV07						EP08	
		AF60	AA08						EP09	
		AF61	AA09						EP10	
		AF62	AN10							
			EA01							
			EA07							
			EA08							
			EA09							
			EA10							

Project 1335-2 - Technical Note 7

Originator: Nelson Beebe
Alice Crema

Date: June 19, 1973

Subject: Program Description of Subroutine SDTAPE
Revision 1

TABLE OF CONTENTS

PROGRAM DESCRIPTION
of
SUBROUTINE SDTAPE

PURPOSE

To be used by all programs that need input/output from Plasma parameter data sets on disk or tape.

CALLING SEQUENCE

CALL SDINIT to initialize routine

CALL SDTAPE to obtain data

SDINIT must be called before any call to SDTAPE in a run, and must be called again whenever the user wishes to change the list of desired variables, keys, or other user-supplied parameters.

COMMON BLOCKS

```
COMMON/SDVAR/IDVAR (100), VARVAL(100)
COMMON/SDKEYS/IDKEY(5), KEYVAL(2, 5)
COMMON/SDINFO/NDATRC, NDESRC, KTAPUN, KPRTUN, JTYPE,
      NWDS, ISPARE(12)
COMMON/SDNEWV/LVNEW(100)
COMMON/SDFLAG/LEOF, LERR, LOVFL, LBADIN, LPRTEX,
      LRD1, LBADDR, LSPARE(13)
COMMON/SDDATA/DATA(1024)
```

INPUT/OUTPUT

User Input

a. Variable identifiers of all desired variables should be put in successive words of array IDVAR. At least the first unused word (if any) must be set = -1.

b. IDKEY should be filled with the variable numbers (subscript of IDVAR) of variables to be checked to determine acceptability of a record. KEYVAL should contain the lowest and highest values within which that variable must lie to accept a record containing it. To be accepted, a record must satisfy all the applicable keys. Unused positions in IDKEY should be padded with -1's (first unused one is enough). Only variables appearing on the tape in integer format should be used. To use JTYPE as a key, IDKEY should contain a 0.

c. FORTRAN data set reference number for the input tape (KTAPUN) and the printout data set (KPRTUN) should be supplied.

d. If the user desires explanatory tape records to be printed out, he should set LPRTEX to .TRUE. Otherwise, it should be set to .FALSE. Explanatory tape records are numbered in the record type by IRT+1000, where IRT is the record type of the record being explained. Each element in an explanatory record contains an 80-byte verbal explanation of the variable occupying its position in the explained record.

Subroutine Output

a. For each call to SDTAPE, the tape will be read until a data record is encountered which contains at least one desired variable, and which satisfies all the keys which appear in it. All desired variables in the record will have their values stored in VARVAL, in the same relative position as their identifiers in IDVAR.

b. Upon return from SDTAPE, NDATRC will contain the number of data records read in the last call, NDESRC will contain the number of descriptor records read, and JTYPE will contain the type of the last record read.

c. Upon return from SDTAPE, LVNEW will contain a .TRUE. in every position corresponding to variables in VARVAL receiving new values, and a .FALSE. elsewhere.

d. Upon return from SDTAPE, LEOF will be .TRUE. if an end-of-file was reached; LERR will be .TRUE. if an error condition occurred in reading the tape, LOVFL will be .TRUE. if more record types contain desired variables than the table can handle (25).

e. Upon return from SDINIT, LBADIN will be set to .TRUE. if some erroneous condition was found in the user's input. (e.g., no variables requested).

f. The subroutine reads into array DATA, so upon return, it contains the last tape record, and NWDS contains the number of words in it.

g. If the user sets LRD1 to .TRUE., SDTAPE will return after processing one record, whether or not it contains any relevant variable.

h. LBADDR will be set .TRUE., by SDTAPE if:

1. a second descriptor for a record type is read
2. a data record shorter than its descriptor is read
3. an undescribed data record is read

KEY VARIABLES

Common Variables

Name	Type	Description
DATA(1024)	Real	Array containing all data from record read
IDKEY(5)	integer	Filled with the variable numbers (subscripts of IDVAR) of variables to be checked to determine acceptability of a record.
IDVAR(100)	integer	User input of code names of parameters desired
ISPARE(12)	integer	Spare array
JTYPE	integer	Type of data record read
KEYVAL(2, 5)	integer	Contains the lowest highest values for variables in IDVAR to accept a record containing it
KPRTUN	integer	Unit number of print output definition
KTAPUN	integer	Unit number of data definition
LBADIN	logical	Flag for bad user input
LBADDR	logical	Flag for bad descriptor record
LEOF	logical	Flag when end of file is reached
LERR	logical	Flag when there exist an error in data read
LOVFL	logical	Flag for an overflow of descriptor records (25 allowed)
LPRTEX	logical	Flag for printing explanatory records
LRD1	logical	Flag used to return after each record read
LSPARE(13)	logical	Spare array
LVNEW(100)	logical	Flag set for new values put into array VARVAL
NDATRC	integer	The number of data records read since last call
NDESRC	integer	The number of descriptor records read since last call
NWDS	integer	The number of words in the last record read
VARVAL(100)	integer	Parameter data corresponding to code names in IDVAR

Internal Variables of Importance

VPTR(2,100,25) Array linking IDVAR subscript to record subscript for each record type. Thus for record type j, with RTPTR (j) = k, VPTR(1, i, k) contains the subscript of the record where the desired variable is stored.

VPTR (2, i, k) contains the subscript of VARVAL to which that value (i. e., data (VPTR(1, i, k))) should be moved.

The subscript i is used to store the cross-reference pairs for a record. Thus if this record contains n desired variables the program will loop from i=1, n moving data from DATA (VPTR(1, i, k)) to VARVAL (VPTR(2, i, k)).

KPTR(2,5,25) Same as VPTR, but links to KEYVAL pairs instead of VARVAL.

RTPTR(500) For each record type code i, RTPTR (i) will contain:

-1 if no descriptor for that type has been encountered

0 if a descriptor record has been encountered, and the record has no desired variables

k> 0 points to that value of the 3rd subscript of VPTR, KPTR, RTLEN, NVP, and NKP where references to that record type (i) are found

Internal Variables of Importance

MCRT =(25)	The largest allowed number of record types with desired data which the program can handle. Equals the dimension of NVP, NKP, RTLEN, and the third dimension of VPTR and KPTR
RTLEN(25)	RTLEN (j) contains the length (not counting length and rec. type words) of records of type i, where RTPTR (i) = j
NKP(25)	Contains the no. of IDKEY variables contained in the record
NVP(25)	Contains the no. of IDVAR variables contained in the record
NCRT	Contains the current no. of active record types (i. e., the last j used where RTPTR (i) = j for the last record type i processed with desired variables).

EXPLANATORY RECORD

Note that explanatory records still must be less than 1024 words long. Therefore, any record type of length >51 words cannot presently be "explained".

One variation on the explanation record is to have only the first 4 bytes of explanation for each variable in record type IRT+1000 then have additional records with types IRT+2000, IRT+3000, etc. each containing an additional 4 bytes of explanation. What the maximum number of such records should be, and whether or not a number less than the max. is considered an error, should be decided. I suggest a max. of 32 with less being permitted and encouraged.

METHOD

There are two parts to the routine, the initialization,(SDINIT entry) and the read (SDTAPE entry).

When entry is via SDINIT the user initialization information (IDVAR, IDKEY, KEYVAL) is checked for consistency and, if bad, LBADIN is set .TRUE. , a message is printed, and return is made. If consistent we just return.

When entry is via SDTAPE, a record is read from KTAPUN. The record type is checked. If it is an explanatory record, (type >1000) the content is printed if LPRTEX is .TRUE., and in either case we read the next record from KTAPUN.

If the record type is negative (descriptor record), array RTPTR is checked with the subscript of the absolute value of the record type. If that value is 0 or positive, an error exit is taken, otherwise (that is, if this record type has not been encountered before) RTPTR (type) is set to NCRT + 1, NCRT is incremented, and the record is scanned for matches to one of the IDVAR variable names. When a match is found, NVP(j) (where j = RTPTR (type) = NCRT) is incremented, and VPTR (1, i, j), where i = NVP(j) is set to k, where k is the current subscript of DATA being examined, and VPTR (2, i, j) is set to m, where m is the subscript of IDVAR at which the match was found. If no matches are found, NCRT is decremented and RTPTR (type) is set to 0. The scan goes from 1 to NRP(j) (= length of the record) and is repeated for IDKEY filling KPTR.

If the record type is positive (data record) an error exit is taken if RTPTR (type) is -1, and the next record is read if RTPTR (type) = 0.

If RTPTR (type) is positive, the program loops from i = 1 to NKP (j) where j = RTPTR (type) checking DATA (KPTR (1, i, j)) against IDVAL (k, KPTR(2, i, j)), k=1, 2, and if all checks are successful, it loops from 1 to NVP(j) moving DATA (VPTR(1, i, j)) to VARVAL (VPTR(2, i, j)).

If LRD1 is .TRUE. the routine returns after one record is processed regardless of type.

```
      SUBROUTINE SDTAPE
C
C     *          ROUTINE TO READ SELF-DESCRIBING TAPES COMPOSED OF VARIABLE LENGTH RECORDS PRE
C     *          RECORDS PROVIDING KEYS TO LINK EACH POSITION IN RECORDS OF A SPECIFIC 'TYPE' T
C     *          VARIABLE IDENTIFIERS.  USER PROVIDES LIST OF VARIABLES DESIRED, AND IS RETURNE
C     *          ALL SUCH VARIABLES IN EACH RECORD.
C
      IMPLICIT INTEGER(A-K,M-Z), LOGICAL (L)
C
      COMMON /SDVAR/ VARID(100), VAR(100)
C
      COMMON /SDKEYS/ KNU(5), KVAL(2,5)
C
      COMMON /SDINFO/ NDATR, NDESR, TAPE, PRT, JRT, NW, ISPAR1(12)
C
      COMMON /SDNEWV/ LVN(100)
C
      COMMON /SDFLAG/ LEOF, LERR, LOVFL, LBADIN, LPRTX, LRD1, LBADDR, LSPAR(13)
C
      COMMON /SDDATA/ REC(1024)
C
C
      INTEGER ERR(5,5)/' NO ','VARI','ABLE','S AS','KED.',' KEY',' > L','IST ','LENG','TH. ',
     1          ' KVA','L2 <',' KVA','L1. ','    ', 'DUPL','ICAT','E KE','YS. ',6*'    '/
C
      INTEGER MCRT/25/, VPTR(2,100,25), KPTR(2,5,25), RTPTR(500), NVP(25), NKP(25)
      INTEGER RTLEN(25)
C
C
C     *          ENTRY SDTAPE - COMES AFTER INITIALIZATION - CODE IS THUS LATER ON
      GO TO 50
C
      ENTRY SDINIT
      LBADIN=.FALSE.
      LCKRT=.FALSE.
C
C     *          CLEAR THE DECKS FOR ACTION
```

```
      NDATR=0
      NDESR=0
      DO 10 I=1,500
      RTPTR(I)=0
   10 CONTINUE
      NCRT=3
C
C     *          CHECK CALLER PARAMETERS
      DO 15 I=1,100
      IF(VARID(I).EQ.-1)GO TO 16
   15 CONTINUE
      NVAR=100
      GO TO 18
   16 IF(I.EQ.1)GO TO 40
      NVAR=I-1
   18 DO 20 I=1,5
      IF(KND(I).EQ.-1)GO TO 21
      IF(KND(I).GT.NVAR)GO TO 41
      IF(KVAL(2,I).LT.KVAL(1,I))GO TO 42
      IF(KND(I).NE.0)GO TO 20
      IF(LCKRT)GO TO 43
      LCKRT=.TRUE.
      JRTCK=I
   20 CONTINUE
      NK=5
      GO TO 900
   21 NK=I-1
      GO TO 900
C
C     *          BAD INPUTS
   40 IERR=1
      GO TO 45
   41 IERR=2
      GO TO 45
   42 IERR=3
      GO TO 45
   43 IERR=4
      GO TO 45
```


3600

```
700   C
800       4  WRITE(PRT,1000)IERR,(ERR(J,IERR),J=1,5
900    1000 FORMAT(//' INPUT ERROR',I2,',',5A4//)
1000          LBADIN=.TRUE.
1100          GO TO 900.
1200  C
1300  C     *           SDTAPE ITSELF - FIRST READ THE RECORD, OBVIOUSLY.
1400       5C LEOF=.FALSE.
1416          NDESR=0
1437          NDATR=0
1500          LERR=.FALSE.
1600          LOVFL=.FALSE.
1700          LBADDR=.FALSE.
1800          DO 51 I=1,100
1900          LVN(I)=.FALSE.
2000       51 CONTINUE
2100  C
2200       52 READ(TAPE,END=810,ERR=820)NW,JRT,(REC(I),I=1,NW)
2300  C
2400  C     *           CHECK RECORD TYPE
2500          IF(JRT)200,830,54
2600       54 IF(JRT.GE.1000)GO TO 700
2700  C
2800  C     *           DATA RECORD
2900  C
3000  C     *           CHECK IF RELEVANT
3100          NDATR=NDATR+1
3200          IF(RTPTR(JRT))500,840,55
3300  C
3400  C     *           RECORD TYPE WITH INFO WE WANT
3500       55 IRT=RTPTR(JRT)
3600  C
3700  C     *           CHECK KEYS, IF ANY
3712          IF(.NOT.LCKRT)GO TO 60
3715  C
3724  C     *           CHECK RECORD TYPE AGAINST KEYS
3736          IF(JRT.LT.KVAL(1,JRTCK).OR.JRT.GT.KVAL(2,JRTCK))GO TO 500
3800       60 IF(NKP(IRT).LE.0)GO TO 70
```

```
25          M1=NKP(IRT)
60 100000   DO 56 I=1,M1
16          M2=KPTR(1,I,IRT)
32          M3=KPTR(2,I,IRT)
CC          IF(REC(M2).LT.KVAL(1,M3).OR.REC(M2).GT.KVAL(2,M3))GO TO 500
00    C
00    C     *           PASSES KEY TEST
90       56 CONTINUE
:C    C
60    C     *           FILL VAR ARRAY
CC       70 IF(RTLEN(IRT).GT.NW)GO TO 860
25  10625   M4=NVP(IRT)
50          DO 75 I=1,M4
25          M5=VPTR(1,I,IRT)
43          M6=VPTR(2,I,IRT)
50          VAR(M6)=REC(M5)
50 11400    LVN(M6)=.TRUE.
30       75 CONTINUE
50    C
00    C     *           ALL DONE FOR A GOOD DATA RECORD
00          GO TO 900
:0    C
00    C     *           DESCRIPTOR RECORD
90       200 KRT=-JRT
50          NDESR=NDESR+1
00    C
20    C     *           NEW RECORD TYPE - IS THERE ROOM?
0012,000    IF(NCRT.GE.MCRT)GO TO 850
25          IF(RTPTR(KRT).NE.0)GO TO 870
30 12100    NCRT=NCRT+1
00          RTPTR(KRT)=NCRT
50          RTLEN(NCRT)=NW
CC    C
CC 12500    NVP(NCRT)=0
50          NKP(NCRT)=0
:C    C
CC    C     *           FILL VPTR, THEN KPTR
50          DO 220 I=1,NW
```

```
3000          DO 205 J=1,NVAR
3100          IF(REC(I).EQ.VARID(J))GO TO 210
3200      205 CONTINUE
3300          GO TO 220
3400      210 NVP(NCRT)=NVP(NCRT)+1
3425          M7=NVP(NCRT)  NVP
3500          VPTR(1,M7,NCRT)=I
3600          VPTR(2,M7,NCRT)=J
3625          IF(NK.LE.0)GO TO 220
3700          DO 215 K=1,NK
3800          IF(KNO(K).EQ.J)GO TO 216
3900      215 CONTINUE
4000          GO TO 220
4100      C
4200      C     *           FOUND A KEY
4300      216 NKP(NCRT)=NKP(NCRT)+1
4325          M8=NKP(NCRT)
4400          KPTR(1,M8,NCRT)=I
4500          KPTR(2,M8,NCRT)=K
4600      C
4700      220 CONTINUE
4800          IF(NVP(NCRT).GT.0)GO TO 500
4816          NCRT=NCRT-1
4832          RTPTR(KRT)=-1
4900          GO TO 500
5000      C
5100      C     *           DONE WITH THIS RECORD - READ NEXT UNLESS LRD1 IS SET
5200      500 IF(LRD1)GO TO 900
5300          GO TO 52
5400      C
5500      C     *           EXPLANATORY RECORD - PRINT IF LPRTEX SET
5600      700 IF(.NOT.LPRTX)GO TO 500
5700          J=JRT-1000
5800          N=NW/20
5900          WRITE(PRT,1003)J,N
6000     1003 FORMAT(//' EXPLANATORY RECORD FOR RECORD TYPE',I4,' - ',I5,' ITEMS'/)
6100          DO 710 I=1,N
6125          M9=(I-1)*20+1
```

check to see if there are any key conditions

```
      M10=I*20
      WRITE(PRT,1001)(REC(J),J=M9,M10)
 1001 FORMAT(1X,20A4)
  710 CONTINUE
      GO TO 500
C
C     *           ERROR CONDITIONS
C
C     *           EOF
  810 LEOF=.TRUE.
      GO TO 900
C
C     *           ERROR ON READ
  820 LERR=.TRUE.
      GO TO 900
C
C     *           RECTYP OF ZERO
  830 LBADDR=.TRUE.
      GO TO 900
C
C     *           DATA RECORD NOT PREVIOUSLY DESCRIBED
  840 GO TO 830
C
C     *           RECORD TYPE TABLE OVERFLOW
  850 LOVFL=.TRUE.
      GO TO 900
C
C     *           RECORD SHORTER THAN DESCRIPTION
  860 GO TO 830
C
C     *           DESCRIPTOR RECORD FOR TYPE ALREADY DESCRIBED
  870 GO TO 830
C
C
  900 RETURN
      END
```

Cross Reference List

SDTAPE1 8/31/72 12:06:27

SYMBOL CROSS-REFERENCE LIST

SYMBOL	DEFINITIONS	REFERENCES
CHCRD1		0
CHCIA1		6800 8200 15900 16200
CHCIE1		6800 8200 15900 16200
CHCIU1		6800 8200 15900 16200
ERR		2400 6800
I	3700 4300 5000 7800 8200 9900 10700 12900 16100	3800 4400 4800 4900 5100 5336 5700 7900 8200 9916 13500 14400 16125 16143
IERR	6100 6300 6500 6612	6800 6800
IRT	9500	9800 9825 9916 9932 10600
ISPAR1		1500
J	6800 13000 15700 16200	6800 13100 13600 13800 15
JRT	8200	1500 8500 8600 9200 9500
JRTCK	5336	9736 9736
K	3700	13800 14500
KNO		1300 5100 5200 5312 1380
KPTR	14400 14500	2700 9916 9932
KRT	11600	12025 12200 14832
KV1		1300 5300 5300 9736 9736

REFERENCES

0

6800 8200 15900 16200

6800 8200 15900 16200

6800 8200 15900 16200

2400 6800

)00 8200 9900 10700 12900 16100 3800 4400 4800 4900 5100 5200 5300 5300 5312
5336 5700 7900 8200 9916 9932 10725 10743 13100
13500 14400 16125 16143

512 6800 6800

9800 9825 9916 9932 10600 10625 10725 10743

1500

16200 6800 13100 13600 13800 15900 16200

1500 8500 8600 9200 9500 9736 9736 11600 15700

9736 9736

13800 14500

1300 5100 5200 5312 13800

2700 9916 9932

12025 12200 14832

1300 5300 5300 9736 9736 10000 10000

LBADDR	7700	17800	1900
LBADIN	3400	7000	1900
LCKPT	3425	5324	5315 9712
LEOF	7400	17000	1900
LERR	7500	17400	1900
LOVFL	7600	18500	1900
LPRTX			1900 15600
LRD1			1900 15200
LSPAR			1900
LVN	7900	10900	1700
MCRT			2700 12000
M1	9825		9900
M10	16143		16200
M2	9916		10000 10000
M3	9932		10000 10000
M4	10625		10700
M5	10725		10600
M6	10743		10800 10900
M7	13425		13500 13600

SDTAPE$$.00/310/72 12:06:27

MB	14325	14400 14500
MS	16125	16200
N	15800	15900 16100
NCRT	4000 12100 14816	12000 12100 12200 123 13425 13500 13600 143 14800 14816
NDATP	3616 7432 9100	1500 9100
NDESP	3632 7416 11700	1500 11700
NK	5500 5700	13625 13700
NKP	12600 14300	2700 9800 9825 14300
NVAR	4600 4900	5200 13000
NVP	12500 13400	2700 10625 13400 13425
NW	8200	1500 8200 10600 12300
PRT		1500 6800 15900 16200
REC	8200	2100 10000 10000 10800
RTLEN	12300	2750 10600
RTPTR	3800 12200 14832	2700 9200 9500 12025
SDDATA	2100	
SDFLAG	1900	
SDINFO	1500	
SDINIT	3300	

T

14400 14500

16200

15900 16100

16 12000 12100 12200 12300 12500 12600 13400 13400
13425 13500 13600 14300 14300 14325 14400 14500
14800 14816

1500 9100

0 1500 11700

13625 13700

2700 9800 9825 14300 14325

5200 13000

2700 10625 13400 13425 14800

1500 8200 10600 12300 12900 15800

1500 6800 15900 16200

2100 10000 10000 10800 13100 16200

2750 10600

32 2700 9200 9500 12025

SDTAPE$$,08/31/72 12:06:27

SDKEYS	1300	
SDNEWV	1700	
SDTAPE	100	
SDVAR	1100	
TAPE		1500 8200
VAR	10800	1100
VARID		1100 4400 13100
VPTR	13500 13600	2700 10725 10743

SDTAPE$$.08/31/72 12:06:27

LABEL CROSS-REFERENCE LIST

LABEL	DEF	REFERENCES
00010	3900	3700
00015	4500	4300
00016	4800	4400
00018	5000	4700
00020	5400	5000 5312
00021	5700	5100
00040	6100	4800
00041	6300	5200
00042	6500	5300
00043	6612	5315
00045	6800	6200 6400 6600 6624
00050	7400	3100
00051	8000	7800
00052	8200	15300
00054	8600	8500
00055	9500	9200
00056	10300	9900
00060	9800	9712
00070	10600	9800
00075	11000	10700
00200	11600	8500
00205	13200	13000
00210	13400	13100
00215	13900	13700
00216	14300	13900

00220	14700	12900	13300	13625	14000				
00500	15200	9200	9736	10000	14800	14900	15600	16500	
00700	15600	8600							
00710	16400	16100							
00810	17000	8200							
00820	17400	8200							
00830	17800	8500	18200	18900	18936				
00840	18200	9200							
00850	18500	12000							
00860	18900	10600							
00870	18936	12025							
00900	19200	5600	5800	7100	11300	15200	17100	17500	17900	18600
01000	6900	6800							
01001	16300	16200							
01002	16000	15900							

Appendix A

EXAMPLE PROGRAM USING SDTAPE

Example program using routine SDTAPE to read parameter data tapes:

```
C***        THIS PROGRAM WILL PRINT PROTON PARAMETERS FOR
C           CYCLES WITH CHI-SQUARES LESS THAN 100.
C
            DIMENSION INPT(11), XVARVL(100)
C
C***        THE FOLLOWING COMMONS USED WITH SDTAPE
            COMMON/SDVAR/IDVAR(100), VARVAL(100)
            COMMON/SDINFO/NDATRC, NDESR, KTAPUN, KPRTUN,
                  JTYPE, NWDS, ISPARE(12)
            COMMON/SDFLAG/LEOF, LERR, LOVFL, LBADIN, LPRTEX,
                  LRD1, LBADDR, LSPARE(13)
            COMMON/SDDATA/DATA(1024)
            COMMON/SDKEYS/IDKEY(5), KEYVAL(2, 5)
C
            LOGICAL *4 LEOF, LERR, LOVFL, LBADIN, LPRTEX, LRD1
            LOGICAL *4 LBADDR, LSPARE
            INTEGER VARVAL
            EQUIVALENCE (VARVAL, XVARVL)
C
C***        CODE NAMES TO BE USED BY SDTAPE
            DATA INPT/'TI01','TI02','TI06', TI91','TI92','CS01',
                  'PT01','PV07','PA08','PA09','PN10' /
C
C***        PUT DATA INPT INTO ARRAY IDVAR FOR SDTAPE
            DO 5 I = 1, 11
            IDVAR (I) = INPT (I)
      5     CONTINUE
C***        NEXT ARRAY MEMBER FOLLOWING INPT CODE NAMES
C           IS SET TO -1
            IDVAR (12) = -1
C
C***        DEFAULTS TO BE SET TO RETURN AFTER EACH CYCLE
            READ FROM SDTAPE
            IDKEY (1) = -1
            LRD1 = .TRUE.
```

```
C***        OUTPUT PRINT UNIT
            KPRTUN = 6
C***        INPUT TAPE UNIT
            KTAPUN =12
C***        PRINT HEADER LINE
            WRITE (KPRTUN, 500)
500         FORMAT
            (1H1, 2X, 'YEAR DAY TM-IN-MS GEN DATE      CHI',
             6X, 'PT01', 6X, 'PV07', 6X, 'PA08', 6X, 'PA09', 6X, 'PN10')
C***        ROUTINE SDTAPE NEEDS TO BE INITIALIZED FIRST TIME THRU
            CALL SDINIT
100         CALL SDTAPE
C***        CHECK FOR END OF FILE
            IF (LEOF) GO TO 200
C***        CHECK FOR READ ERROR
            IF(LERR) GO TO 210
C***        CHECK FOR CHI-SQUARE TO BE LESS THAN 100.
            IF (XVARVL(6).GT.100.) GO TO 100
C
C***        PRINT CYCLE INFO
            WRITE (KPRTUN, 505) (VARVAL(I), I=1, 11)
505         FORMAT (1H, 2X, 2I4, I9, I4, '/', I3, 1X, 6(6X, F10.5))
C
            GO TO 100
C
200         WRITE (6, 201)
201         FORMAT (1H0, 2X, 'END OF FILE REACHED')
            GO TO 999
210         WRITE (6, 211)
211         FORMAT (1H0, 2X, 'ERROR ON TAPE')
C
999         STOP
            END
```

A parameter tape is a one file 9-track tape with DCB as follows:

```
DEN = 2 (800 BPI)
RECFM = VS (Variable)
LRECL = 4112
BLKSIZE = 4116
```

NOTE: In program, code names can be put in any consecutive order in array IDVAR and their values are put in the corresponding array member in VARVAL. e.g.,

```
DIMENSION INPT(6)
DATA INPT/'TI01','TI02','PN10','CS01','TI06','PT01' /
```

PFRSUM

Rdc. Summary Rev

dd SB#PR.LIB.CNTL(BIT2RTSM)

6.2 PROGRAM INPUT

All input to the program is read using the NAMELIST convention of FORTRAN IV.

There are two different NAMELISTs in the Rates Summary Program, NAMELIST INIT and NAMELIST PERIOD. When initializing a new rates summary data set, both NAMELISTs must be present and NAMELIST PERIOD input must precede NAMELIST INIT. When updating a rate summary, neither NAMELIST need necessarily be present,[1] and the NAMELIST INIT should not be present.

6.2.1 NAMELIST PERIOD Input[2]

The first card of the NAMELIST PERIOD input stream must contain the character string &PERIOD in columns 2 through 8, followed by a blank. The blank is followed by data items separated by commas. The end of the NAMELIST input stream is indicated by the character &END.

The following items may be specified in NAMELIST PERIOD:

- DRTAPE--The labels of rates tapes containing the data to be summarized. Each tape label must be enclosed in single quotes (') and separated from the previous one by a comma. Up to 10 labels may be listed. It is recommended that this option be used only when necessary, i.e., only when the data cannot be obtained through the DRS catalog.

[1] It is recommended that both be omitted, if possible.

[2] Column 1 of all cards must be blank.

- NINT--An integer specifying the number of consecutive summary intervals to be processed.

  Default: 10000 (Note that processing will end when rates data is exhausted.)

- QSPEC--Logical variable

  = T, status information is to be read from logical unit 32
  = F, status information is to be read from logical unit 8

  Default: F

- NTPMAX--An integer specifying the maximum number of rates tapes from the DRS catalog that may be mounted for input. Program will terminate after processing NTPMAX tapes. This input is ignored if rates tapes were specified by the DRTAPE parameter above.

  Default: 20

- NCATP--If the DRS catalog pointer data set is not to be read, then the user should input here an integer between -1 and -4, whose absolute value will become the DRS catalog pointer.

  Default: 0 and the DRS pointer is read from the pointer data set.

- HTB--Up to six two-digit integers, separated by commas, specifying the year, month, day, hour, minute, and second, respectively, for the start of the first summary interval of the summary. This parameter is meaningless when not creating a new summary.

  Default: The summary is started with the earliest available data.

- SOURCE--'SOLDPS', 'SDR', or 'BUFOP'. Input must be one of these three parameters, or should be omitted entirely. Note that both the DRS catalogs (if specified) and the rates summary to be

updated must contain only data from the source specified. Any mismatch results in a error and program termination.

= 'BUFOP', origin of data is BUFOP tapes

= 'SDR', origin of data is SDR tapes

= 'SOLDPS', origin of data is SOLDPS tapes

If data is from standard EDR tapes, the source parameter should be omitted.

6.2.2 NAMELIST INIT Input[1]

NAMELIST INIT is used only when the user wishes to create a new rates summary.

Card 1 must contain &INIT in columns 2 through 6 followed by a blank and then the following data items:

- DTAPES--The label of a blank tape to contain the rates summary. This label must be input, and should be enclosed in single quotes (').
- QTRCHK--Logical variable

  = T, readouts that failed trend check will not be included in the summary

  = F, readouts that failed trend check will be included in summary

  Default: T
- INTHR--Integer; number of whole hours to be contained in one summary interval.

  Default: 1

[1] Column 1 of all cards must be blank.

- INTMIN--Integer; number of additional whole minutes to be contained in summary.

  Default: 0

- INTSEC--Integer; number of additional seconds to be contained in summary.

  Default: 0

- HID--Logical variable

  = 'F', rates summary is to contain data only for Pioneer-F
  = 'G', rates summary is to contain data only for Pioneer-G

  Default: F

6.3 ABNORMAL PROGRAM MESSAGES

Following is a list of program messages with appropriate user response.

1. *** INVALID CATALOG POINTER. GIVEN VALUE WAS XX
   RUN TERMINATED.

   Cause: NCATP was specified incorrectly on the PERIOD card with a value of less than -4.

   User Response: If NCATP is to be specified, it must be an integer in the range -1 to -4, whose absolute value will become the catalog pointer.

2. NO DATA ON THE GIVEN TAPE XXXXXXXX SINCE THIS IS THE FIRST OF THE GIVEN RATES TAPES. THIS RUN IS TERMINATED.

   Cause: The first rates tape mounted for input contained no input.

   User Response: If the user specified a rates tape, the wrong tape was specified.

3. SPACECRAFT ID'S DO NOT MATCH X X X.

Cause: The spacecraft ID on the DRS pointer data set does not match the spacecraft ID on the DRS catalog, or does not match the spacecraft ID on the summary status record. The IDs are printed in the following order: summary status ID, pointer ID, DRS catalog ID.

User Response: The wrong summary data set was specified on the FT08F001 or FT32F001 DD card, or the wrong combination of FT20F001 - FT24F001 DD cards is being used, or the user has specified NCATP for a G summary, which cannot be done. If a G catalog not pointed to by the pointer must be used, renumber the FT21F001 - FT24F001 DD cards so that the desired DRS catalog is pointed to.

4. END OF FILE ON THE CATALOG UNIT. ERROR CONDITION.

Cause: An end of file was hit reading the DRS catalog.

User Response: Correct the DRS pointer data set if not valid. If inputing NCATP, check that the data set pointed to has been written on.

5. END OF FILE ENCOUNTERED WHILE READING THE STATUS DATA SET.

Causes: DATA SET NOT INITIALIZED, OR A BAD DD STATEMENT.

User Response: Either correct the status data set DD statement (FT08 or FT32) or perform an initialization run for this data set.

6. NO VOLUME-SERIAL NUMBER FOR THE SUMMARY TAPE WAS GIVEN. RUN TERMINATED.

Cause: In creating a new data set, the user failed to specify the label of the tape to contain the rates summary.

User Response: Input tape label using the DTAPES parameter in NAMELIST INIT.

7. I/O ERROR DETECTED WHILE READING THE

{STATUS DATA SET
DRS CATALOG
RECORD XXXXX OF RATES TAPE XXXXX}

RUN TERMINATED. DESCRIPTION OF ERROR FOLLOWS:
(Message from FTIO is printed.)

Cause: As described.

User Response: Consult with the FTIO Manual for interpretation of the printed error description.

8. ERROR - INPUT ON PERIOD CARD INDICATED THAT DATA SOURCE WAS XXXXXX, HOWEVER, THE GO.FTXXF001 DD CARD CONTAINED THE DSNAME OF THE XXXXXX DRS CATALOG. RUN TERMINATED.

Cause: The user-specified source does not match the source indicated by at least one of the DRS catalog DD cards.

User Response: Input the source parameter on NAMELIST PERIOD card.

9. ERROR - SOURCE SPECIFIED WAS XXXXXXXX, THIS MAKES NO SENSE. RUN HALTING.

Cause: The user-specified source does not match the source indicated by at least one of the DRS catalog DD cards.

User Response: Input the source parameter on NAMELIST PERIOD card.

10. ERROR - THE PERIOD CARD INDICATED A DATA SOURCE OF XXXXXX, BUT THE STATUS RECORD WAS FOR A SOURCE OF XXXXXX. EXECUTION TERMINATING.

Cause: While the specified source was compatible with the DRS catalog DSNAME, the rates summary status record indicates that the data on the summary tape is from some other source.

User Response: Either the status record DD statement (FT08 or FT32) must be changed, or the DRS catalog DD cards and the source parameter or NAMELIST PERIOD card must be changed.

6.4 DATA SETS REFERENCED

The program references the following data sets (by DDNAMES).

Data Set	Description
FT08F001	Defines rates summary status data set if QSPEC = F or QSPEC is not specified on a period card. If cataloged, only DSNAME and DISP parameters must be specified; otherwise, UNIT and VOL parameters must also be specified
FT09F001	Defines rates tapes data set. A seven-track tape drive (UNIT = 2400-7) with deferred mounting and a dummy volume serial number should be specified. The user must also specify DSN = PIORAT, DISP = SHR
FT10F001	Defines rates summary data set. A dummy volume serial number should be specified. Following parameters should be specified exactly as given: UNIT = (2400-9,,DEFER) DISP = (MOD, KEEP, KEEP) DSN = PIOFRSUM (for Pioneer-F) DSN = PIOGRSUM (for Pioneer-G)

Comrates

The purpose of the Pioneer 7/9 Comrate list program is to provide a formatted listing of selected rates and times from rates tapes. Input to this program are Rates tapes and jcl & data cards

B. JCL.

1. The program requires 200 bytes of main storage and approximately .5 min of CPU time + .5 min of I/O to process approximately 100 records. A description of all the DD cards required by the program, the purpose of each and when it is required follows.

DD NAME	PURPOSE OF DATA SET	INPUT/OUTPUT	DEVICE TYPE	CODE
FT06F001 FT08F001	FORMATTED LISTING	OUTPUT	PRINTER	A
↓	Copy from source listing			
28				
PIOCRATE	TAPE UNIT	INPUT	TAPE [redacted]	A
PCATALOG	CATALOG POINTER	↓	DISK	↓
PENCTLG1	CATALOG 1		↓	
2	2			
3	3			
4	4			
SYSDUMP	ABEND DUMP	OUTPUT	PRINTER	A
DATA5	DATA CARDS	INPUT	CARDS	A

The meaning of code is as follows:

A = Always.

O = optional (If rates ~~[redacted]~~ selected are not intended in this unit it may be dummied out.)

```
//* CCMRATES
//* TWISCATE
// EXEC LINKGO,REGION,GO=200K
//LINK.SYSLIB DD DSN=K3.SBCID.SB001.OPICNEER,DISP=SHR
// DD DSN=K3.SBCID.SBCC1.OPICTEMP,DISP=SHR
//LINK.SYSLIN DD *
 INCLUDE SYSLIB(CCMRAT)
//GO.FT06F001 DD UNIT=(C250,,DEFER),LABEL=(1,SL),DSN=OUTPUT,
// DCB=(RECFM=VBA,LRECL=137,BLKSIZE=3429,DEN=3),DISP=(NEW,KEEP),
// VOL=SER=PGD001
//GO.FT08F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT09F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT10F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT11F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT12F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT13F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT14F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT15F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT16F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT17F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT18F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT19F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT20F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT21F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT22F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT23F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT24F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT25F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT26F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT27F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT28F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.PICFRATE DD UNIT=(C250,,DEFER),DISP=SHR,DCB=(EROPT=ACC,DEN=3),
// LABEL=(,SL,,IN),DSN=PICRAT,VOL=SER=PICTAP
//GO.PCATALOG DD DSN=K3.SBCID.SB001.PFRECCRP,DISP=SHR
//GO.PENCTLG1 DD DSN=K3.SBCID.SBCC1.PFRECCR1,DISP=SHR
//GO.PENCTLG2 DD DSN=K3.SBCID.SB001.PFRECCR2,DISP=SHR
//GO.PENCTLG3 DD DSN=K3.SBCID.SB001.PFRECCR3,DISP=SHR
//GO.PENCTLG4 DD DSN=K3.SBCID.SB001.PFRECCR4,DISP=SHR
//GO.SYSUDUMP DD SYSOUT=A
//GO.DATAS DD *
```

There are 2 types of data cards for this program. They are the TIME card and the SELECT card.

C. Data Cards.

The data cards follow the last DD card in the program setup and are read using the Namelist convention of Fortran IV. The first column in each card must be blank, the next 6 columns of the first card must contain the characters '&TIME' followed by a blank. The blank is followed by data items separated by commas. The end of a group of cards is signaled by the characters '&END'. One or more groups of time cards, each identifing a unique time may be submitted each run.

C. Data Cards.

1. The data cards follow the last DD card in the deck setup and are read using the Namelist convention of Fortran IV. There are 2 types of data cards for the program: Time card (which defines the start time + stop time; Select card (which define the [illegible] rates to be plotted.

The [illegible] required parameters on the time card are as follows:

HMONS	START MONTH
HSDY	START DAY
HYRS	START YEAR
HOURS	START HOURS.
HMINS	START MINS.
HSECS	START SECONDS
HMONE	END MONTH
HEDY	END DAY
HYRE	END YEAR
HOURE	END HOUR
HMINE	END MIN.
HSECE	END SEC

The required parameters on the select card are as follows.

QTREND = T OR F
RATES = 'RATE OR RATES DESIRED'
(Rates must appear in quotes).

EXAMPLE OF DATA CARDS IS AS FOLLOWS:

```
&TIME HMONS=10,HSDY=01,HYRS=76,HOURS=00,HMINS=00,HSECS=00,
      HMONE=01,HEDY=01,HYRS=77,HOURE=01,HMINE=01,HSECE=01 &END
&SELECT QTREND=F,RATES='R1','R2A','R2B','R3A','R11A','R11B', &END
```

*** [illegible] REQUESTS PROCESSED.

D. Output.

The primary output from the Comrate program is a formatted listing of the data selected in Hexidecimal + decimal form. this program should be run for very short periods of selected times due to the large consumption of paper required ~~by this program~~.

E. When a job is terminated abnormaly with a user abend completion code refer to the User Abend Section of the "IBM System 360 General I/O Package".

Magsec.

A. Description

The purpose of the Pioneer F&G. Magnetic Field + Sectored Rates List program is to create a formated listing of selected Sectored rates merged with magnetic Field coordinates ~~for given times~~. Input to this program is ~~[illegible]~~ Rates tapes and a Magnetic Field tape. ~~This program was designed for the purposes [illegible]~~. The ~~purpose [illegible]~~

B. JCL

The ~~purpose~~ Magsec program is currently in executable load module form on 'SYS2.LOADLIB' named SEHGDMAG, The program requires .5 min CPU + .5 min I/O ~~for~~ to process approximately 10 days worth of data. ~~The DD cards required by the Magsec program~~

A ~~[illegible]~~ discription of the DD cards required by the program, the purpose of each and when it is required follows:

DD NAME	PURPOSE OF DATA SET	INPUT/OUTPUT	DEVICE	CODE
STEPLIB	USER PROGRAM	INPUT	DISK	A
FT05F001	DEFINE DATA	INPUT	DISK	A
FT06F001	FORMATTED LISTING	OUTPUT	PRINTER	A
FT10F001	MAGNETIC FIELD TAPE	INPUT	TAPE	A
PIOFRAT	RATES TAPE	INPUT	TAPE	A
PCATALOG	CATALOG POINTER	INPUT	DISK	A
PENCTLG1	DRS CATALOGS 1	↓	↓	A
2	2			A
3	3			A
4	4			A
SYSUDUMP	ABEND LISTING	OUTPUT	PRINTER	A
DATA5	PARAMETER CARDS	INPUT	READER	A

SOPIO.LIB.CNTL = PIONEER

TABLE HEADING DEFINITIONS

PRODUCTION GROUP- THE FIRST CHARACTER OF THE T50 ~~CRBE~~ LIBRARY MNEMONIC WHICH DEFINES THE SUB-SYSTEM TO WHICH IT IS ASSOCIATED.
THE PRODUCTION GROUPINGS ARE AS FOLLOWS:
(L)-LIBRARY GENERATOR SYSTEM PROCEDURES
(D)-DATA REDUCTION SYSTEM PROCEDURES
(F)-FLUX DATA BASE GENERATOR SYSTEM PROCEDURES
(A)-ANALYSIS PROGRAMS AND STANDARD ANALYSIS PARAMETER CARDS
(U)-UTILITY PROCEDURES IE. JOB CARDS,SYSTEM UTILITIES,ETC.
(S)-SPECIAL REQUEST IE NON-STANDARD PROCEDURES

PIONEER/HELIOS MNEMONIC- NAME OF THE ACTUAL LIBRARY MEMBER WHERE THE FIRST CHARACTER IS THE PRODUCTION GROUP INDICATOR AND THE LAST CHARACTER IS THE SATELITE IDENTIFIER(A=HELIOS-A;B=HELIOS-B;F=PIONEER-F; G=PIONEER-G) IF THE MEMBER IS SATELITE DEPENDANT

MEMBER TYPE- EITHER 'OTHER' OR 'FLIST'. 'OTHER' TYPE FILES EITHER REQUIRE EDITING BEFORE SUBMITTAL OR ADDITIONAL MEMBERS DURING SUBMITTAL , AND ALWAYS REQUIRE A JOB CARD AND NOTIFY. 'FLIST' MEMBERS CONTAIN BOTH JOB CARD AND NOTIFY , REQUIRE NO EDITING AND MAY BE SIMPLY SUBMITTED USING THE CORRECT TIME ESTIMATE.

FUNCTION OF MEMBER- GIVES A SHORT DESCRIPTION OF THE PURPOSE OF EACH MEMBER

TIME EST. 360/75- THE TIME NECESSARY TO EXECUTE THE PROGRAM ON THE 360/75.

F/V-WEATHER THE TIME ESTIMATE IS FIXED (CONSTANT) OR VARIABLE FROM RUN TO RUN. IF VARIABLE THE TIME EST. GIVEN WILL BE THE MAXIMUM.

ODUCTION GROUP	PIONEER MNEMONIC	HELIOS MNEMONIC	MEMBER TYPE	FUNCTION OF MEMBER	TIME EST. 360/75	F / V
L-LIBRARY GENERATOR SYSTEM (HELIOS ONLY)	NONE	LGINA	OTHER	LOG IN 'A' INPUT EDR TAPES	H00H00	F
		LGINB	OTHER	LOG IN 'B' INPUT EDR TAPES	H00H00	F
		LGOUTA	OTHER	LOG IN 'A' OUTPUT LIB TAPES	H00H00	F
		LGOUTB	OTHER	LOG IN 'B' OUTPUT LIB TAPES	H00H00	F
		LGEXECA	OTHER	JCL TO RUN 'A' LIBGEN	005030	V
		LGEXECB	OTHER	JCL TO RUN 'B' LIBGEN	005030	V
		LTSCPYA	OTHER	COPY 'A' LIBRARY TAPES	H02H06	V
		LTSCPYB	OTHER	COPY 'B' LIBRARY TAPES	H02H06	V
		LBCKINCT	FLIST	BACKUP 'A&B' INDEX&CATALOG	H00001	F
		LRLINCTA	FLIST	RELOAD 'A' INDEX&CATALOG	H00001	F
		LRLINCTB	FLIST	RELOAD 'B' INDEX&CATALOG	H00001	F
L-END						
D-DATA REDUCTION SYSTEM	DRSLSTPF	DRSLSTF	FLIST	LIST 'F' PHA DRS CATALOG	H00H00	F
	DRSLSTRF		FLIST	LIST 'F' RATE DRS CATALOG	H00H00	F
	DRSLSTG		FLIST	LIST 'G' DRS CATALOG	H00H00	F
	DRSADDPF	DRSADDF	OTHER	ADD TAPES 'F' PHA CATALOG	H00H00	F
	DRSADDRF		OTHER	ADD TAPES 'F' RATE CATALOG	H00H00	F
	DRSADDG		OTHER	ADD TAPES 'G' DRS CATALOG	H00H00	F
	DRSRESPF	DRSRESF	OTHER	RESTORE 'F' PHA DRS CATALOG	H00H00	F
	DRSRESRF		OTHER	RESTORE 'F' RATE DRS CATALOG	H00H00	F
	DRSRESG		OTHER	RESTORE 'G' DRS CATALOG	H00H00	F
		DRSLSTA	FLIST	LIST 'A' DRS CATALOG	H00H00	F
		DRSLSTB	FLIST	LIST 'B' DRS CATALOG	H00H00	F
		DRSADDA	OTHER	ADD TAPES 'A' DRS CATALOG	H00H00	F
		DRSADDB	OTHER	ADD TAPES 'B' DRS CATALOG	H00H00	F
		DRSRESA	OTHER	RESTORE 'A' DRS CATALOG	H00H00	F
		DRSRESB	OTHER	RESTORE 'B' DRS CATALOG	H00H00	F
	DRPPHAF	DRPF	OTHER	JCL TO DRP 'F' ~~PHA~~ DATA	010020	V
	DPHADATF		OTHER	INPUT CARDS FOR DRPPHAF		

```
                DRPRATF                 OTHER   JCL TO DRP 'F' RATE DATA     010020 V
                DRATDATF                OTHER   INPUT CARDS FOR DRPRATF
                DRPG                    OTHER   JCL TO DRP 'G' DATA          010020 V
                DDATAG                  OTHER   INPUT CARDS FOR DRPG
                        DRPPROC         OTHER   JCL PROC FOR 'A&B' DRP RUNS
                        DRPACCA         OTHER   EXEC CARD FOR 'A' ACCDM7N    015020 V
                        DRPREJA         OTHER   EXEC CARD FOR 'A' REJDM7N    020030 V
                        DRPACCB         OTHER   EXEC CARD FOR 'B' ACCDM7N    015020 V
                        DRPREJB         OTHER   EXEC CARD FOR 'B' REJDM7N    020030 V
                        DQLKA           OTHER   'A' QUICKLOOK PARAMETERS     005005 V
                        DQLKB           OTHER   'B' QUICKLOOK PARAMETERS     005005 V
                LOADVBS                 FLIST   LOAD PGM=SEHGDVBS LOAD MOD.  H00001 F
                DCPYPHA                 OTHER   BACKUP 'F&G' PHA TAPES       001002 F
                DCPYRAT                 OTHER   BACKUP 'F&G' RATE TAPES      001002 F
                        DCPYPHAA        OTHER   BACKUP 'A' PHA TAPE 1 TO 1   001002 F
                        DCPYRATA        OTHER   BACKUP 'A' RATE TAPE 1 TO 1  001002 F
                        DCPYPHAB        OTHER   BACKUP 'B' PHA TAPES 3 TO 1  H01005 V
                        DCPYRATB        OTHER   BACKUP 'B' RAT TAPES 3 TO 1  H01005 V
D-END
F-FLUX DATA
  BASE          FLUXDBGF                OTHER   JCL TO UPDATE 'F' FLUX D B   030030 V
  GENERATOR     FINSERTF                OTHER   OPTIONAL INSERT CARD FOR 'F'
  SYSTEM        FLUXDBGG                OTHER   JCL TO UPDATE 'G' FLUX D B   030030 V
                FINSERTG                OTHER   OPTIONAL INSERT CARD FOR 'G'
                        FLUXDBGA        OTHER   JCL TO UPDATE 'A' FLUX D B   030030 V
                        FINSERTA        OTHER   OPTIONAL INSERT CARD FOR 'A'
                        FLUXDBGB        OTHER   JCL TO UPDATE 'B' FLUX D B   030030 V
                        FINSERTB        OTHER   OPTIONAL INSERT CARD FOR 'B'
                FCATLST         FCATLST FLIST   LIST CONTENTS OF FLUX CATLG. H00H00 F
                FCATADDF                OTHER   ADD BLANKS 'F' FLUX CATALOG  H00H00 F
                FCATADDG                OTHER   ADD BLANKS 'G' FLUX CATALOG  H00H00 F
                        FCATADDA        OTHER   ADD BLANKS 'A' FLUX CATALOG  H00H00 F
                BAK     FCATADDB        OTHER   ADD BLANKS 'B' FLUX CATALOG  H00H00 F
                FCATBACK        FCATBACK FLIST  COPIES FLUX CATALOG TO TAPE  H00H00 F
                FCATRELO        FCATRELO FLIST  RELOAD FLUX CATALOG TO DISK  H00H00 F
                RLD
                FTPCPY          FTPCPY  OTHER   BACKUP A FLUX TAPE           H00H01 F
                F4TO1F                  OTHER   CREATE 'F' 6250 FLUX TAPE    H02006 V
                F4TO1G                  OTHER   CREATE 'G' 6250 FLUX TAPE    H02006 V
                FTPCPY4                 OTHER   BACKUP F&G 6250 FLUX TAPE    H02006 V
                F6250                   OTHER   UPDATE 'F&G' 6250 SOURCE     H00001 F
F-END
A-ANALYSIS
  PROGRAMS      AMATRIX         AMATRIX OTHER   MATRICES WITH DATA SUMMARY   008007 V
  AND           AMATRIX1        AMATRIX1 OTHER  MATRICES W/O  DATA SUMMARY  008007 V
  PARAMETERS    A26DYMXF                OTHER   'F' MATRIX DATA CARD SETUP
                A26DYMXG                OTHER   'G' MATRIX DATA CARD SETUP
                        ALETMXA         OTHER   'A' LET MATRICES DATA CARDS
                        AHETMXA         OTHER   'A' HET MATRICES DATA CARDS
                        ALETMXB         OTHER   'B' LET MATRICES DATA CARDS
                        AHETMXB         OTHER   'B' HET MATRICES DATA CARDS
                AFLXPROC        AFLXPROC OTHER  JCL PROC FOR FLUX PLOT PROG  008007 V
                A26DYFXF                OTHER   'F' 26 DAY FLUX LIST SETUP
                ARTPLTF                         'F' STANDARD RATEPLOT SETUP
                A26DYFXG                        'G' 26 DAY FLUX LIST SETUP
                AR9AG                           'G' R9A RATE LIST SETUP
                AR10FG                          'G' R10F RATEPLOT SETUP
                AELECTG                         'G' ELECTRON FLUXPLOT SETUP
                APROTONG                        'G' PROTON FLUXPLOT SETUP
                ARTPLTG                         'G' STANDARD RATEPLOT SETUP
                ASATURNG                        'G' SATURN ENC ANALYSIS
                        ARTPLTA                 'A' STANDARD RATEPLOT SETUP
                        AFLXPLTA                'A' STANDARD FLUXPLOT SETUP
                        ASRTPLTA                'A' ST. SECT RATEPLOT SETUP
                        ARTPLTB                 'B' STANDARD RATEPLOT SETUP
                        AFLXPLTB                'B' STANDARD FLUXPLOT SETUP
```

	ASRTPLTB		'B' ST. SECT RATEPLOT SETUP		
	ASXRPLTA		'A' SECT XRAY JCL & DATA CD	020020	V
	ASXRPLTB		'B' SECT XRAY JCL & DATA CD	020020	V
A-END					
U-UTILITIES					
UJC	UJC	OTHER	JOB CARDS OF VARIOUS TYPES		
UNOTE	UNOTE	OTHER	CRBE NOTIFY STATEMENT		
UCRBE	UCRBE	FLIST	LISTS CRBE LIBRARY	H00001	F
USAVEDS	USAVEDS	FLIST	OPEN & CLOSE ALL DATA SETS	H00001	F
URATLST		OTHER	LIST F&G RATES TAPE	001002	V
UPHALST		OTHER	LIST F&G PHA TAPE	001002	V
	URATLSTA	OTHER	LIST 'A' RATES TAPE	001002	V
	URATLSTB	OTHER	LIST 'A' PHA TAPE	001002	V
	UPHALSTA	OTHER	LIST 'B' RATES TAPE	001002	V
	UPHALSTB	OTHER	LIST 'B' PHA TAPE	001002	V
UFLXTPL	UFLXTPL	OTHER	LIST FLUX TAPE	001002	V
	UBCKDRPL	FLIST	SEE THE JCL COMMENTS IN LIB.	001002	F
	UDRPRELO	FLIST	SEE THE JCL COMMENTS IN LIB.	H00001	F
	UEDITPHA	OTHER	EDITS PHA TAPES (TIMED2)	H01H02	V
	UEDITRAT	OTHER	EDITS RATE TAPES (TIMED2)	H01H02	V
	UIEHMOVE	OTHER	LOAD MOD. BACKUP UTILITY	007005	F
	ULOGMNT	OTHER	CLEANS UP LOGISTICS CATA.	001002	F
ULSTEDR	ULSTEDR	OTHER	LISTS HEL/PIO EDR TAPES	001002	F
	UORBIT	OTHER	LISTS HELIOS ORBIT TAPES	001002	F
	UORBITB	OTHER	CLINES ORBIT-B LIST PROG.	001002	F
	URATSCNA	OTHER	SCANS 'A' RATE TP. OVERLAPS	001H01	V
	URATSCNB	OTHER	SCANS 'B' RATE TP. OVERLAPS	001H01	V
URESTORE	URESTORE	OTHER	RESTORES LOGISTICS CATALOG	001002	F
	UTPSQPHA	OTHER	SQUEEZES 'A' PHA TAPES+CATA.	002008	V
	UTPSQPHB	OTHER	SQUEEZES 'B' PHA TAPES+CATA.	002008	V
	UTPSQRTA	OTHER	SQUEEZES 'A' RAT TAPES+CATA.	002008	V
	UTPSQRTB	OTHER	SQUEEZES 'B' RAT TAPES+CATA.	002008	V
ULOGISTF		OTHER	COMPRESS 'F' LOGISTICS CATA.	001002	V
ULOGISTG		OTHER	COMPRESS 'G' LOGISTICS CATA.	001002	V

TAPESCAN 4.1 - GSFC TAPE ANALYSIS AND COPYING PROGRAM 03/12/80 12:28:08.2

NOT IN LIBRARY

VOLUME TABLE OF CONTENTS FOR HGDCPY 9 TRACK

NO	DATA SET NAME	RECFM	LRECL	BLKSIZE	DEN	TRTCH	MAX BLK	MIN BLK	AVG BL
01	K3.SBCID.OHELIOSA	VS	07310	07314	6250		07217	00060	0176
02	ID.SB001.OPIOTEMP	VS	07310	07314	6250		07313	00060	0342
0003	ID.SB007.OPLTPPKP	VS	07310	07314	6250		07232	00060	0128
0004	JS.SB001.OPIOMISC	VS	07310	07314	6250		07177	00060	0270
0005	NL.SB001.OPIOFRAT	VS	07310	07314	6250		07248	00060	0288
0006	NL.SB001.PIORDISP	VS	07310	07314	6250		07284	00060	0246
0007	EHGD.SD002.MACLIB	VS	03376	03380	6250		03380	00020	0201
0008	JTD.SB008.OIMPLIB	VS	07310	07314	6250		07212	00060	0248
0009	ID.SB001.OPIONEER	VS	07310	07314	6250		07313	00020	0262
0010	NL.SB001.FLUXPLOT	VS	07310	07314	6250		06541	00060	0324
0011	NL.SB001.OPIOTEMP	VS	07310	07314	6250		07313	00020	0249
0012	NL.SD001.MACROLIB	VS	00816	00820	6250		00820	00060	0069
0013	NL.SD002.OFLUXLIB	VS	07310	07314	6250		06728	00060	0217
0014	NL.SD002.OPIONEER	VS	07310	07314	6250		06334	00060	0213
0015	ID.SB001.OPLTPPKS	VS	07310	07314	6250		06464	00060	0161
0016	ID.SB007.SOURCEMC	VS	03536	03540	6250		03072	00060	0157
0017	NL.SD001.OPIONEER	VS	07310	07314	6250		07128	00060	0259
0018	2NL.SD002.TESTLIB	VS	00816	00820	6250		00820	00020	0077
0019	NL.SD002.OANALYZE	VS	07310	07314	6250		07154	00060	0228
0020	NL.SD002.OGENERAL	VS	07318	07322	6250		06753	00060	0199
0021	L.SB001.FLUX.LOAD	VS	07310	07314	6250		06965	00020	0191
0022	B001.PIONEER.LOAD	VS	07310	07314	6250		07284	00020	0461
0023	SD001.FLUX.MACLIB	VS	03376	03380	6250		03380	00020	0201
0024	SD001.HELIOS.LOAD	VS	07310	07314	6250		07284	00020	0462
0025	GD.SD002.SFLUXLIB	VS	03536	03540	6250		03540	00020	0301
0026	NL.SB001.ORATPLOT	VS	07310	07314	6250		07245	00060	0292
0027	SB001.FLUX.SOURCE	VS	03536	03540	6250		03540	00020	0310
0028	B2NL.NEWPLOT.LOAD	VS	07310	07314	6250		07057	00060	0398
0029	ZB2NL.NEWMAT.LOAD	VS	07310	07314	6250		05746	00060	0164
0030	ZB2NL.NEWMAT.ASM	VS	03536	03540	6250		03540	00020	0294
0031	SEHGDVBS	VS	06160	06164	6250		03866	00060	0111
0032	SEHGDMAG	VS	06160	06164	6250		06164	00020	0331
0033	ZB2NLPER	VS	06160	06164	6250		06164	00060	0404
0034	SEHGDFLD	VS	06160	06164	6250		06164	00040	0354
0035	SEHGDFLX	VS	06160	06164	6250		06148	00060	0348
0036	SBCIDPIO	VS	07310	07314	6250		07312	00060	0543
0037	ZB2NLBMG	VS	07310	07314	6250		07103	00060	0417
0038	CRBE.LIB.ZBGEM	VS	00808	00812	6250		00812	00020	0071
0039	CRBE.LIB.ZBAIS	VS	00808	00812	6250		00812	00020	0070
0040	CRBE.LIB.SEHGD1	VS	00808	00812	6250		00812	00020	0076
0041	SEDWRPL1	VS	06160	06164	6250		06164	00040	0390
0042	SEDWRPL2	VS	06160	06164	6250		06164	00040	0373
0043	SEDWRXMP	VS	06160	06164	6250		06164	00060	0382
0044	SEDWRXMS	VS	06160	06164	6250		06164	00040	0404
0045	LOG.DATA.BACKUP	F	00000	07232	6250		00064	00064	0006

HGDCPY contains backups of various production load module partitioned data sets.

COMRATES

The purpose of the Pioneer F/G COMRATES list program is to provide a formatted listing of selected rates and times from rates tapes. Input to this program are Rate tapes, JCL, and data cards.

B. JCL

1) The program requires 200 bytes of main storage and approximately .5 minutes of CPU time + .5 minutes of I/O to process approximately 100 records. A description of the DD cards required by the program, the purpose of each, and when it is required follows.

DD NAME	PURPOSE OF DATA SET	INPUT/OUTPUT	DEVICE TYPE	CODE
FT06F001	Formatted Listing	Output	Printer	A
Ft08F001				
28	COPY FROM SOURCE LISTING (over)			
PIOFRATE	Tape Unit	Input	Tape	A
PCATALOG	Catalog Pointer	Input	Disk	A
PENCTLG1	Catalog 1	Input	Disk	A
PENCTLG2	Catalog 2	Input	Disk	A
PENCTLG3	Catalog 3	Input	Disk	A
PENCTLG4	Catalog 4	Input	Disk	A
SYSDUMP	Abend Dump	Output	Printer	A
DATA 5	Data Cards	Input	Cards	A

The meaning of code is as follows:

A = Always

O = Optional (If rates selected are not included in this unit, it may be dummied out.)

```
//* CCNRATES
//* THISDATE
// EXEC LINKGO,REGION=200K
//LINK.SYSLIB DD DSN=K3.SBCID.SB001.OPICNEER,DISP=SHR
// DD DSN=K3.SBCID.SBCC1.OPICTEMP,DISP=SHR
//LINK.SYSLIN DD *
 INCLUDE SYSLIB(CCNRAT)
//GO.FT06F001 DD UNIT=(3250,,DEFER),LABEL=(1,SL),DSN=OUTPUT,
// DCB=(RECFM=VBA,LRECL=137,BLKSIZE=3429,DEN=3),DISP=(NEW,KEEP),
// VOL=SER=PGD001
//GO.FT08F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT09F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT10F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT11F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT12F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT13F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT14F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT15F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT16F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT17F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT18F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT19F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT20F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT21F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT22F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT23F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT24F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT25F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT26F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT27F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.FT28F001 DD SYSOUT=A,DCB=*.FT06F001
//GO.PICPRATE DD UNIT=(3250,,DEFER),DISP=SHR,DCB=(EROPT=ACC,DEN=3),
// LABEL=(,SL,,IN),DSN=PICRAT,VOL=SER=PICTAP
//GO.PCATALOG DD DSN=K3.SBCID.SB001.PFRECCRF,DISP=SHR
//GO.PENCTLG1 DD DSN=K3.SBCID.SBCC1.PFRECCR1,DISP=SHR
//GO.PENCTLG2 DD DSN=K3.SBCID.SB001.PFRECCR2,DISP=SHR
//GO.PENCTLG3 DD DSN=K3.SBCID.SB001.PFRECCR3,DISP=SHR
//GO.PENCTLG4 DD DSN=K3.SBCID.SB001.PFRECCR4,DISP=SHR
//GO.SYSUDUMP DD SYSOUT=A
//GO.DATAS DD *
```

C. Data Cards

1) The data cards follow the last DD card in the deck setup and are read using the Namelist convention of Fortran IV. There are two types of data cards for the program: Time Card (which defines the start time and stop time; Select Card (which defines the rates to be listed). The required parameters on the time card are as follows:

HMONS	START MONTH
HSDY	START DAY
HYRS	START YEAR
HOURS	START HOURS
HMINS	START MINUTES
HSECS	START SECONDS
HMONE	END MONTH
HEDY	END DAY
HYRE	END YEAR
HOURE	END HOUR
HMINE	END MINUTE
HSECE	END SECOND

The required parameters on the select card are as follows:

QTREND = T or F
RATES = 'Rate or Rates Desired'
(Rates must appear in quotes).

Example of Data cards is as follows:

```
&TIME HMONS=10,HSDY=01,HYRS=76,HOURS=00,HMINS=00,HSECS=00,
      HMONE=01,HEDY=01,HYRS=77,HOURE=01,HMINE=01,HSECE=01 &END
&SELECT QTREND=F,RATES='R1','R2A','R2B','R3A','R11A','R11B', &END
```

D. Output

The primary output from the COMRATES program is a formatted listing of the rates selected in hexidecimal and decimal form. This program should be run for very short periods of selected times due to the large consumption of paper required.

When a job is terminated abnormally with a user completion code, refer to the User Abends Section of the "IBM System 360 General I/O Package".

A. Description

The purpose of the Pioneer F&G Magnetic Field and Sectored Rates List program is to create a formatted listing of selected sectored rates merged with magnetic field coordinates. Input to this program is Rates tapes and a Magnetic Field tape.

B. JCL

The MAGSEC program is currently in executable load module form on 'SYS2.LOADLIB' named SEHGDMAG. The program requires .5 minutes CPU and .5 minutes I/O to process approximately 10 days' worth of data. A description of the DD cards required by the program, the purpose of each, and when it is required follows:

DD NAME	PURPOSE OF DATA SET	INPUT/OUTPUT	DEVICE	CODE
STEPLIB	User Program	Input	Disk	A
FT05F001	Define Data	Input	Disk	A
FT06F001	Formatted Listing	Output	Printer	A
FT10F001	Magnetic Field Tape	Input	Tape	A
PIOFRAT	Rates Tape	Input	Tape	A
PCATALOG	Catalog Pointer	Input	Disk	A
PENCTLG1	DRS Catalogs 1	Input	Disk	A
PENCTLG2	DRS Catalogs 2	Input	Disk	A
PENCTLG3	DRS Catalogs 3	Input	Disk	A
PENCTLG4	DRS Catalogs 4	Input	Disk	A
SYSUDUMP	Abend Listing	Output	Printer	A
DATA5	Parameter Cards	Input	Reader	A

APLTAP - User's Guide

5/22/78

GENERATION OF APL TAPES OF PIONEER DATA

1) Change the JCL comment card to reflect the proper rate. (This has no effect on the program operation.)

2) Change FT10F001 to reflect proper tape information: (A) the volume=serial number; (B) the file number; and (C) the disposition, i.e. whether this is a new file, or whether it should be appended to the existing file. disp=(,keep) specifies beginning of file. disp=(mod,keep) specifies appending to existing file

3) Change FT15F001 to reflect the proper (A) start date and time of the data to be processed; (B) the end date and time of that data; and (C) the satellite identification (='F' or ='G').

4) Change FT05F001 to reflect the proper rate. (start in column 2 of the input line.)

NOTES:

Only one rate can be run at a time.

Only one job can be executing at a time since the input tape(s) and/or output tape(s) may be the same tapes for both jobs. Using the RELEASE function (to release the next job in a series from the hold queue) can facilitate submitting several jobs at once and having them run successively.

SENAL.LIB.CNTL(APLRUN)

```
//SENALALL JOB (SB00'27E9F,P,SA000',00T00F),BFR,MSGLEVEL=1,TYPRUN=HOLD
//*
//*P10-R10F
//CHECK1 EXEC SRCHDS,DSN='SENAL.APLOAD'
// EXEC PGM=IEBPR1A,COND=(0,EQ,CHECK1.SEARCH),REGION=20K
//D1 DD DSN=SENAL.APLOAD,VOL=SER=KTUSR3,UNIT=2314,DISP=(,CATLG),
// SPACE=(TRK,(40,,1),RLSE)
// EXEC BACKUP,COND=(0,EQ,CHECK1.SEARCH),PGMC=APLOAD,TAPEDSN='APLTAP',
// TAPEVOL=NAL04,FILESEQ=7,LABEL=NL,LIB='SENAL.APLOAD'
// EXEC PGM=APLTAP,REGION=300K
//STEPLIB DD DSN=SENAL.APLOAD,DISP=SHR
//FT06F001 DD SYSOUT=A,SPACE=(CYL,(40)),
// DCB=(RECFM=VBA,LRECL=137,BLKSIZE=7265,BUFNO=1)
//FT25F001 DD SYSOUT=A
//FT09F001 DD DSN=PICPAT,UNIT=(1600,,DEFER),VOL=SER=RATEP,
// DISP=SHR,DCB=DEN=3
//FT10F001 DD DSN=APLTPE,VOL=SER=DWRCB,LABEL=(01,NL),
//    DISP=(MOD,KEEP),UNIT=(1600,,DEFER),
//    DCB=(RECFM=VS,BLKSIZE=7294,DEN=3)
//FT11F001 DD UNIT=2314,SPACE=(TRK,(600)),
// DCB=(BLKSIZE=5960,BUFNO=1)
//FT12F001 DD SYSOUT=A,SPACE=(CYL,(40)),
// DCB=(RECFM=VBA,LRECL=137,BLKSIZE=7265,BUFNO=1)
//FT13F001 DD UNIT=2314,SPACE=(1680,100),DCB=BLKSIZE=1688
//FT14F001 DD SYSOUT=A,DCB=(RECFM=VBA,LRECL=137,BLKSIZE=1100)
//FT20F001 DD DSN=KJ.ZB2NL.SD001.PICPRTLT,DISP=SHR
//FT30F001 DD DSN=KJ.ZB2NL.SD001.PICPRTLT,DISP=SHR
//FT40F001 DD DSN=KJ.ZBRXB.SB001.PFREDCP1,DISP=SHR
//FT41F001 DD DSN=KJ.ZBRXB.SB001.PFREDCP1,DISP=SHR
//FT42F001 DD DSN=KJ.ZBRXB.SB001.PFREDCP2,DISP=SHR
//FT43F001 DD DSN=KJ.ZBRXB.SB001.PFREDCP3,DISP=SHR
//FT44F001 DD DSN=KJ.ZBRXB.SB001.PFREDCP4,DISP=SHR
//FT45F001 DD DSN=KJ.ZBRXB.SB001.PGDRSCT0,DISP=SHR
//FT46F001 DD DSN=KJ.ZBRXB.SB001.PGDRSCT1,DISP=SHR
//FT47F001 DD DSN=KJ.ZBRXB.SB001.PGDRSCT2,DISP=SHR
//FT48F001 DD DSN=KJ.ZBRXB.SB001.PGDRSCT3,DISP=SHR
//FT49F001 DD DSN=KJ.ZBRXB.SB001.PGDRSCT4,DISP=SHR
//SG4060ZZ DD DSN=NULLFILE,UNIT=1600,,LABEL=(1,NL),
// DCB=(RECFM=F,BLKSIZE=240,DEN=3),DISP=(NEW,KEEP),
// VOL=SER=SCRTCH
//SYSUDUMP DD SYSOUT=A
//FT15F001 DD *
 &PLOT HTB=73,11,29,00,00,HTE=73,12,09,20,00,QLIST=T,HID='F',
 QTRCHK=F
                                                    &END
//FT05F001 DD *
R10F
// EXEC NOTIFYTS
//
```