Time Estimates for a New ISEE
Datapool Encyclopedia Database System

Pam Schuster, CSC

Preliminary Specifications for the
Datapool ENCGEN and ENCMRG and LSELECT Programs

Eunice Eng, GSFC

December 1982

Proofed By

Job Number

This memorandum responds to the request for a time estimate to design, code, and test programs that would be required to generate a new ISEE Datapool encyclopedia data base. (see Appendix A.)

This processing system is modelled after the existing Voyager/ ISEE ENCYGEN system, and is illustrated in Figure 1. The current DATAPOOL Processing System is shown in Figure 2.

A comparison of Figures 1 and 2 shows that:

1) The WRTLST.SORT function of the current system would be replaced with the use of the Datapool LOG in creating the ENCY data base. (see Appendix C for an explanation.)

2) The programs DPT ENCYGEN, DPT ENCMRG, and DPT LSELECT need to be coded. The ENCGEN and ENCMRG programs would access the LOG in their operation. The LSELECT program would not have to access the log, unless desired.

3) The existing DPTSUM program would need to be at least 80% rewritten. It would need to access the new ENCY data record format. The data selection and timing segments would be entirely different (i.e. putting together summary intervals of perhaps 15 minutes, or putting together raw data type verses to make 64-minute summaries, or perhaps generalized time summaries, instead of simply processing one 64-minute DPT record at a time as it currently does.)

4) The FOURIER program would make or obtain whatever magnetic field time summaries it required from the ENCY tapes, replacing the current MAGSUM program function.

From experience with other cosmic ray satellite system programs, one can estimate expected program sizes depending on the degree of complexity in the data processing.

For the DATAPOOL data, two factors will determine the relative complexity of the coding for the new system:

a) The inclusion and proper handling of timeline changes.

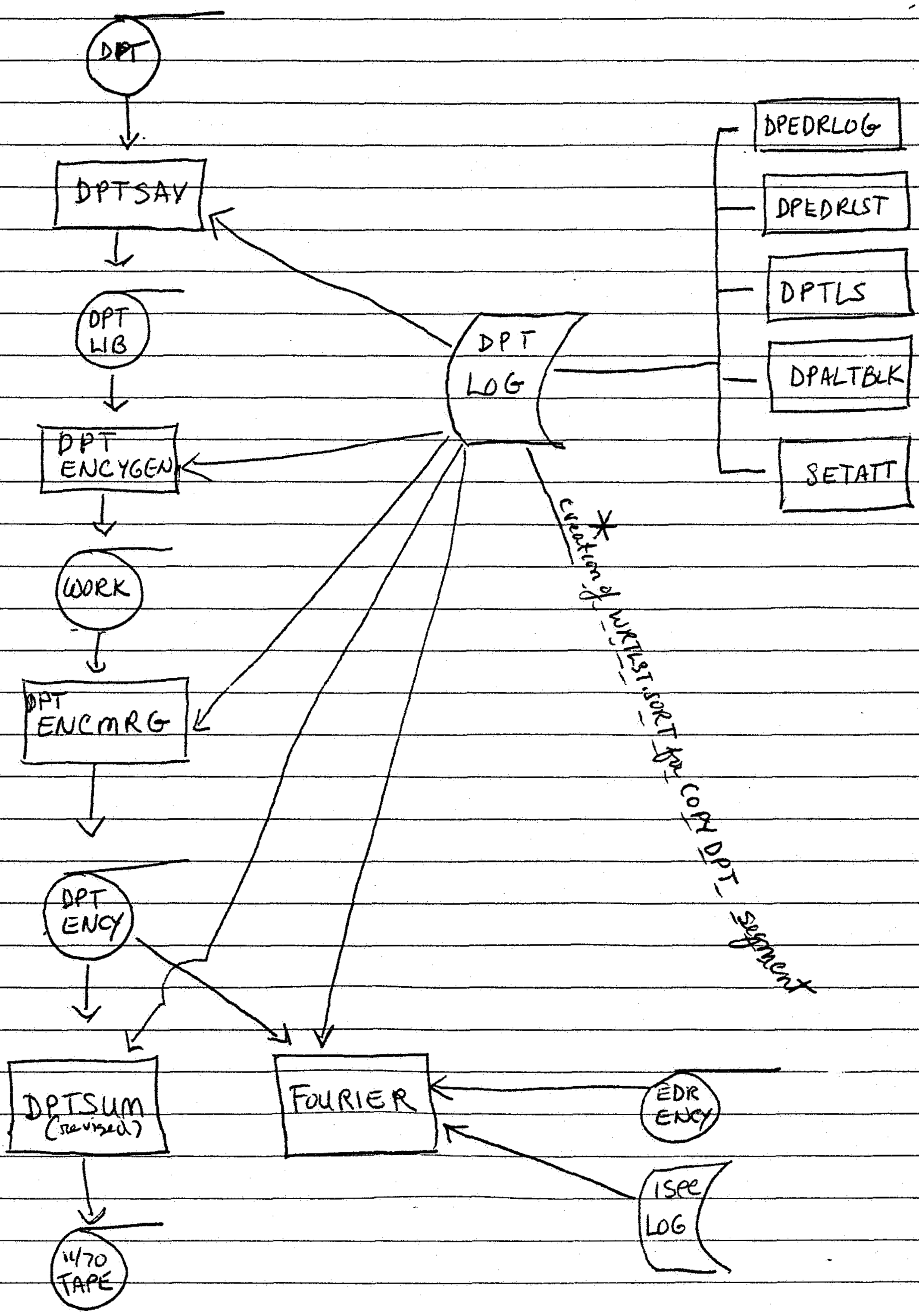
   Data records which begin a new timeline are flagged in that record; that data record number along with the record start time are stored in the DATAPOOL tape file header.

   DATAPOOL file header timeline information can be used along with the start time of the record just before the timeline change to establish how much of the latter record is valid data (in time).

Proofed By
Joh Hun...

Figure 1

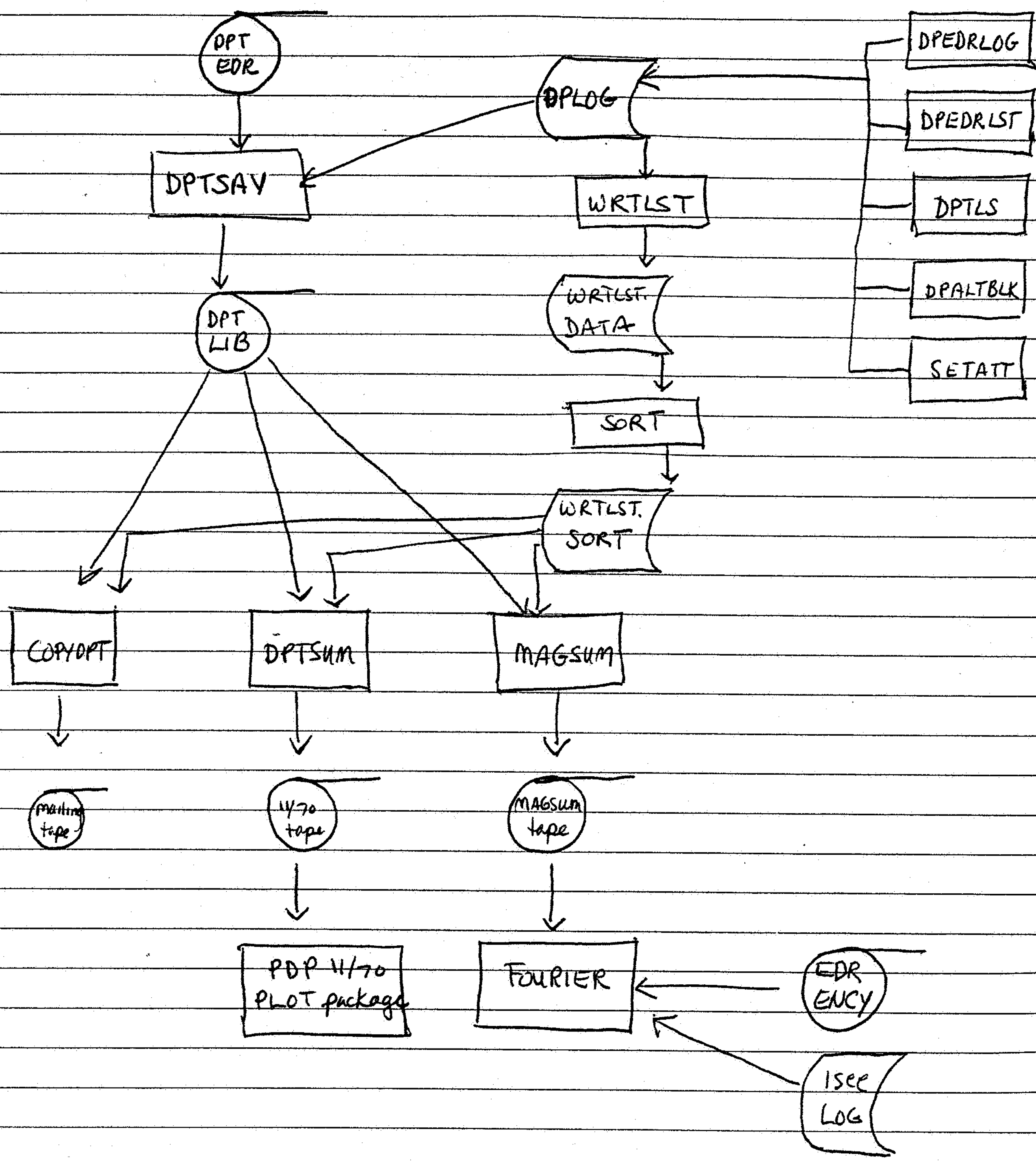
DESIRED DATAPOOL PROCESSING SYSTEM



An LSELECT type Listing program for the ENCY database is needed

* The COPYDPT program would operate, as in the current DATA POOL Processing system, through the WRTLST. SORT dataset and SETATT program software.

Current DATAPOOL PROCESSING SYSTEM



utility programs available:

DPTLIST formatted listing of DPT EDR or LIB tapes
MSLIST formatted listing of MAGSUM output tapes
various foreground executing CLISTS to assist with some processing applications

b) The type of data summaries, if any, required for each of the nine experimenter quantities.

Time handling may not be the same for all quantities. Some occur 64, 30, 12, 4, 3, etc. times per 64-minute DATAPOOL tapes (DPT) record, depending on the frequency of readout. It is necessary to know how to handle quantities which are read out every 16 minutes, or 21-1/3 minutes, etc., especially if a 'raw rates' type verse is to include all data types within the 15-minute volume context.

The start time of the data record is the start time of sampling interval number 1 for all frequencies of readout. The start times of subsequent intervals are computed relative to interval 1, with the interval duration equal to about 64 minutes/N where N is the number of readouts per record. (There is an error of about +2 seconds by the end of the 64-minute record, i.e., for magnetic field data [start time of first record + (60 entries * 64 seconds per entry) = start time of next record + ~2 seconds]).

For data which are later replaced in updated DPT tapes, merging new data or parts of previously missing data may be somewhat complicated if the record start times are much different from the prior data start times.

Given these observations, and our knowledge of the size of the existing DPTSUM and MAGSUM programs and their limitations, time estimates were made for completion of the work mentioned in points 2) and 3) above. They are given in Table 1. Appendix B provides the rationale for these estimates. CSC assumes these are FORTRAN programs and that LOG access routines for all phases of the work are available in black box form.

Table 1. DATAPOOL System Program Completion Estimates (exclusive of work required on log access routines)

	approximate upper limit new lines linecount	÷10/day	approximate lower limit new lines linecount		÷10/day
DPTENC	395*	40 days	245*		25 days
DPTMRG	300*	30 days	200*	(minimum # of data quantities)	20 days
ENCYLIST	200	20 days	200	(minimum # of data quantities)	20 days
		90 days			65 days

	new lines linecount	
DPTSUM (essentially new)	316	32 days

*FORTRAN programs, exclusive of log interface subroutines

CSC believes that the upper limit is closer to the time required to complete the ENCYCLOPEDIA production system. We believe 32 days is a maximum requirement for a new DPTSUM. Time estimates include a basic level of documentation which does not currently exist - i.e. a system description and user's guides, as well as internal code PDL and Prologues.

The ENCYLIST programming would be mostly done by a Junior Programmer.

Attachment I, supplied by the government, is a preliminary summary of the proposed DPT ENCYGEN system program flow. It lists some undefined system variables which would affect the relative complexity of the code for all programs, and thus the time estimates in Table 1, but probably not by more than 10 to 15%, with the possible exception of the following:

There is a problem with the log access routines which the MERGE program would require. These access routines exist, but they are in ASSEMBLER language and are not FORTRAN callable. The government should decide how these routines are to be provided:

1) Do we duplicate the corresponding routines in FORTRAN?

Functions	
MOW	open work block
MOE	open ency block
MCW	close work block
MCE	close ency block
MTERM	close log

2) Do we write a FORTRAN callable ASSEMBLER interface to the existing log routines?

3) Do we write the merge program in ASSEMBLER?

Typical log interface routines for the ISEE Encyclopedia generator (ENCGEN) which are all in FORTRAN, are about 80 executable statements each.

The desired ENCY record format needs to be supplied by the government, as well as the type of summarizing done and the kind of 'raw rates' quantities to be put on the ENCY tape.

A method of accepting merging data needs to be decided. The government must decide what constitutes better data.

These questions are the principle unanswered questions at the time of this writing. Programming estimates provided are somewhat dependent on the answers. LOG access routine development time estimates should be considered separately from this work.

Appendix B. Rationale for Programming Time Estimates

The proposed DPT ENCGEN program would need to perform similar functions to the current DPTSUM and MAGSUM programs. Both of those programs are limited in their processing capabilities:

1) DPTSUM summarizes one record at a time as an entity. MAGSUM summarizes each 15 minutes of magnetic field data including some BAME data. Neither program can be easily generalized to make other time period summaries, although MAGSUM could be changed to produce 64-second summaries and DPTSUM could get multiples of 64-minute summaries.

2) Neither DPTSUM or MAGSUM are currently coded to process timeline changes. Each and every record is assumed to contain 64 minutes of contiguous data. Timeline changes will generally result in records whose start times are before the projected 64-minute end time of the previous record. The entire first timeline change record is treated as if it were overlap data, which is bypassed. Effectively, a false data gap is created.

3) Both programs are limited in the way they must be used to handle replacement or insertion data. See Appendix C for an explanation.

CSC expects the new DPT ENCGEN program to be more complex than either DPTSUM or MAGSUM. Table 2 gives the basic functions within those programs along with the executable code linecount needed to perform the functions.

Comments on Table 2:
To estimate the DPT ENCGEN program size:

1) All basic MAGSUM functions would be necessary except the COPYTAPE and WRTLST.SORT type functions.

2) Additional code to take care of timeline changes needs to be added - about 80 lines.

3) Code to search and correct for year changes within one DPT record needs to be added; DPLOG interface code needs to be added - about 50 lines total.

4) Any other DPT data quantities desired must be allowed for, along with the creation of the 'raw rates' type verse(s) - about 150 lines if all other 8 quantities are treated in some fashion.

Table 2.

Current DPTSUM functions and their executable code linecounts

DPTSUM has a total of about 270 executable lines of code as follows:

Function		Approximate Linecount
Initialization and input processing		32
Data location using the WRTLST.SORT dataset		20
Establish times		20
Summarize data	8 types @ ~17 lines ea.	136
	1 (Smith)	30
Write data records	(12 data + 20 header)	32
End of job		10
		~280

Current MAGSUM functions and their executable code linecounts

MAGSUM has 320 executable lines as follows:

Function	Approximate Linecount	
Initialization and input processing	32	
Copy tape option	25	
Data Location + start processing using WRTLST.SORT	30	
Establish times for data summaries	50	
Summarize magnetic data	100	
End of files, outputting	20	
		257
Subroutine LIBMOU mount data tapes	40	
Subroutine DYNMBR time conversion	15	
		≈312

Proofed By

Job Number

Estimated lines for the new program might then fall between two approximate limits:

upper: 320 - (25 + 30) + 80 + 50 + 150 = 545 lines

lower: 320 - (25 + 30) + 80 + 50 = 395 lines

If the magnetic summarizing segment were used, it would be necessary to redo to conform the existing MAGSUM code to the ENCY output record format. About 100 lines of MAGSUM might be used, but redone in that way.

The LIBMOU and DYNMBR subroutine might be used, if the DPT ENCGEN structure allows it - about 55 lines total.

Subtracting these linecounts from the estimated totals above, give 545 - 150 = 395 new lines (upper limit) and 395 - 150 = 245 new lines (lower limit).

Using the CSC estimate of 10 lines per day for new code with documentation (for a non-complex system) gives about 25 to 40 days for the two limits. CSC believes the upper limit is closer to the time that will be required.

For the DPT ENCMRG program time estimate, we believe the program size will also be comparable to the existing MAGSUM program. The following general functions need to be done:

Initialization and input	30 lines
Locate and mount tapes	30 lines
Read input records Compare data (use pad as an indicator? Compare each time for all quantities? Or simply replace old with new?)	>100 lines
Write output and update log	>20 lines
	>180 lines exclusive of log interface subroutines

Different merge code may be required for the nine different data quantities. We might add 8* ~15 lines = 120 lines or so more to the above estimate.

The expected upper and lower limits for the merge program would then be between 200 and 300 executable statements. Completion time would range from 20 (lower limit) to 30 (upper limit) days.

For the ENCYLIST program (LSELECT type) time estimate, we note the following:

The current DPTLIST program has 200 executable statements.
The current MSLIST program has 75 executable statements.

Table 3 summarizes these programs in their basic functions.

Table 3.

Current DPTLIST functions and their executable code line-counts

DPTLIST has a total of 202 executable lines of code as follows:

Function	Approximate Linecount
Initialization and input	20
Select + mount tape	20
Write file header	10
Locate first requested list time	20
Establish data times, check for year change	20
Write start of data record information	12
Write individual data quantities ~10 lines for 10 quantities = maneuver information + Havestadt, Smith, Steinberg, Anderson, Barne, Scarf, Von Rosenvinge, deFecter, Meyer	100
	202 total statements

Current MSLIST functions and their executable code line-counts

MSLIST has a total of 75 executable lines of code as follows:

Function	Approximate Linecount
Initialization and input	17
Mount tape, get requested start time and end time	7
Read data, get data times, look for 1st time request	15
Write out data - 2 options possible	35
	~74

Comments on Table 3:
With the executable linecount needed for each function, neither program is well formatted, although DPTLIST is better.

The ENCYLIST program would have to be able to do both the MSLIST and the DPTLIST functions entirely, if all data quantities are put onto the ENCY tape. Therefore, we expect the new program to be at least 200 statements long. Completion time would be about 20 days.

The DPTSUM program would need to be about 80% rewritten because of the different expected ENCY record format and timing considerations (putting together into 64-minute summaries (or a generalized time summary) data from 15-minute summaries or raw rates type verses). LOG routine access code would also need to be added. The revised program would be at least as large as the current DPTSUM program and perhaps as much as 80 - 100 statements larger, to account for the above differences. Assuming then (.80 x 270) + 100 = 316 statements as an expected size, we would predict 316/10 = 32 days to redo DPTSUM, assuming the current DPTSUM philosophy of operation is preserved. Note: Encyclopedia blocks within the LOG would have to be searched by desired volume number.

Appendix C. Merge and Replacement data handling limitations in the current DATAPOOL processing system

In order to merge or replace data generated from DPTSUM, routines on the PDP 11/70 must be used. There is no automatic process for inserting data into an existing large DATAPOOL file, although add-on data handling is straight forward. Inserting/replacing data involves splitting the existing file and then combining the pieces with the new data. Dr. R. McGuire's program FLXMAN is used. He has considered the problem of automatic insertion, but has not implemented in FLXMAN. Therefore, whenever replacement DPT data arrive, DPTSUM must regenerate the entire 64-minute summary data base, and an entirely new PDP 11/70 file should replace any existing file.

The MAGSUM program output data base has a similar problem for handling replacement or insertion data. The program user must keep track, exactly, of what data are located in each file of the multi-file output tape. Replacement/Insertion of data must be done on a file basis. The user must copy all existing files, up to the time of the new file, onto a new tape, copy the new MAGSUM file, and then copy any remaining 'old' files onto the new permanent MAGSUM tape. In addition, an exact correspondence must be maintained between the entries of the WRTLST.SORT data set (which contains a time-ordered list of library tape files and is used by DPTSUM and MAGSUM to determine file order of processing) and the files of the output tape. Inserting previously non-existent files of data, would require re-doing all the data base beyond the insertion point, in order to maintain the 1:1 correspondence.

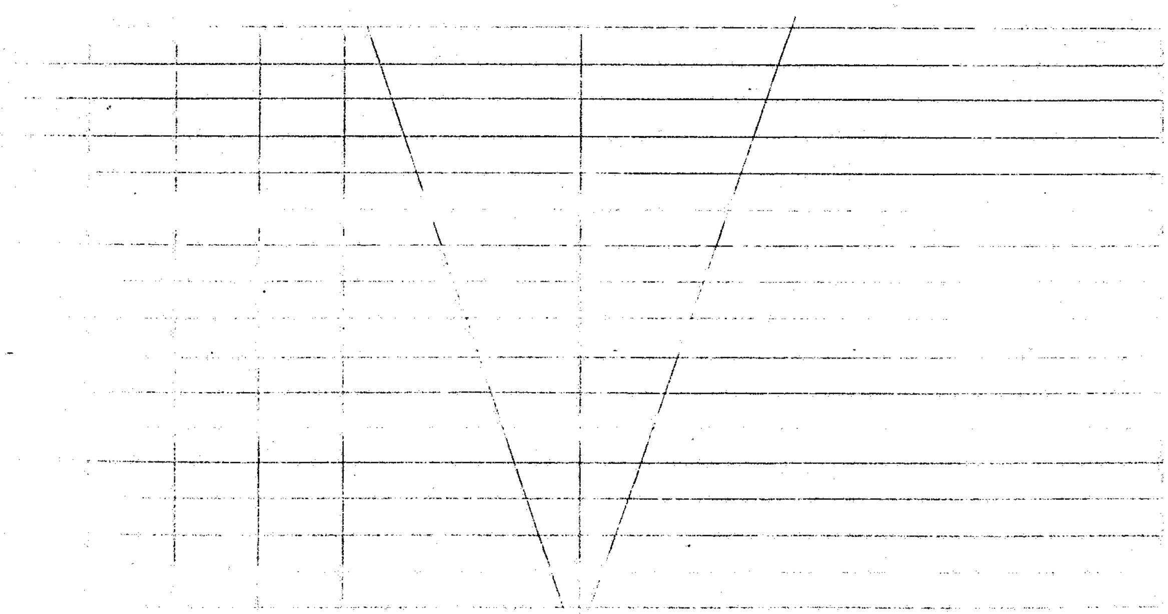


Figure 2: Desired Datapool System

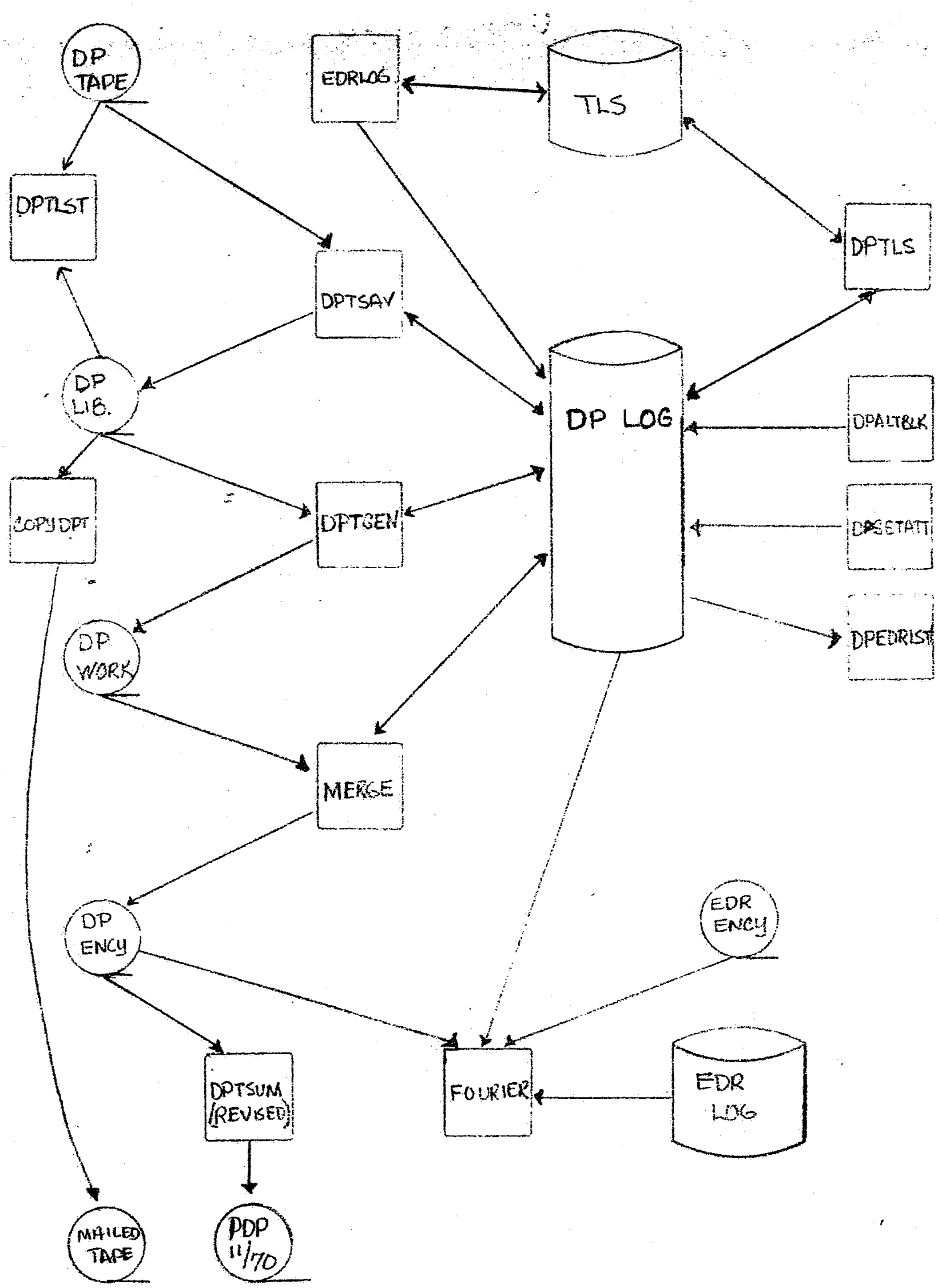
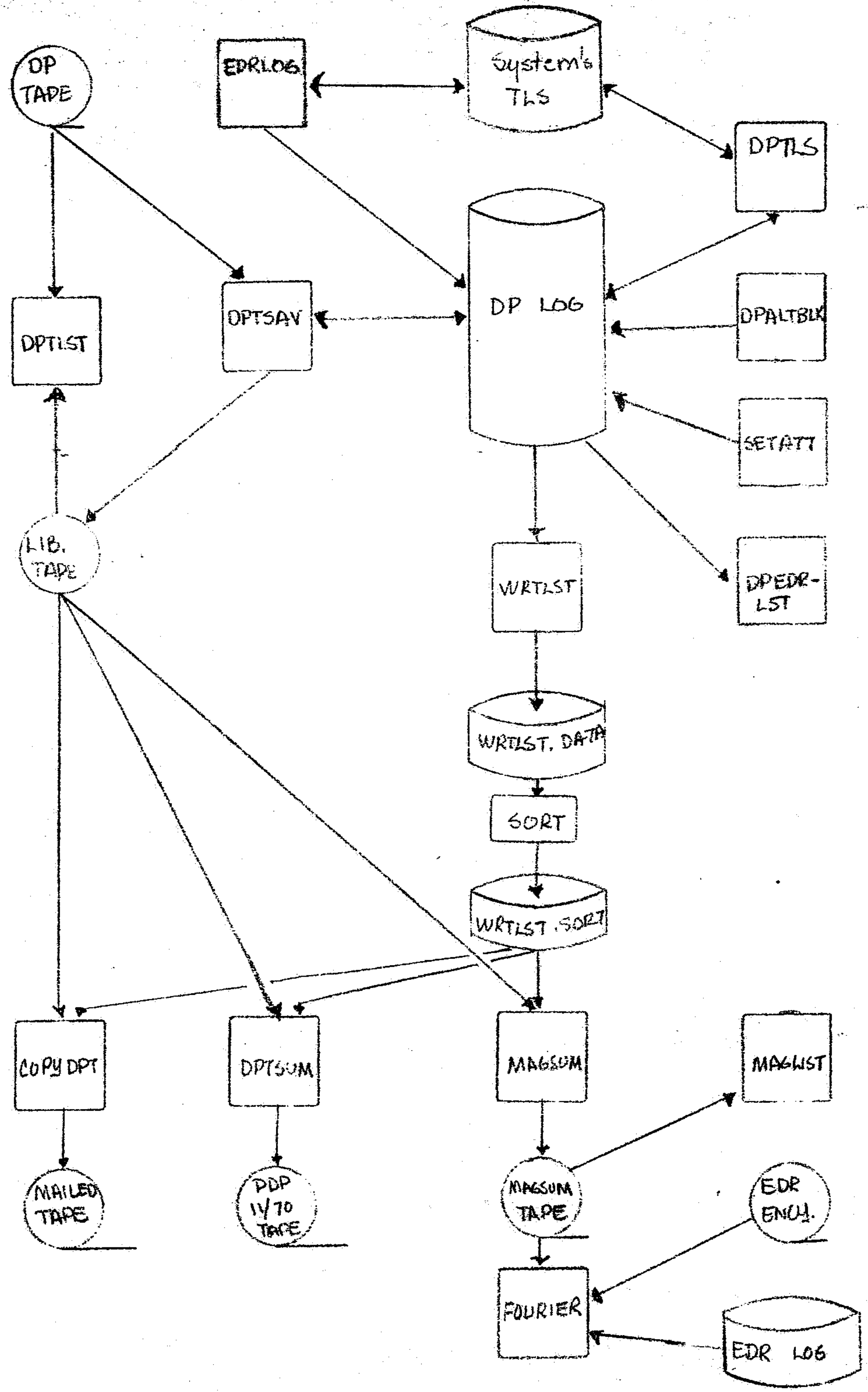


Figure 1: Current Datapcol System



4/23/86
ENG

COPY ISEE-3 DATA POOL LIBRARY TAPES

—Purpose:

The objective is to create new data pool library tapes using less IOEXCPs.

—Proposed solution:

The current library tape has the following physical attribute

Record format: F
Record length: 3240 BYTES
Block size: 3240 BYTES

IO EXCPs are counts of block I/O's. Therefore if the library tapes were blocked as follows the EXCP charges would be cut by 10.

Record format: FB
Record length: 3240 BYTES
Block size: 32400 BYTES (max 32768)

—Methods:

System utilities

- TAPESCAN, copy facility will not change the tape's DCB's. TAPESCAN makes exact copies.

- IEBGENER, copies only one file at a time.
  - Maximum number of files on a tape = 165.
  - Number of tapes to be copies = 4

- PATRICK, PATRICK will copy multiple files as long as the file Data set names (DSN's) are identical. However ISEE DP library files have unique DSNs.

None of the above system utilities are satisfactory. User routine would best copy the above tapes.

— Program requirement.

1. Interface with the DATA POOL automated processing log to get a list of all DSNs for each file, in ascending file order, for a given volume.

2. Read input library tape records

3. Output input library tape records. The output will be blocked by JCL.

4. Position to next file at the end of each file, if another file exists. POSN w/ DSN

• What information will not be preserved.

1. The input volume serial will not be copied

2. The standard labels will not be copied. Therefore the actual file creation date will not be preserved.

— Procedure

A. original
↓
B. backup (blocked data; new vol ser)
↓
C. new original (blocked data) (vol-ser of A) with compare to B

D. Compare A and C

E. Compare B and old Back-up E

A
compare
↕
C

F. If compare D successful replace A

G. If compare E successful replace E.

H. Change DPTSAVE JCL to output library tapes with a blocking factor

I Change DPTENCY to read from blocked data

J Change [illegible]

```
**** TSC FOREGROUND HARDCOPY ****
DSNAME=ZBPAS.DPGEN.SOURCE                                  ($BUILD$ )

//ZBPASBLD JOB (SE013,EF3,09),ISEE,MSGCLASS=A,TIME=(0,30),
//    NOTIFY=ZBPAS,CLASS=0
//*ENCYELD DPTGEN
/*JOBPARM QUEUE=FETCH,LINES=50,COPIES=2
//FORT EXEC FORTRANH,PARM='XREF,NAME=DPTIC'
//SOURCE.SYSIN DD DSN=SEJRC.DPTENC.FORT(MAIN),DISP=SHR
//             DD DSN=SEJRC.DPTENC.FORT(OUCHPT),DISP=SHR
//*            DD DSN=SEEKE.DPGEN.SOURCE(INIT),DISP=SHR
//             DD DSN=SEEKE.DPGEN.SOURCE(DTUPK),DISP=SHR
//             DD DSN=SEEKE.DPGEN.SOURCE(SETTIM),DISP=SHR
//FORT EXEC FORTRANH,PARM='XREF'
//SOURCE.SYSIN DD DSN=ZBPAS.DPGEN.SOURCE(READER),DISP=SHR
//             DD DSN=ZBPAS.DPGEN.SOURCE(TIMCHK),DISP=SHR
//             DD DSN=ZBPAS.DPGEN.SOURCE(YEARCK),DISP=SHR
//             DD DSN=ZBPAS.DPGEN.SOURCE(PROCTM),DISP=SHR
//             DD DSN=ZBPAS.DPGEN.SOURCE(TIMLIN),DISP=SHR
//             DD DSN=ZBPAS.DPGEN.SOURCE(DATCHK),DISP=SHR
//FORT EXEC FORTRANH,PARM='XREF'
//SOURCE.SYSIN DD DSN=SEEKE.DPGEN.SOURCE(INITAN),DISP=SHR
//             DD DSN=SEEKE.DPGEN.SOURCE(ANALYZ),DISP=SHR
//             DD DSN=SEEKE.DPGEN.SOURCE(OUTPUT),DISP=SHR
//             DD DSN=SEEKE.DPGEN.SOURCE(OUMSUM),DISP=SHR
//             DD DSN=SEEKE.DPGEN.SOURCE(OUVOLN),DISP=SHR
//FORT EXEC FORTRANH,PARM='XREF'
//SOURCE.SYSIN DD DSN=SEEKE.DPGEN.SOURCE(OURAW),DISP=SHR
//             DD DSN=SEEKE.DPGEN.SOURCE(COLFIL),DISP=SHR
//             DD DSN=SEEKE.DPGEN.SOURCE(COLFRS),DISP=SHR
//             DD DSN=SEEKE.DPGEN.SOURCE(COLLT1),DISP=SHR
//             DD DSN=SEEKE.DPGEN.SOURCE(COLLT3),DISP=SHR
//             DD DSN=SEEKE.DPGEN.SOURCE(COLNXD),DISP=SHR
//             DD DSN=SEEKE.DPGEN.SOURCE(DPTERM),DISP=SHR
//             DD DSN=SEEKE.DPGEN.SOURCE(UPFEET),DISP=SHR
//FORT EXEC FORTRANH,PARM='XREF'
//SOURCE.SYSIN DD DSN=SEEKE.DPGEN.SOURCE(LOL),DISP=SHR
//             DD DSN=SEEKE.DPGEN.SOURCE(BLDSER),DISP=SHR
//*            DD DSN=SEEKE.DPGEN.SOURCE(RDIRET),DISP=SHR
//             DD DSN=SEEKE.DPGEN.SOURCE(LREAD),DISP=SHR
//             DD DSN=SEEKE.DPGEN.SOURCE(LOW),DISP=SHR
//             DD DSN=SEEKE.DPGEN.SOURCE(LCL),DISP=SHR
//             DD DSN=SEEKE.DPGEN.SOURCE(LCW),DISP=SHR
//             DD DSN=SEEKE.DPGEN.SOURCE(EOPROL),DISP=SHR
//             DD DSN=SEEKE.DPGEN.SOURCE(LTERM),DISP=SHR
//SOURCE EXEC ASMG
//SOURCE.SYSLIB DD DSN=SEEKE.DPGEN.SOURCE,DISP=SHR
//             DD DSN=SYS2.MACLIB,DISP=SHR
//             DD DSN=SYS1.MACLIB,DISP=SHR
//SOURCE.SYSIN DD DSN=SEEKE.DPGEN.SOURCE(FNEXT),DISP=SHR
//* LOADMOD SEEKE
// EXEC LINK
//LINK.SYSLIB DD DSN=SB#VG.GENERAL.LOAD,DISP=SHR
//LINK.SYSLMOD DD DSN=SEEKE.DPGEN.LOAD,DISP=SHR
//LINK.OBJECT DD *
  INCLUDE SYSLIB(RDIRET)
  ENTRY DPTIC
  NAME DPTGEN(R)
// EXEC NOTIFYTS
```

```
     CALLING SEQUENCE FOR SUBROUTINES OF THE ISEE DATAPOOL ENCYCLOPEDIA
          GENERATOR PROGRAM

     ROUTINE                                   DESCRIPTION

   DPTENC ......................             MAIN PROGRAM

     KTIME                                   RETURN JOB RUN TIME FROM SYSTEM
     YDMD                                    GET CALENDER TIME FORM OF JOB RUN TIME

     INIT ....................               LOCATE AVAILABLE DATA FROM LOG, MOUNT
                                             FIRST INPUT TAPE AND MOUNT OUTPUT TAPE

        LCL                                  SEARCH LOG LIBRARY BLOCKS FOR INPUT

           BLDSER                            BUILD VOL-SER OF INPUT LIB TAPE
              INCORE                         CONVERT DATA TO EBCDIC
              KOC                            COMPARE LOGICAL BYTES
           RDIRET                            LOCATE SATELLITE BLOCK OF LOG
           LREAD/LWRITE                      READ/WRITE A LOG BLOCK

        LCW                                  ALLOCATE THE LOG WORK BLOCK FOR THE JOB

           BLDSER                            BUILD VOL-SER OF INPUT LIB TAPE
              INCORE                         CONVERT DATA TO EBCDIC
              KOC                            COMPARE LOGICAL BYTES
           EOPROL                            SET PROLOG VARIABLES OF LOG BLOCK
              KTIME                          RETURN JOB RUN TIME FROM SYSTEM
              YDMD                           GET CALENDER TIME FROM JOB RUN TIME
           RDIRET                            LOCATE SATELLITE BLOCK OF LOG
           LREAD/LWRITE                      READ/WRITE A LOG BLOCK

        MOUNT -                              MOUNT TAPES
        PARM                                 RETRIEVE MAXFIL PARM FROM EXEC JCL CARD

        INCORE                               CONVERT PARM FROM EXEC CARD TO EBCDIC
        KMVC                                 MOVE BYTES OF DATA

     READER ..................               READ IN TAPE DATA CORRESPONDING TO
                                             ONE LIBRARY BLOCK

        FREAD                                READ A TAPE RECORD
        SETTIM ...............               RETURN SECONDS FROM START OF 1977

        TIMCHK ...............               DRIVER FOR TIME CHECK PROCESSING

           YEARCK ............               READ THROUGH DATA RECORDS SEARCHING
                                             FOR A YEAR CHANGE; VERIFY AND
                                             ASSIGN A YEAR TO EACH DATA RECORD
              SETTIM .........               RETURN SECONDS FROM START OF 1977

           PROCTM ............               ASSIGN A START AND STOP RECORD AND DATA
                                             TIME TO EACH INPUT RECORD CORE IMAGE
              SETTIM .........               RETURN SECONDS FROM START OF 1977
              DTUPK                          GIVEN MILLISECONDS FROM THE START OF
                 YDMC                        GET CALENDER DATE OF TIME
                                             1977, RETURN THE CALENDAR DATE AND
                                             TIME

           TIMLIN ............               PROCESS TIMELINE CHANGES BY RESETTING
                                             RECORD STOP DATA TIME IF NEEDED
              SETTIM .........               RETURN SECONDS FROM START OF 1977

        DATCHK ...............               CROSS CHECK DATA ; FLAG RECORDS TO BE
                                             SKIPPED ; FLAG GAP AND OVERLAP DATA

     INITAN ..................               INITIALIZE THE ANALYZ FUNCTION

     ANALYZ ..................               DEFINE CHAPTER BOUNDARIES
        CCLNXD ...............               GET NEXT VALID RECORD AND DATA POINT

     OUTPUT ..................               OUTPUT ONE VOLUME OF DATA

        OUVOLN ...............               OUTPUT VOLUME HEADER

           DTUPK                             GIVEN MILLISECONDS FROM THE START OF
              YDMC                           GET CALENDER DATE OF TIME
                                             1977, RETURN THE CALENDAR DATE AND
                                             TIME
```

```
          DTMJS                     GIVEN MILLISECONDS FROM THE START OF
                                    1977, RETURN YEAR, HOUR OF YEAR,
                                    SECOND OF HOUR, AND MILLISECOND OF
                                    SECOND
                                    (HOUR OF YEAR IS DEFINED SUCH THAT
                                     HOUR/24 = DAY OF YEAR, JULIAN)
          UPFEET .............      CALCULATE TAPE FEET WRITTEN
          KMVC                      MOVE BYTES OF DATA
          FWRITE                    WRITE A TAPE RECORD

       OUCHPT ................      OUTPUT A CHAPTER INTRODUCTION

          UPFEET .............      CALCULATE TAPE FEET WRITTEN
          DTMJS                     GIVEN MILLISECONDS FROM THE START OF
                                    1977, RETURN YEAR, HOUR OF YEAR,
                                    SECOND OF HOUR, AND MILLISECOND OF
                                    SECOND
                                    (HOUR OF YEAR IS DEFINED SUCH THAT
                                     HOUR/24 = DAY OF YEAR, JULIAN)
          FWRITE                    WRITE A TAPE RECORD

       OURAW .................      OUTPUT THE RAW DATA VERSE

          COLLT1 .............      COLLECT TYPES 1 & 2 DATA
             COLNXD ..........      GET NEXT VALID RECORD AND DATA POINT
             COLFRS ..........      FIND THE NEXT DATA POINT SOUGHT
                COLNXD .......      GET NEXT VALID RECORD AND DATA POINT
             COLFIL ..........      SUPPLY PADDED DATA

          COLLT3 .............      COLLECT TYPE 3 DATA
             COLNXD ..........      GET NEXT VALID RECORD AND DATA POINT
             COLFRS ..........      FIND THE NEXT DATA POINT SOUGHT
                COLNXD .......      GET NEXT VALID RECORD AND DATA POINT
             COLFIL ..........      SUPPLY PADDED DATA

          UPFEET .............      CALCULATE TAPE FEET WRITTEN
          FWRITE                    WRITE A TAPE RECORD
          KMVC                      MOVE BYTES OF DATA ENMASSE

       OUMSUM ................      OUTPUT THE MAGNETIC SUMMARY VERSE

          UPFEET .............      CALCULATE TAPE FEET WRITTEN
          FWRITE                    WRITE A TAPE RECORD

    FNEXT XXXXXXXXXXXXXXXXXX        LOCATE THE NEXT INPUT DATA TO
                                    BE PROCESSED, IF ANY

       LCW                          CLOSE THE JOB WORK BLOCK
          LREAD/LWRITE              READ/WRITE A LOG BLOCK
          RDIRET                    LOCATE SATELLITE BLOCK OF LOG
                                    BE PROCESSED, IF ANY
          KCLC                      COMPARE BYTES OF DATA
          ABEND                     TERMINATE THE JOB ABNORMALLY

       LCL                          CLOSE THE CURRENT LIBRARY BLOCK
          LREAD/LWRITE              READ/WRITE A LOG BLOCK
          KTIME                     RETURN JOB RUN TIME FROM SYSTEM
          YDMD                      GET CALENDER TIME FORM OF JOB RUN TIME
          KCLC                      COMPARE BYTES OF DATA
          KOC                       LOGICALLY COMPARE BYTES OF DATA

       LOL                          OPEN A LIBRARY BLOCK FOR PROCESSING
          BLDSER                    BUILD VOL-SER OF INPUT LIB TAPE
             INCORE                 CONVERT DATA TO EBCDIC
             KOC                    COMPARE LOGICAL BYTES
          RDIRET                    LOCATE SATELLITE BLOCK OF LOG
          LREAD/LWRITE              READ/WRITE A LOG BLOCK

       LEAVE                        CLOSE TAPE DCB
       MOUNT                        MOUNT A TAPE
       MSG                          PRINT A MESSAGE
       MSGMAC                       PASS PRINT ARGUMENTS
       UNLOAD                       UNLOAD A TAPE
       POSN                         POSITION TO A FILE

    UPTERM ...................      TERMINATE JOB PROCESSES
       LTERM                        TERMINATE LOG PROCESSES
          LREAD/LWRITE              READ/WRITE A LOG BLOCK
             DREAD                  READ A DISK RECORD
             DWRITE                 WRITE A DISK RECORD
             KMVC                   MOVE BYTES OF DATA
          KCLC                      COMPARE BYTES OF DATA
          ABEND                     TERMINATE THE JOB ABNORMALLY
```

```
ROUTINES DESIGNED FOR THE DPTENC PROGRAM ARE INDICATED BY '******'

THE FOLLOWING CSECT/DSECT AREAS ARE USED BY DPTENC :

          ENCMCB/FENCMCB    MAIN PROGRAM COMMON AREA
          LOGDAT/FLOGDAT    COMMON AREA DESCRIBING THE LOG
          DATIB             COMMON AREA DESCRIBING THE DATAPOOL DATA
          VOLIB             COMMON AREA DESCRIBING THE VOLUME HEADER
          CHPIB             COMMON AREA DESCRIBING THE CHAPTER HEADER
          RDATA             COMMON AREA CONTAINING THE CORE IMAGE OF
                            DATAPOOL LIBRARY TAPE DATA


ROUTINES SLIGHTLTY MODIFIED FROM THE ISEE ENCYCLOPEDIA GENERATOR SYSTEM
ARE INDICATED BY 'XXXXXX'

THE FOLLOWING ISEE ENCYCLOPEDIA GENERATOR ASSEMBLER MACROS ARE USED BY
FNEXT, ENCMCB, LOGDAT

                GETDATE, INVOKE, MSGMAC, GLOBAL, LOPROLOG

THE FOLLOWING ISEE ENCYCLOPEDIA GENERATOR ROUTINES ARE USED IN THE
DPTENC SYSTEM :

           LOG ROUTINES :
               LOL, LCL, LOW, LCW, BLDSER, RDIRET, EOPROL,
               LREAD/LWRITE, LTERM

           OTHER ROUTINES :
               CTLPK/CTMJS, MSG


THE FOLLOWING SACC SYSTEM ROUTINES ARE USED :

               KTIME, YDMD, KCLC, KMVC, KOC, FTIO ROUTINES,
               DAIC ROUTINES, ABEND, INCORE, PARM
```

Attachment I.

a.

Eunice Eng
11/18/82

ISEE DATA POOL ENCYCLOPEDIA GENERATOR
A SOFTWARE SYSTEM SPECIFICATION

OVERVIEW.

The object of this software system is to process, in time order, high density (6250 bpi), library tape files Each file of the library tape is a copy of the data pool tape. The library tape files are not in time order.

The output, 'disposable', single-filed work tape will be the input to the next data processing step, the encyclopedia merge. This encyclopedia generator system will not merge data over library file boundaries. If data overlap occurs within a library tape file, the encyclopedia generator will have to handle intra-file data overlaps.

Involved in the ISEE data pool tape processing is an automatic processing log (DP Log) which keeps track of the processing status of the data pool information. Since the ISEE DP Log is identical, in structure, to the ISEE Experimentor Data Record Automatic Processing Log (EDR Log), this system could use the EDR Log interface routines whenever possible.

FUNCTION:

```
Call an EDR Log interface routine to get the first (the earliest)
  DP library file to process (Subroutine LOL)
  IF (no DP library file marked for processing)
     Write a message
     STOP
  ELSE (DP library file marked for processing)
   Mount an output work tape
   IF (no output work tape available)
     Write message requesting user action
     STOP
   END IF
```

```
DO UNTIL ((maximum number of library files are processed) or
  (no more library files to process, for this run))
  note:  This system will not pick-up files with start
         times before the end time of the last library
         file processed.  The skipped files will have
         to be picked up in the next run.
  Get library file to be processed
    IF (library tape mounted)
      Position library tape to file to be processed
    ELSE (library tape not mounted)
      Mount the library tape
      Position library tape to file to be processed
    End IF
  Process a library tape file to a single filed,
    time ordered work tape.  A sixty-four minute
    input library file record will be processed into
    4. fifteen minute work records.
```

```
    TO BE DETERMINED.
    1.   Output Format.
    2.   Number of output records for each 15-minute record
         opt. 1.   15-minute raw data followed by 15-minute
                   summary data (two records per 15-minutes)
         opt. 2.   15-minute header followed by a 15-minute
                   raw data record followed by a 15-minute
                   summary data record (three records per
                   15 minutes)
         opt. 3.   a record of 15-minute raw data and
                   15 minute summary data (one record per
                   15 minutes)
         opt. 4.   a 15-minute volume header followed by
                   rates data records, one data record
                   per datatype (multiple, but a fixed
                   amount of records for every
                   15 minutes).  This may involve very
                   small records
         opt. 5.   other
    3.   How to Handle Time Lines and Time Line Changes
         opt. 1.   Give the time line a black box treatment.
                   i.e. pull a subroutine out of the existing
                   DP Magnetic Summary program
                   COMMENT: (PAS) It would be difficult
                   to extract a time line subroutine.
         opt. 2.   Define algorithms to handle time line,
                   time line changes and the associated
                   data points.

         opt. 3.   other.
```

4. Define Padded Data.
The documentation defines padded data as a negative value outside the range of the data type. Therefore a valid range for each data type must be defined or padded data must be determined for each data type. When would the programmer need to know about padded data?
    1. to handle data during time line changes
    2. to derive summary data
5. How to Handle Questionable data, with respect to summary data.
6. How to Derive Summary data
opt. 1. black box treatment by extracting a subroutine from the DP Magnetic summary program
COMMENT: (PAS) It would be, again, difficult to extract a subroutine.
opt. 2. define an algorithm for each data type
opt. 3. other.
* 7. Start record time and event time problem.
If each output record contains the start time of the 15-minute interval, the time of the first event of the 64 minute interval is lost. Also, in a 15-minute interval, all the data types could have differing times for their first event.

DECISIONS:

1. To help-facilitate the merge process, there should be a data quality flag:
a. a count of all the non-padded and non-questionable data and, perhaps,
b. a count of all the questionable data
in the 15-minute interval. There should also be a maximum count of data points per 15-minute interval, since all 15-minute records will not have the same amount of data due to the 64-second minute.

2. Each datatype should have associated with it
   a. a count of the number of data points for the 15-minute interval,
   b. a time associated with the first data point other than the start of the 15-minute interval and,
   c. the interval, in seconds, between data points.

   This is necessary. The time of the first data point is different for each datatype. The timeing information must be preserved to allow for summaries to be taken over intervals other than the raw rate level and the 15-minute leve.

```
    At End of Library file
      Temporarily close library block of DP Log using an EDR
        Log interface routine (Subroutine LCL)
      Temporarily close the work block of the DP Log using an
        EDR Log interface routine (Subroutine LCW)
      Get next, time sequential, library file to process using
        an EDR Log interface routine (Subroutine LOL)
  End DO
  Permanently close the DP Log library and work blocks using
    an EDR Log interface routine (Subroutine LTERM)
  Dismount magnetic tapes
  Close DP Log
 STOP
```

Performance:
None. Since this system involves magnetic tapes, the SAIC requires that it be run as a background job. Under normal conditions, a job should run within twenty-four hours.

Constraints.
Computer processing jobs that change the DP Log can not run concurrently.

Whenever possible, limit the number of library tape mounts.

Acceptance Test Considerations.

Tests will be run on actual data. The tester should check for the following

1. how the system performs when there are more than the allowed amount of library files to process.
   a) check the order in which the library files were processed. i.e. check that files were not picked with overlapping data at the file boundaries.
2. how the system performs when there are no files to process
3. look for any overlaps within a library tape file
4. correct data was included in the fifteen minute record. Look at more than fifteen output records at a multiple of fifteen and the associated input records

As a learning experience, run BOOLE and BABBAGE with the trace option, if available. The user, and designer could learn to estimate computer response times and how much time each routine uses.

n. lal

n. laubental
p. schuster
t. von rosenvinge

Eunice Eng
11/19/82

ISEE DATA POOL MERGE ENCYCLOPEDIA
A SOFTWARE SYSTEM SPECIFICATION

OVERVIEW.

The object of this software system is to merge data from one work tape created by the encyclopedia generator program and an encyclopedia tape from the encyclopedia data base.

A work tape is an analyzed, time ordered, intermediate and temproary data base of data with like attributes. The work tapes are single filed and of density 6250 BPI.

The encyclopedia tapes are a time ordered, permanent data base of analyzed data of the same attributes. There may be as many encyclopedia data bases as there are declared data attributes in the Data Pool Automatic Log (DP Log).

The DP Log will keep track of the work tapes, encyclopedia tapes, the encyclopedia data base(s) and the merge processing.

FUNCTION.

```
Quiz the DP Log to see if there are any merge jobs to perform
  IF (no merge jobs outstanding)
    Write Message
    STOP
  ELSE (one merge job to do)
    Quiz the DP Log for next available encyclopedia output
      tape
    IF (no encyclopedia tapes available)
      Write a message requesting operator action
      STOP
    ELSE (encyclopedia output tape available)
      Mount output encyclopedia tape
      Mount input work tape
      Mark the DP Log's work block process begun
      Quiz the DP Log to get appropriate input
        encyclopedia tape of same attribute
```

```
        IF (no input encyclopedia tape)
          Copy work tape to output encyclopedia tape
          Update DP Log's new encyclopedia block with
            necessary infromation
          Update DP Log's work block with necessary information
        ELSE (input encyclopedia tape to be merged)
          DO UNTIL (all data merged)
            DO CASE
              1.   IF (missing data on encyclopedia tape)
                   DO UNTIL (overlap found .OR. end of work
                     tape)
                     Copy work tape in 15-minute chunks
                   End DO
              2.   IF (missing data on work tape)
                   DO UNTIL (overlap found .OR. end of
                     encyclopedia tape)
                     Copy encyclopedia tape in 15-minute chunks
              3.   IF (overlapping data)
                   DO UNTIL (no more overlap .OR. end of work
                     tape .OR. end of encyclopedia tape)
                     Copy onto output encyclopedia tape the
                       15-minute chunk with the better data
                       quality
                   End DO
            End DO
            IF (end out volume on the output encyclopedia
              tape)
              Update the DP Log's current new encyclopedia
                block
              Open a new DP Log new encyclopedia block
            End IF
          End DO CASE
          Update DP Log's new encyclopedia block
          Update DP Log's old encyclopedia block
          Update DP Log's work block
        End IF
    End IF
  End IF
STOP
```

DESIGN CONSIDERATION.

The ISEE EDR Merge system and the EDR Log interface routines are written in Assembly Language (ASM). The EDR Log interface routines are not FORTRAN callable. Therefore, the EDR Log routines may have to be modified to make the FORTRAN compatible.

PERFORMANCE.

None. Since this system involves magnetic tapes, the SACC requires that it be run as a background job. Under normal conditions, a job should run within twenty-four hours.

CONSTRAINTS.

Computer processing jobs that change the DP Log can not run concurrently.

ACCEPTANCE TEST CONSIDERATIONS.

The tests will be run on actual data. Besides checking to see that the resultant new encyclopedia tape(s) has time ordered data of the same attribute, the tester should check for the following:

1. how the system handles the situation when there are no input encyclopedia tapes,
2. how the system handles the situation when there are no work tapes to merge,
3. how the system handles overlap data, check to see that the volume with the better data quality was kept,
4. how the system handles end of volume on an input encyclopedia tape, and
5. how the system handles an incomplete or interrupted merge process. Hopefully, in the case of an incomplete or interrupted merge process, the work block is still marked for processing or process begun but not completed, the new DP Log encyclopedia blocks are lost (i.e. they are not linked to previously existing encyclopedia blocks), and the old DP Log encyclopedia block is still marked active.

As a learning experience, run BOOLE and BABBAGE with the trace option, if available. The user and designer could learn to estimate computer response times and how much time each routine uses.

n. lal
n. laubental
p. schuster
t. von rosenvinge

C.

Eunice Eng
11/19/82

ISEE DATA POOL FORMATTED ENCYCLOPEDIA TAPE DUMP
A SOFTWARE PROGRAM SPECIFICATION

OVERVIEW.

This routine will produce a formatted dump of a Data Pool (DP) processing work tape or a DP encyclopedia tape. Both the DP work tape and the DP encyclopedia tape have the same data format, same tape density and both are single filed.

This program will dump only one tape per run. The user will specify the tape volume-serial number. The user will specify which span of records are to be dumped by one of two methods:

1. the first and last record volume number

   or

2. the first and last date and time.

The user may specify which datatypes are to be dumped. The user may also want to have the option of a hexidecimal dump.

The encyclopedia data format has not yet been defined.

FUNCTION.

Given the 1) tape volume-serial number, 2) the span of data records to be dumped, 3) the datatypes to dump and 4) the hexidecimal dump option, the program will give a formatted and/or hexidecimal dump of the specified datatypes within the span of records given.

The program is to give a return code of 0 only if the program is completely successful.

PERFORMANCE.

The real numbers are to be written in exponential form with a precision of _______ decimal places.

CONSTRAINTS.

The viewer of the output listing should be able to quickly distinguish one 15-minute interval of data from another. Within the 15-minute interval, the header(s) and datatypes should also be easily distinguished.

The output should conserve as much paper as possible.

???????? QUESTION. Would the user like to limit the output to 80 columns to be better viewed on a CRT?

ACCEPTANCE TESTING.

The tester could use a system utility program PATRICK to compare the tape dump against a PATRICK hexidecimal dump.

n. lal

n. laubenthal

[illegible]. schuster

[illegible]. von rosenvinge

Time Estimates for a new
ISEE Datapool Encyclopedia
database system

Pam Schuster, CSC

and

Preliminary specifications for the
DATAPOOL ENCGEN, ENCMRG and
LSELECT programs

Eunice Eng, GSFC

Preliminary
Draft

November 19, 1982

CSC was asked to make time estimates for the design, coding + testing of programs required to generate a new ISEE Datapool encyclopedia database.

That processing system, as it would follow the existing Voyager / ISEE ENCYGEN system, is illustrated in Figure 1. The current DATAPOOL PROCESSING SYSTEM is shown in Figure 2.

Comparison of Figures 1 and 2 shows that:

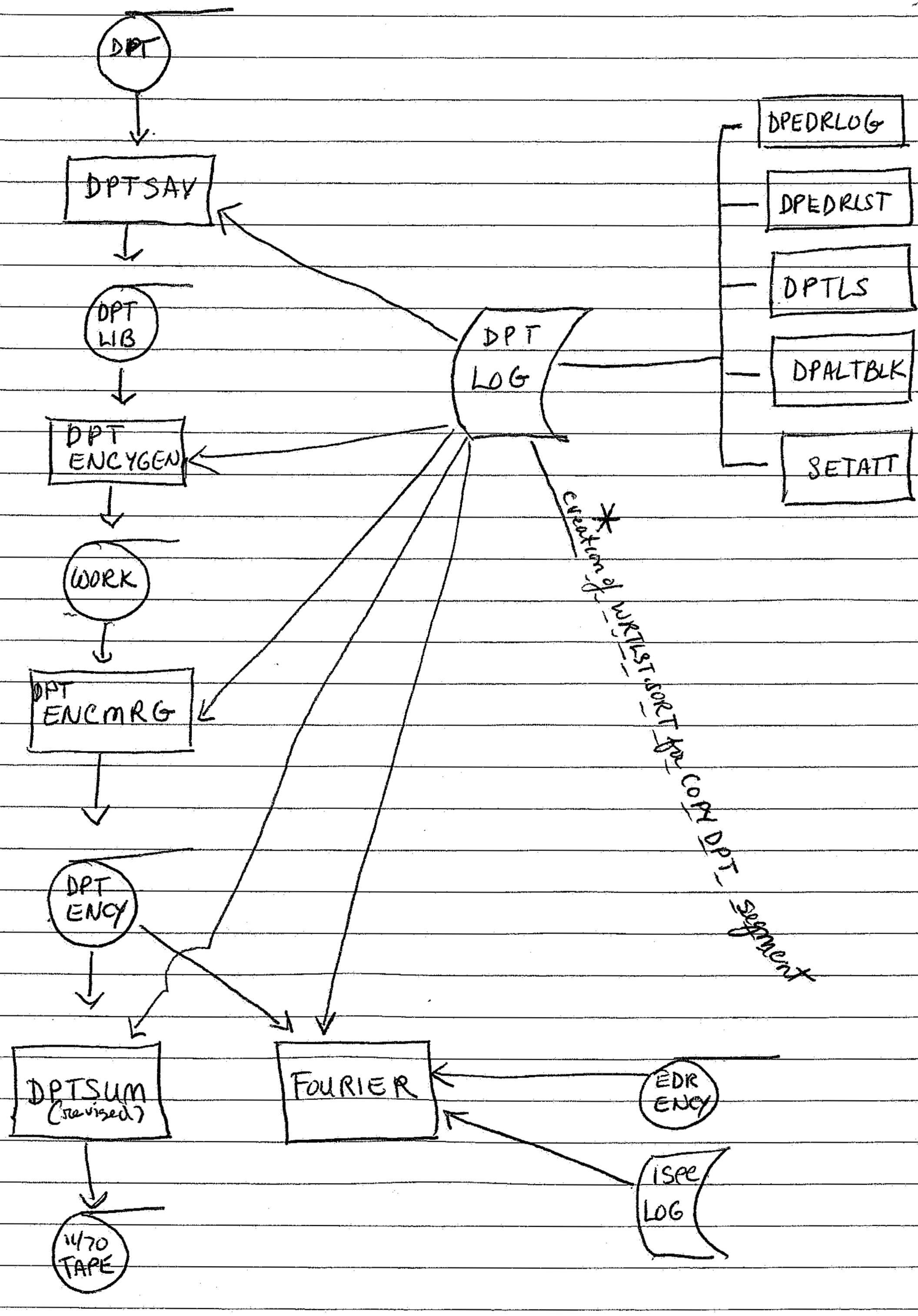
1) The WRTLST.SORT function of the current system would be replaced with the use of the Datapool LOG in creating the ENCY database. (See Appendix B for an explanation.)

2) PROGRAMS DPT ENCGEN, DPT ENCMRG, and DPT LSELECT need to be coded. The ENCGEN and ENCMRG programs would access the LOG in their operation. The LSELECT program would not have to access the log, unless desired.

3) The DPTSUM program in existence would need to be mostly rewritten - at least 80% rewritten. It would need to access the new ENCY datarecord format. The data selection and timing segments would be entirely different (i.e. putting together summary intervals of perhaps 15 minutes, or putting together raw data type verses to make 64 minute summaries, or perhaps generalized ~~time~~ summaries, instead of simply processing one 64 minute DPT record at a time as it currently does.)

4) The FOURIER program would make or obtain whatever magnetic field time summaries it required from the ENCY tapes, replacing the current MAGSUM program function.

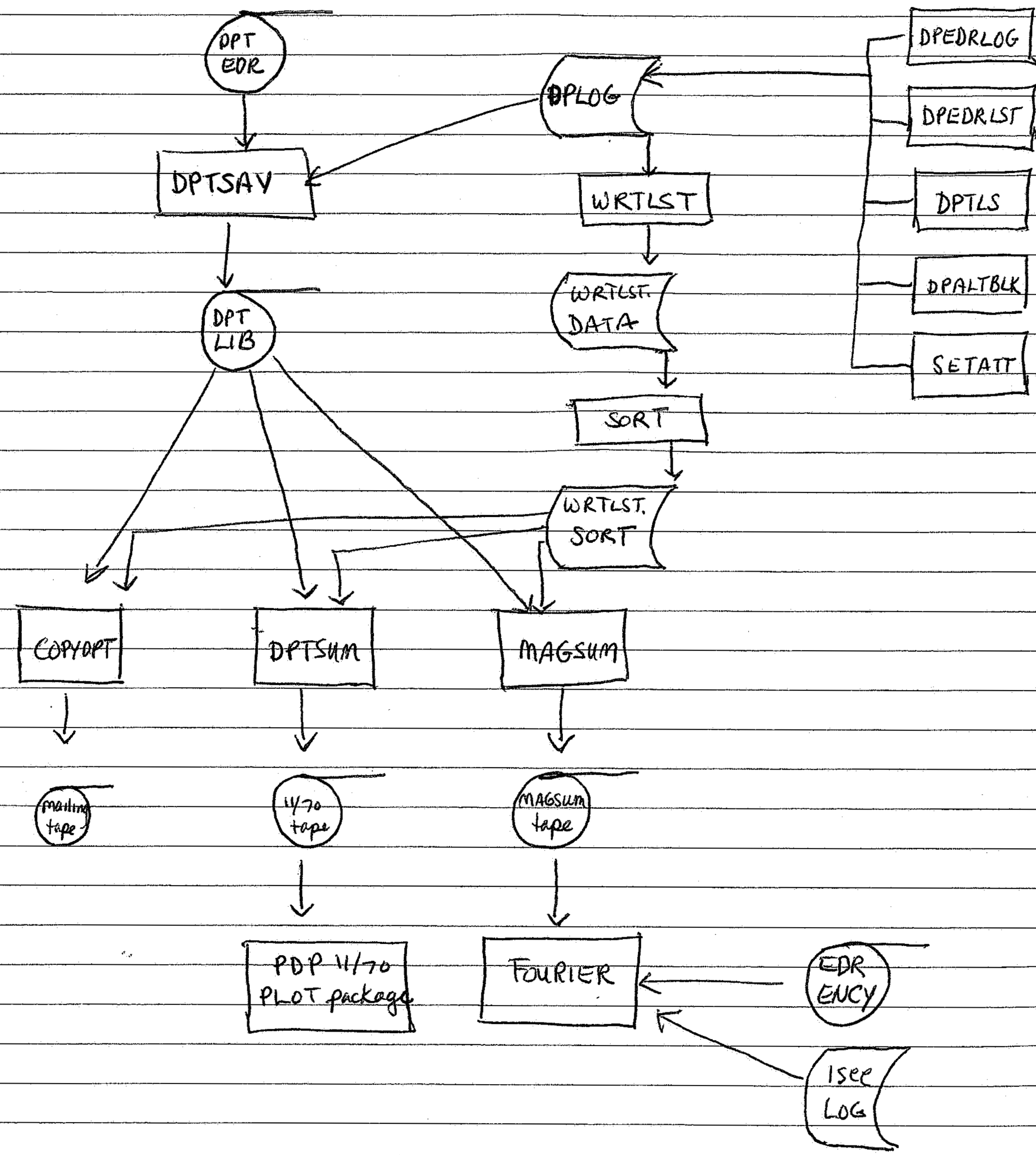
DESIRED DATAPOOL PROCESSING SYSTEM



An LSELECT type listing program for the ENCY database is needed

* The COPYDPT program would operate, as in the current DATA POOL Processing system, through the WRTLST.SORT dataset and SETATT program software.

Current DATAPOOL PROCESSING SYSTEM



utility programs available:

DPTLIST formatted listing of DPT EDR or LIB tapes
MSLIST formatted listing of MAGSUM output tapes
various foreground executing CLISTS to assist with some processing applications

From experience with other cosmic ray satellite system programs one can estimate certain generally expected program sizes depending on the degree of complexity in the data processing.

For the DATAPOOL data, two factors will determine the relative complexity of the coding for the new system:

a) the inclusion and proper handling of timeline changes.
Data records which begin a new timeline are flagged in that record; that data record number along with the record start time are stored in the DATAPOOL tape file header.
DATAPOOL file header timeline information can be used along with the start time of the record just before the timeline change to establish how much of the latter record is valid data (in time).

b) the kind of data summarizing, if any, required for each of the nine experimenter quantities.
Time handling may not be the same for all quantities. Some occur 64, 30, 12, 4, 3, etc times per 64 minute DATAPOOL tape (DPT) record, depending on the frequency of readout. We need to know how to handle quantities which are read out every 16 minutes, or 21 1/3 minutes, etc. especially if a 'raw rates' type verse is to include all data types within the 15 minute volume context.
The start time of the data record is the start time of sampling interval number 1 for all frequencies of readout. The start times of subsequent intervals are computed relative to interval 1, with the interval duration equal to about 64 minutes / N where N is the number of readouts per record.
[There is an error of about +2 seconds by the end of the 64 minute record. i.e. for magnetic field data Start time of first record + (60 entries * 64 seconds per entry)
= Start time of next record + ~2 seconds]

For data which is later replaced in updated DPT tapes, merging new data or parts of previously missing data may be somewhat complicated if the record start times are much different from the prior data start times

Given these observations, and our knowledge of the size of the existing DPTSUM and MAGSUM programs and their limitations, time estimates were made for completion of the work mentioned in points 2) and 3) above. They are given in Table 1. Appendix A gives a rationale for those estimates. CSC assumes these programs are FORTRAN programs, and that LOG access routines for all phases of the work are available in black box form.

Table 1. here

Table 1. Data Pool System Program Completion Estimates exclusive of work required on log access routines

	approximate upper limit		approximate lower limit	
	new lines linecount	÷10/day	new lines linecount	÷10/day
DPTENC	395*	40 days	245*	25 days
DPTMRG	200*	30 days	200* (minimum # of data quantities)	20 days
ENCYLIST	200	20 days	200 (minimum # of data quantities)	20 days
		90 days		65 days
	new lines linecount			
DPTSUM (essentially new)	316	32 days		

* Fortran programs, exclusive of log interface subroutines

CSC believes that the upper limit is closer to the time required to complete the ENCYCLOPEDIA production system. We believe 32 days is a maximum requirement for a new DPTSUM. Time estimates include a basic level of documentation which does not currently exist - i.e. a system description and user gides, as well as internal code PDL and Prologues.

ENCYLIST programming days would be a Junior Programmer level for the most part.

Attachment I, supplied by the government, is a preliminary summary of the proposed DPT ENCYGEN system program flow. It lists some undefined system variables which would affect the relative complexity of the code for all programs, and thus the time estimates in Table 1, but probably not by more than +10 to 15%, with the possible exception of the following:

There is a problem with the log access routines which the MERGE program would require. These access routines exist, but they are in ASSEMBLER language and are not FORTRAN callable. The government should decide how these routines are to be provided:

1) do we duplicate the corresponding routines in FORTRAN? Functions

Functions	MOW	open work block
	MOE	open ency block
	MCW	close work block
	MCE	close ency block
	MTERM	close log

2) do we write a Fortran callable ASSEMBLER interface to the existing log routines?

3) do we write the merge program in ASSEMBLER?

Typical log interface routines for the ISEE Encyclopedia generator (ENCGEN), which are all in FORTRAN, are about 80 executable statements each.

The desired ENCY record format needs to be supplied by the government, as well as the type of summarizing done and the kind of 'raw rates' quantities to be put on the ENCY tape.

A method of accepting merging data needs to be decided upon. The government must decide what constitutes better data.

These questions are the principle unanswered questions at the time of this writing. Programming estimates provided are somewhat dependent on the answers. LOG ACCESS routine development time estimates should be considered separately from this work

Appendix A. Rationale for programming time estimates

The proposed DPT ENCGEN program would need to perform similar functions to the current DPTSUM and MAGSUM programs. Both of those programs are limited in their processing capabilities:

1) DPTSUM summarizes one record at a time as an entity. MAGSUM summarizes each 15 minutes of magnetic field data including some BAME data. Neither program can be easily generalized to make other time period summaries, although MAGSUM could be changed to produce 64 second summaries, and DPTSUM could get multiples of 64 minute summaries.
2) Neither DPTSUM or MAGSUM is currently coded to process timeline changes. Each and every record is assumed to contain 64 minutes of contiguous data. Timeline changes will generally result in records whose start times are before the projected 64 minute end time of the previous record. The entire first timeline change record is treated as if it were overlap data, which is bypassed. Effectively, a false datagap is created.

3) Both programs are limited in the way they must be used to handle replacement or insertion data. See Appendix B for an explanation.

CSC expects the new DPT ENCGEN program to be more complex than either DPTSUM or MAGSUM. Table 2 gives the basic functions within those programs along with the executable code linecount needed to perform the functions.

Table 2.

Current DPTSUM functions and their executable code linecounts

DPTSUM has a total of about 270 executable lines of code as follows:

function		approximate linecount
initialization and input processing		32
data location using the WRTLST.SORT dataset		20
establish times		20
summarize data	8 types @ ~17 lines each	136
	1 (smth ?)	30
write data records	(12 data + 20 header)	32
end of job		10
		~280

Current MAGSUM functions and there executable code linecounts

MAGSUM has 320 executable lines as follows:

function		approximate linecount
initialization and input processing		32
Copy tape option		25
data location + start processing using WRTLST.SORT		30
establish times for data summaries		50
summarize magnetic data		100
end of files, outputting		20
		257
subroutine LIBMOU	mount data tapes	40
subroutine DYNMBR	time conversion	15
		~312

To estimate the DPT ENCGEN program size:

1. all basic MAGSUM functions would be necessary except the COPYTAPE and WRTLST, SORT type functions

2. Additional code to take care of timeline changes needs to be added - about 80 lines

3. code to search and correct for year changes within one DPT record needs to be added; DPLOG interface code needs to be added - about 50 lines total

4. Any other DPT data quantities desired must be allowed for, along with the creation of the 'raw rates' type verse(s). - about 150 lines if all other 8 quantities are treated in some fashion.

Estimated lines for the new program might then fall between two approximate limits;

upper: $320 - (25+30) + 80 + 50 + 150 = 545$ lines

lower: $320 - (25+30) + 80 + 50 = 395$ lines

OF the existing MAGSUM code, if the magnetic summarizing segment were used it would need to be redone to conform to the ENCY output record format. About 100 lines of MAGSUM might be used, but redone in that way.
The LIBMOU and DYNMBR subroutines might be used, if the DPT ENCGEN structure allows it - about 55 lines total.

Subtracting these linecounts from the estimated totals gives $545 - 150 = 395$ new lines (upper limit) and $395 - 150 = 245$ new lines (lower limit).

Using the CSC estimate of 10 lines per day for new code with documentation (for a non-complex system) gives about 25 to 40 days for the two limits. CSC believes the upper limit is closer to the time that will be required.

For the DPT ENGMRG program time estimate we believe the program size will be comparable to the existing MAGSUM program also. The following general functions will need to be done:

initialization and input	30 lines
locate and mount tapes	30 lines
read input records compare data (use pad as an indicator? compare each time for all quantities? or simply replace old with new?)	> 100 lines
write output and update log	> 20 lines
	>180 lines exclusive of log interface subroutines

Different merge code may be required for the nine different data quantities. We might add 8 * ~15 lines = 120 lines or so more to the above estimate.

The expected upper and lower limits for the merge program would then be between 200 and 300 executable statements. Completion time would range from 20 (lower limit) to 30 (upper limit) days.

For the ENCYLIST program (LSELECT type) time estimate we note the following:

The current DPTLIST program has 200 executable Statements
The current MSLIST program has 75 executable Statements.

Table 3 summarizes these programs in their bases for two

Table 3.

Current DPTLIST functions and their executable code linecounts.

DPTLIST has a total of 202 executable lines of code as follows:

function	approximate linecount
initialization and input	20
select + mount tape	20
write file header	10
locate first requested list time	20
establish data times, check for year change	20
write start of data record information	12
write individual data quantities ~10 lines for 10 quantities	= 100
maneuver information + Hovestadt, Smith, Sternberg, Anderson, Bame, Scarf, Von Rosenvinge, deFeiter, Meyer	
	202 total Statement

Current MSLIST functions and their executable code linecounts

MSLIST has a total of 75 executable lines of code as follows:

function	approximate linecount
initialization and input	17
mount tape, get requested start time + end time	7
read data, get data times, look for 1st time request	15
write out data - 2 options possible	35
	~74

With the executable line count needed for each function, neither program is well formatted, although DPTLIST is better.
The ENCYLIST program would have to be able to do both the MSLIST and the DPTLIST functions entirely, if all data quantities are put onto the ENCY tape. Therefore we expect the new program to be at least 200 statements long. Completion time would be about 20 days.

The DPTSUM program would need to be about 80% rewritten because of the different expected ENCY record format and timing considerations (putting together into 64 minute summaries (or a generalized time summary) data from 15 minute summaries or raw rates type verses). LOG routine access code would need to be added too. The revised program would be at least as large as the current DPTSUM program and perhaps as much as 80 - 100 Statements larger, to account for the above differences. Assuming then $(.80 \times 270) + 100 = 316$ statements as an expected size we would predict $316 / 10 = 32$ days to redo DPTSUM, assuming the current DPTSUM philosophy of operation is preserved. notes Encyclopedia blocks within the LOG would have to be searched by desired volume number.

Appendix B. Merge and Replacement data handling limitations in the current DATAPOOL processing system.

To merge or replace data generated from DPTSUM, routines on the PDP 11/70 must be used. There is no automatic process for inserting data into an existing large DATAPOOL file, although add-on data handling is straight forward. Inserting/replacing data involves splitting the existing file and then combining the pieces with the new data. Dr. R McGuires program FLXMAN is used. He has considered the problem of automatic insertion, but it is not implemented in FLXMAN. Therefore, whenever replacement DPT data arrives, DPTSUM must regenerate the entire 64 minute summary database, and an entirely new PDP 11/70 file should replace any existing file.

The MAGSUM program output database has a somewhat similar problem of handling replacement or insertion data. The program user must keep track, exactly, of what data is located in each file of the multi-filed output tape. Replacement/Insertion of data must be done on a file basis. The user must copy existing files, up to the time of the new file, onto a new tape, then copy the new MAGSUM file, and then copy any remaining 'old' files onto the new permanent MAGSUM tape. In addition an exact correspondence must be maintained between the entries of the WRTLST.SORT dataset (which contains a time-ordered list of library tape files and is used by DPTSUM and MAGSUM to determine file order of processing.) and the files of the output tape. Inserting previously non-existant files of data would require re-doing all the database beyond the insertion point, in order to maintain the 1:1 correspondence.

Clarifying ~~Special~~ Notes on the DPT Library tape data base and DPT data for READER function

Time-Line Changes

When a time-line change occurs a new data record is started in the DPT input file (new record time-line flag = 7).

That means that only part of the data in the previous record (the one just before the timeline change record) is valid data. The rest of the words in that record are padded to the length of 810 words.

Since the start time of the time-line change is noted in the file header information, the extent of valid data in the previous record can be calculated:

previous record end time =

(previous record start time) - (time-line record start time)

If this end time is greater than 64 minutes, a gap has occurred. If it is less than 64 minutes, assume that valid data exists in the previous record up to the calculated end time.

Several records in a row may contain a time-line change

Reading in input records

~~Keep~~ Put all records in core. This is because of time-line checking needs. (E.g. do not lose 1st record of time-line because of projected (calculated) overlap; or do not lose any (n-1) type records)

Check for year changes, both in time-line checking and in normal 'settim' processing.

dimension FILHDR(810), TIMLNS (6,80)
equivalence (FILHDR(127), TIMLNS(1,1))

# of timelines is in word 126 of fileheader info

Check for year change in timeline segment; summary info is day, sec of day only

Stop time of data record = Start time + (60 * 64) seconds unless there is a timeline correction

Allow 4 seconds overlap between stop of previous + start of next. This is not an actual overlap, but only a calculated effect that comes from assuming 64 second readouts for the magnetic field data.

dimension FILHDR(810), TIMLNS(6,80)
equivalence (FILHDR(127), TIMLNS(1,1))

# of timelines is in word 126 of fileheader info

Check for year change in timeline segment; summary info is day, sec of day only

Stop time of data record = Start time + (60 * 64) seconds unless there is a timeline correction

Allow 4 seconds overlap between stop of previous + start of next. This is not an actual overlap, but only a calculated effect that comes from assuming 64 second readouts for the magnetic field data.

If timeline occurs just before year change, (within 64 words)

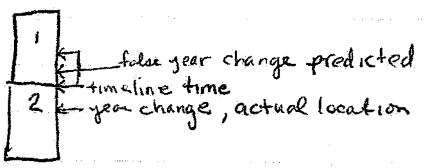
year change must be in timeline record itself because they begin a new record at a timeline

That implies that records 1 + 2 must have the same start day, unless there is a large gap just before start of record 2

If the timeline change occurs just right at, or after a year change records 1 + 2 will have different start days

(A) cases

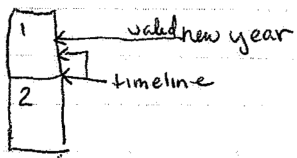
contiguous data



start time, however if only start times are being calculated present code could handle OK, placing year in record 2 (↑ in any verification attempt)

1 + 2 have same start day

(B) contiguous data — year change in (m-1) record



present code will get year change OK; year times will be calculated OK

1 + 2 have different start days

(m-1) record case B only

gap between 1 + 2 for either A or B should be O.K.

DATAPOOL ENCYGEN SYSTEM

MAIN PROGRAM : DPTENC

Purpose: main driver for program
check return codes from various program segments
& determine if processing continues

Common ENCMCB

Subroutines called:

RTIME	retrieve system execution clock time
YDMD	time → month day year time
INIT.	search for data, mount first tapes
READER	read in data
INITAN	initialize analyze function
ANALYZ	determine data boundaries
OUTPUT	output data to tape
FNEXT	search for next input data
	close out program functions

interface variables

variable	location	type	I/O	description
ENCDAT	ENCMCB		I	data available flag
ISTAT	local		I/O	=8 terminate current volume # =12 terminate current file processing
ITERM	local		I/O	terminate job if = yes
ENEXDT	ENCMCB		see below	

Code for time retrieval

```
Call RTIME (ENCGYR, ENCGDOY, ENCGHR, ENCGMIN, ENCGSECH)
Call YDMD (ENCGYR, ENCGDOY, ENCGMON, ENCGDAY)
        { ENEXDT(1)  =  1900 + ENCGYR
ENCMCB— {   "   (2)  =  ENCGMON
        {   "   (3)  =  ENCGDAY
```

DPTENC

```
Set ITERM to NO
Set ISTAT to zero
Retrieve job run time ; set ENCMCB time variables ENEXDT
Write job run time message
Call INIT to search for data and mount tapes
IF    ENCDAT = yes (1) and ENRTCODE is zero [FROM INIT]
      Do while ENCDAT is yes
            Call Reader to get input data
            IF ENERROR is zero and ISTAT < 12 and
                        ITERM is no (=0) [From READER]
                  Call INITAN to prepare for ANALYZ function
                  IF there is valid data (ISTAT < 12)
                        Do until all volumes processed
                              and if ISTAT < 12
                              Set ISTAT to zero
                              Call ANALYZ to define chapters
                              IF   IStat ≠ 12 (not EOF)
                                    IF IStat ≠ 8 (empty volume)
                                          Call Output
                                    ELSE (get next volume)
                                    FI
                                    Get next volume
                                          increment volume #
                                          incr. Start time
                                          incr. Stop time
                              ELSE
                                    (ISTAT = 12 ; fall thru to FNEXT)
                              FI
                        OD  all volumes processed
                  ELSE
                        Print INITAN fail message
                  FI
            ELSE
                  Print Reader error message
            FI
            IF   ITERM is not yes (1)
                  Call FNEXT to get next input
                  Set ISTAT to zero
            FI
      OD
      Call DPTERM to close log, dismount tape
ELSE
      Print message indicating no processing was done
```

SUBROUTINE INIT

Purpose — data to process is searched for
Initial input and output tapes are mounted
Starting job messages are written

Called by — main program, once per job

Common — ENCMCB — required for communication with FNEXT log search routines used later in the program

Special data initializations — Local variables ATTRIB and STIME, used in the call to LOL, must be set to zero here before the call to LOL.

references needed — LOG Documentation on LIBRARY and WORK BLOCK formats
page included showing LOL and LOW calling sequence

Start INIT
Call LOL to search for any data available for processing
IF return code from LOL is zero
Set ~~the~~ data available flag in ENCMCB, ENCDAT, to YES = T
ELSE
set ENCDAT to NO = F
Write a message so stating
FI
IF data is available to process
write job startup messages (program, ENCMCB version, time)
Obtain vol-ser, DSN, filseq, etc from ENLIBBLK area of ENCMCB
Call mount for LIBRARY input tape + write message
FI
IF data is available to process
Call LOW to open a WORK BLOCK for output tape
IF successful (retcode ≠ 8)
Obtain vol-ser, DSN, filseq from ENWORKBLK area of ENCMCB
Call Mount for WORK output tape + write message
Else
Print message ~~so~~ indicating error status from LOW
FI
FI
END INIT

additional:
input unit # is defined in ENCMCB = ENEUNIT
output unit # needs to be added to ENCMCB, replacing ENENEKIT and ENJFCB areas

The FORTRAN common ENCMCB names have yet to be made.

over for LOW + LOL calling sequences, errors

Call LOL (ATTRIB, STIME, ENATTRIB, ENLIBBLK, ENRTCODE)

return code
- 0 = found data to process
- 4 = no data marked to process
- 8 = error

ATTRIB	Local	I*4	I	1
STIME	Local	R*8	I	1
ENATTRIB	ENCMCB	I*2	O	32
ENLIBBLK	"	I*2	O	32
ENRTCODE	"	I*4	O	1

Call LOW (ENATTRIB, ENWRKBLK, ENRTCODE)

return code:
- = 0 normal
- = 4 no new block allocated
- = 8 error

ENATTRIB	ENCMCB	I*2	I	32
ENWRKBLK	"	I*2	O	32
ENRTCODE	"	I*4	O	1

↑ variable name

↑ Location

↑ length

↑ input output

↑ # of array locations if an array

DATAPOOL ENCYGEN SYSTEM

Preliminary

SUBROUTINE READER

Purpose: Reader reads records from the input LIBRARY tape file. It makes a modified core image of an entire weeks (= 1 LIB tape file = 1 LIB BLOCK entry) data.
Reader checks for time-line changes and overlap/gap data and sets flags accordingly.
Reader counts the number of data records in the LIBRARY tape file.

Called by: Main program
Reader is called once for each LOG LIBRARY BLOCK which log routines find available for processing. (The Log system has an internal limitation of processing no more than 15 LIBRARY BLOCKS per job.)

Common: ENCMCB — ENCGEN program subsystems main interface
DATA (810, 170) — allows 170 64minute records per week

Subroutines called:

FREAD	FTIO	read input tape records
FMOVE	system	move bytes en-masse
SETTIM		given year, day, sec, return seconds from start of 1977
TIMCHK		process time-line changes + time check

necessary input variables *

ENCDAT	in ENCMCB	data available flag
ENEUNIT	"	input unit # for tape

interface output variables

(SCIREC) ENCSEBR	ENCMCB	# of data records in file
ENSMH1	"	Smith data
ENSMH2	"	↓
ENSMH3	"	↓
ENSMH4	"	↓
~~ISTAT~~	~~argument~~	~~return code from READER~~
FSTAT	argument	Stop code for job

* add to ENCMCB a variable name for the output tape unit #

DPT ENCYGEN SYSTEM

READER (continued)

FIXED location variables which READER section must set:

For each input data record / core image

	word	type	description
	73	~~I*4~~	Seconds from 1977, start of record
an/ou	74	~~I*4~~	" " " start of data
	77	~~I*4~~ Real*4	" " " end of record
an/ou	78	~~I*4~~	" " " end of data
	75	~~I*4~~	year of start time

Start of record should equal start of data unless some unusual data condition is found
End of record may not equal end of data, and should only be true if a time-line change has occurred, unless some unusual data condition is found

	word	type	description
	62	I*4	=1 this record is the one just before a timeline change
			=0 normal within a timeline
	65	I*4	=0 normal time continuity
			=1 gap is indicated
			=2 overlap is indicated
AN/OU	68	I*4	=0 normal acceptance of record for processing
			=16 skip this record
	70	I*4	=0 normal record
			= Values to be determined, if used. miscellaneous classification of data condition.

References Data Pool Tape format

accompanying comments for READER processing

DPT ENCYGEN SYSTEM

READER (return code, Stop code)

```
IF data is available
    Set return code to zero
    Read file header of Input
        IF  I/O error
                Set return code to 16  (ISTAT)
                Write message
        ELSE
                IF  EOF   (would assume EOV here)
                    set return code to 12 (ISTAT)
                    Write message
                FI
        FI
    IF return code is zero
        Transfer header info to ENCMCB as required
        do header year change check (see attached page)
    FI
    IF  return code is zero process data records
        Do until EOF on input tape
            FREAD data
                IF I/O error
                        set return code to 10 (or 16?)
                        Write message
                FI
            IF  return code is zero
                count # of data records (SCIREC) for
                Set bitrate, note all changes      <-- set words 62,65,68,70 to zero
                move data into proper core image
                location
            FI
        OD  all records read in
        IF  no errors
            Set SCIREC in ENCMCB
            Call  TIMCHK  to  set record times
                  (need Header year check variables as arguments)

            Call  DATCHK to flag any records
                  for skip, etc.
                  (argument would include any bitrate array)
        FI
    FI
FI
END
```

Header processing

	time locations		
Start	header year 48	51	{ stop
	header day 49	52	
	" sec 50	53	

data record word 1 = day of year
2 = seconds of day

Read in Header

```
syear = Header(48)
Sday  = header(49)
Ssec  = header(50)
eyear = header(51)
eday  = header(52)
esec  = header(53)
Call settim (syear, sday, ssec, stotse, err)
Call settim (eyear, eday, esec, etotse, err1)
IF (eyear - syear) .EQ. 0) ~~goto~~ DO sameyear   QHDYR = .FALSE.
ELSE
      IF (eyear .NE. (syear+1)) seterror to exit reader?
      ELSE
        set year change from file header variable to true
                                  QHDYR=.TRUE.
      FI
FI
Sameyear  Set year change from file header variable to false

              SMH1 = HEADER(59)
                 2 =    "  (60)
                 3 =    "  (61)
                 4 =    "  (62)
       SID = Header(46)
       IF ( SID .NE. SATID ) set return to exit processing of file
```

Completion of header processing

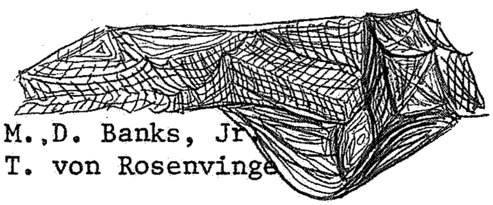
might want to double check day values

Pam - this is perhaps a cleaner write-up

NOTES ON THE

ISEE-3

DATA POOL TAPE



M.,D. Banks, Jr
T. von Rosenvinge

May 1979

I. Introduction

The International Sun-Earth Explorer Mission is a joint NASA/ESA project intended to study the earth's magnetosphere and its response to disturbances in the solar wind. The ISEE-3 spacecraft is positioned ~240 earth radii upstream of the earth's bow-shock and observes the solar wind flowing towards the earth while the ISEE-1 and 2 spacecraft make observations in or near the magnetosphere. This project has been described in detail by Ogilvie, et al. (1977).* The primary purpose of the ISEE-3 data pool tape is to make basic quantities measured at ISEE-3 readily available beyond the individual experiment groups making the measurements. This is particularly desirable since the emphasis of this mission is on utilizing simultaneous data from all three spacecraft. The data pool tape does have some limitations, however. For example, the time resolution and selection of data is limited. Also, the algorithms for transforming measured quantities into physical units are generally not as complex as those that experimenters may eventually use in reducing their data. On the other hand, many users will profit more by quick access to somewhat imperfect data than by eventual access to more refined data. For instance, an experimenter can use the data pool tape to identify interesting time periods and hence greatly reduce the volume of refined data which he may request from another experimenter.

The data pool tape is produced at Goddard Space Flight Center by the Information Processing Division (IPD) using algorithms provided by each experimenter. IPD does its best to process the data accordingly, but it is staffed by programmers and not scientists and hence cannot be held responsible for identifying, for example, subtle changes in instrument performance, limitations of experiment response, or interference to an experiment.

In order, then, for a user to make sensible use of the data pool tape he requires a good description of each experiment, a description of the tape format and a description of the algorithms used. The first of these has been provided in the IEEE transactions on Geoscience Electronics, July 1978, Volume GE-16. It is intended that the remaining items be supplied in part by the present document. Following this introduction, we have provided notes on the ISEE-3 data pool tape together with the tape format. This precedes sections which have short write-ups from each experimenter regarding the method by which their data is reduced to yield the quantities on the tape and appropriate caveats. Finally there is a list of Principal Investigators with addresses and telephone numbers.

* Ogilvie, K. W., von Rosenvinge, T. T. and Durney, A. C., Science, 198, #131, 1977.

NOTES ON THE ISEE-3 DATA POOL TAPE

GENERAL DESCRIPTION AND USAGE

1. Structure of the Data Pool Tape

(A) Tapes

Each data pool user receives one reel of tape per week. This tape may be 7-track 556 cpi, 9-track 1600 cpi, or 9-track 800 cpi, depending upon the user's preference. All tapes are odd parity. Inter-record gaps are .65 inch for 9-track tapes and .75 inch for 7-track tapes.

(B) Files

Data pool information for a 7-day period is presented as a single tape file, beginning with a label record and ending with a standard tape mark (EOF mark). All records, including the label, are of the same length. The data pool file contains approximately 160 data records spanning 7 days of telemetry data. Redundant telemetry data (due to overlap in ground station coverage) has been removed. The data pool file coincides in time with a "shipping group" of the usual telemetry data (decom tapes) which is sent to each experimenter.

The data pool file appears 3 times on the tape for redundancy backup. See Figure 1.

(C) Data Words

(1) Word Size

Each data pool tape is written in computer words of a length compatible with the intended user's computer. Tapes thus constructed can be read directly into the user's computer with no reformatting. This holds true for both 7-track and 9-track tapes. For example, records intended for use with a CDC 6000 series computer would appear as packed strings of 60-bit fields. On a 9-track tape, each such 60-bit field occupies 7-1/2 tape characters. But the total number of words per record is an even number, so that the 60-bit fields can all be written in pairs, 15 tape characters per pair, and thus can be read normally by the CDC. (Other combinations of word size and tape type work out in a similar fashion).

- ONE TAPE PER 7-DAY DATA GROUP.
- ONE FILE PER DATA GROUP, REPEATED 3 TIMES FOR BACKUP.
- DATA POOL QUANTITIES ARE IN USER COMPATIBLE FLOATING POINT AND USER WORD LENGTH.
- DATA RECORDS ARE APPROXIMATELY 1 HOUR.

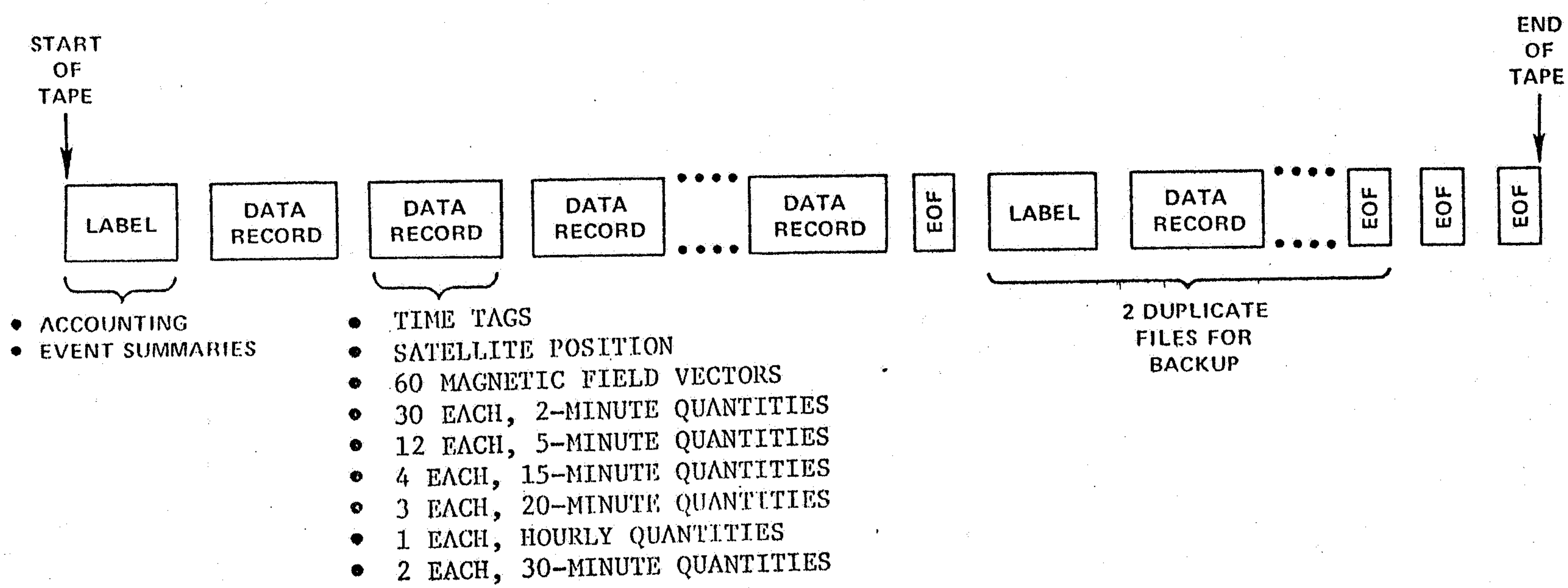


Figure 1. Data Pool Tape

(2) Floating Point Format

The entire data pool tape, the file label record and all data records, are in floating point format. (This provides a uniform, standard appearance of all information, facilitates conversion to the various computer word sizes, and simplifies tape printing and verification). The floating point representation used on each tape is specified by the user, and, like the word size, is compatible with the computer which will process the tape.

Most data pool tape quantities are originally computed in floating point and should be interpreted as such. Certain items, however, are obvious as integer values converted to floating point representation (day-of-year, seconds-of-day, clock, various indicator flags, etc.). The user may interpret such items either as floating point (in which case appropriate precautions should be taken to prevent possible truncation during conversion to integer).

(3) Missing Data

Missing data items are indicated by a negative value fill code in place of the missing item. The exact value of the fill code is dependent on the type of floating point used, but will in all cases be outside the legitimate range of any data item.

Missing data pool items may be the result of gaps in data recovery at the ground station, or the result of data being rejected by one of the experiment processing algorithms.

It is possible that uncertain conditions may lead to a data pool result of questionable validity. Rather than be rejected, such results may, in certain cases, be presented as the negative of the actual result. Thus, a negative number other than the fill code, if in a field which should normally be positive, represents a doubtful result and may be used, but with caution. (Note that this does not apply to those values which may normally be negative, such as magnetic field components).

Data taken when the spacecraft is in engineering format is rejected by the data pool program.

II. Contents of the Data Pool Tape

(A) Time

The time values given on the data pool tape are UTC (Universal Time Code) at the time the data was transmitted from the spacecraft (i.e. transmission lag time has been removed). Times have been smoothed to remove random errors and then verified by intercomparison among all the ground stations. Time is given as Julian day (1-366) and seconds-of-day.

(B) Clock

The clock used on the data pool tape is a minor frame counter. It is constructed by combining the 24-bit raw spacecraft clock with part of the frame counter. (This is the same clock as used on the telemetry data tapes).

Since the full clock will not fit in all types of floating point works without truncation, it is broken into 2 pieces, the low order 21 bits in one word, the remaining high order portion in another. The full clock can be reconstructed by converting both clock pieces to integer, then adding them together.

It should be noted that time, rather than clock, is the primary reference for data pool items. Clock is subject to rollover within the data pool file, as well as to unpredicted jumps forward or backward.

(C) Timelines

The time versus clock relationship, may not be linear throughout the entire data pool file. Breaks occur if the bit rate changes, at end-of-year, and if the clock jumps or rolls over. A segment of data in which time versus clock is linear is defined as a time/clock baseline or "timeline."

Data accumulation intervals for experiment algorithms do not continue across timelines; that is, each data pool quantity is computed using data which begins and ends in the same timeline. See Figure 2.

(D) Data Records

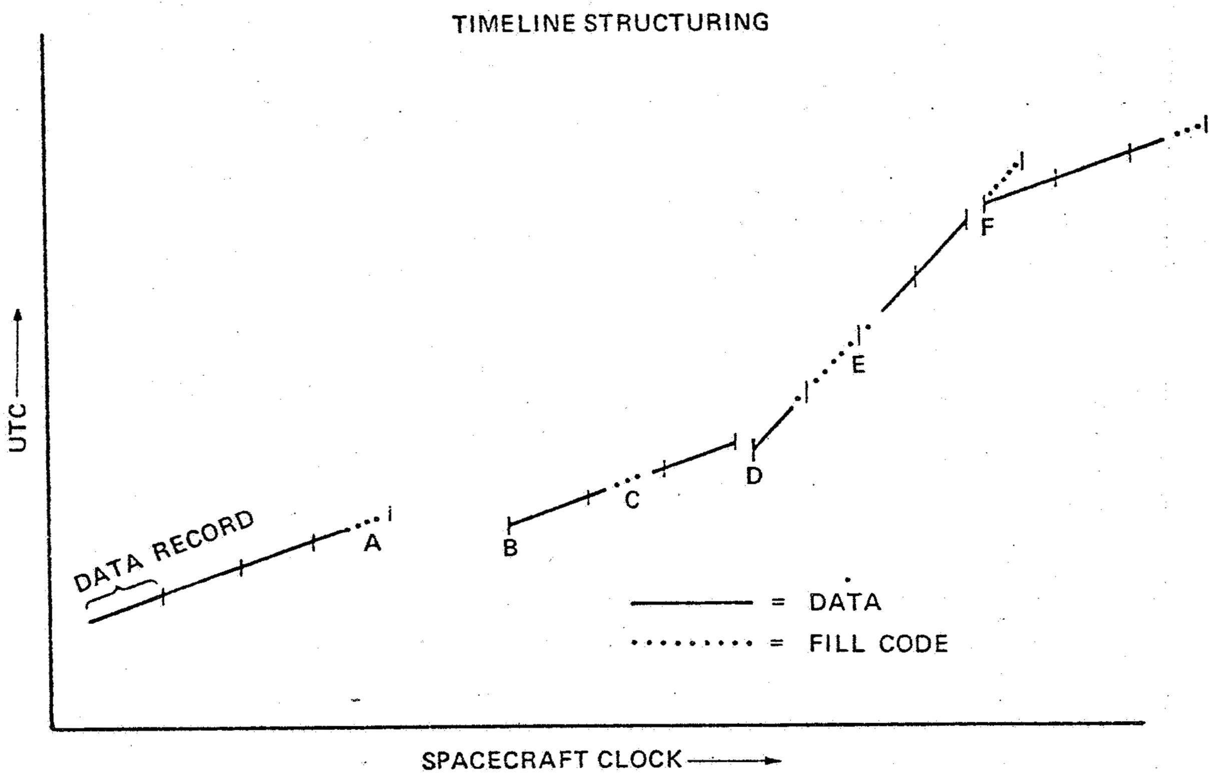
Data records are fixed length, 810 words long. A full record holds 64 minutes of information.

Within a timeline, each data record represent
tition in time. Data items are positioned within the records

When a new timeline starts, the uniform 64-minute spacing of records is interrupted and a new sequence of 64-minute records is established. The first record of the new timeline will not, in general, increment by 64 minutes from the previous record. Subsequent records will increment by 64 minutes until another timeline begins. Data records which begin a new timeline are flagged both in the records themselves and in the file label.

version contained one additional

These were never taken into account in DPTSUM, or MAGSUM



Spacecraft Clock

A. Last record of timeline contains fill beyond the end of the timeline.

B. A new timeline begins due to a clock reset. No gap in data coverage.

C. Data gap within a record results in fill code.

D. New timeline begins due to a bit rate change.

E. Large data gap results in an entire record of fill code. Note that the record following the all-fill record begins with fill, but has a start time assigned as if data were present.

F. New timeline begins due to a bit rate change.

Figure 2. Timeline Structuring

The format of the data record is given in Table 2.

(E) Time Tagging

There are seven types of data pool information, according to frequency of readout: (1) 60 per record, 1-minute intervals, (2) 30 per record, 2-minute intervals; (3) 12 per record, 5-minute interval; (4) 4 per record, 15-minute intervals; (5) 3 per record, 20-minute intervals; (6) 2 per record, 30-minute intervals; (7) once per record. ("Minute," as used here, means an ISEE minute," or 64-seconds independent of bit-rate.

The start time of the data record (words 1 and 2) is the start time of sampling interval number at all seven frequencies of readout. The start times of subsequent intervals are computed relative to interval number 1.

Examples (Refer to Table 3):

Example 1 -- The magnetic field vector {Bz(1), Bx(1), By(1)}, in words 201-203 was computed for the 64-second interval beginning at the record start time.

The vector {Bz(60), Bx(60), By(60)} , in words 555-557, was computed over the 64-second interval beginning at t=(record start time) + (59 x 64 seconds). Similarly, vector {Bz(3), Bx(3), By(3)} , words 213-215, was computed over the 64-second interval beginning at t=(record start time) + (128 seconds).

Example 2 -- Find the energetic particles flux, energy >15 MeV, at a point in time 20 minutes from the start of the record.

The required information set is from the Anderson algorithm, words 681-682, labeled EFLUX (1) - EFLUX (12). This data is given at intervals of 5 ISEE-minutes or every 320 seconds. Let RST equal the record start time. Then,

RST + 20 min = RST + 1200 sec = RST + 3.75 intervals

The desired value would thus be best approximated by interval No. 4, word 684.

(F) File Label

The file label record, Table 1, contains identification and accounting information for the data pool file. It also contains the minimum and maximum spin periods encountered over the 7-day file period, an index to all timelines in the file, and magnetometer parameters. The record is padded to the length of the data records.

As indicated in Table 1, the first 1440 bits should be ignored by the user. These bits are used for internal accounting purposes by Goddard Space Flight Center.

Table 1: DATA POOL FILE LABEL

WORD NUMBER*	DESCRIPTION (ALL VALUES ARE FLOATING POINT)
1	
	1440 BITS FOR GSFC INTERNAL USE.
N	
N+1	SATELLITE ID NUMBER
N+2	INTENDED RECIPIENT OF THIS TAPE. (SEE TABLE 2)
N+3	YY,START OF FILE, 2 DIGITS OF YEAR.
N+4	DDD,START OF FILE,DAY OF YEAR.
N+5	SSSSS,START OF FILE,SECONDS OF DAY.
N+6	YY,END OF FILE,2 DIGITS OF YEAR.
N+7	DDD,END OF FILE,DAY OF YEAR.
N+8	SSSSS,END OF FILE,SECONDS OF DAY.
N+9	HIGH ORDER BITS. CLOCK AT START OF THE DATA POOL FILE.
N+10	LOW ORDER 21 BITS. CLOCK AT START OF THE DATA POOL FILE.
N+11	GROUP NUMBER (CORRESPONDING TO THE TELEMETRY DATA TAPE GROUP NUMBER)
N+12	MINIMUM VALUE OF SPIN PERIOD FOUND WITHIN THIS FILE IN SECONDS.
N+13	MAXIMUM VALUE OF SPIN PERIOD FOUND WITHIN THIS FILE IN SECONDS.
N+14	SMH1 Z-OFFSET USED FOR THIS RUN.
N+15	SMH2 NUMBER OF ESTIMATES MADE FOR Z-OFFSET ABOVE.
N+16	SMH3 ALPHA USED FOR Z-OFFSET ABOVE.
N+17	SMH4 GROUP NUMBER OF THE DATA GROUP USED TO DETERMINE Z-OFFSET.
N+18	
•	
•	SPARES.
•	
N+80	
N+81	NUMBER OF TIME LINES (MAXIMUM OF 80)
N+82	START DAY OF YEAR (1).
N+83	START SECONDS OF DAY (1).
N+84	HIGH ORDER BITS OF THE SPACECRAFT CLOCK (1).
N+85	LOW ORDER 21 BITS OF START SPACECRAFT CLOCK (1).
N+86	BIT RATE (1.0 FOR 512 BPS, 2.0 FOR 1024 BPS AND, 4.0 FOR 2048 BPS)
N+87	START RECORD NUMBER.
•	
•	
•	
N+656	START DAY OF YEAR (80).
N+657	START SECONDS OF DAY (80).
N+658	HIGH ORDER BITS OF START SPACECRAFT CLOCK (80).
N+659	LOW ORDER 21 BITS OF START SPACECRAFT CLOCK (80).
N+660	BIT RATE (1.0 FOR 512 BPS, 2.0 FOR 1024 BPS AND, 4.0 FOR 2048 BPS)
N+661	START RECORD NUMBER (80).
N+662	
•	FILL TO EQUAL DATA RECORD LENGTH.
810	

Table 2: DATA POOL - DATA RECORD

WORD NUMBER	DESCRIPTION (ALL VALUES ARE FLOATING POINT)		
1	DAY OF YEAR,RECORD START		
2	SECONDS OF DAY,RECORD START		
3	CLOCK,RECORD START. HIGH ORDER PORTION		
4	CLOCK.RECORD START. LOW ORDER 21 BITS		
5	RECOVERY FACTOR: (MINOR FRAMES PROCESSED)/(7.5 X 256.).FOR 512 BPS (MINOR FRAMES PROCESSED)/(15 X 256.).FOR 1024 BPS (MINOR FRAMES PROCESSED)/(30 X 256.).FOR 2048 BPS		
6	BIT RATE: 1.0 = 512 BPS (BACKUP) 2.0 = 1024 BPS (LOW) 4.0 = 2048 BPS (HIGH)		
7	DUMMY RECORD INDICATOR: 0.0 = AT LEAST ONE MINOR FRAME OF DATA WITHIN THIS RECORD'S SP 7.0 = NO DATA WITHIN THE SPAN OF THIS RECORD. A DUMMY RECORD.		
8	TIMELINE INDICATOR: 0.0 = THIS RECORD LIES ON AN EXISTING TIMELINE 7.0 = THIS RECORD BEGINS A NEW TIMELINE		
9	DATA RECORD NUMBER		
10 - 12	SPARES		
13	BO-X	OFFSET USED FOR SMH BX	
14	BO-Y	OFFSET USED FOR SMH BY	
15			
16	WORDS 15 TO 19 FOR SMH USE ONLY		
17			
18			
19			
20	SPIN PERIOD AVERAGE,PREVIOUS HOUR.		
21	GSE-X		
22	GSE-Y	SATELLITE POSITION VECTOR IN GSE COORDINATES	
23	GSE-Z	(AT TIME OF FIRST POINT IN THIS RECORD)	
24-168	SPARES		
	* * * * * * * * * HOVESTADT ALGORITHM * * * * * * * * * *		
169	PROLP(1)	0.17-0.4MEV PROTONS	1ST OF 4
.			
.			
172	PROLP(4)	0.17-0.4MEV PROTONS	4TH OF 4
173	ALFLA(1)	0.12-0.25MEV ALPHAS	1ST OF 4
.			
.			
176	ALFLA(4)	0.12-0.25MEV ALPHAS	4TH OF 4
177	HEAVYS(1)	HEAVIES (Z>2) GT 0.1MEV	1ST OF 4
.			
.			
180	HEAVYS(4)	HEAVIES (Z>2) GT 0.1MEV	4TH OF 4
181	PROHP1(1)	5-10MEV PROTONS	1ST OF 4
.			
.			
184	PROHP1(4)	5-10MEV PROTONS	4TH OF 4

Table 2 (Continued)

185	PROHP2(1)	10-20MEV PROTONS	1ST OF 4
.			
188	PROHP2(4)	10-20MEV PROTONS	4TH OF 4
	* * * * * * * * * MANEUVER INFORMATION * * * * * * * * * *		
189	MANUVR(1)	MANEUVER INDICATORS FOR EACH OF THE TWELVE	
190	MANUVR(2)	5 MINUTE(APPROX) INTERVALS OF THIS RECORD:	
191	MANUVR(3)	0.0 = NO MANEUVER IN THIS INTERVAL	
192	MANUVR(4)	7.0 = MANEUVER INDICATED DURING THIS INTERVAL	
193	MANUVR(5)		
194	MANUVR(6)		
195	MANUVR(7)		
196	MANUVR(8)		
197	MANUVR(9)		
198	MANUVR(10)		
199	MANUVR(11)		
200	MANUVR(12)		
	* * * * * * * * * SMITH ALGORITHM (MAGNETOMETER) * * * * * * * * * *		
201	BZ(1)	SPIN AXIS COMPONENT	
202	BX(1)	SATELLITE-SUN LINE COMPONENT	MAG FIELD VECTOR
203	BY(1)	3RD COMPONENT OF TRIAD	1ST OF 60 VECTORS
204	BMAG(1)	MAGNITUDE	
205	BDELTA(1)	LATITUDE	MAGNETIC FIELD VECTOR
206	BPHI(1)	LONGITUDE	1ST OF 60 VECTORS
207-554	2ND THROUGH 59TH MAGNETIC FIELD VECTORS.		
555	BZ(60)	SPIN AXIS COMPONENT	
556	BX(60)	SATELLITE-SUN LINE COMPONENT	MAG FIELD VECTOR
557	BY(60)	3RD COMPONENT OF TRIAD	60TH OF 60 VECTOR
558	BMAG(60)	MAGNITUDE	
559	BDELTA(60)	LATITUDE	MAGNETIC FIELD VECTOR
560	BPHI(60)	LONGITUDE	60TH OF 60 VECTORS
	* * * * * * * * * STEINBERG ALGORITHM * * * * * * * * * *		
561	RAMAP1(1)	AVERAGE VOLTAGE AND RMS (1000KHZ.)	
562	RARMS1(1)	1ST OF 30 SETS	
.			
.			
619	RAMAP1(30)	AVERAGE VOLTAGE AND RMS (1000KHZ.)	
620	RARMS1(30)	30TH OF 30 SETS	
621	RAMAP2(1)	AVERAGE VOLTAGE AND RMS (200KHZ.)	
622	RARMS2(1)	1ST OF 30 SETS	
.			
.			
679	RAMAP2(30)	AVERAGE VOLTAGE AND RMS (200KHZ.)	
680	RARMS2(30)	30TH OF 30 SETS	

Table 2 (Continued)

* * * * * * * * * DE FEITER ALGORITHM * * * * * * * * * *			
797	PROLO1(1)	PROTONS 78-205 KEV	1ST OF 3
798	PROLO2(1)	PROTONS 536-1400KEV	1ST OF 3
799	ISOTRO(1)	ISOTROPY INDEX	1ST OF 3
800	QUAD(1)	QUADRANT	1ST OF 3
•			
•			
•			
805	PROLO1(3)	PROTONS 78-205 KEV	3RD OF 3
806	PROLO2(3)	PROTONS 536-1400KEV	3RD OF 3
807	ISOTRO(3)	ISOTROPY INDEX	3RD OF 3
808	QUAD(3)	QUADRANT	3RD OF 3
* * * * * * * * * MEYER ALGORITHM * * * * * * * * * *			
809	LOWEE(1)	LOW ENERGY 5-150MEV ELECTRONS RATE	1 OF 2
810	LOWEE(2)	LOW ENERGY 5-150MEV ELECTRONS RATE	2 OF 2

page missing — Anderson/Bame/Scarf/VonRosenvinge is next

Table 2 (Continued)

* * * * * * * * * ANDERSON ALGORITHM * * * * * * * * * *			
681	EFLUX(1)	PARTICLE FLUX, ENERGY > 15KEV	1ST OF 12
•			
•			
•			
692	EFLUX(12)	PARTICLE FLUX, ENERGY > 15KEV	12TH OF 12
693	XRAY(1)	COUNTS PER SECOND,20-37 KEV	1ST OF 12
•			
•			
•			
704	XRAY(12)	COUNTS PER SECOND,20-37 KEV	12TH OF 12
* * * * * * * * * BAME ALGORITHM * * * * * * * * * *			
705	IONPD(1)	ION PSEUDO-DENSITY (PARTICLES/CC)	1ST OF 1
•			
•			
•			
716	IONPD(12)	ION PSEUDO-DENSITY (PARTICLES/CC)	12TH OF 1
717	WINDPS(1)	SOLAR WIND PSEUDO-SPEED (KM/SEC)	1ST OF 1
•			
•			
•			
728	WINDPS(12)	SOLAR WIND PSEUDO-SPEED (KM/SEC)	12TH OF 1
729	WINDPA(1)	SOLAR WIND PSEUDO FLOW ANGLE (DEG)	1ST OF 1
•			
•			
•			
740	WINDPA(12)	SOLAR WIND PSEUDO FLOW ANGLE (DEG)	12TH OF 1
* * * * * * * * * SCARF ALGORITHM * * * * * * * * * *			
741	PLA31(1)	PLASMA WAVE 31HZ MAX. VOLTAGE	1ST OF 12
742	PLA1K(1)	PLASMA WAVE 1KHZ MAX VOLTAGE	1ST OF 12
743	PLA31K(1)	PLASMA WAVE 31KHZ MAX VOLTAGE	1ST OF 12
744	PLANT(1)	PLASMA WAVE ANTENNA STATUS	1ST OF 12
•			
•			
•			
785	PLA31(12)	PLASMA WAVE 31HZ MAX. VOLTAGE	12TH OF 12
786	PLA1K(12)	PLASMA WAVE 1KHZ MAX VOLTAGE	12TH OF 12
787	PLA31K(12)	PLASMA WAVE 31KHZ MAX VOLTAGE	12TH OF 12
788	PLANT(12)	PLASMA WAVE ANTENNA STATUS	12TH OF 12
* * * * * * * * * VON ROSENVINGE ALGORITHM * * * * * * * * * *			
789	PARTLO(1)	PARTICLES, LOW RANGE 4-57MEV/NUCLEON	1ST OF
•			
•			
792	PARTLO(4)	PARTICLES, LOW RANGE 4-57MEV/NUCLEON	4TH OF
793	PARTHI(1)	PARTICLES, HIGH RANGE 18-70MEV/NUCLEON	1ST OF
•			
•			
796	PARTHI(4)	PARTICLES, HIGH RANGE 18-70MEV/NUCLEON	4TH OF

ISEE-3

MAGNETIC FIELD INVESTIGATION:

A Brief Description

of the

Experiment

and the

Data Pool

Processing Algorithm

A. M. A. Frandsen

2 October, 1978

TABLE OF CONTENTS

MAGNETIC FIELD INVESTIGATION

Principal Investigator: Edward J. Smith

Address: Jet Propulsion Laboratory
Building 183, Room 401
4800 Oak Grove Drive
Pasadena, California 91103
(213) 354-2248 (Commercial)
792-2248 (FTS)

Co-Investigators: Leverett Davis, Jr.
California Institute of Technology

Douglas E. Jones
Brigham Young University

George L. Siscoe
University of California, Los Angeles

Bruce T. Tsurutani
Jet Propulsion Laboratory

A. BRIEF EXPERIMENT DESCRIPTION

1. Experiment Objectives

The objectives of the ISEE-3 magnetic field experiment are to investigate the following:

(a) The effect of solar activity on the interplanetary magnetic field and solar wind parameters.

(b) The persistence, as well as changing character, of corotating features in the solar wind.

(c) Changes in large scale solar wind features over great distances, by correlating with data from planetary missions (Pioneer Venus, Pioneer Saturn and Voyager).

(d) Waves and other irregularities in the solar wind, their frequency content and phase relationships.

(e) The velocity of propagation and dispersion of field fluctuations between ISEE-3 and ISEE-1 & 2, while the latter are in the interplanetary medium.

(f) Plasma instabilities in the solar wind.

(g) The response of the magnetosphere to solar wind variations.

(h) The relationship between solar wind parameters and the extent and properties of the earth's magnetosheath.

2. Instrument Description

The ISEE-3 magnetometer experiment consists of a sensor mounted at the outboard end of a 3-meter boom and an electronics package located within the main body of the spacecraft. A Project-supplied data processing unit (DPU) provides 2-way buffering between the vector helium magnetometer (VHM) electronics package and the spacecraft data system. It conditions the spacecraft command and control signals for use by the instrument and it buffers the magnetometer outputs as they are read into the main telemetry stream. In addition, the DPU generates a magnetic sector pulse for on-board use by other experiments.

To simplify data reduction, the VHM sensor is mounted with its sensitive axes parallel to the principal axes of the spacecraft. The sensor detects the three components of the ambient steady-state magnetic field vector and fluctuations in these components up to 3 Hz. The operating principles of the VHM sensor are based on the effect of an external magnetic field upon the efficiency with which a population of metastable helium atoms may be optically pumped. Figure 1 illustrates

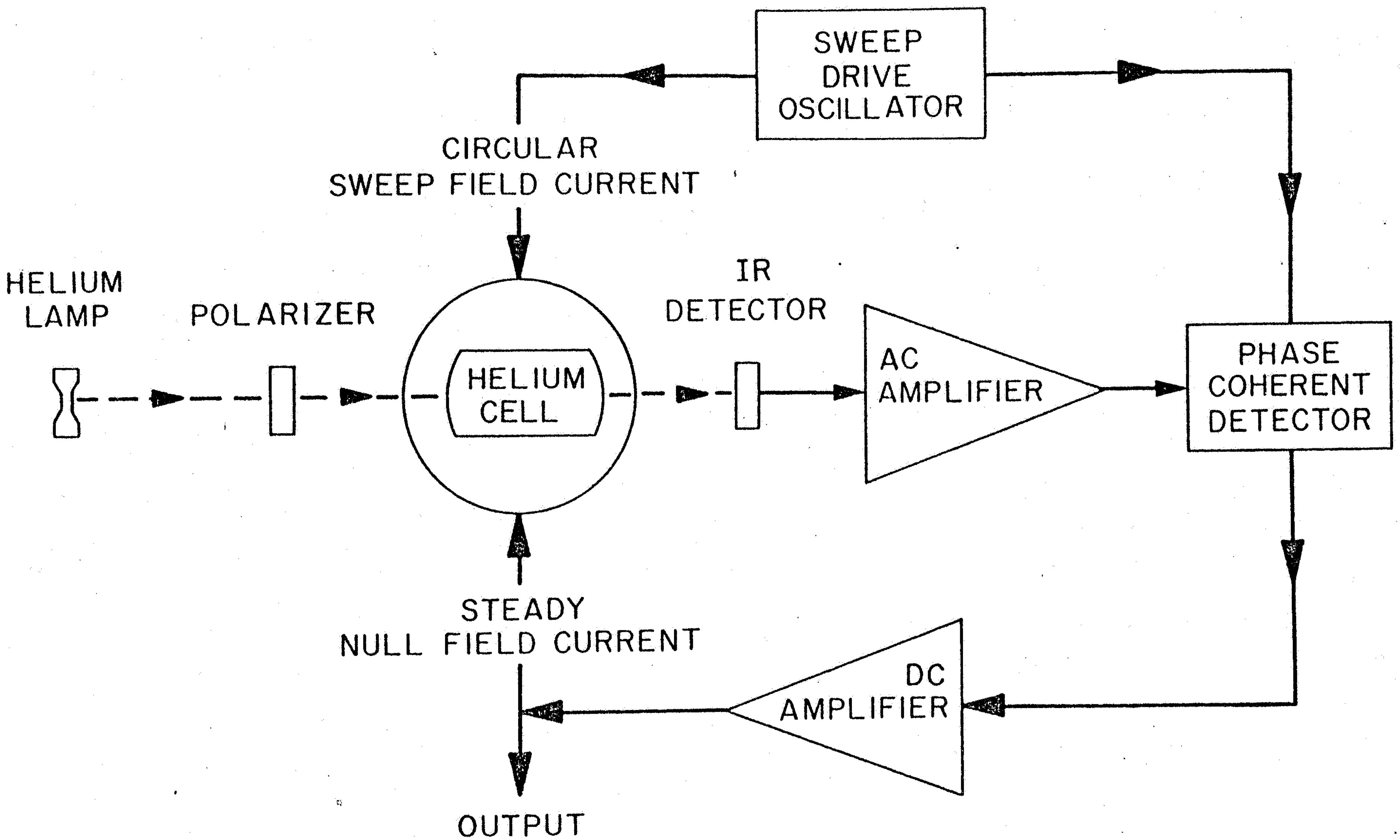


FIGURE 1 VECTOR HELIUM MAGNETOMETER SIMPLIFIED SCHEMATIC

several of the essential parts of the instrument. In the sensor, a collimated beam of infrared (IR) radiation from a helium lamp is circularly polarized to optically pump metastable helium atoms in the absorption cell. Properly phased sweep currents drive three mutually orthogonal coil pairs surrounding the cell and create a constant-amplitude rotating field vector. The sweep vector intensity modulates the emerging IR radiation. Brightness variations are then converted into a low frequency electrical signal at the IR detector. When no external magnetic field is present, only the second harmonic of the applied sweep frequency appears at the detector output. The main electronic assembly is designed to sense departures from the pure second harmonic signal. Synchronous demodulators are used to generate currents that are fed back to the helmholtz coils at the sensor. In this closed loop mode of operation, the ambient magnetic field acting upon the sensor is nulled to zero, and the three feedback currents accomplishing the nulling action are read as outputs. A multiplexer and a 9-bit analog-to-digital converter (ADC) are used in the process.

The three field components are sampled in rapid succession with less than 10 milliseconds of skewing. Each 9-bit conversion includes the sign of the component and eight bits of amplitude information. A 27-bit vector measurement is then combined with three additional bits that convey which of eight ranges the instrument is operating on. These data are held in a 30-bit storage register until a readout is requested by the spacecraft data system. At the nominal telemetry rate, (2048 information bits per second), six vector readings are acquired each second. They are nearly equally spaced in time (e.g., to within 4 milliseconds at 2048 ibps). The instrument's sampling rate scales directly with the prevailing telemetry rate.

The VHM has eight linear operating ranges giving a wide, effective dynamic range. Normally, it is operated in a mode where the upranging and downranging take place automatically. However, it can be commanded to any specific range through a sequence of ground commands. In either mode, all three axes switch range at the same time. Table I lists the eight operating ranges and the number of nanotesla (gamma) per least significant bit (LSB).

The instrument's calibration is checked in flight on a weekly basis by commanding the VHM in-flight calibration (IFC) sequence. Prelaunch mechanical alignment of the sensor was determined within 0°.1.

B. MAGNETOMETER DATA POOL ALGORITHM DESCRIPTION

1. Data Conversion

The ISEE-3 data pool algorithm for the magnetometer experiment converts telemetry counts into field units, performs intermediate calculations and averages the results. The basic relationship used for the conversion is:

$$B_i = K_{i,r} \times (M_i - 255.5) - 0_{i,r}, \text{ for } i = X,Y \text{ or } Z, \text{ and } r = 0,1, \ldots 7$$

Where: (a) M_i is the telemetry count of the ith component of a given measurement.

(b) $K_{i,r}$ is the ith axis scale factor (gamma/count) for VHM operating range r.

TABLE I

VHM OPERATING RANGES

Range Number	Full-Scale gamma (1 gamma = 10^{-9} tesla)	Sensitivity gamma/LSB
0	± 4	0.015
1	± 13	0.051
2	± 43	0.17
3	± 145	0.57
4	± 632	2.5
5	$\pm 3,870$	15.0
6	$\pm 22,630$	88.0
7	$\pm 136,000$	530.0

(c) 255.5 is the nominal zero level count for each 9-bit component measurement.

(d) $0_{i,r}$ is the offset field (gamma) at the sensor location with respect to the nominal zero level.

Both $K_{i,r}$ and $0_{i,r}$ are stored as 3 x 8 arrays. Provision is made in the algorithm for automatically updating the elements of $0_{i,r}$ using the magnetometer data from the previous time interval processed. The scale factors $K_{i,r}$ are checked weekly in flight but are not automatically updated since they have been found on earlier missions using similar instruments not to change significantly.

Once the measurements have been converted from telemetry counts into magnetic field units, each vector is transformed into an inertial coordinate system based upon the spacecraft spin axis and the direction toward the sun. At the nominal 2048 ibps telemetry rate, the data pool algorithm acquires updated atitude information from the spacecraft's sun sensor once every 32 seconds. For each field measurement, the time difference is determined between it and the last occurring valid sun pulse readout. Using telemetered knowledge of the spacecraft spin rate, the algorithm then performs a matrix multiplication which rolls each measurement back through an angle corresponding to the time difference. Vector-by-vector, the components are transformed into the inertial coordinate system based upon $\hat{S}$, the direction toward the sun and $\hat{I}$, the + Z spin axis of the ISEE spacecraft. These inertial coordinates are called the ISEE Inertial (IS) system and are illustrated in Figure 2.

After the magnetometer data have been despun into ISEE inertial coordinates, a vector magnitude and two angles are computed for every set of component values. Every vector measurement is

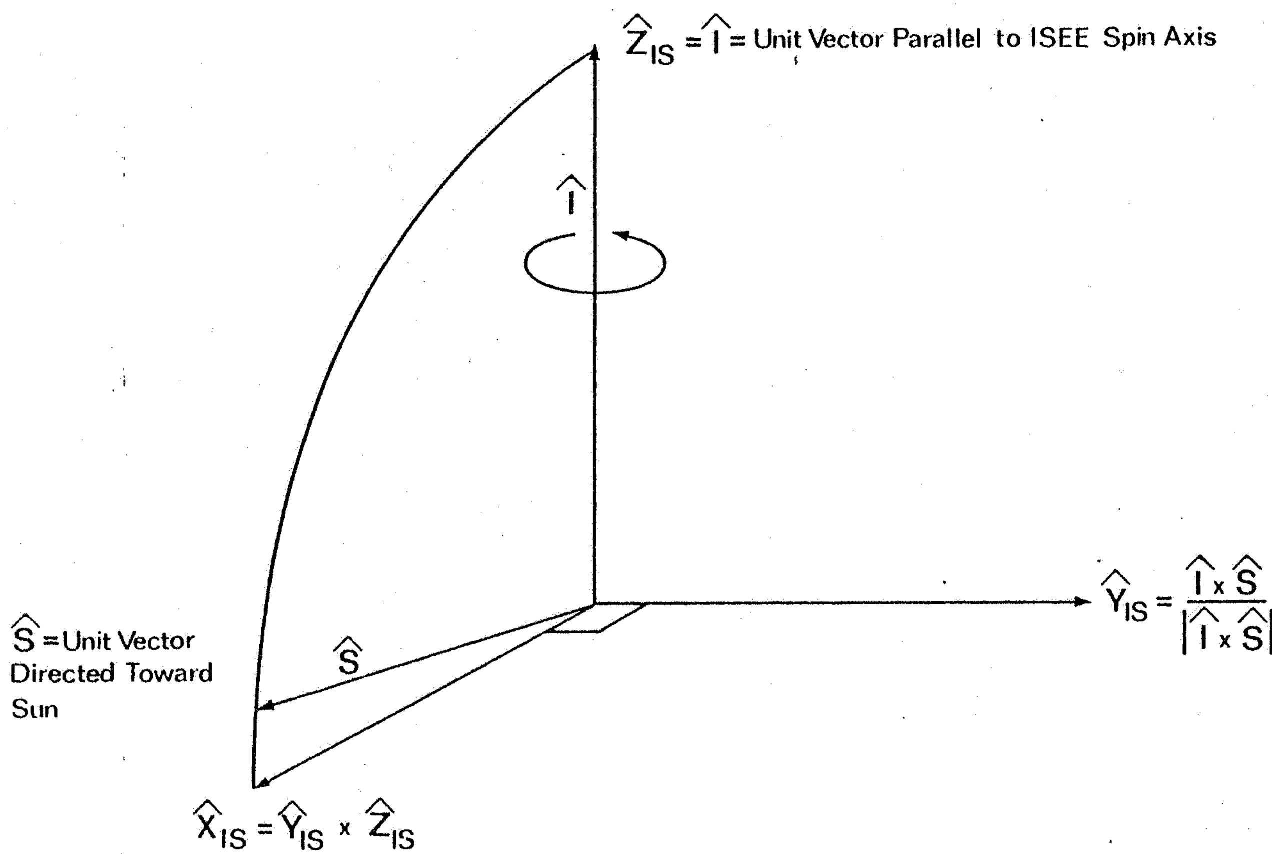


FIGURE 2 INERTIAL (IS) COORDINATE SYSTEM

therefore represented in both rectangular and spherical form. The angle δ_B is the field latitude in the IS system and ϕ_B the longitude angle. The six quantities thus determined are then separately averaged to form 1-minute averages appearing on the data pool tape:

$$\overline{B_{XIS}},\quad \overline{B_{YIS}},\quad \overline{B_{ZIS}},\qquad \overline{|B|},\quad \overline{\delta_B},\quad \overline{\phi_B}$$

Precautions are taken to avoid errors which can result from averaging azimuth angles that lie in the semicircle containing the 0°/360° branch cut.

Both spherical and rectangular field component averages are included in the pooled data because the two representations convey different information. The average field magnitude, for example, could be large and nearly constant during a natural disturbance while at the same time the field direction is so variable that the cartesian components average to small values. Thus, the field magnitude determined from the three component averages could be quite small. The difference in the two representations of B magnitude gives the power in the fluctuations, i.e.,

$$\sigma_B^2 = \left[\overline{|B|}\right]^2 - \left[\bar{B}_{XIS}^2 + \bar{B}_{YIS}^2 + \bar{B}_{ZIS}^2\right]$$

The user of the pooled data should be aware however that the six field averages are liable to be somewhat in error owing to the preliminary nature of the offset estimates used in the data reduction.

2. Offset Determinations

It is expected that when the pooled data are processed, the magnetometer offset will be known to 0.1 gamma. However, the precise values will always be lagging because an extended interval of data is required to determine them accurately, (e.g., ~ 1 month). When the algorithm uses biased estimates of the offset field components, the spacecraft spin frequency modulates B_{XIS} and B_{YIS}. The spin modulation is averaged out in the pooled data variables $\bar{B}_{XIS}$ and $\bar{B}_{YIS}$, but not in

$\overline{|B|}$, the average of the instantaneous vector magnitudes. Futhermore, biased estimates of the Z axis offset field component are reflected directly in the quantity $\bar{B}_{ZIS}$.

In order to keep the offset errors low, the algorithm computes hourly estimates of the two offset field components in the spacecraft spin plane by separately averaging the X and Y axis measurements. The new values are then used to update the spin plane offsets in the next hour's processing. The offset component parallel to the spacecraft spin axis is also estimated hourly by determining the value of Z axis field which minimizes the variance in the square of the field magnitude. However, from hour to hour these estimates are characterized by a fair amount of statistical scatter. Thus, the hourly spin axis offset field estimates are themselves averaged over one week and the result is used as the Z axis offset in the next week's processing. Many of the magnetometer parameters on the data pool tape are sums or products of the instantaneous vector components used in determining the offsets to be used on the following tape. The offset values used in creating the current tape are also given so that, if it becomes necessary, results may be corrected when more accurate values become available.

Attempts have been made in the data pool algorithm to subtract small interference fields known to originate on the spacecraft. Telemetered knowledge of the state of interfering subsystems is used for accessing a look-up table that gives the values to be subtracted from the magnetometer data. The algorithm also edits out data for time intervals during which the instrument is undergoing an in-flight calibration sequence, or when a spacecraft attitude maneuver is underway.

3. Sources of Error

Errors in the apparent field direction can come about not only when incorrect offset field estimates are used,

$$(\text{e.g., } \tan^{-1}\left[0.1\gamma \text{ error}/5\gamma \ |B|\right] \approx 1°),$$

but also through timing errors. Incorrect determinations of the occurrence of a sun pulse, or the time at which a vector sample was acquired, or an incorrect estimate of the spacecraft spin period can all lead to an error in the inertial field longitude. The algorithm assumes the nominal values for the spacecraft clock frequency and for the telemetry frame rate. Early indications are that the actual values are within 0.1% of the nominal frequencies. The resulting roll error in the pooled data is therefore less than 0.001 x 360° (max), or less than one-third of a degree due to this cause.

The algorithm also assumes the nominal mechanical alignment of the two oppositely directed spacecraft sun sensors, and of the VHM sensor itself. Prelaunch optical sighting and electronic calibrations showed the pulse coming from each sun sensor to be within 0.35° of nominal. Furthermore, all three magnetometer axes were found to be within 0°.1 of nominal. Generally speaking then, the probable error in field direction determined from the data pool tapes is approximately a degree or two. Larger errors can occur, however, when the ambient field becomes small compared to the typical interplanetary value at 1 AU. This error comes about because, in a fixed analysis time on a multirange instrument, the uncertainty in the offset field estimates is not necessarily reduced in direct proportion to the ambient field.

THE 3-D RADIO MAPPING EXPERIMENT -- STEINBERG

Experiment Description

The principal purpose of this experiment is to map the trajectories of type III solar bursts by determining the angular coordinates of a localized source as a function of frequency and time. The radial distance may be obtained by triangulation with observations from another satellite, or from assumptions about the density of the interplanetary medium.

Two perpendicular dipole antennas are used. A 90 m tip-to-tip antenna in the spin plane of ISEE-3, referred to as the S antenna, sees a signal which is modulated because of the changing aspect of the source due to the spacecraft's rotation. The Z antenna is 14 m tip-to-tip, along the spin axis. From the S measurements, the azimuth and strength of the source are obtained. Comparison of S and Z observations provides the elevation of the source from the spin plane and an estimate of its angular diameter.

Measurements are made in 12 frequency channels, between 30 and 1980 kHz, in each of two bandwidths, 10 kHz (B), and 3kHz (N). Every 1.5 seconds (which is nearly one-half spin), one measurement of Z and 11 of S are made for one frequency channel in each bandwidth, interleaving B and N observations. This provides nearly the full range of modulation possible from the S antenna. (At data rates lower than 2048 bps, proportionally fewer S samples are taken.) The frequency channel is selected according to a fixed 72 step program, designed to observe each frequency at uniform intervals but with shorter intervals for the higher frequencies. Alternate modes of observation are possible using only the B or only the N bandwidth.

For any single measurement, the signal passes a logarithmic square-law detector so that the output receiver voltage V is roughly proportional to the logarithm of the antenna temperature T_A. V is digitized into 256 steps between 0 and 5.

Algorithm for the Data Pool Tape

Average voltages at two frequencies and the rms values are supplied on the data pool tape. The averages are made of all the S B samples at the selected frequency which were obtained in successive 128 second intervals (one major frame at the top data rate). The tape record contains 30 averages (one "ISEE hour") at 1000 kHz, each followed by its rms value, after which come 30 averages at 233kHz, each followed by its rms value. (if only N band observations are being made, averages are of all S N samples, and the lower frequency is 188kHz. There is no flag on the tape to denote this.)

The average voltage measures the strength of the signal. The rms values are a good indicator of the degree of modulation present; the larger the rms value, the more deeply modulated is the signal, denoting a narrow source near the spin plane. The full data record is needed for direction finding.

An approximate transformation between voltage and antenna temperature is

$$\log T_A = 5.938 + 0.712*V + 0.132*V^2 \qquad 1000 \text{ kHz}$$

$$\log T_A = 6.217 + 0.712*V + 0.132*V^2 . \qquad 233 \text{ kHz}$$

NOTES ON THE SOLAR WIND PLASMA PARAMETERS ON THE ISEE-3 DATA POOL TAPE

Quantities have been derived by approximate algorithms only; to distinguish these from more accurate quantities we have referred to these quantities as the pseudo wind speed, pseudo density, etc. The following caveats and comments apply:

1) IONPD (ion density) is in units of cm^{-3} and can have values ranging from 0 to 100 cm^{-3}. Because of a number of factors, including the fact that only relatively simple-minded algorithms can be used for the data pool tape, we can't quote an accuracy any better than a factor of two for the ion density.

2) WINDPS (speed) is in km sec^{-1} and we expect values ranging from 250 to 850 km sec^{-1}, with an accuracy of $\pm$ 5%.

3) WINDPA (direction in the plane of the ecliptic) is in degrees, and we expect values ranging between $\pm$ 15° of the solar direction with an accuracy of $\pm$ 3°.

No background corrections are made and the values are instantaneous as opposed to averages.

DESCRIPTIVE EXPLANATION - PLASMA WAVE DATA ON THE ISEE-3 COMMON DATA POOL TAPE

by

Frederick L. Scarf

The ISEE-C plasma wave instrument has three spectrum analyzers with a total of 19 different frequency channels that cover the range from 0.3 Hz to 100 kHz. The main analyzer, which uses 16 continuously active automatic gain control amplifiers, gives two complete spectral scans per second in each of 16 filter channels. The instrument sensors include a high sensitivity magnetic search coil (B), and electric antennas with effective lengths of 45 meters (U-axis or V-axis) and 0.6 meter (short-E) [see Scarf et al. (1978) for a detailed description of the instrumentation]. The plasma wave output on the common data pool tape contains the peak values (uncalibrated output voltages; 5-minute accumulations) from three channels of the main 16-channel analyzer (31 Hz, 1 kHz and 31 kHz), along with an indicator of the antenna in use (U, V, short-E or B). The calibration data and a brief interpretation of the ISEE-3 plasma wave output on the data pool tape are given below.

The top part of Figure 1 contains a typical production plot of the full output from the 16-channel analyzer, and this serves as a reference for the data pool tape explanation. For all channels, this production plot has both peaks (isolated dots) and averages (the tops of the blackened regions) computed using 128-second accumulations of telemetry output values ($0 \le \phi \le 5$ volts), and the sensor-in-use is indicated below the time axis. This plot shows whistler mode activity (17.8, 31 and 56 Hz channels), impulsive ion acoustic wave bursts (sporadic activity with high peak-to-average ratios in the 311 Hz to 5.6 kHz channels) along with solar radio bursts, electron plasma oscillations and auroral kilometric radiation (on the 31 kHz, 56 kHz and 100 kHz channels). The bottom figure contains a plot of the common data pool tape output from the same day, and a comparison of the two drawings gives a general indication of the way to interpret the data pool plots.

The calibrated E-field spectral densities can readily be derived from the telemetry output plots because for each channel the E value [in volts/meter $(Hz)^{1/2}$] is related to the telemetry output (ϕ - 0 to 5 volts) by

$$E = ae^{b\phi}$$

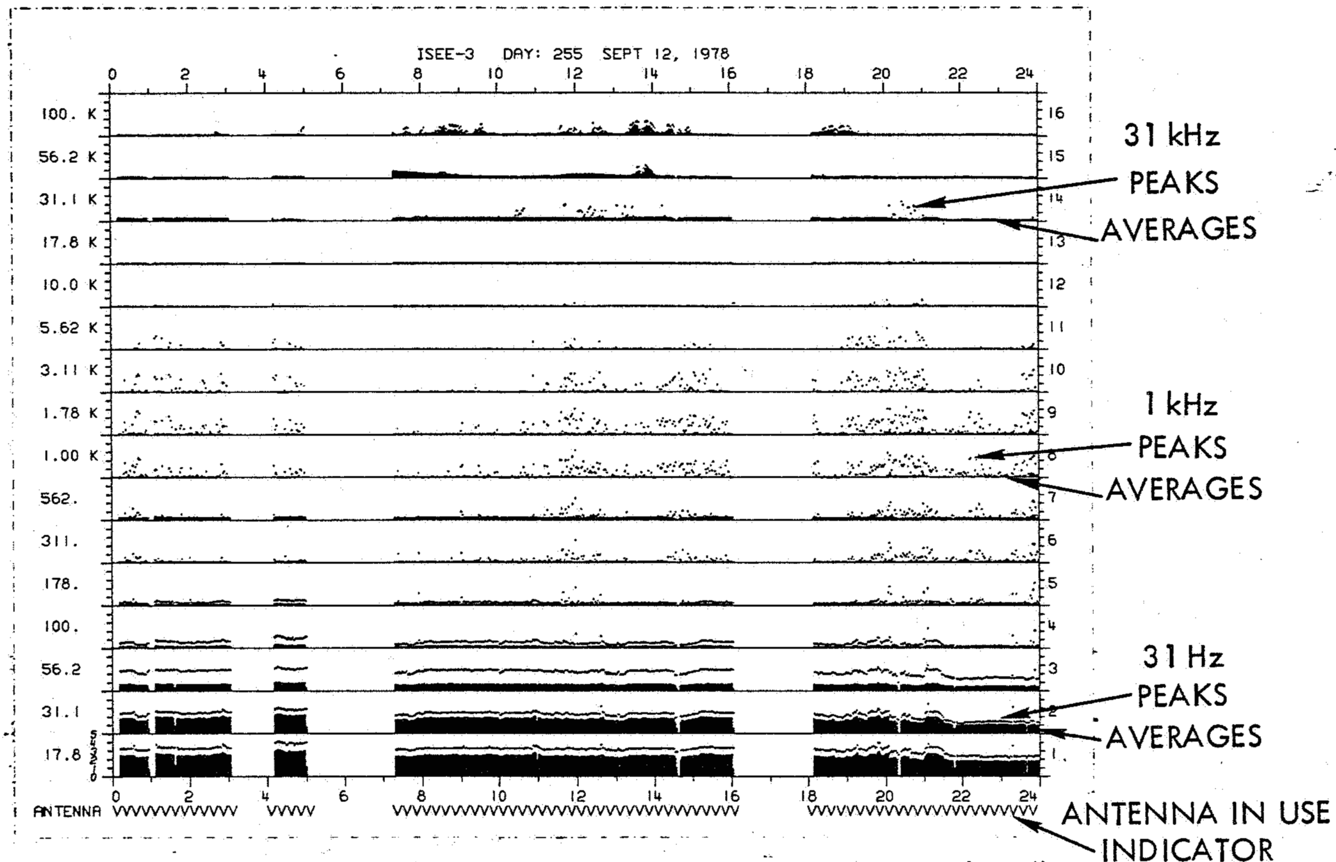
For either of the long (U- or V-axis) antennas, the calibration table is

Channel	31 Hz	1 kHz	31 kHz
a	1.06×10^{-7}	1.97×10^{-8}	3.26×10^{-9}
b	1.9217	1.9567	1.9616

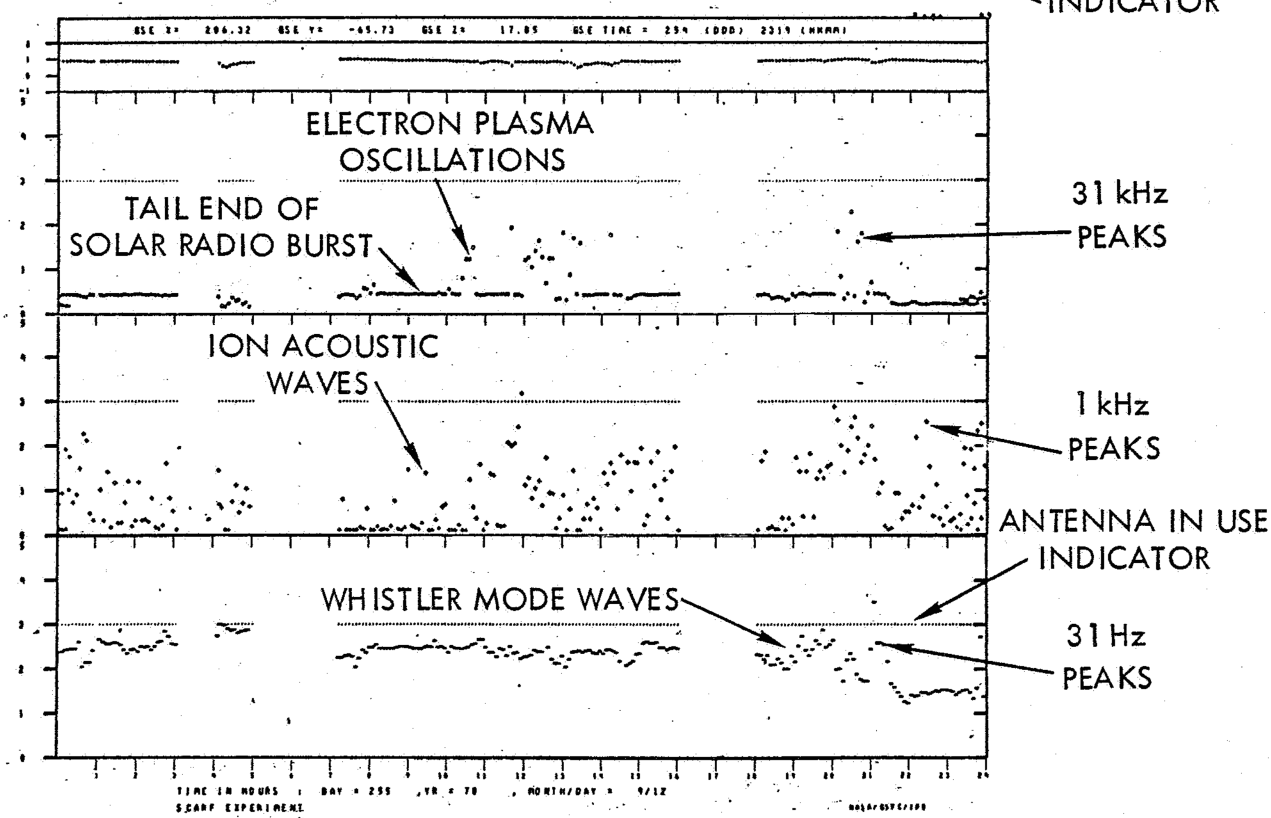
For the short antenna, we must use E(short) = 74 x E(long). The B-sensor can be connected to the 16-channel analyzer and in this case we can use the same b-values with a(B) = 3.2×10^{-5} $\gamma/(Hz)^{1/2}$ at 31 Hz and a(B) = 9.7×10^{-7} $\gamma/(Hz)^{1/2}$ at 1 kHz (the search coil is not sensitive at 31 kHz). For the common data_pool plots the sensor-in-use is given by the horizontal dotted line, according to the following code: 1 volt level for short electric, 2 volt level for U antenna, 3 volt level for V antenna, and 4 volt level for search coil.

ISEE-3
PRODUCTION
PLOT
16-CHANNEL
ANALYZER

(128 SECOND
PEAKS AND
AVERAGES)



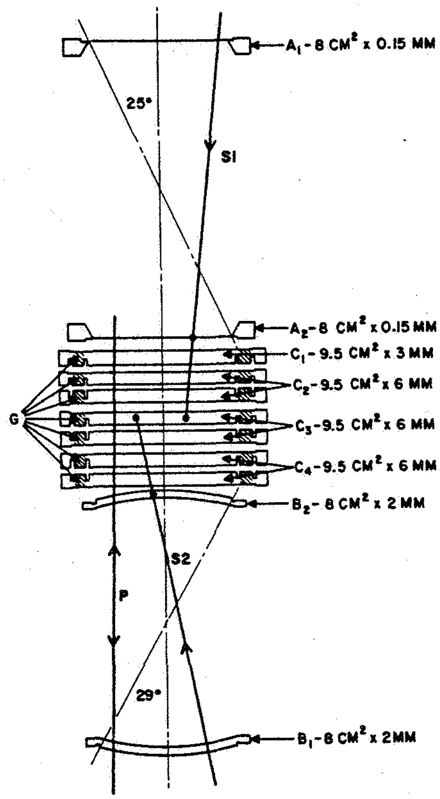
PLOT FROM
ISEE-3
COMMON
DATA POOL
TAPE
(5-MINUTE
PEAKS)



NOTES ON THE 4-57 MeV AND 18-70 MeV
PROTON + ALPHA COUNTING RATES

by
T. von Rosenvinge

The ISEE-3 Medium Energy Cosmic Ray Experiment has two High Energy Telescopes (HETs) which are designed to measure the charge composition of energetic particles from charge 1 to charge 26 over an energy range from a few MeV/nucleon to several hundred MeV/nucleon. One of these telescopes is shown schematically in Figure 1; the telescope is cylindrically symmetric around the vertical axis in the figure. Protons (and alphas) which enter A_1 and A_2 but not C_4 or the guard lie in the energy interval 4-57 MeV/nucleon (the guard "detector" is the composite of the ring detectors which encircle each of the "C" detectors; cf. Figure 1). Such events are counted by rate counters during the half of the time when each telescope is in high gain. This rate is characterized by the coincidence condition $A_1 \cdot A_2 \cdot \overline{C_4} \cdot \overline{G_1}$. Particles detected to satisfy this condition are referred to as A-Stopping (or AST) events. Protons (and alphas) which enter B_1 and B_2 but not C_1 or the guard lie in the energy interval 18-70 MeV/nucleon. Such B-stopping (or BST) events are characterized by the coincidence conditions $B_1 \cdot B_2 \cdot SB \cdot \overline{C_1} \cdot \overline{G_1}$. These rates are also available less frequently as sectored rates.



HIGH ENERGY TELESCOPE (HET)

Figure 1

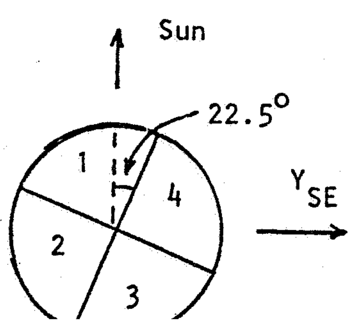
The spirit of the data pool tapes is that it should be "quick and dirty", i.e. in return for simplified algorithms for data analysis it will be possible for a wide community to gain access to data from a variety of experiments simultaneously and long before detailed data reduction can be completed. In this spirit we provide 15 minute averaged values of the counting rates $A_1 \cdot A_2 \cdot \overline{C_4} \cdot \overline{G_1}$ and $B_1 \cdot B_2 \cdot SB \cdot \overline{C_1} \cdot \overline{G_1}$ from one of the HETs (HET-II). In our own detailed data reduction we will use pulseheight analysis data for the A-stopping and B-stopping events to remove background events, to separate each charge, to correct for edge effects, and to take into account the energy dependence of the geometry factor (for AST the geometry factor for each telescope varies from .82 to 1.24cm^2-steradian and for BST the geometry factor for each telescope varies from .82 to 1.69cm^2-steradian). We will also examine sectored rates to determine spin aliassing and we will assess dead-time corrections at high counting rate levels ($\gtrsim$5,000 counts/sec). None of these is considered in the algorithm which has been used for the Data Pool Tape.

Low Energy Proton Experiment

The low energy proton experiment on ISEE-3 consists of three identical telescopes, inclined at 30°, 60° and 135° respectively, to the spin axis of the spacecraft. Each telescope measures protons in the energy range 35 keV to 1. MeV, in 8 logarithmically spaced channels. The output of each telescope is separately stored for each of 8 equi-angular sectors, and this applies to every energy channel except channel 8, which has only 4 equi-angular sectors. A general outline of the experiment is given by Balogh et al (Geoscience Electronics GE16, 176, 1978). Detailed descriptions are to be found in Balogh and Iversen (Space Sci.Instrum. 3, 187, 1977) and van Rooijen et al (Space Sci.Instrum. - to be published 1979).

The data on the Data Pool tape is taken from the telescope inclined at 60° to the spin axis i.e. 30° from the equatorial plane of the spacecraft. Two particle fluxes are supplied, obtained from the spin-averaged values for Channel 3 and Channel 4 combined, and from Channel 8. These correspond to energy ranges 91-237 keV, and 1.0 MeV to 1.6 MeV, respectively. The data are averaged over 21 minutes and 20 seconds, so that three sets of values are supplied per data record on the tape. The start of the first averaging period corresponds to the first time in the record. The fluxes are expressed in p cm^{-2} $sterad^{-1}$ sec^{-1}, using a geometrical factor of 0.05 cm^2 sterad.

A measure of particle anisotropy is also supplied. It is taken from Channel 8 and consists of the value $\frac{I_{MAX} - I_{MIN}}{I_{MAX} + I_{MIN}}$, where I_{MAX} and I_{MIN} are the maximum and minimum counting rates observed in the four sectors of Channel 8. Also given is the number of the quadrant in which I_{MAX} was observed i.e. 1 to 4. If $I_{MAX} \sim I_{MIN}$, the number 5 is inserted in this value. The convention for the numbering of quadrants with respect to the spacecraft-sun line is shown in figure 1, which is a projection in the ecliptic plane, looking down from the positive Z direction (S.E co-ordinate system).



In interpreting the data caution needs to be used, and it is strongly recommended that all users of the data verify and/or clarify with the Principle Investigator any uncertainties or peculiarities concerning the data.

To assess the statistical accuracy of the data it should be remembered that converting the flux back into a counting rate, using the geometric factor and the time averaging period will only give the minimum error, since some data may have been rejected in forming the average.

So far as the physical significance of the data is concerned it should be remembered that many of the events seen by this experiment are strongly collimated in azimuth as well as longitude, so that the intensity seen by one telescope may be significantly lower than the intensity observed in the direction of maximum intensity relative to the ecliptic plane. With regard to the anisotropy measurement it should be remembered that many of the events observed have steep energy spectra, so that the statistics in Channel 8 may be insufficient to record a very strong anisotropy which exists in the lower energy channels.

ISEE-3 PRINCIPAL INVESTIGATORS

Dr. Kinsey A. Anderson
University of California
Space Science Laboratory
Berkeley, California 94720
Tel. 415-642-1313

Dr. S. J. Bame
Los Alamos Scientific Lab.
P. O. Box 1663
Los Alamos, New Mexico 87545
Tel. 505-667-5308

Dr. Harry H. Heckman
University of California
Lawrence Berkeley Lab.
Berkeley, California 94720
Tel. 415-843-5685

Dr. D. Hovestadt
Max-Planck-Institut
fur Physik and Astrophysik
Institut fur Extraterrestrische
Physik
8046 Garching bei Munchen
Germany
Tel. 49-089-329-9817

Dr. Robert Hynds
Dept. of Physics
Imperial College of Science & Tech.
10 Prince's Gardens
London, England, SW7-INA

Dr. Peter Meyer
University of Chicago
Enrico Fermi Institute
933 E. 56th Street
Chicago, Illinois 60637
Tel. 312-753-8543

Dr. K. W. Ogilvie
Code 692
NASA/GSFC
Greenbelt, Maryland 20771
Tel. 301-344-5904

Dr. F. L. Scarf
TRW Systems Group, R1/1176
One Space Park
Redondo Beach, California 90278
Tel. 213-536-2015

Dr. E. J. Smith
Jet Propulsion Laboratory
4800 Oak Grove Drive
Pasadena, California 91103
Tel. 213-354-2248

Dr. J. L. Steinberg
Dept. de Recherches Spatiales
Observatoire de Meudon
92190 Meudon
France
Tel. 626-16-30 X677

Dr. E. C. Stone
Calif. Institute of Tech.
Physics Department
Mail Station 220-47
Pasadena, California 91109
Tel. 213-795-6841 X1516

Dr. Tycho von Rosenvinge
Code 661
NASA/GSFC
Greenbelt, Maryland 20771
Tel. 301-344-6721

Dr. J. M. Wilcox
Stanford University
Stanford Electronics Lab.
Via Crespi Street
Stanford, California 94305
Tel. 415-497-2300

SEICC.DPTLIST.SOURCE (DPLIBLST)

NOTES ON THE

ISEE-3

DATA POOL TAPE

M..D. Banks, Jr.
T. von Rosenvinge

May 1979

I. Introduction

The International Sun-Earth Explorer Mission is a joint NASA/ESA project intended to study the earth's magnetosphere and its response to disturbances in the solar wind. The ISEE-3 spacecraft is positioned ~240 earth radii upstream of the earth's bow-shock and observes the solar wind flowing towards the earth while the ISEE-1 and 2 spacecraft make observations in or near the magnetosphere. This project has been described in detail by Ogilvie, et al. (1977).* The primary purpose of the ISEE-3 data pool tape is to make basic quantities measured at ISEE-3 readily available beyond the individual experiment groups making the measurements. This is particularly desirable since the emphasis of this mission is on utilizing simultaneous data from all three spacecraft. The data pool tape does have some limitations, however. For example, the time resolution and selection of data is limited. Also, the algorithms for transforming measured quantities into physical units are generally not as complex as those that experimenters may eventually use in reducing their data. On the other hand, many users will profit more by quick access to somewhat imperfect data than by eventual access to more refined data. For instance, an experimenter can use the data pool tape to identify interesting time periods and hence greatly reduce the volume of refined data which he may request from another experimenter.

The data pool tape is produced at Goddard Space Flight Center by the Information Processing Division (IPD) using algorithms provided by each experimenter. IPD does its best to process the data accordingly, but it is staffed by programmers and not scientists and hence cannot be held responsible for identifying, for example, subtle changes in instrument performance, limitations of experiment response, or interference to an experiment.

In order, then, for a user to make sensible use of the data pool tape he requires a good description of each experiment, a description of the tape format and a description of the algorithms used. The first of these has been provided in the IEEE transactions on Geoscience Electronics, July 1978, Volume GE-16. It is intended that the remaining items be supplied in part by the present document. Following this introduction, we have provided notes on the ISEE-3 data pool tape together with the tape format. This precedes sections which have short write-ups from each experimenter regarding the method by which their data is reduced to yield the quantities on the tape and appropriate caveats. Finally there is a list of Principal Investigators with addresses and telephone numbers.

* Ogilvie, K. W., von Rosenvinge, T. T. and Durney, A. C., Science, **198**, #131, 1977.

NOTES ON THE ISEE-3 DATA POOL TAPE

GENERAL DESCRIPTION AND USAGE

1. Structure of the Data Pool Tape

(A) Tapes

Each data pool user receives one reel of tape per week. This tape may be 7-track 556 bpi, 9-track 1600 bpi, or 9-track 800 bpi, depending upon the user's preference. All tapes are odd parity. Inter-record gaps are .65 inch for 9-track tapes and .75 inch for 7-track tapes.

(B) Files

Data pool information for a 7-day period is presented as a single tape file, beginning with a label record and ending with a standard tape mark (EOF mark). All records, including the label, are of the same length. The data pool file contains approximately 160 data records spanning 7 days of telemetry data. Redundant telemetry data (due to overlap in ground station coverage) has been removed. The data pool file coincides in time with a "shipping group" of the usual telemetry data (decom tapes) which is sent to each experimenter.

The data pool file appears 3 times on the tape for redundancy backup. See Figure 1.

(C) Data Words

(1) Word Size

Each data pool tape is written in computer words of a length compatible with the intended user's computer. Tapes thus constructed can be read directly into the user's computer with no reformatting. This holds true for both 7-track and 9-track tapes. For example, records intended for use with a CDC 6000 series computer would appear as packed strings of 60-bit fields. On a 9-track tape, each such 60-bit field occupies 7-1/2 tape characters. But the total number of words per record is an even number, so that the 60-bit fields can all be written in pairs, 15 tape characters per pair, and thus can be read normally by the CDC. (Other combinations of word size and tape type work out in a similar fashion).

- ONE TAPE PER 7-DAY DATA GROUP.
- ONE FILE PER DATA GROUP, REPEATED 3 TIMES FOR BACKUP.
- DATA POOL QUANTITIES ARE IN USER COMPATIBLE FLOATING POINT AND USER WORD LENGTH.
- DATA RECORDS ARE APPROXIMATELY 1 HOUR.

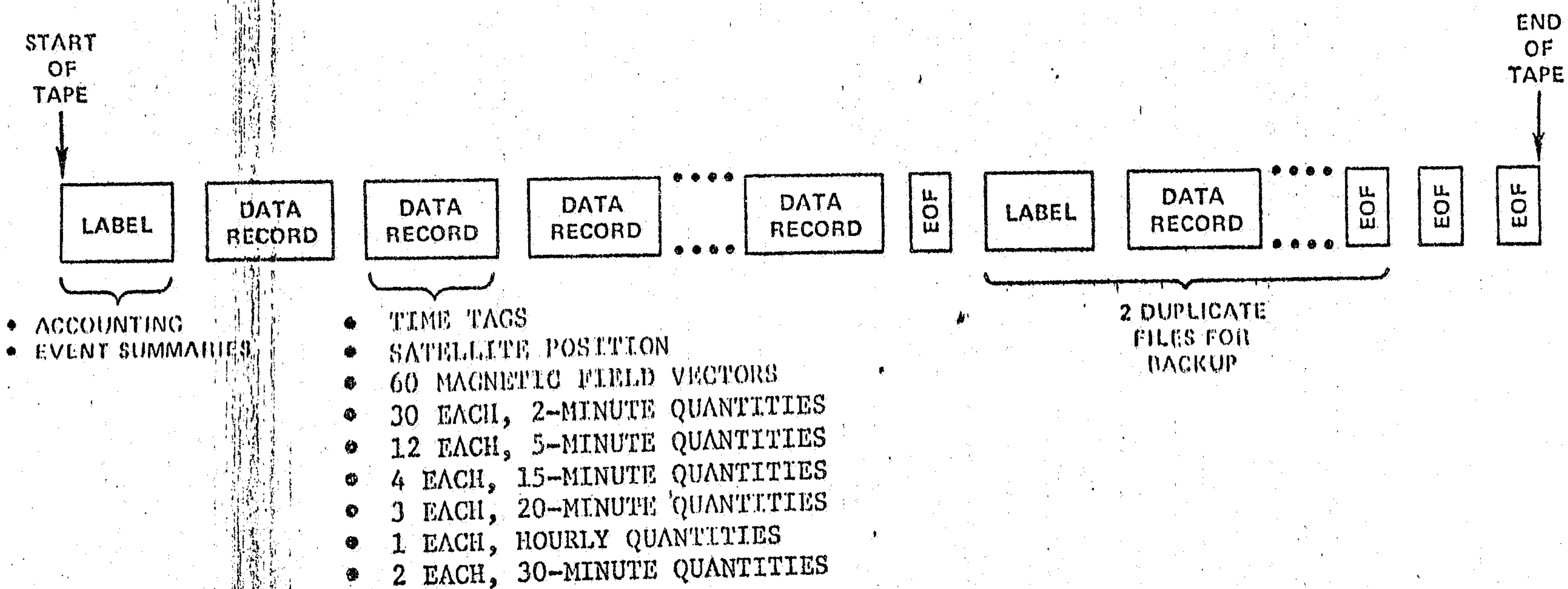


Figure 1. Data Pool Tape

(2) Floating Point Format

The entire data pool tape, the file label record and all data records, are in floating point format. (This provides a uniform, standard appearance of all information, facilitates conversion to the various computer word sizes, and simplifies tape printing and verification). The floating point representation used on each tape is specified by the user, and, like the word size, is compatible with the computer which will process the tape.

Most data pool tape quantities are originally computed in floating point and should be interpreted as such. Certain items, however, are obvious as integer values converted to floating point representation (day-of-year, seconds-of-day, clock, various indicator flags, etc.). The user may interpret such items either as floating point (in which case appropriate precautions should be taken to prevent possible truncation during conversion to integer).

(3) Missing Data

Missing data items are indicated by a negative value fill code in place of the missing item. The exact value of the fill code is dependent on the type of floating point used, but will in all cases be outside the legitimate range of any data item.

Missing data pool items may be the result of gaps in data recovery at the ground station, or the result of data being rejected by one of the experiment processing algorithms.

It is possible that uncertain conditions may lead to a data pool result of questionable validity. Rather than be rejected, such results may, in certain cases, be presented as the negative of the actual result. Thus, a negative number other than the fill code, if in a field which should normally be positive, represents a doubtful result and may be used, but with caution. (Note that this does not apply to those values which may normally be negative, such as magnetic field components).

Data taken when the spacecraft is in engineering format is rejected by the data pool program.

II. Contents of the Data Pool Tape

(A) Time

The time values given on the data pool tape are UTC (Universal Time Code) at the time the data was transmitted from the spacecraft (i.e. transmission lag time has been removed). Times have been smoothed to remove random errors and then verified by intercomparison among all the ground stations. Time is given as Julian day (1-366) and seconds-of-day.

(B) Clock

The clock used on the data pool tape is a minor frame counter. It is constructed by combining the 24-bit raw spacecraft clock with part of the frame counter. (This is the same clock as used on the telemetry data tapes).

Since the full clock will not fit in all types of floating point works without truncation, it is broken into 2 pieces, the low order 21 bits in one word, the remaining high order portion in another. The full clock can be reconstructed by converting both clock pieces to integer, then adding them together.

It should be noted that time, rather than clock, is the primary reference for data pool items. Clock is subject to rollover within the data pool file, as well as to unpredicted jumps forward or backward.

(C) Timelines

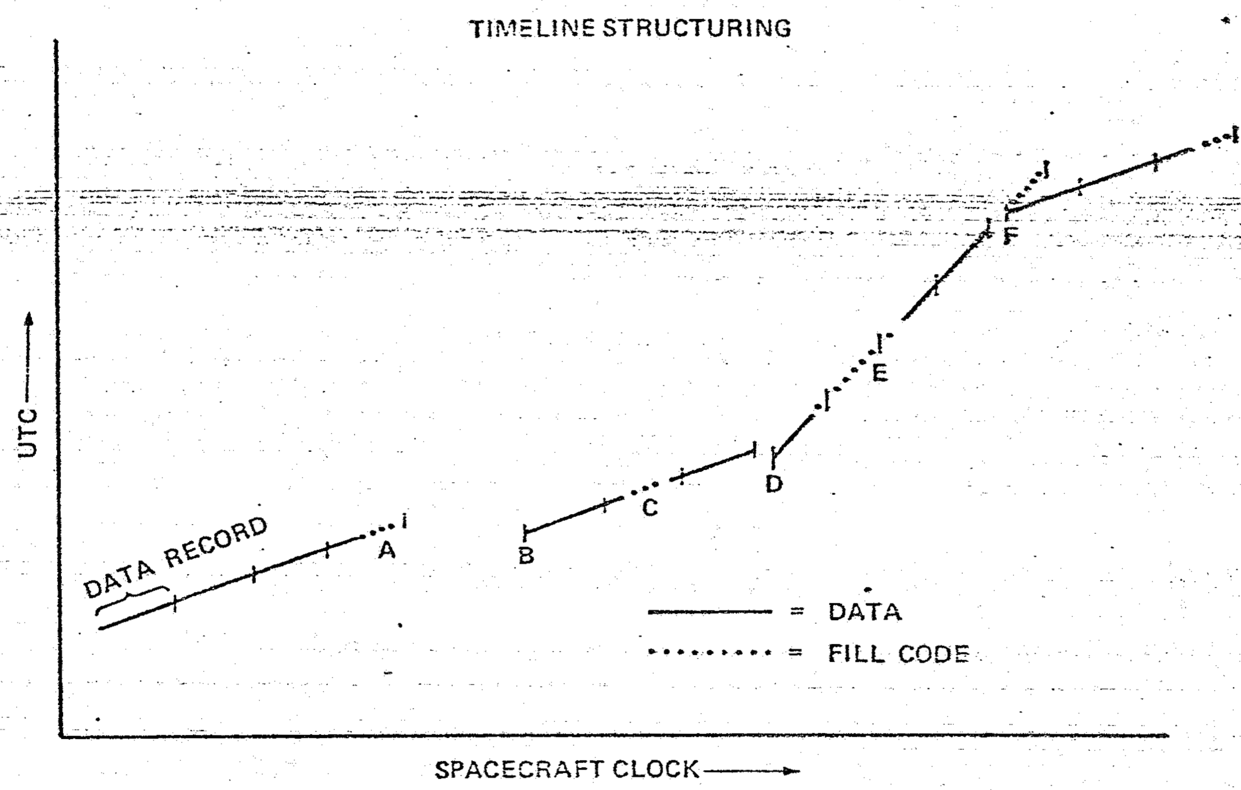
The time versus clock relationship, may not be linear throughout the entire data pool file. Breaks occur if the bit rate changes, at end-of-year, and if the clock jumps or rolls over. A segment of data in which time versus clock is linear is defined as a time/clock baseline or "timeline."

Data accumulation intervals for experiment algorithms do not continue across timelines; that is, each data pool quantity is computed using data which begins and ends in the same timeline. See Figure 2.

(D) Data Records

Data records are fixed length, 810 words long. A full record holds 64 minutes of information.

Within a timeline, each data record represent tition in time. Data items are positioned within the records by time, relative to the start of the record (see "Time Tagging," below). Fill code is substituted where data is unavailable. If a gap in data coverage greater than 64 minutes occurs, it is possible that an entire record will be fill code. In this case the dummy record indicator is turned on.



Spacecraft Clock

A. Last record of timeline contains fill beyond the end of the timeline.

B. A new timeline begins due to a clock reset. No gap in data coverage.

C. Data gap within a record results in fill code.

D. New timeline begins due to a bit rate change.

E. Large data gap results in an entire record of fill code. Note that the record following the all-fill record begins with fill, but has a start time assigned as if data were present.

F. New timeline begins due to a bit rate change.

Figure 2. Timeline Structuring

The format of the data record is given in Table 2.

(E) Time Tagging

There are seven types of data pool information, according to frequency of readout: (1) 60 per record, 1-minute intervals, (2) 30 per record, 2-minute intervals; (3) 12 per record, 5-minute interval; (4) 4 per record, 15-minute intervals; (5) 3 per record, 20-minute intervals; (6) 2 per record, 30-minute intervals; (7) once per record. ("Minute," as used here, means an ISEE minute," or 64-seconds independent of bit-rate.

The start time of the data record (words 1 and 2) is the start time of sampling interval number at all seven frequencies of readout. The start times of subsequent intervals are computed relative to interval number 1.

Examples (Refer to Table 3):

Example 1 — The magnetic field vector {Bz(1), Bx(1), By(1)}, in words 201-203 was computed for the 64-second interval beginning at the record start time.

The vector {Bz(60), Bx(60), By(60)} , in words 555-557, was computed over the 64-second interval beginning at t=(record start time) + (59 x 64 seconds). Similarly, vector {Bz(3), Bx(3), By(3)} , words 213-215, was computed over the 64-second interval beginning at t=(record start time) + (128 seconds).

Example 2 — Find the energetic particles flux, energy >15 MeV, at a point in time 20 minutes from the start of the record.

The required information set is from the Anderson algorithm, words 681-682, labeled EFLUX (1) - EFLUX (12). This data is given at intervals of 5 ISEE-minutes or every 320 seconds. Let RST equal the record start time. Then,

$$RST + 20 \text{ min} = RST + 1200 \text{ sec} = RST + 3.75 \text{ intervals}$$

The desired value would thus be best approximated by interval No. 4, word 684.

(F) File Label

The file label record, Table 1, contains identification and accounting information for the data pool file. It also contains the minimum and maximum spin periods encountered over the 7-day file period, an index to all timelines in the file, and magnetometer parameters. The record is padded to the length of the data records.

As indicated in Table 1, the first 1440 bits should be ignored by the user. These bits are used for internal accounting purposes by Goddard Space Flight Center.

Table 1: DATA POOL FILE LABEL

Handwritten	WORD NUMBER*	DESCRIPTION (ALL VALUES ARE FLOATING POINT)
	1	
		1440 BITS FOR GSFC INTERNAL USE.
45	N	
46	N+1	SATELLITE ID NUMBER
47	N+2	INTENDED RECIPIENT OF THIS TAPE. (SEE TABLE 2)
48	N+3	YY,START OF FILE, 2 DIGITS OF YEAR.
49	N+4	DDD,START OF FILE,DAY OF YEAR.
50	N+5	SSSSS,START OF FILE,SECONDS OF DAY.
51	N+6	YY,END OF FILE,2 DIGITS OF YEAR.
52	N+7	DDD,END OF FILE,DAY OF YEAR.
53	N+8	SSSSS,END OF FILE,SECONDS OF DAY.
	N+9	HIGH ORDER BITS. CLOCK AT START OF THE DATA POOL FILE.
	N+10	LOW ORDER 21 BITS. CLOCK AT START OF THE DATA POOL FILE.
	N+11	GROUP NUMBER (CORRESPONDING TO THE TELEMETRY DATA TAPE GROUP NUMBER)
	N+12	MINIMUM VALUE OF SPIN PERIOD FOUND WITHIN THIS FILE IN SECONDS.
	N+13	MAXIMUM VALUE OF SPIN PERIOD FOUND WITHIN THIS FILE IN SECONDS.
	N+14	SMH1 Z-OFFSET USED FOR THIS RUN.
	N+15	SMH2 NUMBER OF ESTIMATES MADE FOR Z-OFFSET ABOVE.
	N+16	SMH3 ALPHA USED FOR Z-OFFSET ABOVE.
	N+17	SMH4 GROUP NUMBER OF THE DATA GROUP USED TO DETERMINE Z-OFFSET.
	N+18 . . . N+80	SPARES.
	N+81	NUMBER OF TIME LINES (MAXIMUM OF 80)
	N+82	START DAY OF YEAR (1).
	N+83	START SECONDS OF DAY (1).
	N+84	HIGH ORDER BITS OF THE SPACECRAFT CLOCK (1).
	N+85	LOW ORDER 21 BITS OF START SPACECRAFT CLOCK (1).
	N+86	BIT RATE (1.0 FOR 512 BPS, 2.0 FOR 1024 BPS AND, 4.0 FOR 2048 BPS)
	N+87	START RECORD NUMBER.
	. . .	
	N+656	START DAY OF YEAR (80).
	N+657	START SECONDS OF DAY (80).
	N+658	HIGH ORDER BITS OF START SPACECRAFT CLOCK (80).
	N+659	LOW ORDER 21 BITS OF START SPACECRAFT CLOCK (80).
	N+660	BIT RATE (1.0 FOR 512 BPS, 2.0 FOR 1024 BPS AND, 4.0 FOR 2048 BPS)
	N+661	START RECORD NUMBER (80).
	N+662 . 810	FILL TO EQUAL DATA RECORD LENGTH.

Table 2: DATA POOL – DATA RECORD

WORD NUMBER	DESCRIPTION (ALL VALUES ARE FLOATING POINT)		
1	DAY OF YEAR,RECORD START		
2	SECONDS OF DAY,RECORD START		
3	CLOCK,RECORD START. HIGH ORDER PORTION		
4	CLOCK,RECORD START. LOW ORDER 21 BITS		
5	RECOVERY FACTOR: (MINOR FRAMES PROCESSED)/(7.5 X 256.),FOR 512 BPS (MINOR FRAMES PROCESSED)/(15 X 256.),FOR 1024 BPS (MINOR FRAMES PROCESSED)/(30 X 256.),FOR 2048 BPS		
6	BIT RATE: 1.0 = 512 BPS (BACKUP) 2.0 = 1024 BPS (LOW) 4.0 = 2048 BPS (HIGH)		
7	DUMMY RECORD INDICATOR: 0.0 = AT LEAST ONE MINOR FRAME OF DATA WITHIN THIS RECORD'S SP 7.0 = NO DATA WITHIN THE SPAN OF THIS RECORD. A DUMMY RECORD.		
8	TIMELINE INDICATOR: 0.0 = THIS RECORD LIES ON AN EXISTING TIMELINE 7.0 = THIS RECORD BEGINS A NEW TIMELINE		
9	DATA RECORD NUMBER		
10 - 12	SPARES		
13	B0-X	OFFSET USED FOR SMH BX	
14	B0-Y	OFFSET USED FOR SMH BY	
15			
16	WORDS 15 TO 19 FOR SMH USE ONLY		
17			
18			
19			
20	SPIN PERIOD AVERAGE,PREVIOUS HOUR.		
21	GSE-X		
22	GSE-Y SATELLITE POSITION VECTOR IN GSE COORDINATES		
23	GSE-Z (AT TIME OF FIRST POINT IN THIS RECORD)		
24-168	SPARES		
	* * * * * * * * * HOVESTADT ALGORITHM * * * * * * * * * *		
169	PROLP(1)	0.17-0.4MEV PROTONS	1ST OF 4
.			
.			
172	PROLP(4)	0.17-0.4MEV PROTONS	4TH OF 4
173	ALFLA(1)	0.12-0.25MEV ALPHAS	1ST OF 4
.			
.			
176	ALFLA(4)	0.12-0.25MEV ALPHAS	4TH OF 4
177	HEAVYS(1)	HEAVIES (Z>2) GT 0.1MEV	1ST OF 4
.			
.			
180	HEAVYS(4)	HEAVIES (Z>2) GT 0.1MEV	4TH OF 4
181	PROHP1(1)	5-10MEV PROTONS	1ST OF 4
.			
.			
184	PROHP1(4)	5-10MEV PROTONS	4TH OF 4

Table 2 (Continued)

```
185        PROHP2(1)             10-20MEV PROTONS                    1ST OF 4
 .
 .
188        PROHP2(4)             10-20MEV PROTONS                    4TH OF 4

 * * * * * * * * * MANEUVER INFORMATION * * * * * * * * * *

189        MANUVR(1)      MANEUVER INDICATORS FOR EACH OF THE TWELVE
190        MANUVR(2)      5 MINUTE(APPROX) INTERVALS OF THIS RECORD:
191        MANUVR(3)                  0.0 = NO MANEUVER IN THIS INTERVAL
192        MANUVR(4)                  7.0 = MANEUVER INDICATED DURING THIS INTERVAL
193        MANUVR(5)
194        MANUVR(6)
195        MANUVR(7)
196        MANUVR(8)
197        MANUVR(9)
198        MANUVR(10)
199        MANUVR(11)
200        MANUVR(12)

 * * * * * * * * * SMITH ALGORITHM (MAGNETOMETER) * * * * * * * * *

201        BZ(1)                 SPIN AXIS COMPONENT
202        BX(1)                 SATELLITE-SUN LINE COMPONENT   MAG FIELD VECTOR
203        BY(1)                 3RD COMPONENT OF TRIAD             1ST OF 60 VECTORS
204        BMAG(1)               MAGNITUDE,
205        BDELTA(1)             LATITUDE          MAGNETIC FIELD VECTOR
206        BPHI(1)               LONGITUDE             1ST OF 60 VECTORS

207-554    2ND THROUGH 59TH MAGNETIC FIELD VECTORS.

555        BZ(60)                SPIN AXIS COMPONENT
556        BX(60)                SATELLITE-SUN LINE COMPONENT    MAG FIELD VECTOR
557        BY(60)                3RD COMPONENT OF TRIAD             60TH OF 60 VECTOR
558        BMAG(60)              MAGNITUDE
559        BDELTA(60)            LATITUDE          MAGNETIC FIELD VECTOR
560        BPHI(60)              LONGITUDE             60TH OF 60 VECTORS

 * * * * * * * * * STEINBERG ALGORITHM * * * * * * * * * *

561        RAMAP1(1)             AVERAGE VOLTAGE AND RMS (1000KHZ.)
562        RARMS1(1)             1ST OF 30 SETS
 .
 .
619        RAMAP1(30)            AVERAGE VOLTAGE AND RMS (1000KHZ.)
620        RARMS1(30)            30TH OF 30 SETS

621        RAMAP2(1)             AVERAGE VOLTAGE AND RMS (250KHZ.)
622        RARMS2(1)             1ST OF 30 SETS
 .
 .
679        RAMAP2(30)            AVERAGE VOLTAGE AND RMS (250KHZ.)
680        RARMS2(30)            30TH OF 30 SETS
```

(2) - Sean Rami (3) Debby (4) Rami |B| θ φ measured from?

Table 2 (Continued)

(5)

* * * * * * * * * ANDERSON ALGORITHM * * * * * * * * * *

681	EFLUX(1)	PARTICLE FLUX, ENERGY > 15KEV	1ST OF 12
.			
.			
.			
692	EFLUX(12)	PARTICLE FLUX, ENERGY > 15KEV	12TH OF 12
693	XRAY(1)	COUNTS PER SECOND,20-37 KEV	1ST OF 12
.			
.			
.			
704	XRAY(12)	COUNTS PER SECOND,20-37 KEV	12TH OF 12

(6)

* * * * * * * * * BAME ALGORITHM * * * * * * * * * *

705	IONPD(1)	ION PSEUDO-DENSITY (PARTICLES/CC)	1ST OF 1
.			
.			
.			
716	IONPD(12)	ION PSEUDO-DENSITY (PARTICLES/CC)	12TH OF
717	WINDPS(1)	SOLAR WIND PSEUDO-SPEED (KM/SEC)	1ST OF
.			
.			
.			
728	WINDPS(12)	SOLAR WIND PSEUDO-SPEED (KM/SEC)	12TH OF 1
729	WINDPA(1)	SOLAR WIND PSEUDO FLOW ANGLE (DEG)	1ST OF 1
.			
.			
.			
740	WINDPA(12)	SOLAR WIND PSEUDO FLOW ANGLE (DEG)	12TH OF 1

(7)

* * * * * * * * * SCARF ALGORITHM * * * * * * * * * *

741	PLA31(1)	PLASMA WAVE 31HZ MAX. VOLTAGE	1ST OF 12
742	PLA1K(1)	PLASMA WAVE 1KHZ MAX VOLTAGE	1ST OF 12
743	PLA31K(1)	PLASMA WAVE 31KHZ MAX VOLTAGE	1ST OF 12
744	PLANT(1)	PLASMA WAVE ANTENNA STATUS	1ST OF 12
.			
.			
.			
785	PLA31(12)	PLASMA WAVE 31HZ MAX. VOLTAGE	12TH OF 12
786	PLA1K(12)	PLASMA WAVE 1KHZ MAX VOLTAGE	12TH OF 12
787	PLA31K(12)	PLASMA WAVE 31KHZ MAX VOLTAGE	12TH OF 12
788	PLANT(12)	PLASMA WAVE ANTENNA STATUS	12TH OF 12

(8)

* * * * * * * * * VON ROSENVINGE ALGORITHM * * * * * * * * * *

789	PARTLO(1)	PARTICLES, LOW RANGE 4-57MEV/NUCLEON	1ST OF
.			
.			
792	PARTLO(4)	PARTICLES, LOW RANGE 4-57MEV/NUCLEON	4TH OF
793	PARTHI(1)	PARTICLES, HIGH RANGE 18-70MEV/NUCLEON	1ST OF
.			
.			
796	PARTHI(4)	PARTICLES, HIGH RANGE 18-70MEV/NUCLEON	4TH OF

Table 2 (Continued)

(9)

* * * * * * * * * DE FEITER ALGORITHM * * * * * * * * * *

797	PROLO1(1)	PROTONS 78-205 KEV	1ST OF 3
798	PROLO2(1)	PROTONS 536-1400KEV	1ST OF 3
799	ISOTRO(1)	ISOTROPY INDEX	1ST OF 3
800	QUAD(1)	QUADRANT	1ST OF 3
•			
•			
•			
805	PROLO1(3)	PROTONS 78-205 KEV	3RD OF 3
806	PROLO2(3)	PROTONS 536-1400KEV	3RD OF 3
807	ISOTRO(3)	ISOTROPY INDEX	3RD OF 3
808	QUAD(3)	QUADRANT	3RD OF 3

(10)

* * * * * * * * * MEYER ALGORITHM * * * * * * * * * *

809	LOWEE(1)	LOW ENERGY 5-150MEV ELECTRONS RATE	1 OF 2
810	LOWEE(2)	LOW ENERGY 5-150MEV ELECTRONS RATE	2 OF 2

ISEE-3

MAGNETIC FIELD INVESTIGATION:

A Brief Description

of the

Experiment

and the

Data Pool

Processing Algorithm

A. M. A. Frandsen

2 October, 1978

TABLE OF CONTENTS

MAGNETIC FIELD INVESTIGATION

Principal Investigator: Edward J. Smith

Address: Jet Propulsion Laboratory
Building 183, Room 401
4800 Oak Grove Drive
Pasadena, California 91103
(213) 354-2248 (Commercial)
792-2248 (FTS)

Co-Investigators: Leverett Davis, Jr.
California Institute of Technology

Douglas E. Jones
Brigham Young University

George L. Siscoe
University of California, Los Angeles

Bruce T. Tsurutani
Jet Propulsion Laboratory

A. BRIEF EXPERIMENT DESCRIPTION

1. Experiment Objectives

The objectives of the ISEE-3 magnetic field experiment are to investigate the following:

(a) The effect of solar activity on the interplanetary magnetic field and solar wind parameters.

(b) The persistence, as well as changing character, of corotating features in the solar wind.

(c) Changes in large scale solar wind features over great distances, by correlating with data from planetary missions (Pioneer Venus, Pioneer Saturn and Voyager).

(d) Waves and other irregularities in the solar wind, their frequency content and phase relationships:

(e) The velocity of propagation and dispersion of field fluctuations between ISEE-3 and ISEE-1 & 2, while the latter are in the interplanetary medium.

(f) Plasma instabilities in the solar wind.

(g) The response of the magnetosphere to solar wind variations.

(h) The relationship between solar wind parameters and the extent and properties of the earth's magnetosheath.

2. Instrument Description

The ISEE-3 magnetometer experiment consists of a sensor mounted at the outboard end of a 3-meter boom and an electronics package located within the main body of the spacecraft. A Project-supplied data processing unit (DPU) provides 2-way buffering between the vector helium magnetometer (VHM) electronics package and the spacecraft data system. It conditions the spacecraft command and control signals for use by the instrument and it buffers the magnetometer outputs as they are read into the main telemetry stream. In addition, the DPU generates a magnetic sector pulse for on-board use by other experiments.

To simplify data reduction, the VHM sensor is mounted with its sensitive axes parallel to the principal axes of the spacecraft. The sensor detects the three components of the ambient steady-state magnetic field vector and fluctuations in these components up to 3 Hz. The operating principles of the VHM sensor are based on the effect of an external magnetic field upon the efficiency with which a population of metastable helium atoms may be optically pumped. Figure 1 illustrates

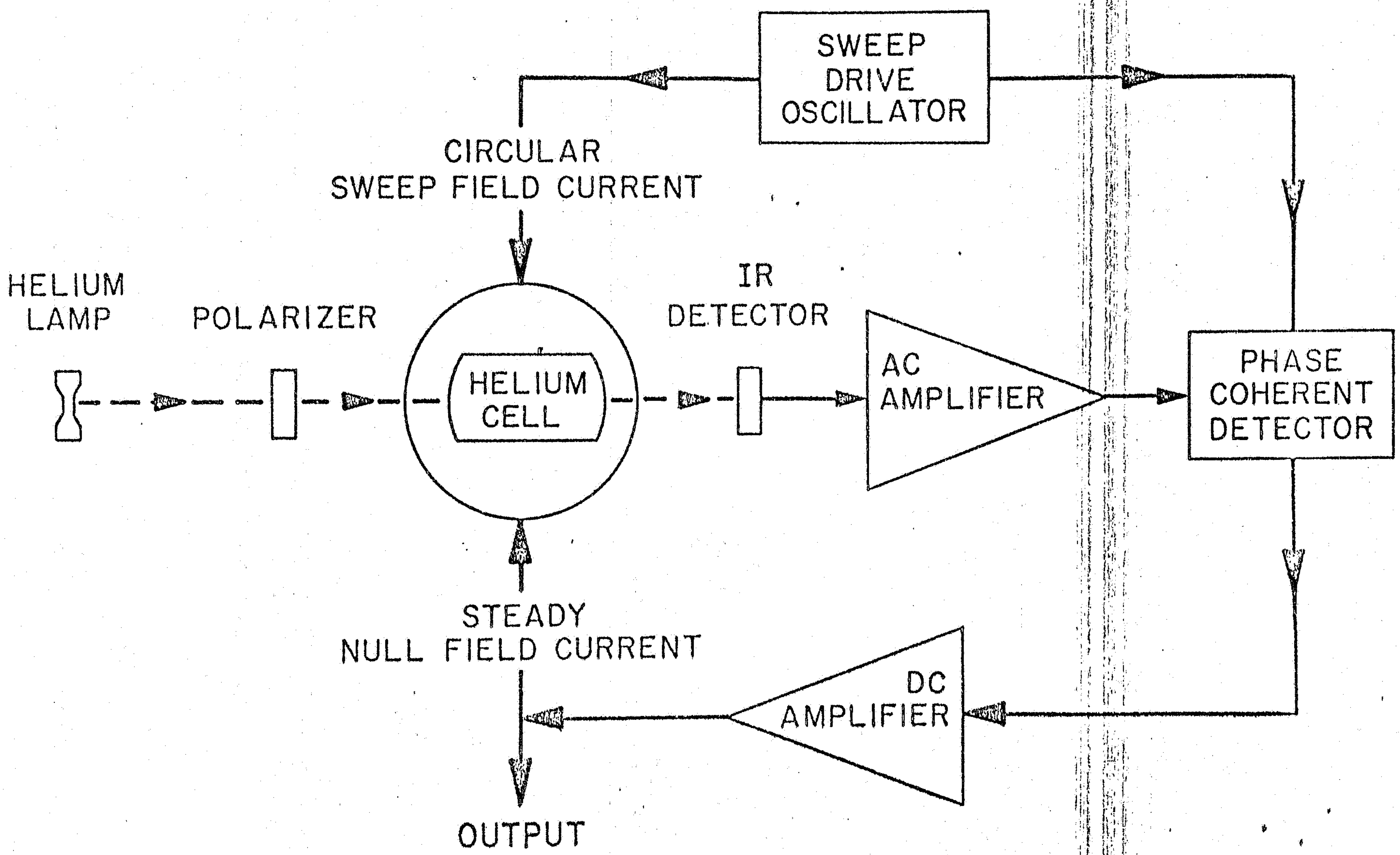


FIGURE 1 VECTOR HELIUM MAGNETOMETER SIMPLIFIED SCHEMATIC

several of the essential parts of the instrument. In the sensor, a collimated beam of infrared (IR) radiation from a helium lamp is circularly polarized to optically pump metastable helium atoms in the absorption cell. Properly phased sweep currents drive three mutually orthogonal coil pairs surrounding the cell and create a constant-amplitude rotating field vector. The sweep vector intensity modulates the emerging IR radiation. Brightness variations are then converted into a low frequency electrical signal at the IR detector. When no external magnetic field is present, only the second harmonic of the applied sweep frequency appears at the detector output. The main electronic assembly is designed to sense departures from the pure second harmonic signal. Synchronous demodulators are used to generate currents that are fed back to the helmholtz coils at the sensor. In this closed loop mode of operation, the ambient magnetic field acting upon the sensor is nulled to zero, and the three feedback currents accomplishing the nulling action are read as outputs. A multiplexer and a 9-bit analog-to-digital converter (ADC) are used in the process.

The three field components are sampled in rapid succession with less than 10 milliseconds of skewing. Each 9-bit conversion includes the sign of the component and eight bits of amplitude information. A 27-bit vector measurement is then combined with three additional bits that convey which of eight ranges the instrument is operating on. These data are held in a 30-bit storage register until a readout is requested by the spacecraft data system. At the nominal telemetry rate, (2048 information bits per second), six vector readings are acquired each second. They are nearly equally spaced in time (e.g., to within 4 milliseconds at 2048 ibps). The instrument's sampling rate scales directly with the prevailing telemetry rate.

The VHM has eight linear operating ranges giving a wide, effective dynamic range. Normally, it is operated in a mode where the upranging and downranging take place automatically. However, it can be commanded to any specific range through a sequence of ground commands. In either mode, all three axes switch range at the same time. Table I lists the eight operating ranges and the number of nanotesla (gamma) per least significant bit (LSB).

The instrument's calibration is checked in flight on a weekly basis by commanding the VHM in-flight calibration (IFC) sequence. Prelaunch mechanical alignment of the sensor was determined within 0°.1.

B. MAGNETOMETER DATA POOL ALGORITHM DESCRIPTION

1. Data Conversion

The ISEE-3 data pool algorithm for the magnetometer experiment converts telemetry counts into field units, performs intermediate calculations and averages the results. The basic relationship used for the conversion is:

$B_i = K_{i,r} \times (M_i - 255.5) - 0_{i,r}$, for i = X,Y or Z, and r = 0,1, ...7

Where: (a) M_i is the telemetry count of the ith component of a given measurement.

(b) $K_{i,r}$ is the ith axis scale factor (gamma/count) for VHM operating range r.

TABLE I

VHM OPERATING RANGES

Range Number	Full-Scale gamma (1 gamma = 10^{-9} tesla)	Sensitivity gamma/LSB
0	±4	0.015
1	±13	0.051
2	±43	0.17
3	±145	0.57
4	±632	2.5
5	$\pm3,870$	15.0
6	$\pm22,630$	88.0
7	$\pm136,000$	530.0

(c) 255.5 is the nominal zero level count for each 9-bit component measurement.

(d) $0_{i,r}$ is the offset field (gamma) at the sensor location with respect to the nominal zero level.

Both $K_{i,r}$ and $0_{i,r}$ are stored as 3 x 8 arrays. Provision is made in the algorithm for automatically updating the elements of $0_{i,r}$ using the magnetometer data from the previous time interval processed. The scale factors $K_{i,r}$ are checked weekly in flight but are not automatically updated since they have been found on earlier missions using similar instruments not to change significantly.

Once the measurements have been converted from telemetry counts into magnetic field units, each vector is transformed into an inertial coordinate system based upon the spacecraft spin axis and the direction toward the sun. At the nominal 2048 ibps telemetry rate, the data pool algorithm acquires updated atitude information from the spacecraft's sun sensor once every 32 seconds. For each field measurement, the time difference is determined between it and the last occurring valid sun pulse readout. Using telemetered knowledge of the spacecraft spin rate, the algorithm then performs a matrix multiplication which rolls each measurement back through an angle corresponding to the time difference. Vector-by-vector, the components are transformed into the inertial coordinate system based upon $\hat{S}$, the direction toward the sun and $\hat{I}$, the + Z spin axis of the ISEE spacecraft. These inertial coordinates are called the ISEE Inertial (IS) system and are illustrated in Figure 2.

After the magnetometer data have been despun into ISEE inertial coordinates, a vector magnitude and two angles are computed for every set of component values. Every vector measurement is

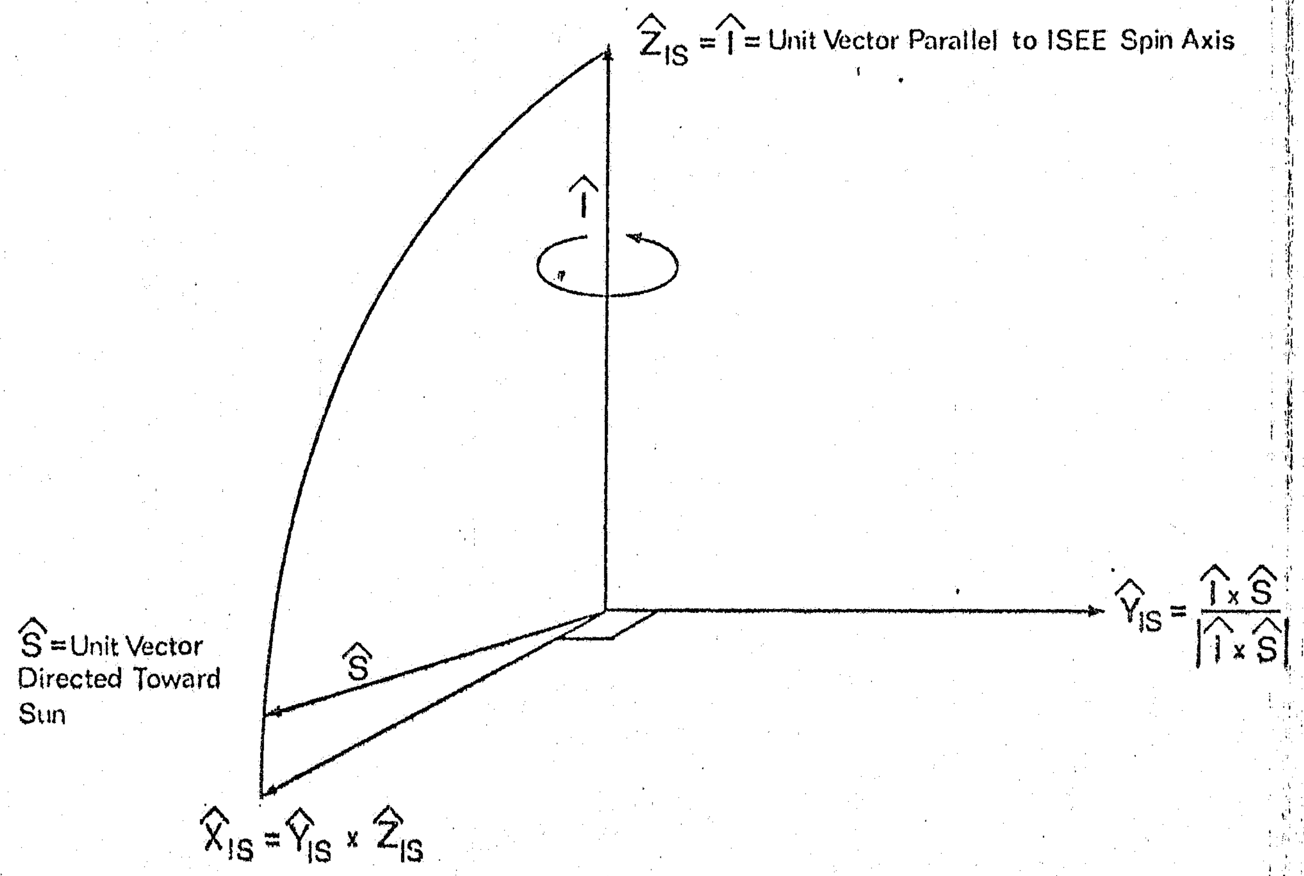


FIGURE 2 INERTIAL (IS) COORDINATE SYSTEM

therefore represented in both rectangular and spherical form. The angle δ_B is the field latitude in the IS system and ϕ_B the longitude angle. The six quantities thus determined are then separately averaged to form 1-minute averages appearing on the data pool tape:

$$\overline{B_{XIS}},\quad \overline{B_{YIS}},\quad \overline{B_{ZIS}},\qquad \overline{|B|},\quad \overline{\delta_B},\quad \overline{\phi_B}$$

Precautions are taken to avoid errors which can result from averaging azimuth angles that lie in the semicircle containing the 0°/360° branch cut.

Both spherical and rectangular field component averages are included in the pooled data because the two representations convey different information. The average field magnitude, for example, could be large and nearly constant during a natural disturbance while at the same time the field direction is so variable that the cartesian components average to small values. Thus, the field magnitude determined from the three component averages could be quite small. The difference in the two representations of B magnitude gives the power in the fluctuations, i.e.,

$$\sigma_B^2 = \left[\overline{|B|}\right]^2 - \left[\bar{B}_{XIS}^2 + \bar{B}_{YIS}^2 + \bar{B}_{ZIS}^2\right]$$

The user of the pooled data should be aware however that the six field averages are liable to be somewhat in error owing to the preliminary nature of the offset estimates used in the data reduction.

2. Offset Determinations

It is expected that when the pooled data are processed, the magnetometer offset will be known to 0.1 gamma. However, the precise values will always be lagging because an extended interval of data is required to determine them accurately, (e.g., ~ 1 month). When the algorithm uses biased estimates of the offset field components, the spacecraft spin frequency modulates B_{XIS} and B_{YIS}. The spin modulation is averaged out in the pooled data variables $\bar{B}_{XIS}$ and $\bar{B}_{YIS}$, but not in

$\overline{|B|}$, the average of the instantaneous vector magnitudes. Futhermore, biased estimates of the Z axis offset field component are reflected directly in the quantity $\bar{B}_{ZIS}$.

In order to keep the offset errors low, the algorithm computes hourly estimates of the two offset field components in the spacecraft spin plane by separately averaging the X and Y axis measurements. The new values are then used to update the spin plane offsets in the next hour's processing. The offset component parallel to the spacecraft spin axis is also estimated hourly by determining the value of Z axis field which minimizes the variance in the square of the field magnitude. However, from hour to hour these estimates are characterized by a fair amount of statistical scatter. Thus, the hourly spin axis offset field estimates are themselves averaged over one week and the result is used as the Z axis offset in the next week's processing. Many of the magnetometer parameters on the data pool tape are sums or products of the instantaneous vector components used in determining the offsets to be used on the following tape. The offset values used in creating the current tape are also given so that, if it becomes necessary, results may be corrected when more accurate values become available.

Attempts have been made in the data pool algorithm to subtract small interference fields known to originate on the spacecraft. Telemetered knowledge of the state of interfering subsystems is used for accessing a look-up table that gives the values to be subtracted from the magnetometer data. The algorithm also edits out data for time intervals during which the instrument is undergoing an in-flight calibration sequence, or when a spacecraft attitude maneuver is underway.

3. Sources of Error

Errors in the apparent field direction can come about not only when incorrect offset field estimates are used,

$$(e.g., \tan^{-1}\left[0.1\gamma \text{ error}/5\gamma \ |B|\right] \approx 1°),$$

but also through timing errors. Incorrect determinations of the occurrence of a sun pulse, or the time at which a vector sample was acquired, or an incorrect estimate of the spacecraft spin period can all lead to an error in the inertial field longitude. The algorithm assumes the nominal values for the spacecraft clock frequency and for the telemetry frame rate. Early indications are that the actual values are within 0.1% of the nominal frequencies. The resulting roll error in the pooled data is therefore less than 0.001 x 360° (max), or less than one-third of a degree due to this cause.

The algorithm also assumes the nominal mechanical alignment of the two oppositely directed spacecraft sun sensors, and of the VHM sensor itself. Prelaunch optical sighting and electronic calibrations showed the pulse coming from each sun sensor to be within 0.35° of nominal. Furthermore, all three magnetometer axes were found to be within 0°.1 of nominal. Generally speaking then, the probable error in field direction determined from the data pool tapes is approximately a degree or two. Larger errors can occur, however, when the ambient field becomes small compared to the typical interplanetary value at 1 AU. This error comes about because, in a fixed analysis time on a multirange instrument, the uncertainty in the offset field estimates is not necessarily reduced in direct proportion to the ambient field.

#4

THE 3-D RADIO MAPPING EXPERIMENT -- STEINBERG

Experiment Description

The principal purpose of this experiment is to map the trajectories of type III solar bursts by determining the angular coordinates of a localized source as a function of frequency and time. The radial distance may be obtained by triangulation with observations from another satellite, or from assumptions about the density of the interplanetary medium.

Two perpendicular dipole antennas are used. A 90 m tip-to-tip antenna in the spin plane of ISEE-3, referred to as the S antenna, sees a signal which is modulated because of the changing aspect of the source due to the spacecraft's rotation. The Z antenna is 14 m tip-to-tip, along the spin axis. From the S measurements, the azimuth and strength of the source are obtained. Comparison of S and Z observations provides the elevation of the source from the spin plane and an estimate of its angular diameter.

Measurements are made in 12 frequency channels, between 30 and 1980 kHz, in each of two bandwidths, 10 kHz (B), and 3kHz (N). Every 1.5 seconds (which is nearly one-half spin), one measurement of Z and 11 of S are made for one frequency channel in each bandwidth, interleaving B and N observations. This provides nearly the full range of modulation possible from the S antenna. (At data rates lower than 2048 bps, proportionally fewer S samples are taken.) The frequency channel is selected according to a fixed 72 step program, designed to observe each frequency at uniform intervals but with shorter intervals for the higher frequencies. Alternate modes of observation are possible using only the B or only the N bandwidth.

For any single measurement, the signal passes a logarithmic square-law detector so that the output receiver voltage V is roughly proportional to the logarithm of the antenna temperature T_A. V is digitized into 256 steps between 0 and 5.

Algorithm for the Data Pool Tape

Average voltages at two frequencies and the rms values are supplied on the data pool tape. The averages are made of all the S B samples at the selected frequency which were obtained in successive 128 second intervals (one major frame at the top data rate). The tape record contains 30 averages (one "ISEE hour") at 1000 kHz, each followed by its rms value, after which come 30 averages at 233kHz, each followed by its rms value. (if only N band observations are being made, averages are of all S N samples, and the lower frequency is 188kHz. There is no flag on the tape to denote this.)

The average voltage measures the strength of the signal. The rms values are a good indicator of the degree of modulation present; the larger the rms value, the more deeply modulated is the signal, denoting a narrow source near the spin plane. The full data record is needed for direction finding.

An approximate transformation between voltage and antenna temperature is

$$\log T_A = 5.938 + 0.712*V + 0.132*V^2 \qquad 1000\ \text{kHz}$$

$$\log T_A = 6.217 + 0.712*V + 0.132*V^2 \qquad 233\ \text{kHz}$$

Bame

#6

NOTES ON THE SOLAR WIND PLASMA PARAMETERS ON THE ISEE-3 DATA POOL TAPE

Quantities have been derived by approximate algorithms only; to distinguish these from more accurate quantities we have referred to these quantities as the pseudo wind speed, pseudo density, etc. The following caveats and comments apply:

1) IONPD (ion density) is in units of cm^{-3} and can have values ranging from 0 to 100 cm^{-3}. Because of a number of factors, including the fact that only relatively simple-minded algorithms can be used for the data pool tape, we can't quote an accuracy any better than a factor of two for the ion density.

2) WINDPS (speed) is in km sec^{-1} and we expect values ranging from 250 to 850 km sec^{-1}, with an accuracy of $\pm$ 5%.

3) WINDPA (direction in the plane of the ecliptic) is in degrees, and we expect values ranging between $\pm$ 15° of the solar direction with an accuracy of $\pm$ 3°.

No background corrections are made and the values are instantaneous as opposed to averages.

DESCRIPTIVE EXPLANATION - PLASMA WAVE DATA ON THE ISEE-3 COMMON DATA POOL TAPE

#7

by

Frederick L. Scarf

The ISEE-C plasma wave instrument has three spectrum analyzers with a total of 19 different frequency channels that cover the range from 0.3 Hz to 100 kHz. The main analyzer, which uses 16 continuously active automatic gain control amplifiers, gives two complete spectral scans per second in each of 16 filter channels. The instrument sensors include a high sensitivity magnetic search coil (B), and electric antennas with effective lengths of 45 meters (U-axis or V-axis) and 0.5 meter (short-E) [see Scarf et al. (1978) for a detailed description of the instrumentation]. The plasma wave output on the common data pool tape contains the peak values (uncalibrated output voltages; 5-minute accumulations) from three channels of the main 16-channel analyzer (31 Hz, 1 kHz and 31 kHz), along with an indicator of the antenna in use (U, V, short-E or B). The calibration data and a brief interpretation of the ISEE-3 plasma wave output on the data pool tape are given below.

The top part of Figure 1 contains a typical production plot of the full output from the 16-channel analyzer, and this serves as a reference for the data pool tape explanation. For all channels, this production plot has both peaks (isolated dots) and averages (the tops of the blackened regions) computed using 128-second accumulations of telemetry output values ($0 \leq \phi \leq 5$ volts), and the sensor-in-use is indicated below the time axis. This plot shows whistler mode activity (17.8, 31 and 56 Hz channels), impulsive ion acoustic wave bursts (sporadic activity with high peak-to-average ratios in the 311 Hz to 5.6 kHz channels) along with solar radio bursts, electron plasma oscillations and auroral kilometric radiation (on the 31 kHz, 56 kHz and 100 kHz channels). The bottom figure contains a plot of the common data pool tape output from the same day, and a comparison of the two drawings gives a general indication of the way to interpret the data pool plots.

The calibrated E-field spectral densities can readily be derived from the telemetry output plots because for each channel the E value [in volts/meter $(Hz)^{1/2}$] is related to the telemetry output (ϕ - 0 to 5 volts) by

$$E = a e^{b\phi}$$

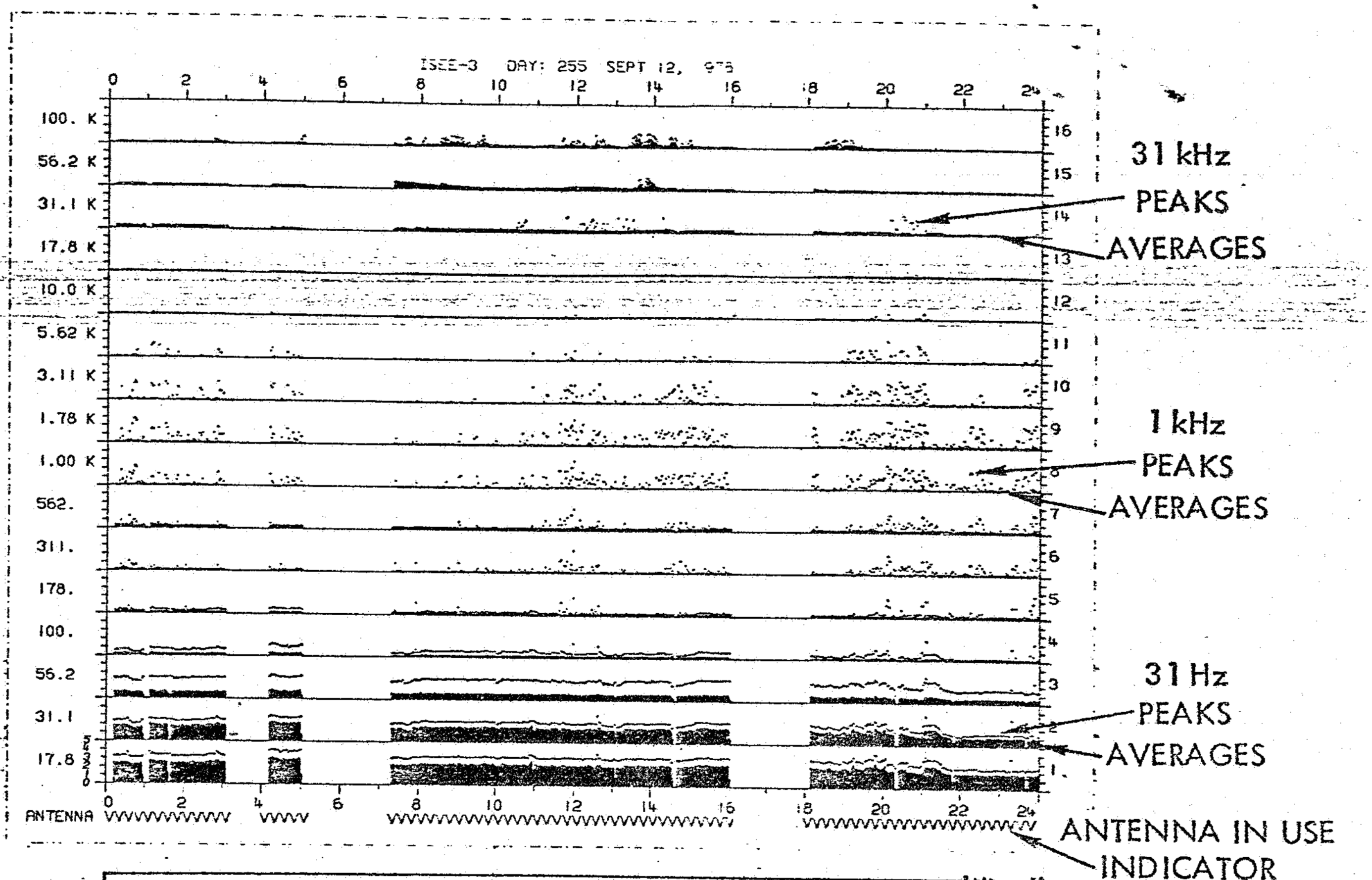
For either of the long (U- or V-axis) antennas, the calibration table is

Channel	31 Hz	1 kHz	31 kHz
a	1.06×10^{-7}	1.97×10^{-8}	3.26×10^{-9}
b	1.9217	1.9567	1.9616

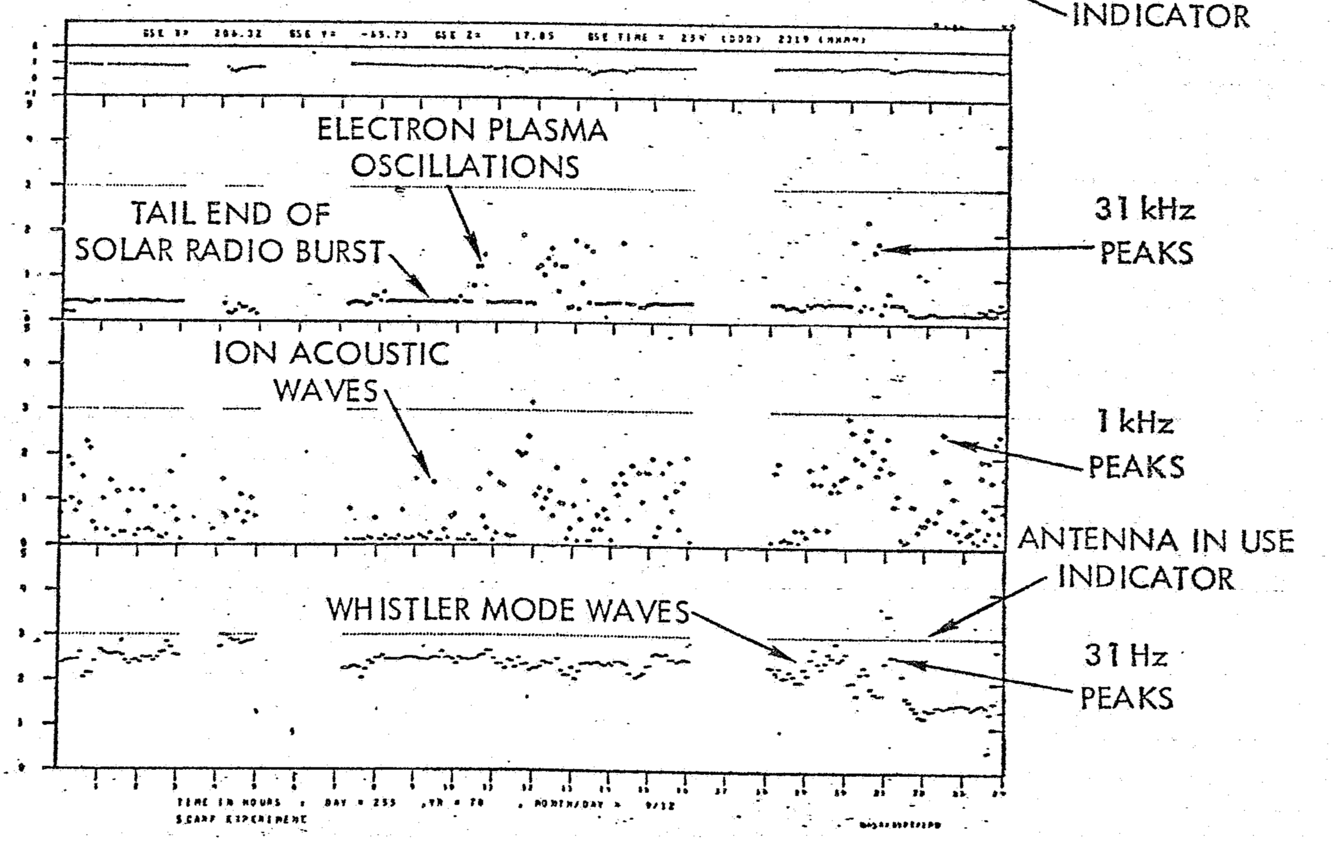
For the short antenna, we must use E(short) = 74 x E(long). The B-sensor can be connected to the 16-channel analyzer and in this case we can use the same b-values with $a(B) = 3.2 \times 10^{-5}\ \gamma/(Hz)^{1/2}$ at 31 Hz and $a(B) = 9.7 \times 10^{-7}\ \gamma/(Hz)^{1/2}$ at 1 kHz (the search coil is not sensitive at 31 kHz). For the common data pool plots the sensor-in-use is given by the horizontal dotted line, according to the following code: 1 volt level for short electric, 2 volt level for U antenna, 3 volt level for V antenna, and 4 volt level for search coil.

ISEE-3
PRODUCTION
PLOT
16-CHANNEL
ANALYZER

(128 SECOND
PEAKS AND
AVERAGES)



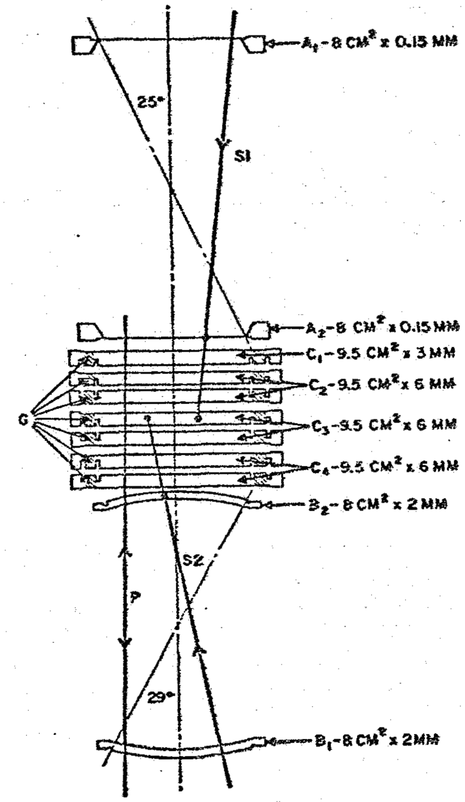
PLOT FROM
ISEE-3
COMMON
DATA POOL
TAPE
(5-MINUTE
PEAKS)



NOTES ON THE 4-57 MeV AND 18-70 MeV
PROTON + ALPHA COUNTING RATES

by
T. von Rosenvinge

The ISEE-3 Medium Energy Cosmic Ray Experiment has two High Energy Telescopes (HETs) which are designed to measure the charge composition of energetic particles from charge 1 to charge 26 over an energy range from a few MeV/nucleon to several hundred MeV/nucleon. One of these telescopes is shown schematically in Figure 1; the telescope is cylindrically symmetric around the vertical axis in the figure. Protons (and alphas) which enter A_1 and A_2 but not C_4 or the guard lie in the energy interval 4-57 MeV/nucleon (the guard "detector" is the composite of the ring detectors which encircle each of the "C" detectors; cf. Figure 1). Such events are counted by rate counters during the half of the time when each telescope is in high gain. This rate is characterized by the coincidence condition $A_1 \cdot A_2 \cdot \overline{C_4} \cdot \overline{G_1}$. Particles detected to satisfy this condition are referred to as A-Stopping (or AST) events. Protons (and alphas) which enter B_1 and B_2 but not C_1 or the guard lie in the energy interval 18-70 MeV/nucleon. Such B-stopping (or BST) events are characterized by the coincidence conditions $B_1 \cdot B_2 \cdot SB \cdot \overline{C_1} \cdot \overline{G_1}$. These rates are also available less frequently as sectored rates.



HIGH ENERGY TELESCOPE (HET)

Figure 1

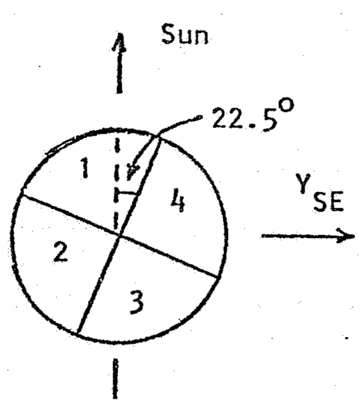
The spirit of the data pool tapes is that it should be "quick and dirty", i.e. in return for simplified algorithms for data analysis it will be possible for a wide community to gain access to data from a variety of experiments simultaneously and long before detailed data reduction can be completed. In this spirit we provide 15 minute averaged values of the counting rates $A_1 \cdot A_2 \cdot \overline{C_4} \cdot \overline{G_1}$ and $B_1 \cdot B_2 \cdot SB \cdot \overline{C_1} \cdot \overline{G_1}$ from one of the HETs (HET-II). In our own detailed data reduction we will use pulseheight analysis data for the A-stopping and B-stopping events to remove background events, to separate each charge, to correct for edge effects, and to take into account the energy dependence of the geometry factor (for AST the geometry factor for each telescope varies from .82 to 1.24cm^2-steradian and for BST the geometry factor for each telescope varies from .82 to 1.69cm^2-steradian). We will also examine sectored rates to determine spin aliassing and we will assess dead-time corrections at high counting rate levels ($\gtrsim$5,000 counts/sec). None of these is considered in the algorithm which has been used for the Data Pool Tape.

Low Energy Proton Experiment   DE FEITER

The low energy proton experiment on ISEE-3 consists of three identical telescopes, inclined at 30^o, 60^o and 135^o respectively, to the spin axis of the spacecraft. Each telescope measures protons in the energy range 35 keV to 1. MeV, in 8 logarithmically spaced channels. The output of each telescope is separately stored for each of 8 equi-angular sectors, and this applies to every energy channel except channel 8, which has only 4 equi-angular sectors. A general outline of the experiment is given by Balogh et al (Geoscience Electronics GE16, 176, 1978). Detailed descriptions are to be found in Balogh and Iversen (Space Sci.Instrum. 3, 187, 1977) and van Rooijen et al (Space Sci.Instrum. - to be published 1979).

The data on the Data Pool tape is taken from the telescope inclined at 60^o to the spin axis i.e. 30^o from the equatorial plane of the spacecraft. Two particle fluxes are supplied, obtained from the spin-averaged values for Channel 3 and Channel 4 combined, and from Channel 8. These correspond to energy ranges 91-237 keV, and 1.0 MeV to 1.6 MeV, respectively. The data are averaged over 21 minutes and 20 seconds, so that three sets of values are supplied per data record on the tape. The start of the first averaging period corresponds to the first time in the record. The fluxes are expressed in p cm^{-2} $sterad^{-1}$ sec^{-1}, using a geometrical factor of 0.05 cm^2 sterad.

A measure of particle anisotropy is also supplied. It is taken from Channel 8 and consists of the value $\frac{I_{MAX} - I_{MIN}}{I_{MAX} + I_{MIN}}$, where I_{MAX} and I_{MIN} are the maximum and minimum counting rates observed in the four sectors of Channel 8. Also given is the number of the quadrant in which I_{MAX} was observed i.e. 1 to 4. If $I_{MAX} \sim I_{MIN}$, the number 5 is inserted in this value. The convention for the numbering of quadrants with respect to the spacecraft-sun line is shown in figure 1, which is a projection in the ecliptic plane, looking down from the positive Z direction (S.E co-ordinate system).



#9

In interpreting the data caution needs to be used, and it is strongly recommended that all users of the data verify and/or clarify with the Principle Investigator any uncertainties or peculiarities concerning the data.

To assess the statistical accuracy of the data it should be remembered that converting the flux back into a counting rate, using the geometric factor and the time averaging period will only give the minimum error, since some data may have been rejected in forming the average.

So far as the physical significance of the data is concerned it should be remembered that many of the events seen by this experiment are strongly collimated in azimuth as well as longitude, so that the intensity seen by one telescope may be significantly lower than the intensity observed in the direction of maximum intensity relative to the ecliptic plane. With regard to the anisotropy measurement it should be remembered that many of the events observed have steep energy spectra, so that the statistics in Channel 8 may be insufficient to record a very strong anisotropy which exists in the lower energy channels.

Document No : 83-27-1

ISEE-3

DATA POOL TAPE PRODUCTION

ENCYCLOPEDIA GENERATOR

USER'S GUIDE

Team Members: J. Callahan, NASA
E. Eng, NASA
P. Pynn, CSC
P. Schuster, CSC

Eunice K. Eng
Code 664
Data Management and Programming Office
NASA/Goddard Space Flight Center
May 12, 1983

TABLE OF CONTENTS

I. PURPOSE

DPTGEN (Data Pool Tape Encyclopedia Generator) converts DP (Data Pool) raw data into DP encyclopedia data format similar to the ISEE-3 EDR encyclopedia format.

II. OVERVIEW

DPTGEN is part of the rigid DP Data Processing system which is monitored by the automatic log 'SB#IC.DPLOG.DATA'. When DP Library Tape file(s) become(s) available for processing, the user may initiate the execution of DPTGEN. DPTGEN will process up to fifteen Library Tape files with like attributes, in time order. DPTGEN outputs a temporary DP Work Tape, updates the automatic log and generates a printed listing.

DPTGEN will not process Library Tape files with overlapping data at file boundaries. The user will have to repeat the execution of DPTGEN to process all overlapping files.

III. MACHINE REQUIREMENTS

A. IBM 3081
B. 800k Core

IV. PREPROCESSING OF DATA

A. Automatic Log
The automatic log determines if there is any outstanding work to be done. The automatic log also determines if there are enough output tapes available for processing. If the user attempts to run DPTGEN without the proper resources that the automatic log manages, the program will issue a message and will terminate.

B. Magnetic Tapes
Each execution of DPTGEN will output a temporary DP Work Tape. If the automatic log requests new DP Work Tapes, old, released DP Work Tapes are to be relabled, using a DP version of the ISEE LABWORK proceedure.

V. PROCESSING TIMES

I/O in EXCPs	3/file (approx.), in Class A estimates
CPU	30 sec/file, in Class A estimates
Class	E for overnight
	F for weekend
	A for weekday, if turn around is critical and job is short

VI. INPUT

A. Library Tape To Be Processed
The automatic log will determine the volume serials and files to be processed. The following describes a DP Library Tape.

Record Format = Fixed
Blocksize = 3240 bytes
Density = 4 (6250 BPI)
Multi-filed

B. PARM Parameter of the JCL EXEC Card
The user may explicitly request a maximum number of library files, of like attributes to be processed.

// EXEC DPTGEN,MAX=0	program default of 15 files maximum
// EXEC DPTGEN	JCL PROC default will be taken
// EXEC DPTGEN,MAX=5	maximum of 5 files
// EXEC DPTGEN,MAX=11	maximum of 11 files

VII. STEPS IN RUNNING THE PROGRAM

To avoid some error messages the user might want to consult the latest automatic log list generated by the DP version of LOGLIST or run the DP version of LOGLIST interactively. The log listing should inform the user of

1. outstanding encyclopedia generation requests and
2. availablity of work tapes.

The user should check the JCL for

1. time estimates,
2. job class, and
3. maximum number of library files to be processed.

The JCL PROC and the PROC EXEC card are in 'SB#IC.LIB.CNTL(DPENCGEN)' as follows. The underlined areas are areas the user should check or the user might possibly change.

```
//SEEKEGEN JOB (SB013,318,15),ISEE,MSGCLASS=A,TIME=(02,30),
//   NOTIFY=SEEKE,CLASS=E
/*JOBPARM QUEUE=FETCH
//DPTGEN PROC,MAX=4
//DPTGEN EXEC PGM=DPTGEN,REGION=800K,PARM='&MAX'
//STEPLIB DD DSN=SEEKE.DPGEN.LOAD,DISP=SHR
//* DPENCGEN LIBTAPE
//FT06F001 DD SYSOUT=A,SPACE=(CYL,(5,5)),DCB=(RECFM=VBA,
//   LRECL=137,BLKSIZE=2560,BUFNO=1)
//FT10F001 DD UNIT=(6250,,DEFER),DSN=DP.NOINFO,DISP=(NEW,KEEP),
//  VOL=SER=ICW000,DCB=(DEN=4,RECFM=VB,BUFNO=1,LRECL=32000,
//  BLKSIZE=32008),LABEL=(1,SL,,OUT)
//FT12F001 DD UNIT=(6250,,DEFER),DISP=SHR,LABEL=(1,SL),
// DCB=(RECFM=F,LRECL=3240,BLKSIZE=3240,BUFNO=1,DEN=4),
// DSN=DP.DSN,VOL=SER=DPL000
//FT25F001 DD  DSN=SB#IC.DPTSTLOG.DATA,DISP=(OLD,KEEP),
//   DCB=(RECFM=F,LRECL=7232,BLKSIZE=7232)
//SYSUDUMP DD SYSOUT=A
//DPTGEN PEND
// EXEC DPTGEN,MAX=5
// EXEC NOTIFYTS
```

The above example requests that DPTGEN process a maximum of 5 library files of like attributes with timing parameters of 15 EXCPs and CPU of 2 minutes and 30 seconds.

After saving the altered DPENCGEN back into 'SB#IC.LIB.CNTL' the user can submit DPTGEN for background execution by typing

submit 'sb#ic.lib.cntl(dpencgen)'

VIII. OUTPUT

A. DP Work Tape

One temporary DP Work tape is produced with the following characteristics.

Record Format = Variable Blocked
Record Length = 32000 bytes
Blocksize = 32008 bytes
Density = 4 (6250 BPI)
Single Filed
Number of Buffers = 1, recommended for such a large blocksize

B. Printed Listing
A printout is produced with process messages indicating
1. processing status,
2. time lines for each library file,
3. minor error conditions,
4. data gaps,
5. data overlaps within a file,
6. inadequate resources, and
7. major error conditions.

IX. POST PROCESSING OF OUTPUT

A. If the system or user return code is not zero, check the printed listing for error messages. A non-zero return code indicates a serious error.

B. If the last massage on the printout is

********** DPTENCY NORMAL TERMINATION **********

the program had been successfully executed. The automatic log will reflect the program status in the library blocks processed and a new work block should have been created.

X. MESSAGES, CODES AND ERRORS

A. Abnormal Codes
The following is a list of abnormal job termination. Those that apply to the programmer will be noted, only.

STOP	Responsibility	Descriptionof Problem
101	User	Input tape does not meet specifications
102	User	Same as STOP 101
103	Programmer	
300	Programmer	
400	Programmer	
500	Programmer	

ABEND	Responsibility	Descpription of Problem
300	Programmer	

B. Messages

There are basically two types of messages; 1) messages for the user and 2) messages for the programmer. The messages for the programmer will usually terminate the program abnomally. The messages for the user will be placed into three catagories as follows:

a) program process messages requiring no user action,
b) program process messages that should be brought to the attention of the programmer for possible programmer action or for data verification, and
c) abnormal termination messages that require user action.

The messages will be grouped as classified above. Most messages follow the general format of: the name of the program that issues the message followed by the message text. Variables within the messages are as follows:

I's for integers
A's for alphanumberic characters
F's for floating point numbers
Z's for hexadecimal characters
G's for general floating point

1. User - No Action

```
ANALYZ ----- EOF -----
ANALYZ ----- EMPTY VOLUME FOR VOLUME NUMBER IIIIIIIIIIIIIIIIIIII
          START   =   IIIIIIIIIIIIIIIIIIII   AND   END   =
              IIIIIIIIIIIIIIIIIIII IN SEC. SINCE 1977
DPTENC RUN BEGAN   II/II/IIII   IIII:IIII:IIII
IN DPTENC - FAILURE OF INITAN HAS OCCURRED
IN DPTENC - FAILURE OF READER HAS OCCURRED
IN DPTENC - NO PROCESSING WAS DONE
**********  DPTENC NORMAL TERMINATION  **********
DPTERM  ENTERED
DTMJS:  INPUT MILLISECOND TOO LARGE: DDDD.DDDDDDDDDDDDDDDDDDDD
        YEAR,MONTH,DAY,HOUR AND MINUTE SET TO 0
DTUPK:  INPUT MILLISECOND TOO LARGE: DDDD.DDDDDDDDDDDDDDDDDDDD
        YEAR,MONTH,DAY,HOUR AND MINUTE SET TO 0
FNEXT  SPECIFIED/DEFAULT NUMBER OF FILES PROCESSED
FNEXT  PROCESSING  OF  FILE  III  OF  TAPE  AAAAAA  COMPLETED
    SUCCESSFULLY
     FEET USED FFFFFFFF
FNEXT  UNLOAD TAPE AAAAAA
FNEXT  LEAVE TAPE AAAAAA AT END OF FILE III
FNEXT  POSITION TAPE AAAAAA TO FILE III
FNEXT  MOUNT TAPE AAAAAA AT FILE III
INIT---THE DEFAULT VALUE OF 15 FILES IS BEING USED
INIT---NORMAL JOB STARTUP
INIT---LIBRARY TAPE AAAAAA IS MOUNTED FOR INPUT
INIT---WORK TAPE AAAAAA IS MOUNT FOR OUTPUT
OUMSUM-----LATITUDE  OR  LONGITUDE  OUT  OF  RANGE  FOR  VOLUME
    IIIIIIIIIIIIIIIIIIII   CHAPTER  II   LATITUDE   LONGITUDE
    IIIIIIIIII
           Z:FFFFFFF.FF X:FFFFFFF.FF Y:FFFFFFF.FF
```

```
SETTIM***ERROR FOUND IN INPUT TIMES***
      VALUES ARE INPUT AS:
      YEAR = IIIII
      DAY OF YEAR = IIIII
      SECONDS = IIIIIIIII
TIMLIN  RECORD PROCESSING IIIII IIII IIII IIII IIII IIII
   IIIIIIIII
  IIIIIIIII  IIIIIIIII
```

2. USER - Notify Programmer for Action or for Data Verification

```
ANALYZ -----SKIPPING MAGNETIC DATA POINTS.
        POSSIBLE OVERLAP DATA. NEXT MAG. PT. AT
           IIIIIIIIIIIIIIIIIIII EXPECTED
           IIIIIIIIIIIIIIIIIIII
ANALYZ ----- ERROR, MORE THAN 8 CHAPTERS FOR VOLUME
   IIIIIIIIIIIIIIIIIIII
        DATA WILL BE LOST UNTIL NEXT VOLUME.
DATCHK  ITIML, IPTREC INCONSISTENT, IIIIIIIIIIIIII
DATCHK  OVERLAP FOUND AT RECORD IIIIIIIIIIIIII IIIIIIIIIIIII
DATCHK  DATAGAP FOUND AT RECORD III IIIIIIIIIIIIII
   IIIIIIIIIIIIII
DATCHK  BITRATE CHANGE AT RECORD IIIII
INITAN ----- UNEXPECTED END OF FILE AFTER IIIIIIIIII RECORDS
   PROCESSED. POSSIBLE DUMMIED FILE.
PROCTM  RECORD IIIII YEAR PROBLEM IIIIIIIIII IIIIIIIII
   IIIIIIIII
 IIIIIIIII   IIIIIIIIIIIIIIIIIIII
PROCTM  RECORD IIII TIME PROBLEM IIIIIIIIII IIIIIIIII IIIIIIIII
 IIIIIIIII   IIIIIIIIIIIIIIIIIIII
PROCTM  WARNING - DTUPK YEAR DISCREPANCY IN RECORD IIIII
   GGGGGGGGG GGGGGGGGGG.G IIII IIII IIII IIII IIII IIII IIII
   IIII IIII
PROCTM  NUMBAD TIMES = IIIII
READER ----- QUESTIONABLE END TIME ON HEADER RECORD. START
   YEAR:III II END YEAR:IIIII
        SUBROUTINE SETTIM DETECTED A DATE AND TIME ERROR IN
        THE HEADER START TIME
        YEAR: IIIII DOY: IIIII SEC. OF DAY: IIIIIIIII
READER ----- NO RECORDS AFTER HEADER
TIMCHK  PROCTM RETURN = IIIII   (notify only if number .ne. 0)
TIMCHK  TIMLIN RETURN = IIIII   (notify only if number .ne. 0)
TIMCHK  WARNING NWYEAR,IEYEAR = IIIII IIII
TIMCHK  UNDETERMINED YEAR STATUS IIII
TIMLIN  RECORD IIII TIME PROBLEM IIIIIIIIII IIIIIIIII IIIIIIIII
        III IIIIII    IIIIIIIIIIIIIIIIIIII
YEARCK  ERROR IN SETTIM NEW YEAR START - RECORD IIIIII IIIII
        IIIII QINREC DEFAULTS TO FALSE
YEARCK  ERROR IN SETTIM PREV REC START - RECORD IIIIIII IIIIIII
        III IIII IIIIIII  QINREC DEFAULTS TO FALSE
YEARCK  YEAR CHANGE NOTED IIIIIIII IIIIIII IIIIIII IIIIIII
        LLLL
YEARCK  NYRCHG = IIII
YEARCK  HEADER/DATA YEAR STAUTS DESCREPANCE IIII III
```

YEARCK ODDITY ARRAY IIII III III III III III III III III III

3. USER - Abnormal Termination Requiring User Action

   a. The following occurs when a work tape reaches it's program determined end of volume. The next DPTGEN run will reprocess the incomplete file.

      ```
      FNEXT ???????????????????????????????
                TERMINATED DUE TO TAPE USAGE  FEET USED = FFFFFFFF
      ```

   b. An I/O error will processing a tape will produce one of the following messages. The problem tape should be checked using tapescan of any other appropriate system utility. DPTGEN will attempt to process a Library file four times. If the Library file is unusable, DPSAVE will have to be rerun on to retrieve the data onto a new Library file.

      ```
      FNEXT  PROCESSING OF FILE III  OF TAPE AAAAAA TERMINATED DUE
          TO I/O ERROR.

      LCL ***** I/O ERROR ENCOUNTERED
      ZZZZZZZZIIIIIIAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
          AAAAAAAAAAAAAAAAAAAAAAAAAA
      ZZZZZZZZ

      LCW ***** I/O ERROR ENCOUNTERED
      ZZZZZZZZIIIIIIAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
          AAAAAAAAAAAAAAAAAAAAAAAAAA
      ZZZZZZZZ

      LOL ***** I/O ERROR ENCOUNTERED
      ZZZZZZZZIIIIIIAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
          AAAAAAAAAAAAAAAAAAAAAAAAAA
      ZZZZZZZZ

      LOW ***** I/O ERROR ENCOUNTERED
      ZZZZZZZZIIIIIIAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
          AAAAAAAAAAAAAAAAAAAAAAAAAA
      ZZZZZZZZ

      LTERM ***** I/O ERROR ENCOUNTERED
      ZZZZZZZZIIIIIIAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
          AAAAAAAAAAAAAAAAAAAAAAAAAA
      ZZZZZZZZ

      READER ----- I/O ERROR WHILE READING HEADER ON UNIT II
      READER -----  I/O ERROR WHILE  READING DATA RECORD  IIIII ON
          UNIT II
      ```

   c. The following messages would indicate a contaminated Library tape. The actual DCB's of the file does not match the program's expected DCBs. The contents of the Library tape need to be checked, possibly with system utility TAPESCAN. In the last message, the program picked-up the wrong Library file from the automatic log. In that last case, the automatic log , the Library file and the data's history should be investigated.

READER ----- READ ERROR, EXPECTED HEADER RECORD LENGTH OF 3240 BUT GOT IIIIIIIII INSTEAD
TERMINATING

READER ----- READ ERROR, EXPECTED DATA RECORD LENGTH OF 3240 BUT GOT IIIIIIIII INSTEAD
TERMINATING

READER ----- PROGRAM SATELLITE ID OF ZZ DOES NOT MATCH TAPE SATELLITE ID OF II

d. The following messages indicate that the automatic log did not find work for DPTGEN. A new listing of the automatic log should confirm the lack of work.

INIT***DATA NOT AVAILABLE FOR PROCESSING***
LOL ***** NO LIBRARY FILE MARKED FOR PROCESSING

e. The following messages are caused by an unfavorable automatic log condition. The first message indicates that the automatic log could not allocate a new work block because it ran out of space. The user must allocate more space for the automatic log. The second message indicates that the user has run out of available work tapes. The user must remove, from the log and the Tape Library System (TLS), work tapes released by the automatic log. The user can accomplish this by running the Data Pool Processing CLIST TLS. The user must then relabel and reassign the recycled work tapes for the automatic log and TLS, by using the Data Pool Processing CLIST RMVWORK. The third message emphasizes the automatic log problem.

LOW ***** NEXT AVAILABLE BLOCK NUMBER EXCEEDS IIIIIIIII
LOW ***** NEXT WORK VOL-SER IIIIII GREATER THAN IIIIII
INIT---WORK BLOCK NOT OPENED***ERROR!

4. PROGRAMMER - Possible Programmer Error
One of the following messages will result in the setting of a non-zero user return code or an abend code. The printed listing should be brought to the attention of the project programmer.

ANALYZ ------ ERROR. ILLEGAL RETURN CODE OF III FROM COLNXD
COLFRS ------ Error. UNEXPECTED RETURN CODE FROM COLNXD WHILE PROCESSING DATATYPE II
COLLT1 ------PROGRAMMER ERROR. INVALID DATATYPE FOR THIS SUBROUTINE!
COLLT1------ERROR. UNEXPECTED RETURN CODE OF IIIII FROM SUBROUTINE COLFRS
COLLT1 ----- ERROR. BIT RATE CHANGE WHILE PROCESSING A CHAPTER IN RECORD NO IIIIIIIII

COLLT3 -----PROGRAMMER ERROR. INVALID DATATYPE FOR THIS SUBROUTINE!

COLLT3-----ERROR. UNEXPECTED RETURN CODE OF IIIII FROM SUBROUTINE COLFRS

COLLT3 ----- ERROR. BIT RATE CHANGE WHILE PROCESSING A CHAPTER IN RECORD NO IIIIIIIII

COLNXD-----ERROR. RECORD IIIIIIIII HAS START TIME OF IIIIIIIIIIII IIIIIII
WHICH IS AFTER THE FIRST MAGNETIC DATA POINT TIME OF IIIIIIIIIIIIII IIIII
POSSIBLE PROGRAMMER ERROR

LCL ***** WORK BLOCK DOES NOT HAVE PRESENT LIBRARY ADDRESS IN IT'S LIST
LIBRARY BLOCK WILL NOT BE UPDATED

****
THE LIBRARY BLOCK USED IN LIBRARY PROCESSING (BLOCK IIIII) HAS BEEN TAMPERED.
ABENDING FROM SUBROUTINE LCL
****

****
THE WORK BLOCK USED IN LIBRARY PROCESSING (BLOCK IIIII) HAS BEEN TAMPERED.
ABENDING FROM SUBROUTINE LCW
****

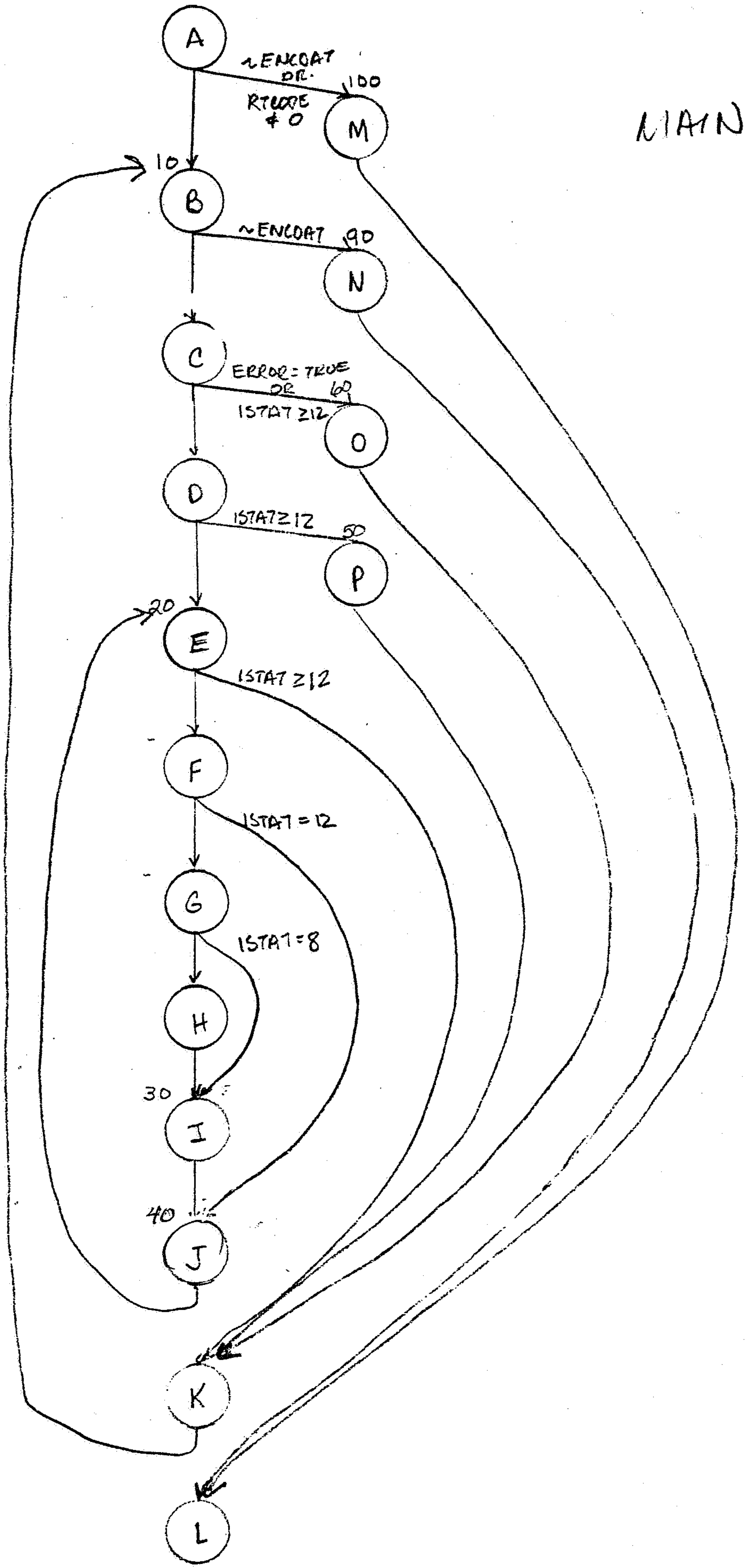
****
THE WORK BLOCK USED IN LIBRARY PROCESSING (BLOCK IIIII) HAS BEEN TAMPERED.
ABENDING FROM SUBROUTINE LTERM
****

READER ----- MORE THAN 195 RECORDS. INCREASE COMMON INDATA.

XI. DATA SET LOCATION

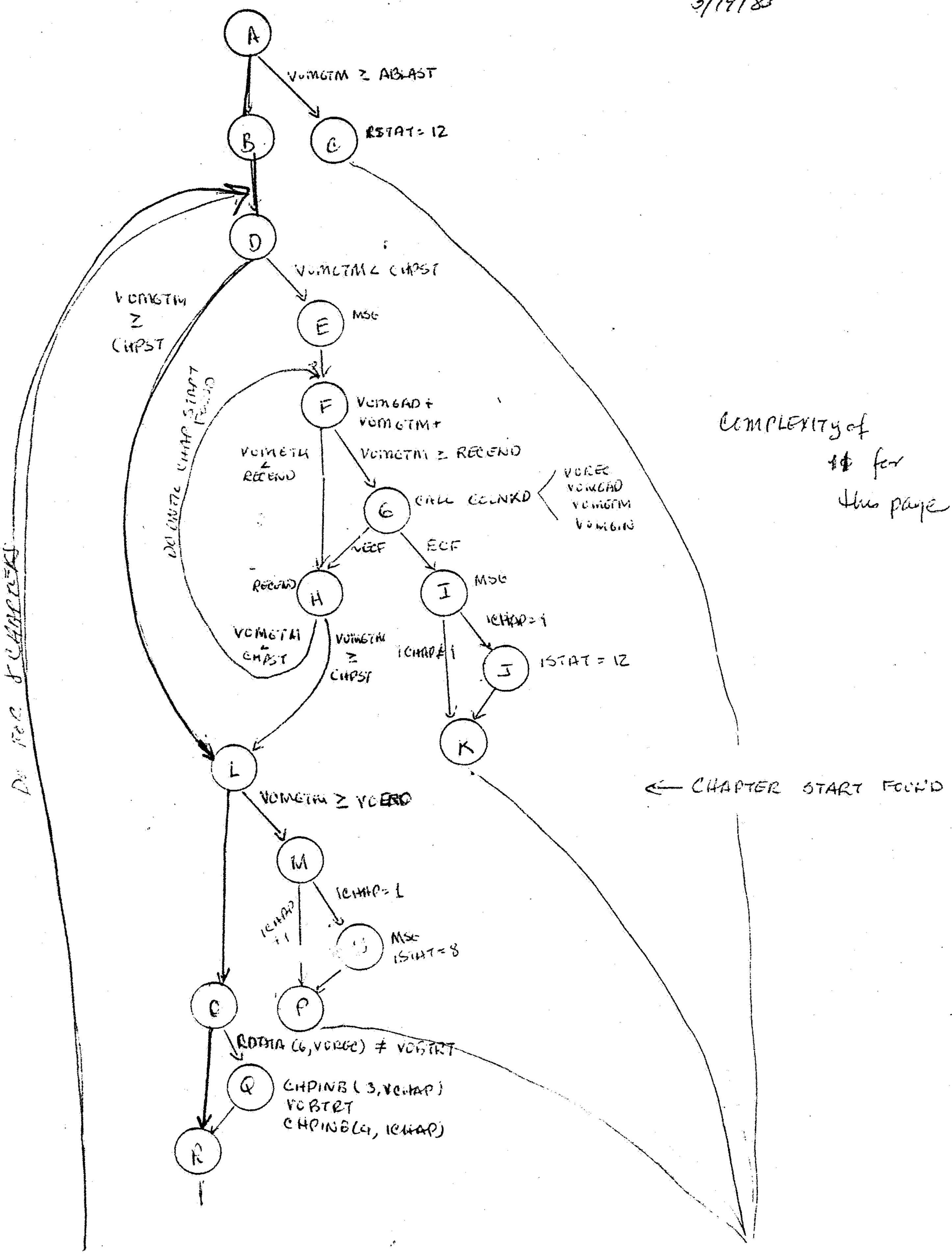
A. Source: 'SB#IC.DPGEN.SOURCE'
B. Load Module: 'SB#IC.LIB.LOAD(DPTGEN)'
C. Run JCL: 'SB#IC.LIB.CNTL(DPENCGEN)'
D. Build JCL: 'SB#IC.DPGEN.SOURCE(BUILD)'
E. User's Guide: 'SB#IC.DPGEN.SOURCE(USERGIDE)'

MAIN



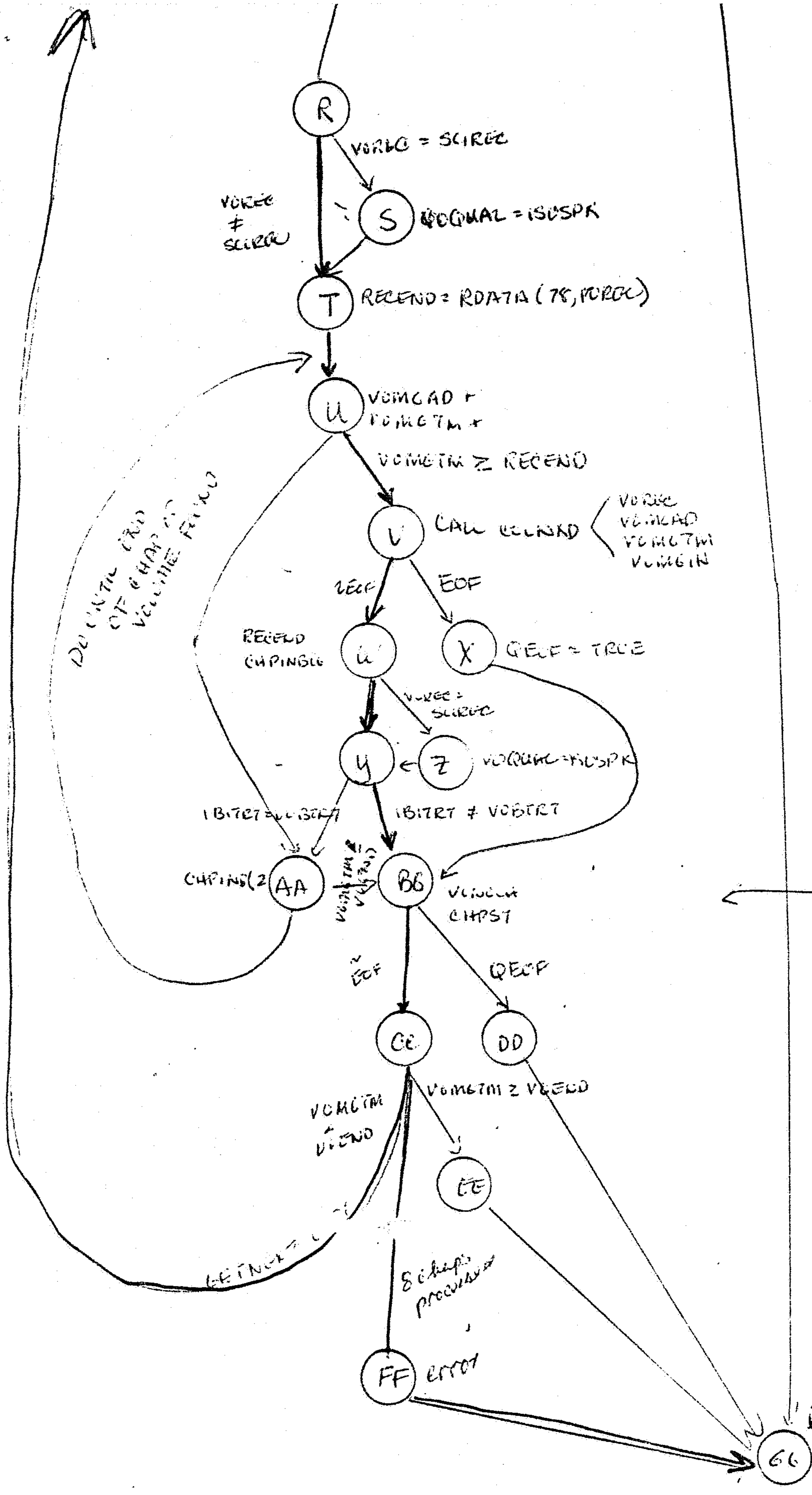
COMPLEXITY = 8

ISEE - ENG ANALYZ
3/19/83



COMPLEXITY of 1$ for this page

ANALYZ



COMPLEXITY of 9
on this pg
total of
complexity
of 17

End of chap Fam

end of volume
one way or
another

Case 2

Expected Results

INVOLIB / Namelist

VOLEN =	VONUM =	VOSTRT =
VOEND =	VOBTRT =	VONOCH =
VOMSTR =	VOREC =	VOMGRD =
VOMGIN =	VOQUAL =	ABLAST =
ISTAT =		

/CHPIB/

START - END CSMC BITRT STREC ENDREC

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.

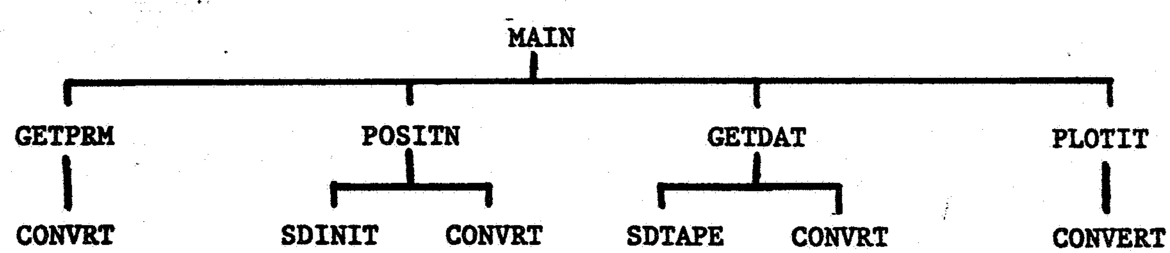
B. Plots:

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

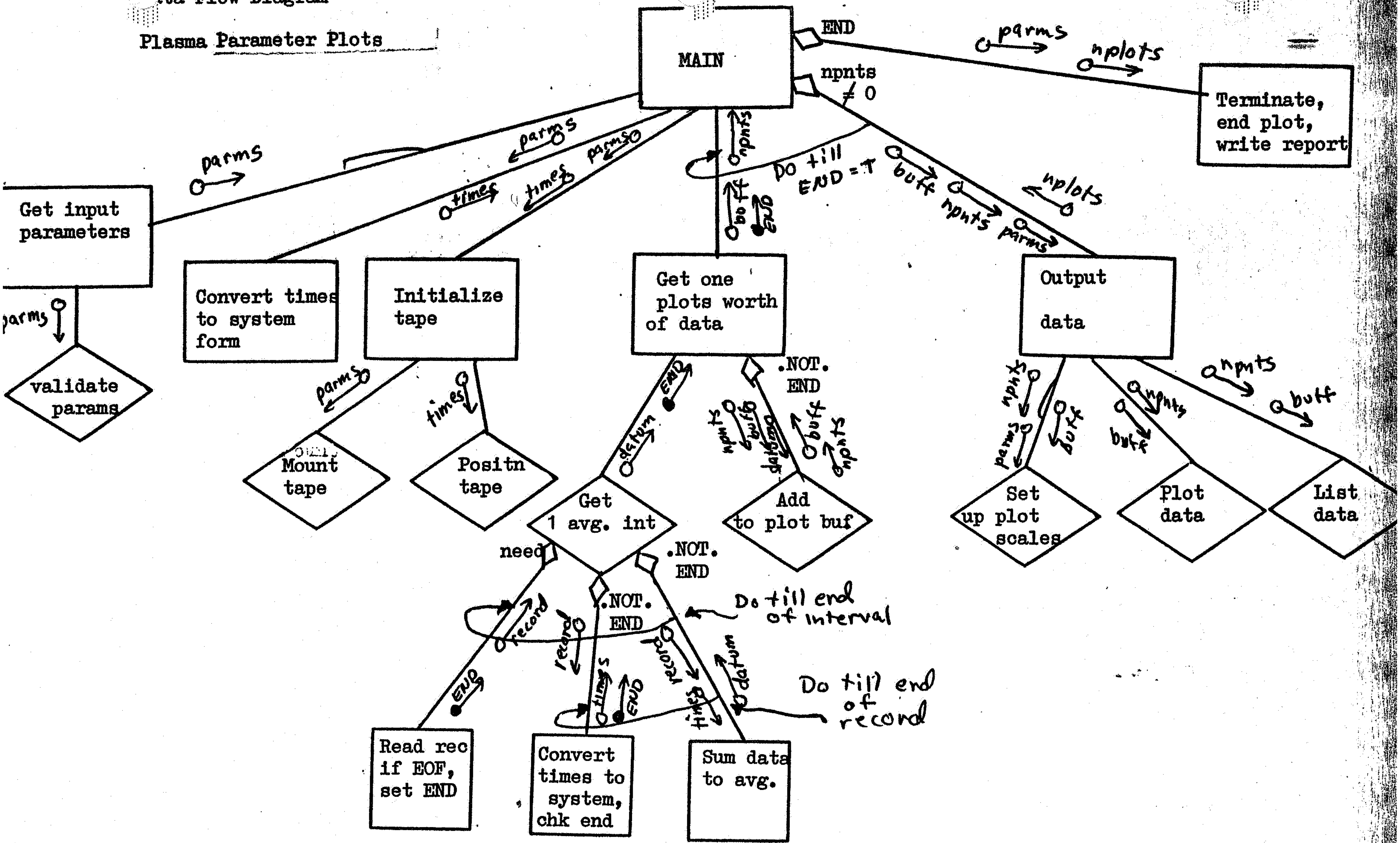


B. Module Definition

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

ta 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 GETPRM PLOTIT POSITN	I,O O I I	Holds plotting paramters
TIMES	CONVRT GETPRM	I O	Holds timing information to convert to and from system times
SDATA,SDKEYS SDINFO	POSITN SDTAPE	O I,O	Holds AMES tape reading parameters
SDVAR,SDFLAG, SDNEWV	GETDAT POSITN SDTAPE	I,O O I,O	Holds AMES tape to be returned variables
VARBLS	GETPRM,POSITN, PLOTIT	I O	Holds plasma parameter information.

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

MAGNETIC FIELD TAPE GENERATOR

I. A. Overview

The magnetic field data for PIONEER and ISEE is to be used in the Fourier Plot Program for listing and plotting. A new data base will consist of the data averaged over a user-specified time interval which would be consistent with the input data base to the Fourier program. Data specific to the Fourier program's use in included, as well as the original data in the input magnetic field tape for processing by other programs. All values will be averaged with a resolution of one minute, with no interpolation. All values are averaged over the interval in a simple manner: $\langle X \rangle = \Sigma X_i/N$, where $N = \#$ if intervals included in the sum of X_i.

The generated data base tape is made of one averaging interval, with resolution of one minute which is the resolution of the input tape. No interpolation will be done for averages of non-integral multiples of one minute due to the expected stability of the field. The times on the output tape will be event times, adjusted from the input tape ground receipt times.

B. Input Required

1. Input magnetic field tape

The tape is multi-filed with each file containing one day's data. The data exists in three average periods: minute, hour, and day averages.

Structure

Header record: 1440 minute average records

24 hour average records
1 day average record
5 spare records

The data is read with FORTRAN formatted reads as follows:

Header record: (3X,I2,2X,F3,4X,A1,15X,6E15.7,30A4)

Variable	Format	Description
	3X	
IYR	I2	Last two digits of year
	2X	
IDAY	I3	Day of year
	4X	
ISC	A1	Spacecraft (F or G)
	15X	
HRANGP	E15.7	Distance of spacecraft from sun (km)

CELLTP	E15.7	Heliocentric celestial latitude of spacecraft (degrees)
CELLNP	E15.7	Heliocentric celestial longitude of spacecraft (degrees)
REARSU	E15.7	Distance of Earth from sun (km)
CELLTE	E15.7	Heliocentric celestial latitude of Earth (degrees)
CELLNE	E15.7	Heliocentric celestial longitude of Earth (degrees)
TEXT	30A4	Text describing the file

The above trajectory data is often filled with zeros, thus a trajectory tape is used to convert ground receipt times to event times.

Data records: (8E15.6,15X,7E15.6)

Variable	Format	Description
DT	E15.7	Number of milliseconds for which data exists in the period over which the average was taken
EV(1)	E15.7	$\langle B_x \rangle$
EV(2)	E15.7	$\langle B_y \rangle$
EV(3)	E15.7	$\langle B_z \rangle$
EV(4)	E15.7	$\langle B^2 \rangle$
EV(5)	E15.7	$\langle B_x B_y \rangle$
EV(6)	E15.7	$\langle B_x B_z \rangle$
EV(7)	E15.7	$\langle B_Y{}^2 \rangle$
	15X	
EV(8)	E15.7	$\langle B_y B_z \rangle$
EV(9)	E15.7	$\langle B_z{}^2 \rangle$
EV(10)	E15.7	$\langle \cos \alpha \rangle = \langle B_x/\|B\| \rangle$
EV(11)	E15.7	$\langle \cos \beta \rangle = \langle B_y/\|B\| \rangle$
EV(12)	E15.7	$\langle \cos \nu \rangle = \langle B_z/\|B\| \rangle$
EV(13)	E15.7	$\langle \|B\| \rangle$
EV(14)	E15.7	$\langle \|B\|^2 \rangle$

2. Trajectory Tape

Tape Characteristics

Blksze = 12640
Lrecl = 1264
Recfm = FB
Den = 4
Label = SL
Track = 9-track

File Formats

Each file corresponds to a particular time period, not necessarily in chronological order. The records are read into a common block as follows:

Header record - skipped
Data records:

Variable	Type	Definition
DTSP50	R*8	Time (sec.) past 0^h Jan. 1, 1950
DJULDT	R*8	Julian date (days)
ITIME(6)	I*4	Year, month, day, hour, min. second
DEC2(153)	R*4	Element #8 = distance of S/C from earth (km)

C. Output Generated

1. Fourier Magnetic Field Data Base

This tape is an option in case only a listing is desired.

Tape Characteristics

Blksize = 7200
Lrecl = 160
Recfm = FB
Den = 3
Label = NL
Track = 9-track

File Format

Each file is one imput nagnetic field tape of about one week of data. It is read into a common block called /MAGN/ (see Common Block Definitions).

There are no file headers.

Byte Field	Type	Description
1-4	I4	Year of start of interval
5-8	I4	Day of start of interval
9-12	I4	Seconds of day of start of interval
13-16	I4	Interval in seconds of the average
17-20	R4	Milliseconds of data in the interval
21-24	I4	Input tape flag: 0=cruise, 1=Jupter, 2=Saturn

25-28	R4	*$\langle\cos\alpha\rangle$ in S/C spin coordinates
29-32	R4	*$\langle\cos\beta\rangle$ in S/C spin coordinates
33-36	R4	*$\langle\cos\gamma\rangle$ in S/C spin coordinates
37-60	24*L1	The Phi sector counts, 15° sectors
61-72	12*L1	The Theta sector counts, 15° sectors
73-76	R4	$\langle B_x\rangle$ in desired coordinate system
77-80	R4	$\langle B_y\rangle$ in desired coordinate system
81-84	R4	$\langle B_z\rangle$ in desired coordinate system
85-88	R4	$\langle B_x\rangle$ in input tape coordinates
89-92	R4	$\langle B_y\rangle$ in input tape coordinates
93-96	R4	$\langle B_z\rangle$ in input tape coordinates
97-100	R4	$\langle B_x^2\rangle$ in input tape coordinates
101-104	R4	$\langle B_xB_y\rangle$ in input tape coordinates
105-108	R4	$\langle B_xB_z\rangle$ in input tape coordinates
109-112	R4	$\langle B_y^2\rangle$ in input tape coordinates
113-116	R4	$\langle B_yB_z\rangle$ in input tape coordinates
117-120	R4	$\langle B_z^2\rangle$ in input tape coordinates
121-124	R4	$\langle\cos\alpha\rangle = B_x/B$ in input tape coordinates
125-128	R4	$\langle\cos\beta\rangle = B_y/B$ in input tape coordinates
129-132	R4	$\langle\cos\gamma\rangle = B_y/B$ in input tape coordinates
133-136	R4	$\langle\|B\|\rangle$
137-140	R4	$\langle\|B^2\|\rangle$
141-160	--	Spare bytes

2. <u>Listing</u>

A listing of the times and above data is generated at user's options.

II. Program Documentation

A. Design Flow

M-Tape = input magnetic field tape
F-Tape = output Fourier data base tape
T-Tape = Trajectory tape

MAGDBG

Read user input

Mount T-tape

Mount F-tape

Process up to 10 input M-tapes

Terminate

Check validity, Stop 1 if bad

Mount M-tape

Collect averaging interval of data

List interval's data

Write to F-tape

If new file, read trajectory tape and adjust time

Read data record

Sum to average

Create PHI, THETA histogram arrays

B. <u>Block Diagram</u>

MAIN

READIN MINVAG INTAPE

TIMADJ TIMJUL JULTIM PROCESS OUTPUT

C. Module Definition

Module	Purpose
MAIN	Controls program flow
INTAPE	Mounts and positions proper input magnetic field tape
JULTIM	Converts modified Julian time (epoch 1972) to year, month, day, hour, minute, second
TIMJUL	Reverses order of JULTIM, converting to modified Julian time (epoch 1972)
TIMADJ	Reads the trajector tape and returns the difference in seconds of ground receipt time and S/C event time
MINAVG	Collects the averaging intervals and processes them from the minute averages of the magnetic field tape
PROCES	Collects the histrograms for PHI and THETA and sends the current data into the average
OUTPUT	Prepares the data for output and lists it if desired

D. Common Blocks

1. Common /TAPE/NVOL,NFILE,ITRFIL(2),QVOLS(6,10),QODB(6,2),QNDB(6,2),QTRAJ(6)

Holds all tape information.

Variable	Type	Description
NVOL	I*4	Number of input tapes used so for this run
NFILE	I*4	The Fourier data base file #
ITRFIL(2)	I*4	Trajectory tape start, end files
QVOLS(6,10)	L*1	100 6-character names of input magnetic field tapes
QODB(6,2)	L*1	2 6-character names of the old data base
QNDB(6,2)	L*1	2 6-character names of the new data base
QTRAJ(6)	L*1	Trajectory tape name

2. Common /FLAGS/QNEW,QTAPE,QPRINT

Holds flags for processing.

Variable	Type	Description
QNEW	L*1	T = first time through program
QTAPE	L*1	T = add the data to the Fourier magnetic field data base
QPRINT	L*1	T = list the data on the line printer

3. COMMON

/MAGN/HYR,MDAY,MSC,INTRVL,DMILLI,ISATRN,COSA,COSB,COSG,QPSECT(24)
QTSECT(12),BX,BY,BZ,AVGS(14),SPARE(5)

The output common for the Fourier data base.
The values corresponds to the Fourier data base system records, with the variable types as follows: IMPLICIT REAL (A-H),INTEGER(I-N),LOGICAL*1(Q)

E. Formulas

1. Distance from S/C to Earth:

$$R = (R_1{}^2 + R_2{}^2 - 2R_1R_2(\cos(A)\cos(B),\cos(C) + \sin(A)\sin(B)))^{1/2}$$

where: R_1 = Distance from S/C to sun

R_2 = Distance from Earth to sun

A = Theta for S/C

B = Theta for Earth

C = $\text{Phi}_{S/C} - \text{Phi}_{Earth}$

2. If u is the unit vector in the direction of B, and if $x = |u|\cos\alpha$ and $y = |u|\cos\beta$, and $z = |u|\cos\gamma$, $|u| = 1$, then Phi, Theta are:

$$\text{Phi} = \phi = \text{ATAN2 } (y/x) = \text{ATAN2 } \left(\frac{\langle\cos\beta\rangle}{\langle\cos\alpha\rangle}\right)$$

$$\text{Theta} = \theta = 90 - \cos^{-1} z = 90 - \cos^{-1} (\langle\cos\gamma\rangle)$$

For PIONEER, x = -y' and y = x', x',y' are from the input tape.

F. Coordinate System

The coordinate system used for the Fourier input is a right handed system, using X axis as the reference direction. The X axis lies in the ecliptic plane, pointing toward the Sun. The Y axis lies in the ecliptic plane, perpendicular to X axis. The Z axis is zenith, perpendicular to the ecliptic plane.

The cosines are then defined as follows:

$\langle\cos\alpha\rangle = \langle B_x/|B|\rangle$
$\langle\cos\beta\rangle = \langle B_y/|B|\rangle$
$\langle\cos\gamma\rangle = \langle B_z/|B|\rangle$

where B_x, B_y, and B_z are the components of the magnetic field in S/C spin coordinates.

The Phi and Theta arrays are oriented to the reference direction such that 0^o is the reference direction, and thus Phi(1) for example is the sector value averaged from 0^o to 15^o centered on 7.5^o in the counter-clockwise direction from the reference direction:

G. Module Documentation

All modules were designed, coded, and tested by Jenny Jacques, Code 664, November 1979.

1. Module: MAIN

Purpose: Controls program flow

a. Calls: READIN,INTAPE,MINAVG

b. Commons:

Common	Variable	Input/Output
TRAJ	none used	
TAPE	ITRFIL	I
QODB	I	
NFILE	I	
QTRAJ	I	
FLAGS	QTAPE	I

c. Local Variables:

Name	Type	Description
QDONE	L*1	T = end of processing

d. Algorithm:

The necessary tapes are mounted, then the data is collected with MINAVG which returns when finished with the entire tape. If no more input tapes are required, the program ends.

2. Module: INTAPE

a. Purpose: Mounts a new magnetic field input tape

b. Arguments:

Name	Type	Input/Output	Description
QDONE	L*1	O	T = no more tapes to mount

c. Called by: MAIN

d. Commons:

Common	Variable	Input/Output
FLAGS	QNEW	I
	QTAPE	I
TAPE	NVOL	I,O
	NFILE	I,O
	QVOLS	I

e. Local Variables:

Name	Type	Description
QASTR	L*1	'*', which means no more entries in the name array.

f. Algorithm:

The name array is checked to see if there are any more input tapes to mount. If not, QDONE is set to .true. and INTAPE returns. If there is another tape, it is mounted, and an EOF is placed on the Fourier data base.

3. Module: JULTIM

a. Purpose: Convert modified Julian time to year, day, seconds of day.

b. Arguments:

Name	Type	Input/Output	Description
IYR	I*4	O	Last two digits of year
IDAY	I*4	O	Day of year

IDAY	I*4	O	Day of year
ISECC	I*4	O	Seconds of day
LTIME	I*4	I	Modified Julian day

c. Called by: MINAVG

d. Commons:

Name	Variable	Input/Output
MAGN	INTRVL	I

e. Local Variables:

Name	Type	Description
RDYCN	R*4	Number of intervals per day
JDAY	I*4	Days since epoch day 1972
JAVGS	I*4	Number of intervals of day
LEAP	I*4	1=not leap year, 2=leap year
IDAYS(16)	I*4	Days since epoch for each year

f. Algorithm:

The number of days since epoch day Jan. 0, 1972 is found, and the number of intervals in the day is calculated. Then the year, day, and seconds are found by simple calculation using a pre-defined array.

4. Module: MINAVG

a. Purpose: To collect minute averages from the magnetic field tape and process them onto the output data base.

b. Arguments: None

c. Called by: MAIN

d. Calls: TIMADJ,OUTPUT,JULTIM,TIMJUL,PROCES

e. Commons:

Name	Variable	Input/Output
FLAGS	all	I
MAGN	MYR,MDAY,MSC, DMILLI,COSA to end	O

f. Local Variables:

Name	Type	Description
DATA(14)	R*4	Input data from magnetic field tape, summed into AVGS(14)

TRAJ(6)	R*4	Input trajectory data from tape
TEXT(30)	R*4	Input text data from tape
JYR JDAY JSC JTIME	I*4	Year, day, seconds of day, and coverted time of the current data
ITOT	I*4	Number of input averages read in for the interval
QDONE	L*1	T=done with the input tape processing
ITIMER	I*4	The averaging interval times, in epoch time
MINUTS	I*4	Counter for the minutes processed in the input tape data record
NUMAVG	I*4	Subset of ITOT; number of input averages accepted into the interval's sums
RMILLI	I*4	Milliseconds of data in the current record

g. Algorithm:

The flags and pointers are initialized, and the header to the day data is read. The time from the header is adjusted so that the time is S/C time instead of ground receipt time. Then the minute averages from the tape are processd as follows:

1. Average read in
2. Check for end of interval. If end, write it to the data base tape
3. Process the day through several steps in PROCES, summing it into the interval

After all the minutes for the day have been processed, the hour and day averages on the tape are skipped and the next day is processed as described above.

5. Module: OUTPUT

a. Purpose: To prepare data for output to data base tape, listing it if desired.

b. Arguments:

Name	Type	Input/Output	Description
NUMAVG	I*4	I	# averages summed into the data
QPRINT	L*1	I	T=print the data
ITOT	I*4	I	# averages read from the tape for this interval

c. Called by: MINAVG

d. Commons:

Name	Variable	Input/Output
MAGN	all but TIMES	I,O

e. Local Variables:

Name	Type	Description
IP	I*4	Pointer to Phi sector with the maximum counts
OUT(6)	R*4	Used to store data which is to be printed out
PHI	R*4	Sector degrees, for calculating the deviation in Phi
THETA	R*4	Sector, degrees for calculating the deviation in Theta

f. Algorithm:

Divide the summed values by the number of intervals. If print is desired, calculate the angles Phi and Theta, and their deviations. The Phi deviation is derived using the maximum counts sector as the middle sector of the formula.

g. Formulas:

The angles are created as:

$$PHI = \left(\tan^{-1} \left(\frac{\cos\beta}{\cos A}\right) \right)^{1/2} \quad (\text{modulus } 360^{\circ})$$

$$THETA = 90 - \cos^{-1} (COSG)$$

The deviations of Phi and Theta are:

$$\left| \frac{\sum_{i=1}^{n} C_i \theta_i^2}{\sum_{i=1}^{n} C_i \theta_i^2} - \frac{(\sum_{i=1}^{n} C_i \theta_i)^2}{(\sum_{i=1}^{n} C_i)^2} \right|$$

where n=24 sectors for Phi, and
n=12 sectors for Theta

6. Module: PROCES

a. Purpose: To analyze each input average interval.

b. Arguments:

Name	Type	Input/Output	Description
RMILLI	R*4	I	Milliseconds of data of the average interval
DATA(14)	R*4	I	Data from input current interval

c. Called by: MINAVG

d. Commons:

Name	Variables	Input/Output
MAGN	AVGS,DMILLI, QTSECT,QPSECT	I,O

e. Local Variables:

Name	Type	Description
ANGL	R*4	Steps through the angles of Phi and Theta to collect the QTSECT and QPSECT arrays
PHI	R*4	The Phi angle for this average
THETA	R*4	The Theta angle for this average

f. Algorithm:

The input record data is summed to the collecting variables. Then the Phi and Theta histogram arrays are added to by the current interval's Phi and Theta.

g. Formulas:

$$\text{Phi} = \tan^{-1}\left(\frac{y}{x}\right), \text{ modulus } 360^{\circ}$$

where y=input x*(-1)
x=input y

7. Module: READIN

a. Purpose: To read in user input options

b. Arguments: None

c. Called by: MAIN

d. Commons:

Common	Variables	Input/Output
FLAGS	all	O
MAGN	INTRVL	O
TAPE	all	O

e. Local Variables: None

f. Algorithm:

Initialize the arrays with asterisks to signal end. Then read in the data via namelist and perform simple validity checks.

8. Module: TIMADJ

a. Purpose: To read the trajectory tape and find the time elapsed between the S/C and ground receipt times, adjusting the time from the data to be S/C time.

b. Arguments:

Name	Type	Input/Output	Description
IYR IDY ISC	I*4	I,O	Year, day, and seconds of day of the current magnetic field data
QDONE	L*1	O	T=end of trajectory tape usage
QNEW	L*1	I	T=first time in routine, skip file header

c. Called by: MINAVG

d. Commons:

Name	Variable	Input/Output
TRAJ	all	O
TAPE	ITRFIL	I,O

e. Local Variables:

Name	Type	Description
IDAY	I*4	Day of year
LEAP	I*4	1=not leap year, 2=leap year
QEOV	L*1	T=end of trajectory file reached
IDIFF	I*4	Seconds differences between ground receipt time (GRT) and S/C time
IMAGT	I*4	Time created from input time to compare with trajectory tape
ITRAJT	I*4	Time created from trajectory record to compare with input time

f. Algorithm:

A trajectory tape record is read and its time compared to the input time to be adjusted. If less, then the next record is read. When a record is found which has a time greater or equal, calculate the distance, hence time, between the earth and the S/C. Add this into the time to be adjusted. If an end of file is encountered, check to see if another file is allowed. If so, then continue. If not, return with flag set to end program.

g. Formulas:

$$\text{Difference in time} = \frac{R_{es}}{299792.5}$$

where R_{es} = distance from earth to S/C, km.

9. Module: TIMJUL

a. Purpose: Converts year, day, and seconds of day into one number for future time comparisons.

b. Arguments:

Name	Type	Input/Output	Description
IYR IDAY ISEC	I*4	I	Input time to convert
INTRVL	I*4	I	Averaging interval in seconds
JTIME	I*4	O	Converted time, # intervals since launch

c. Called by: MINAVG

d. Commons: none

e. Local Variables: none

f. Algorithm:

The time is converted using an array which contains the days elapsed

g. Formulas:

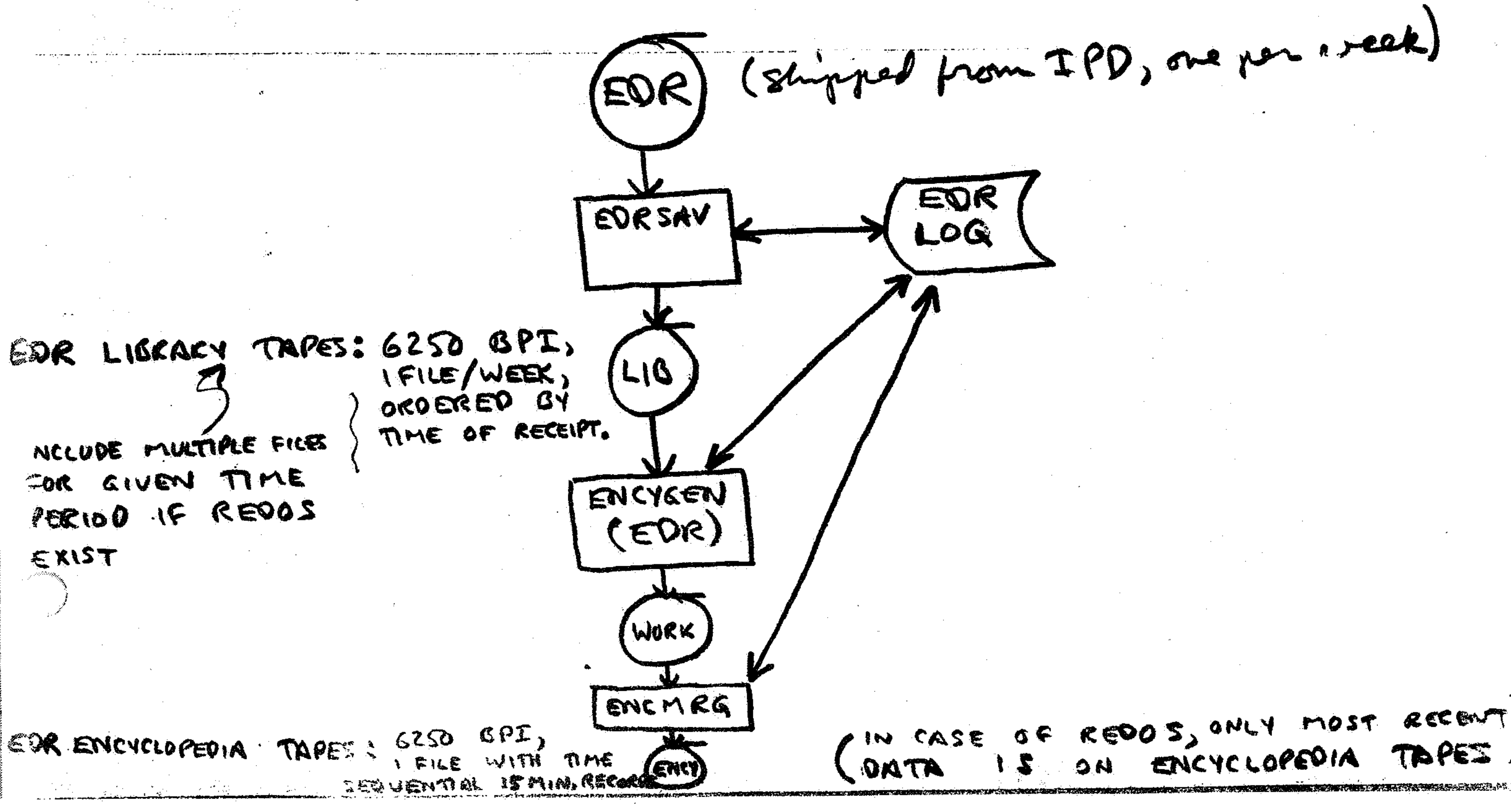
$$\text{Time} = \frac{\text{days since launch}}{\frac{\text{days}}{\text{interval}}} + \frac{\text{Seconds of day}}{\frac{\text{seconds}}{\text{interval}}}$$

EDRICE MTG. W/ JENNY JACQUES & NANU LAL — SUMMARY

TYC

10/20/88

We are trying to find ways to enable program Fourier to process anisotropy data at the highest possible time resolution. This requires accessing the raw rates verse on the encyclopedia tapes instead of the summary verses which program Fourier currently accesses. In the course of studying this problem, it has become apparent that accessing magnetic field data is a problem and that the data pool information doesn't have a proper encyclopedia data base. Consider processing of EDR tapes (Experiment Data Record Tapes):

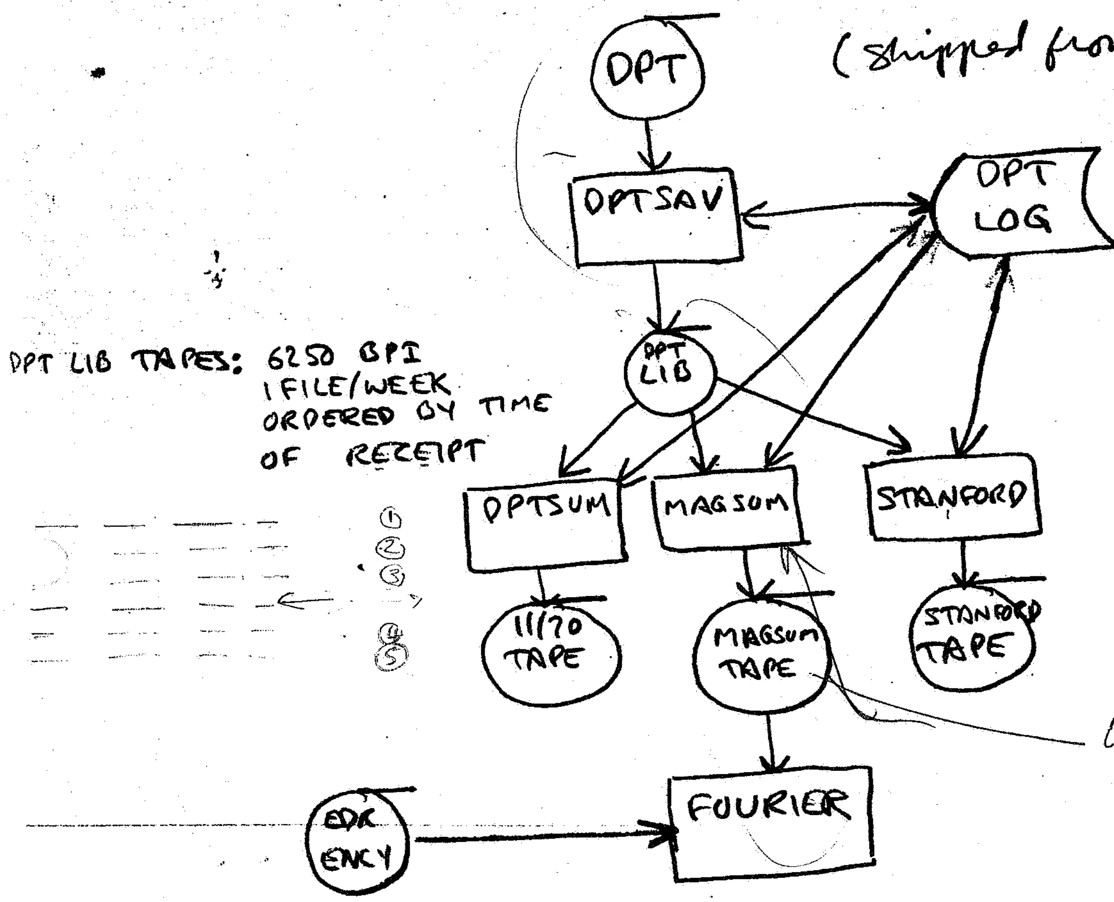


The original philosophy was to have the processing of the data pool tapes follow an identical flow. Apparently the advantages of this were not sufficiently impressed upon CSC and only Data Pool Library tapes (3 to date) and a corresponding LOG (DP LOG) have been created. Three programs currently use the Data Pool Library (DP LIB) tapes; at least 2 of these should have been accessing a Data Pool Encyclopedia (DP ENCY) tape instead. This follows from the facts that (1) they need to process the input data in time-sequential order and (2) there have been a lot of redos of Data Pool Tapes so that the DP LIB tapes have poorly ordered files. The latter fact leads to the result that there are many tape mounts required to access appropriate files in time sequential order. This would not be necessary if the proper DP ENCY tape data base existed. The two programs referred to are DPTSUM, which creates a file of averaged Data Pool quantities for display on the 11/70-VG system, and MAGSUM, which creates a tape containing 15 minute averages + 2 histograms for magnetic field data. This tape is input to program Fourier.

③ The 3rd program which currently accesses the DP LIB tapes creates a time sequential version of the DP LIB tapes which is sent to Stanford (it is not kept and is not logged)

Thus we have the following:

(shipped from IPD, one per ...



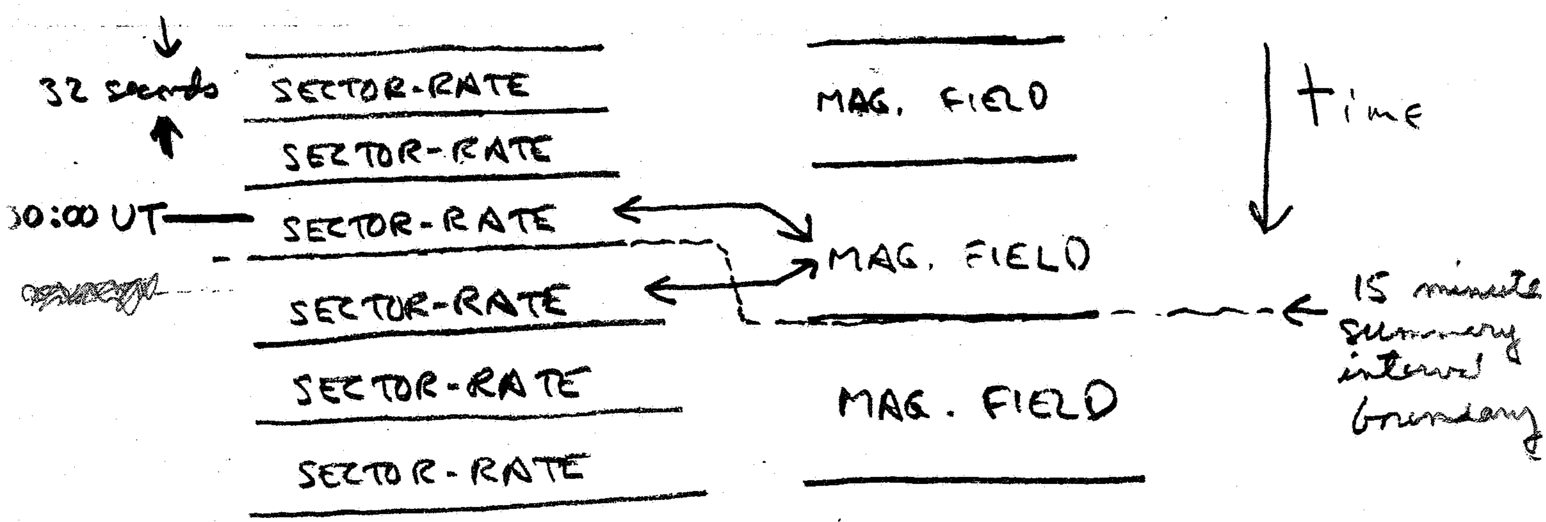
actually the log is not used in DPTSUM or MAGSUM - a time-ordered data file is created (in a separate job) which uses the log as input and outputs a file containing tape volume and file in time-ordered sequence. When any new DPT EDR is received this WRTLST.SORT dataset must be regenerated. DPTSUM & MAGSUM assume a 1:1 correspondence btwn output data files and the WRTLST.SORT DATASET

Why does [illegible] have to run on the whole data base every time?

There are no facilities for easily updating the output tapes for DPTSUM, MAGSUM or STANFORD. If a proper Data Pool Encyclopedia (DP ENCY) Tape existed then the contents of the MAGSUM Tape would just exist as a summary verse and the contents of the Stanford Tape would exist as the equivalent of the raw rates verse on the EDR ENCY tapes. The problems associated wi...

(4) time lines in OPTSUM and MAGSUM would only have to be dealt with once.

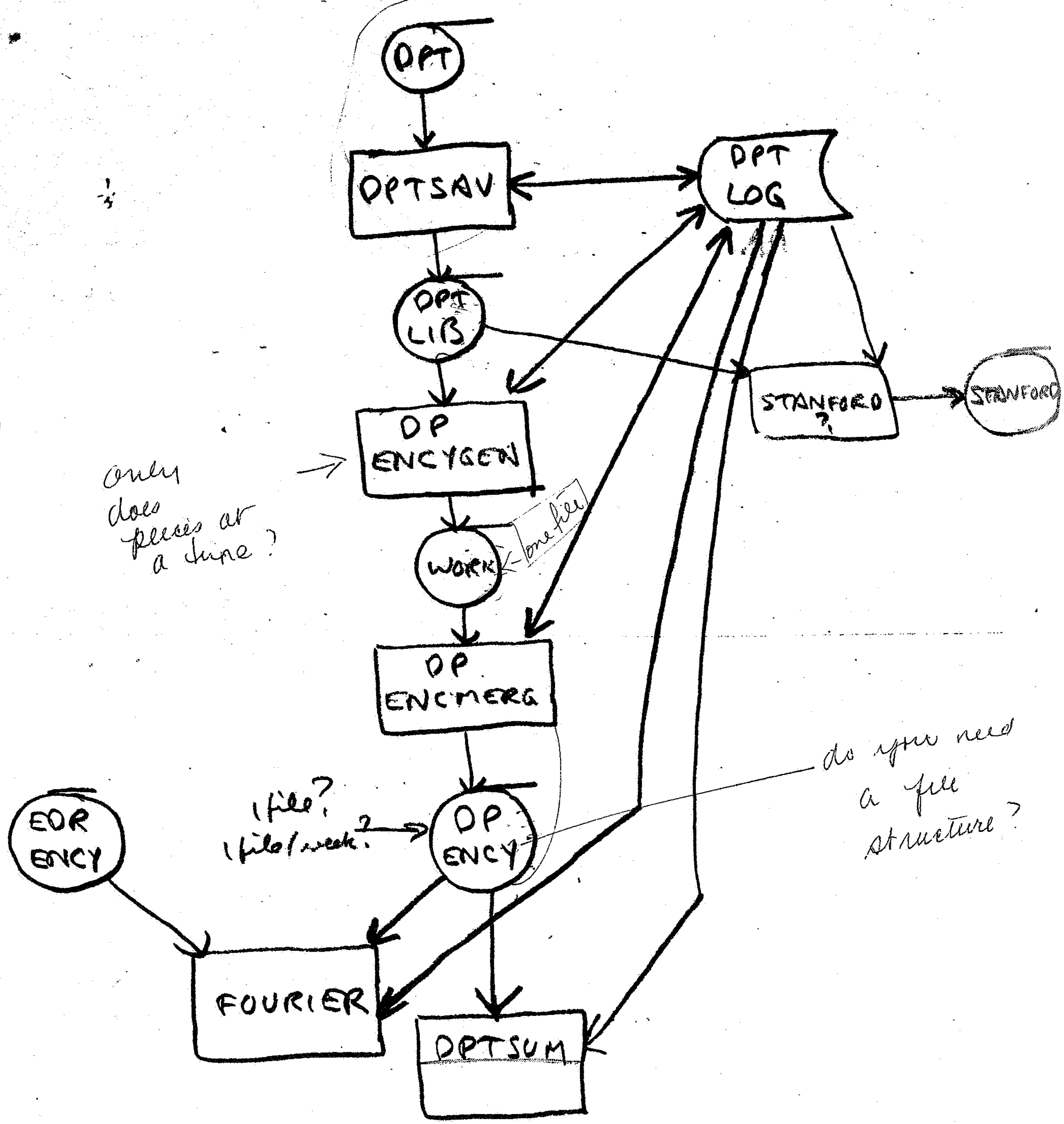
It is suggested that the OP ENCY tapes use a 15 minute summary interval aligned with U.T. hourly boundaries as is the case with the EDR ENCY tapes. Consideration was also given to the possibility of using 16 minute summary intervals because 16 minutes is an exact multiple of 64 seconds while 15 minutes is not. The magnetic field data comes as 64 second averages; sector-rate data comes as 32 second averages. Thus with 15 minute intervals you may have a sector-rate readout in one 15 minute interval that would best be associated with a magnetic field readout in an adjacent 15 minute interval:



This would not occur w/ 16-minute summaries. On the other hand with 15-minute summaries

(5) the summaries on the EDR ENCY and DP ENCY tapes will match; the 15-minute summaries were originally chosen because of the obvious universality of U.T.

Thus what we would like to see is the following:



(6) ACTION ITEMS:

PAM — ESTIMATE TIME TO CREATE OP ENCY DATA BASE

— ESTIMATE TIME TO MODIFY OPTSUM (LOW PRIOR-Y)

EUNICE — MAKE ESTIMATE OF WHEN SHE COULD CREATE OP ENCY DATA BASE.

- DECIDE WHO IS BEST TO CREATE OP ENCY DATA BASE tape

wrtlst . sort ~~file~~ change (tape mounting)

98760586 = time

back + forth ~~[illegible]~~ area

~ 118~~0~~14252

ends

127186247

Should be ~ 80 286 1381.0376

DPL002 file 30

to 80 349 1444.0865

DPL002 file 99