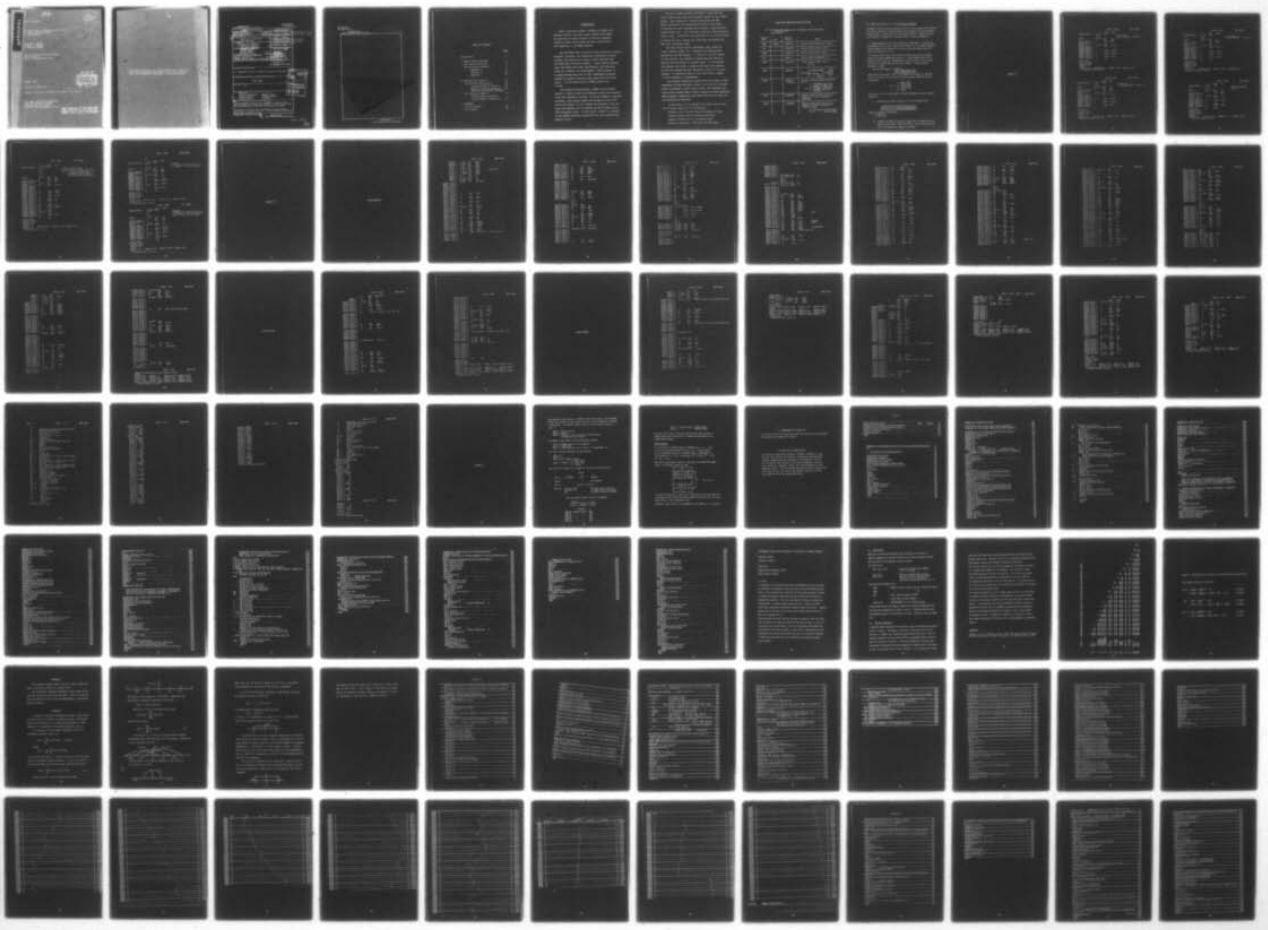
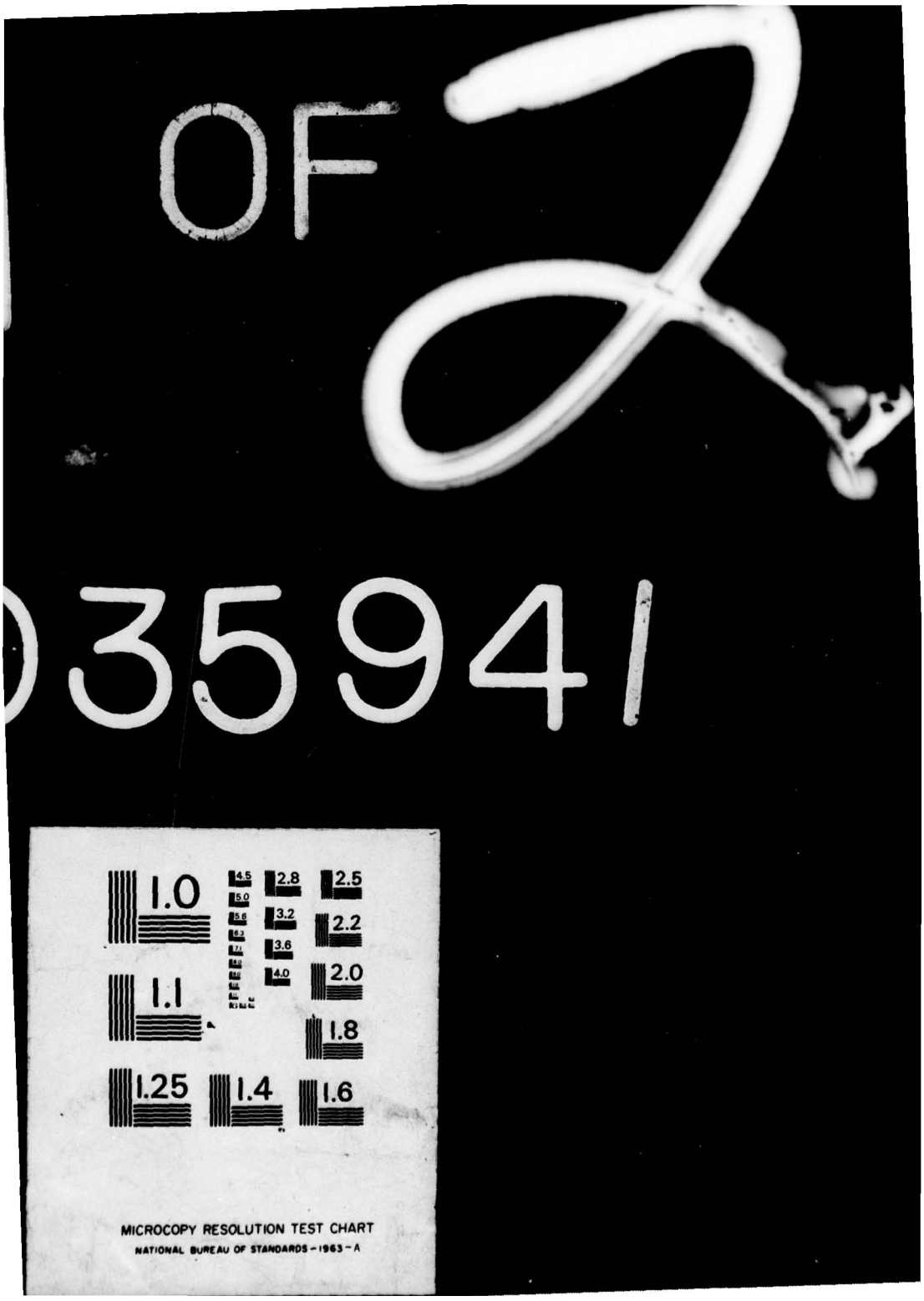
AD-A035 941 EMMANUEL COLL BOSTON MASS F/G 17/6
INVESTIGATION OF MICROPULSATION ACTIVITY. 1. MAGAF SYSTEM ADDIT--ETC(U)
AUG 76 W F BELLEW, C J CANTOR, M P HAGAN F19628-76-C-0013
UNCLASSIFIED SCIENTIFIC-1 AFGL-TR-76-0244 NL

1 OF 2
AD A035941




OF
035941
1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
5.6
6.3
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

AD A035941

AFGL-TR-76-0244

# INVESTIGATION OF MICROPULSATION ACTIVITY
## 1. MAGAF SYSTEM ADDITIONS
## 2. DATA ANALYSIS

WILLIAM F. BELLEW
CHARLES J. CANTOR
M. PATRICIA HAGAN

The Trustees of Emmanuel College
400 The Fenway
Boston, Massachusetts 02115

DDC
RECEIVED
FEB 24 1977
C

August 1976

Scientific Report No. 1

Approved for public release; distribution unlimited.

AIR FORCE GEOPHYSICS LABORATORY
AIR FORCE SYSTEMS COMMAND
UNITED STATES AIR FORCE
HANSCOM AFB, MASSACHUSETTS 01731

COPY AVAILABLE TO DDC DOES NOT
PERMIT FULLY LEGIBLE PRODUCTION

Qualified requestors may obtain additional copies from the Defense Documentation Center. All others should apply to the National Technical Information Service.

MIL-STD-847A
31 January 1973

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

| REPORT DOCUMENTATION PAGE | READ INSTRUCTIONS BEFORE COMPLETING FORM |
|---|---|
| 1. REPORT NUMBER (18) AFGL-TR-76-0244 | 2. GOVT ACCESSION NO. (19) Rept. for 1 Aug 75-30 Jun 76) |
| 4. TITLE (and Subtitle) (6) Investigation of Micropulsation Activity<br>1. MAGAF System Additions<br>2. Data Analysis | 5. TYPE OF REPORT & PERIOD COVERED<br>Scientific No. 1<br>01AUG75 - 30JUN76 |
| | 6. PERFORMING ORG. REPORT NUMBER (14) Scientific-1 |
| 7. AUTHOR(s) (10) WILLIAM F. BELLEW<br>CHARLES J. CANTOR<br>M. PATRICIA HAGAN | 8. CONTRACT OR GRANT NUMBER(s) (15) F19628-76-C-0013 |
| 9. PERFORMING ORGANIZATION NAME AND ADDRESS<br>EMMANUEL COLLEGE<br>400 THE FENWAY<br>BOSTON MA 02115 | 10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS (16) 62101F (17) 7601 08 01 |
| 11. CONTROLLING OFFICE NAME AND ADDRESS<br>AIR FORCE GEOPHYSICS LABORATORY PHG<br>HANSCOM AFB MA 01731<br>CONTRACT MONITOR: ROBERT O. HUTCHINSON/ | 12. REPORT DATE (11) AUGUST 1976<br>13. NUMBER OF PAGES (13) 98p. 97 |
| 14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) | 15. SECURITY CLASS. (of this report)<br>UNCLASSIFIED |
| | 15a. DECLASSIFICATION DOWNGRADING SCHEDULE |

16. DISTRIBUTION STATEMENT (of this Report)

A - Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

TECH, OTHER

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

ARCHIVING
MULTIPLE CORRELATION
MAGNETIC ACTIVITY
MICROPULSATIONS
MAGNETIC FIELD
SPECTRAL ANALYSIS
FILTERING

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The Micropulsation Analysis for Geomagnetic Activity Forecasting (MAGAF) data processing system is in operation at AFGL. System additions are described and programs are listed.

Processing of the data has included multiple correlation, filtering, and spectral analysis. These programs are described and listed in this report.

| ACCESSION for | |
|---|---|
| NTIS | White Section ☑ |
| DDC | Buff Section ☐ |
| UNANNOUNCED | ☐ |
| JUSTIFICATION | |
| BY | |
| DISTRIBUTION/AVAILABILITY CODES | |
| Dist. | AVAIL. and/or SPECIAL |
| A-23 | |

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

128 950
tpg

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

TABLE OF CONTENTS


| | Page |
|---|---|
| Introduction | 4 |
| 1. MAGAF SYSTEM ADDITIONS | 6 |
| (a) Library Functions | 6 |
| (b) Archiving Procedure | 7 |
| Appendix A | 8 |
| Appendix B | 15 |
| Appendix C | 45 |
| 2. MATHEMATICAL ANALYSIS | 48 |
| (a) Multiple Correlation | 48 |
| Program MLTCOR (Program 1) | 49 |
| Report of Results Employing Multiple Correlation Technique | 62 |
| (b) Spectral Analysis | 68 |
| Program UPACK (Program 2) | 73 |
| Program READTP (Progrsm 3) | 91 |
| 3. PROGRAMS | |
| (a) In Progress | 97 |
| (b) Planned | 97 |

## INTRODUCTION

Under a previous contract (F19628-73-C-0081) with Emmanuel College, the final report (AFCRL-TR-75-0588) has detailed the specifications of the mini-computer located at AFGL, which system has been installed for data reception in the MAGAF network.

Now the MAGAF Data Collection system consists of various programs, functions, and subroutines designed to collect, archive, and provide user access to data received from the MAGAF data collection network. These routines operate under the VORTEX real-time, multitasked, disk operating system on a VARIAN V-72 minicomputer. (Multitasking is a scheme whereby many more or less independent programs compete for system resources under some kind of priority setup.) A brief description of MAGAF system flow follows.

Upon system initialization, a MAGAF initialization program is read in from the Foreground Library and executed. Since this program is on disk, it can be changed as required. Currently, this program checks the presence of the 5 kHz and 10 kHz signals to V-72 counter and interrupt clock and syncs the computer and its counter to the 10 second pulse from the master clock. At this point, control is returned to the VORTEX operating system with the core resident DDU monitor active.

The core resident monitor initiates a read into the first of two buffer areas and releases control to the VORTEX system. Upon receipt of a frame of data from the DDU, VORTEX reactivates the monitor which updates the pointer to the data for user programs and initiates a read into the second buffer area. This procedure continues using alternate buffer areas. Subroutines are available on the system library to access the data, convert the time code, and unpack the data into instrument order.

The data access monitor subroutine, when called by a user program, checks for a change in the data pointer provided by the DDU monitor. If there has been no change in this pointer, the program is deactivated for 50 milliseconds and the pointer is checked again. If the pointer has changed, the frame is moved to the data area defined by the user program and the subroutine returns to the user program. It should be noted at this point that a change in this procedure is contemplated.

The interim archiving procedure takes the frame of data, compresses it when possible, stores it on disk in a circular buffer to maintain a short term history, and schedules the tape writing program when a buffer segment has been completed. Both programs, compression and tape writing are on disk and can therefore be modified.

In the future, it is expected that among other things the following items will be implemented:

Software lock on archive tape and protect of tape

Software system reset to protect archiving

Network reliability on a by-station basis

Graphics capability - real time and from tape

## 1. MAGAF DATA PROCESSING SYSTEM ADDITIONS

(a) The library functions listed in Appendix A have been added to the system.

| Function Name | Type | Calling Sequence | Result |
|---|---|---|---|
| IAND | Integer | I=IAND(J,K) | Returns logical product of J and K. |
| IOR | " | I=IOR(J,K) | Returns logical sum of J and K. |
| IXOR | " | I=IXOR(J,K) | Returns exclusive of J and K. |
| NOT | " | I=NOT(J) | Returns logical complement of J. |
| IBIT | " | I=IBIT(J,N) | I=$\{^{0}_{1}\}$ if Nth bit (mod. 16) of J is $\{^{0}_{1}\}$. |
| MASK | " | I=MASK(N) | Returns $\begin{cases} N \text{ leading ones} \\ 0 \\ \lvert N\rvert \text{ leading zeros} \end{cases}$ if N $\begin{cases} >0 \\ =0 \\ <0 \end{cases}$ |
| ISHIFT | " | I=ISHIFT (J,N,ITP) | Returns J shifted N places according to ITP.<br>ITP = 0 - Arithmetic left shift<br>1 - Logical rotation left<br>2 - Aritmetic right shift<br>3 - Logical right shift (0 fill) |
| ISET | " | I=ISET(N) | Returns a mask with bit N set to 1 (others 0). N out of range 0-15 undefined. |
| IRSET | " | I=IRSET(N) | Returns a mask with bit N set to 0 (others 1). N out of range 0-15 is undefined except,<br>N = 17 returns left byte mask<br>N = 18 " right byte mask |

(b) MAGAF data collection interim archiving procedure:

The MAGAF interim archiving procedure consists of two programs which prepare the archive tape and a library subroutine and program for using previously prepared tapes. A description of the programs follows. Program listings are contained in Appendix B. The tape format is given in Appendix C.

I. Program ARCHIV monitors the incoming data, compresses it if possible, and buffers it out to disk. When the disk buffer is full, ARCHIV schedules program ARCTAP, which transfers the data to tape and changes tape units when the current unit is full.

II. Function I$TPY returns tape status and frame data to user programs, scheduling program DCMPRS when necessary. Program DCMPRS reads in one physical record from tape, decompresses the data, writes it out to disk, and returns to I$TPY tape status and the number of logical records contained to I$TPY.

The calling sequence for I$TPY is:

```
IST = I$TPY(IUNIT, IA)
```

where IST is tape status, IUNIT is the logical unit number of the tape, and IA is an array with dimension 245 which contains frame and status information.

IST =
- 1 - good frame
- 0 - device busy
- -1 - end of tape
- -2 - end of file
- -3 - error

The format of IA is as described in AFCRL-TR-75-0588, with the following exceptions:

A. Data words and buffer contents are in the following form:

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| s | s | s | s | s | d | d | d | d | d | d | d | d | d | d | e |

,

where s = sign bit (replicated)
d = data bit
e = error bit

B. IA(242) and IA(243) now contain frame time in seconds and milliseconds respectively, frame time being reset to zero every time real time reaches zero modulo ten seconds.

# Appendix A

```
                                VORTEX  DASMR              0014 HOURS

                        1          NAME     IAND
000000 000000 A         2 IAND     ENTP
                        3 *                          LOGICAL AND
                        4 *                          CALLING SEQUENCE: I=IAND(J,K)
000000                  5 RETU     BES      0
                        6          EXT      $SE
000001 002000 A         7          CALL     $SE
000002 000000 E
000002                  8 CALL     BES      0
000003 000002 A         9          DATA     2,0,0
000004 000000 A
000005 000000 A
000006 006017 A        10          LDAE*    CALL+2
000007 100004 R
000010 006157 A        11          ANAE*    CALL+3
000011 100005 R
000012 001000 A        12          RETU*    RETU
000013 100000 R
                       13          END
ENTRY NAMES
000000 R IAND
EXTERNAL NAMES
000002 E $SE
SYMBOLS
000002 E $SE     000002 R CALL    000000 R IAND    000000 R RETU
  0 ERRORS ASSEMBLY COMPLETE
```

```
                                VORTEX  DASMR              0014 HOURS

                        1          NAME     IOR
000000 000000 A         2 IOR      ENTR
                        3 *                          LOGICAL OR
                        4 *                          CALLING SEQUENCE:  I=IOR(J,K)
000000                  5 RETU     BES      0
                        6          EXT      $SE
000001 002000 A         7          CALL     $SE
000002 000000 E
000002                  8 CALL     BES      0
000003 000002 A         9          DATA     2,0,0
000004 000000 A
000005 000000 A
000006 006017 A        10          LDAE*    CALL+2
000007 100004 R
000010 006117 A        11          ORAE*    CALL+3
000011 100005 R
000012 001000 A        12          RETU*    RETU
000013 100000 R
                       13          END
ENTRY NAMES
000000 R IOR
EXTERNAL NAMES
000002 E $SE
SYMBOLS
000002 E $SE     000002 R CALL    000000 R IOR     000000 R RETU
  0 ERRORS ASSEMBLY COMPLETE
```

```
                                        VORTEX  DASMR              0015 HOURS

                          1            NAME     IXOR
000000 000000 A           2 IXOR       ENTR
                          3 *
                          4 *                                EXCLUSIVE OR
                                                            CALLING SEQUENCE:  I=IXOR(J,K)
000000                    5 RETU       BES      0
                          6            EXT      $SE
000001 002000 A           7            CALL     $SE
000002 000000 E
000002                    8 CALL       BES      0
000003 000002 A           9            DATA     2,0,0
000004 000000 A
000005 000000 A
000006 006017 A          10            LDAE*    CALL+2
000007 100004 R
000010 006137 A          11            ERAE*    CALL+3
000011 100005 R
000012 001000 A          12            RETU*    RETU
000013 100000 R
                         13            END
ENTRY NAMES
000000 R IXOR
EXTERNAL NAMES
000002 E $SE
SYMBOLS
000002 E $SE     000002 R CALL    000000 R IXOR    000000 R RETU
  0 ERRORS ASSEMBLY COMPLETE
```

```
                                        VORTEX  DASMR              0015 HOURS

                          1            NAME     NOT
000000 000000 A           2 NOT        ENTR
                          3 *                                LOGICAL COMPLEMENT
                          4 *                                I=NOT(J)
000000                    5 RETU       BES      0
                          6            EXT      $SE
000001 002000 A           7            CALL     $SE
000002 000000 E
000002                    8 CALL       BES      0
000003 000001 A           9            DATA     1,0
000004 000000 A
000005 006017 A          10            LDAE*    CALL+2
000006 100004 R
000007 005211 A          11            CPA
000010 001000 A          12            RETU*    RETU
000011 100000 R
                         13            END
ENTRY NAMES
000000 R NOT
EXTERNAL NAMES
000002 E $SE
SYMBOLS
000002 E $SE     000002 R CALL    000000 R NOT     000000 R RETU
  0 ERRORS ASSEMBLY COMPLETE
```

```
                                VORTEX  DASMR              0015 HOURS

                        1          NAME     IBIT
000000 000000 A         2 IBIT     ENTR
                        3 *                              J=IBIT(I,N)
                        4 *                              RETURNS IN J THE VALUE(0,1) OF THE
                        5 *                              OF I;  N IS TAKEN MODULO 16.
000000                  6 RETU     BES      0
                        7          EXT      $SE
000001 002000 A         8          CALL     $SE
000002 000000 E
000002                  9 CALL     BES      0
000003 000002 A        10          DATA     2,0,0
000004 000000 A
000005 000000 A
000006 006017 A        11          LDAE*    CALL+3
000007 100005 R
000010 074011 A        12          STX      STX+1
000011 150472 A        13          ANA      0472             MASK 017
000012 005014 A        14          TAX
000013 015421 A        15          LDA      0421,1
000014 006157 A        16          ANAE*    CALL+2
000015 100004 R
000016 001010 A        17          JAZ      *+3
000017 000021 R
000020 005101 A        18          INCP     01
000021 006030 A        19 STX      LDXI     *
000022 000021 R
000023 001000 A        20          RETU*    RETU
000024 100000 R
                       21          END
ENTRY NAMES
000000 R IBIT
EXTERNAL NAMES
000002 E $SE
SYMBOLS
000002 E $SE    000002 R CALL   000000 R IBIT   000000 R RETU
000021 R STX
  0 ERRORS ASSEMBLY COMPLETE
```

```
                                VORTEX  DASMR              0015 HOURS

                      1
                      2
                      3
                      4          NAME     MASK
000000 000000 A       5 MASK     ENTR
                      6 *                                 I=MASK(N) GIVES A MASK OF N 1
                      7 *                                 IF N IS POSITIVE; N 0'S FOLL
                      8 *                                 IF N IS NEGATIVE
000000                9 RETU     BES      0
                     10          EXT      $SE
000001 002000 A      11          CALL     $SE
000002 000000 E
000002               12 CALL     BES      0
000003 000001 A      13          DATA     1,0
000004 000000 A
000005 006017 A      14          LDAE*    CALL+2
000006 100004 R
000007 001010 A      15          JAZ*     RETU
000010 100000 R
000011 074017 A      16          STX      STX+1
000012 005004 A      17          TZX
000013 001002 A      18          JAP      LEAD
000014 000021 R
000015 005211 A      19          CPA
000016 005111 A      20          IAP
000017 006030 A      21          LDXI     041                MASK DISPLACEMENT
000020 000041 A
000021 005311 A      22 LEAD     DAP
000022 150472 A      23          ANA      0472               MASK 017
000023 114010 A      24          ORA      INSTR
000024 054001 A      25          STA      *+2
000025 010440 A      26          LDA      0440                 SIGN BIT
000026 004300 A      27          SRA      0
000027 135420 A      28          ERA      0420,1                 EITHER 0 OR 177777
000030 006030 A      29 STX      LDXI     *
000031 000030 R
000032 001000 A      30          RETU*    RETU
000033 100000 R
000034 004300 A      31 INSTR    ASRA     0
                     32          END

ENTRY NAMES
000000 R MASK
EXTERNAL NAMES
000002 E $SE
SYMBOLS
000002 E $SE     000002 R CALL   000034 R INSTR  000021 R LEAD
000000 R MASK    000000 R RETU   000030 R STX
  0 ERRORS ASSEMBLY COMPLETE
```

```
                                VORTEX  DASMR              0015 HOURS

                          1
                          2          NAME     ISHIFT
000000 000000 A           3 ISHIFT   ENTR
                          4 *                        I=ISHIFT(IWHAT,N,ITYPE)
                          5 *                          SHIFTS IWHAT N PLACES; TYPE OF SHI
                          6 *                        ITYPE: 0=ARITHMETIC SHIFT LEFT
                          7 *                               1=LOGICAL ROTATION LEFT(SIGN
                          8 *                               2=ARITMETIC SHIFT RIGHT
                          9 *                               3=LOGICAL SHIFT RIGHT(SHIFTED
000000                    10 RETU    BES      0
                          11         EXT      $SE
000001 002000 A           12         CALL     $SE
000002 000000 E
000002                    13 CALL    BES      0
000003 000003 A           14         DATA     3,0,0,0
000004 000000 A
000005 000000 A
000006 000000 A
000007 064016 A           15         STB      STB+1
000010 006017 A           16         LDAE*    CALL+4
000011 100006 R
000012 006027 A           17         LDBE*    CALL+3
000013 100005 R
000014 004053 A           18         LRLB     11
000015 004542 A           19         LLSR     2
000016 006010 A           20         LDAI     17
000017 000021 A           *
000020 004447 A           21         LLRL     7
000021 054002 A           22         STA      *+3
000022 006017 A           23         LDAE*    CALL+2
000023 100004 R
000024 005000 A           24         NOP
000025 006020 A           25 STB     LDBI     *
000026 000025 R
000027 001000 A           26         RETU*    RETU
000030 100000 R
                          27         END

ENTRY NAMES
000000 R ISHIFT
EXTERNAL NAMES
000002 E $SE
SYMBOLS
000002 E $SE     000002 R CALL    000000 R ISHIFT 000000 R RETU
000025 R STB
  0 ERRORS ASSEMBLY COMPLETE
```

```
                        1
                        2
                        3
                        4          NAME    ISET
000000 000000 A         5 ISET     ENTP
                        6 *                              J=ISET(N)
                        7 *                               RETURNS A BIT MASK WITH BIT N SE
                        8 *                               1 (OTHERS 0). N OUT OF RANGE 0-1
000000                  9 RETU     BES     0
                       10         EXT     $SE
000001 002000 A        11         CALL    $SE
000002 000000 E
000002                 12 CALL     BES     0
000003 000001 A        13         DATA    1,0
000004 000000 A
000005 074004 A        14         STX     STX+1
000006 006037 A        15         LDXE*   CALL+2
000007 100004 R
000010 015421 A        16         LDA     0421,1
000011 006030 A        17 STX     LDXI    *
000012 000011 R
000013 001000 A        18         RETU*   RETU
000014 100000 R
                       19         END
ENTRY NAMES
000000 R ISET
EXTERNAL NAMES
000002 E $SE
SYMBOLS
000002 E $SE     000002 R CALL    000000 R ISET    000000 R RETU
000011 R STX
  0 ERRORS ASSEMBLY COMPLETE
```



```
                        1          NAME    IRSET
000000 000000 A         2 IRSET    ENTP
                        3 *                              J=IRSET(N)
                        4 *                                RETURNS A BIT MASK WITH BIT N SE
                        5 *                                0 (OTHERS 1). N OUT OF RANGE 0-1
                        6 *                               EXCEPT 17 GIVES LEFT BYTE MASK, 1
000000                  7 RETU     BES     0
                        8          EXT     $SE
000001 002000 A         9          CALL    $SE
000002 000000 E
000002                 10 CALL     BES     0
000003 000001 A        11         DATA    1,0
000004 000000 A
000005 074004 A        12         STX     STX+1
000006 006037 A        13         LDXE*   CALL+2
000007 100004 R
000010 015441 A        14         LDA     0441,1
000011 006030 A        15 STX     LDXI    *
000012 000011 R
000013 001000 A        16         RETU*   RETU
000014 100000 R
                       17         END
ENTRY NAMES
000000 R IRSET
EXTERNAL NAMES
000002 E $SE
SYMBOLS
000002 E $SE     000007 R CALL    000000 R IRSET   000000 R RETU
000011 R STX
  0 ERRORS ASSEMBLY COMPLETE
```

Appendix B

Program ARCHIV

```
000422 A     1 TWO     EQU     0422
000423 A     2 FOUR    EQU     0423
000424 A     3 EIGHT   EQU     0424
000466 A     4 SIX     EQU     0466
000467 A     5 SEVEN   EQU     0467
000443 A     6 M5      EQU     0443          MINUS FIVE
000465 A     7 FIVE    EQU     0465
000464 A     8 THREE   EQU     0464
000472 A     9 FIFTN   EQU     0472
000002 A    10 B       EQU     2
000001 A    11 X       EQU     1
000025 A    12 TUNIT   EQU     21
000041 A    13 DUNIT   EQU     33
            14 ARC     REW     FCB,DUNIT
000000 006505 A
000001 000000 E
000002 100000 A
000003 001441 A
000004 000175 R
000005 000000 A
000006 000000 A
000007 006010 A    15         LDAI    TUNIT
000010 000025 A
000011 050005 A    16         STA     5
000012 006010 A    17         LDAI    DUNIT
000013 000041 A
000014 050006 A    18         STA     6
000015 006010 A    19         LDAI    FCB
000016 000175 R
000017 050004 A    20         STA     4
000020 002000 A    21 ARCH1   CALL    ARCHIV
000021 000211 R
000022 124147 A    22         ADD     NWD1
000023 005012 A    23         TAB
000024 144146 A    24         SUB     DCB
000025 001004 A    25         JAN     ADD
000026 000067 R
000027 005301 A    26         DECR    01
000030 002000 A    27         CALL    OUT
000031 000075 R
000032 014145 A    28         LDA     FCB+3
000033 124154 A    29         ADD     TWOTWO
000034 144152 A    30         SUB     NSECT
000035 054142 A    31         STA     FCB+3
000036 005001 A    32         TZA
000037 054147 A    33         STA     NSECT
000040 014131 A    34         LDA     NWD1
            35         SCHED   31,1,106,'F','AR','CT','AP'
000041 006505 A
000042 000000 E
000043 010137 A
000044 143152 A
000045 140722 A
```

```
000046 141724 A
000047 140720 A
000050 024524 A    36          LDB     NWDS
000051 014126 A    37          LDA     FCB+3
000052 124135 A    38          ADD     TWOTWO
000053 144125 A    39          SUB     FCB+4
000054 001004 A    40          JAN     ADD
000055 000067 R
000056 001010 A    41          JAZ     ADD
000057 000067 R
                   42          REW     FCB,DUNIT
000060 006505 A
000061 000001 E
000062 100000 A
000063 001441 A
000064 000175 R
000065 000000 A
000066 000000 A
000067 054102 A    43 ADD      STB     NWD1
000070 014504 A    44          LDA     NWDS
000071 002000 A    45          CALL    OUT
000072 000075 R
000073 001000 A    46          JMP     ARCH1
000074 000020 R
                   47 *
                   48 *
                   49 *
000075 000000 A    50 OUT      ENTR
000076 054464 A    51          STA     TMP
000077 006057 A    52          STAE*   ABUF
000100 100171 R
000101 001004 A    53          JAN     WRITE
000102 000115 R
000103 014453 A    54          LDA     HDAD
000104 054454 A    55          STA     STRT1
000105 014061 A    56 LFT1     LDA     LEFT
000106 005014 A    57          TAX
000107 144453 A    58          SUB     TMP
000110 005311 A    59          DAR
000111 001002 A    60          JAP     FILL
000112 000144 R
000113 006506 A    61          JSR     FILLIN,B
000114 000156 R
                   62 WRITE    WRITE   FCB,DUNIT
000115 006505 A
000116 000061 E
000117 100000 A
000120 000441 A
000121 000175 R
000122 000000 A
000123 000000 A
000124 044062 A    63          INR     NSECT
000125 014433 A    64          LDA     STRT1
```

```
000126 124040 A    65          ADD     LEFT
000127 054411 A    66          STA     STRT1
000130 014432 A    67          LDA     TMP
000131 144035 A    68          SUB     LEFT
000132 024041 A    69          LDB     DCB+1
000133 064035 A    70          STB     ABUF
000134 024033 A    71          LDB     D120
000135 064031 A    72          STB     LEFT
000136 054424 A    73          STA     TMP
000137 005311 A    74          DAR
000140 001002 A    75          JAP     LFT1
000141 000105 R
000142 001000 A    76          RETU*   OUT
000143 100075 R
000144 005111 A    77 FILL     IAR
000145 054021 A    78          STA     LEFT
000146 034414 A    79          LDX     TMP
000147 006506 A    80          JSR     FILLIN,B
000150 000156 R
000151 014411 A    81          LDA     TMP
000152 124016 A    82          ADD     ABUF
000153 054015 A    83          STA     ABUF
000154 001000 A    84          RETU*   OUT
000155 100075 R
                   85 *
                   86 *
                   87 *
000156 005344 A    88 FILLIN   DXR
000157 006215 A    89          LDAE*   STRT1,X,0200
000160 100561 R
000161 006255 A    90          STAE*   ABUF,X,0200
000162 100171 R
000163 001046 A    91          JXNZ    FILLIN
000164 000156 R
000165 006706 A    92          IJMP    0,B
000166 000000 A
                   93 *
                   94 *
                   95 *
000167 000170 A    96 LEFT     DATA    120
000170 000170 A    97 D120     DATA    120
000171 001253 R    98 ABUF     DATA    BUFR
000172 000000 A    99 NWD1     DATA    0
                  100 DCB      DCB     2560,BUFR
000173 005000 A
000174 001253 R
000175 000000 A
000175            101 FCB      ORG     *-1
                  102          FCB     120,BUFR,3
000175 000170 A
000176 001253 R
000177 001400 A
000200 000000 A
```

```
000201 000000 A
000202 000000 A
000203 000000 A
000204 000000 A
000205 000000 A
000206 000000 A
000207 000000 A   103 NSECT   DATA    0
000210 000026 A   104 TWOTWO  DATA    22
                  105 *
                  106 *
                  107 *
       000211 R   108 ARCHIV  EQU     *
000211 000212 R   109         DATA    *+1
                  110 DELAY   DELAY   1
000212 006505 A
000213 000042 E
000214 001100 A
000215 000001 A
000216 000000 A
000217 010002 A   111 DELY1   LDA     2
000220 144351 A   112         SUB     CUR
000221 001010 A   113         JAZ     DELAY
000222 000212 R
000223 030002 A   114         LDX     2
000224 074345 A   115         STX     CUR
000225 015364 A   116         LDA     244,X
000226 054346 A   117         STA     NWDS
000227 015360 A   118         LDA     240,X
000230 054345 A   119         STA     STAT
000231 015363 A   120         LDA     243,X
000232 054356 A   121         STA     BUFF
000233 020003 A   122         LDB     3               TIME
000234 005001 A   123         TZA
000235 004441 A   124         LLRL    1
000236 004141 A   125         LSRB    1
000237 006170 A   126         DIVI    5000
000240 011610 A
000241 064345 A   127         STB     SEC             SECONDS
000242 120464 A   128         ADD     THREE           ROUND
000243 004560 A   129         LLSR    16
000244 170465 A   130         DIV     FIVE
000245 064342 A   131         STB     MILLI           MILLISECONDS
                  132         EXT     $UNPK
000246 002000 A   133         CALL    $UNPK,(CUR)*,SCX
000247 000000 E
000250 100572 R
000251 000612 R
000252 034322 A   134         LDX     NWDS
000253 005144 A   135         IXR
000254 005344 A   136 DXR1    DXR
000255 006015 A   137         LDAE    BUFF,X
000256 000611 R
000257 004501 A   138         LASR    1
```

```
000260 004304 A  139          ASRA    4
000261 004401 A  140          LASL    1
000262 006055 A  141          STAE    BUFF,X
000263 000611 R
000264 001046 A  142          JXNZ    DXR1
000265 000254 R
000266 014307 A  143          LDA     STAT
000267 001002 A  144          JAP     PACK
000270 000307 R
000271 006010 A  145          LDAI    SEC
000272 000607 R
000273 054265 A  146          STA     STRT1
000274 006010 A  147          LDAI    IPACK
000275 000577 R
000276 054261 A  148          STA     START
000277 010464 A  149          LDA     THREE
000300 124274 A  150          ADD     NWDS
000301 054002 A  151          STA     *+3
000302 006505 A  152          JSR     MOV,X
000303 000540 R
000304 005000 A  153          NOP
000305 001000 A  154          JMP     RET1
000306 000353 R
       000307 R  155 PACK     EQU     *
                 156          EXT     SNAP
000307 006010 A  157          LDAI    SCX
000310 000612 R  *
000311 054246 A  158          STA     START
000312 054246 A  159          STA     STRT1
000313 006010 A  160          LDAI    IPK1
000314 000577 R
000315 054244 A  161          STA     IPKWRT
000316 006010 A  162          LDAI    50
000317 000062 A
000320 002000 A  163          CALL    CMPRS
000321 000360 R
000322 002000 A  164          CALL    CMPRS
000323 000360 R
000324 002000 A  165          CALL    CMPRS
000325 000360 R
000326 006010 A  166          LDAI    10
000327 000012 A
000330 002000 A  167          CALL    CMPRS
000331 000360 R
000332 002000 A  168          CALL    CMPRS
000333 000360 R
000334 002000 A  169          CALL    CMPRS
000335 000360 R
000336 006505 A  170          JSR     MOV,X
000337 000540 R
000340 000011 A  171          DATA    9
000341 006010 A  172          LDAI    23
000342 000027 A
```

```
000343 002000 A  173          CALL    CMPRS
000344 000360 R
000345 010472 A  174          LDA     FIFTN
000346 002000 A  175          CALL    CMPRS
000347 000360 R
000350 006505 A  176          JSR     MOV,X
000351 000540 R
000352 000015 A  177          DATA    13
000353 014204 A  178 RET1     LDA     START
000354 144202 A  179          SUB     HDAD
000355 054217 A  180          STA     NWDS
000356 001000 A  181          RETU*   ARCHIV
000357 100211 R
                 182 *
                 183 *
                 184 *
000360 000000 A  185 CMPRS    ENTR
000361 005311 A  186          DAR
000362 054144 A  187          STA     N
000363 006027 A  188          LDBE*   STRT1
000364 100561 R
000365 002000 A  189          CALL    INSRT
000366 000532 R
000367 005311 A  190          DAR
000370 005014 A  191          TAX
000371 124167 A  192          ADD     STRT1
000372 005012 A  193          TAB
000373 005001 A  194          TZA
000374 054166 A  195          STA     TMP
000375 016001 A  196 LDA      LDA     1,B
000376 146000 A  197          SUB     0,B
000377 004301 A  198          ASRA    1
000400 006055 A  199          STAE    .TEMP,X
000401 001172 R
000402 001002 A  200          JAP     SUB
000403 000410 R
000404 005211 A  201          CPA
000405 001016 A  202          JANZ    *+3
000406 000410 R
000407 005111 A  203          IAR
000410 144152 A  204 SUB      SUB     TMP
000411 001004 A  205          JAN     *+4
000412 000415 R
000413 124147 A  206          ADD     TMP
000414 054146 A  207          STA     TMP
000415 005322 A  208          DBR
000416 005344 A  209          DXP
000417 001046 A  210          JXNZ    LDA
000420 000375 R
000421 010443 A  211          LDA     M5          MINUS FIVE
000422 024140 A  212          LDB     TMP
000423 041020 A  213          JBZ     DONE
000424 000441 R
```

```
000425 005111 A  214 IAR    IAR
000426 001010 A  215        JAZ     DN1
000427 000435 R
000430 001020 A  216        JBZ     DONE
000431 000441 R
000432 004101 A  217        ASRB    1
000433 001000 A  218        JMP     IAR
000434 000425 R
000435 005111 A  219 DN1    IAR
000436 004102 A  220        ASRB    2
000437 001026 A  221        JBNZ    TWEL
000440 000521 R
000441 120465 A  222 DONE   ADD     FIVE
000442 006057 A  223        STAE*   IPKWRT
000443 100562 R
000444 054121 A  224        STA     NBITS
000445 001010 A  225        JAZ     ZBITS
000446 000510 R
000447 005001 A  226        TZA
000450 054114 A  227        STA     N1
000451 024116 A  228        LDB     TWELVE
000452 174113 A  229        DIV     NBITS
000453 064113 A  230        STB     NTOWD
000454 014114 A  231        LDA     LLSR
000455 124110 A  232        ADD     NBITS
000456 054007 A  233        STA     LLS
000457 006010 A  234        LDAI    TEMP
000460 001172 R
000461 054102 A  235        STA     TMP1
000462 034104 A  236        LDX     NTOWD
000463 006017 A  237 NXT1   LDAE*   TMP1
000464 100564 R
000465 044076 A  238        INR     TMP1
000466 004540 A  239 LLS    LLSR    0
000467 005344 A  240        DXR
000470 001046 A  241        JXNZ    NXT1
000471 000463 R
000472 004544 A  242        LLSR    4
000473 002000 A  243        CALL    INSRT
000474 000532 R
000475 014067 A  244        LDA     N1
000476 124070 A  245        ADD     NTOWD
000477 054065 A  246        STA     N1
000500 144026 A  247        SUB     N
000501 001004 A  248        JAN     NXT1-1
000502 000462 R
000503 034023 A  249        LDX     N
000504 006025 A  250        LDBE*   STRT1,X,0200
000505 100561 R
000506 002000 A  251        CALL    INSRT
000507 000532 R
000510 014050 A  252 ZBITS  LDA     STRT1
000511 124015 A  253        ADD     N
```

```
000512 005111 A  254          IAR
000513 054045 A  255          STA      STRT1
000514 014012 A  256 LODX     LDA      N
000515 005111 A  257          IAR
000516 044043 A  258          INR      IPKWRT
000517 001000 A  259          RETU*    CMPPS
000520 100360 R
000521 034046 A  260 TWEL     LDX      TWELVE
000522 006077 A  261          STXE*    IPKWRT
000523 100562 R
000524 044034 A  262          INR      STRT1
000525 006505 A  263          JSR      MOV,X
000526 000540 R
000527 000000 A  264 N        DATA     0
000530 001000 A  265          JMP      LODX
000531 000514 R
000532 000000 A  266 INSRT    ENTR
000533 006067 A  267          STBE*    START
000534 100560 R
000535 044022 A  268          INR      START
000536 001000 A  269          RETU*    INSRT
000537 100532 R
                 270 *
                 271 *
                 272 *
000540 025000 A  273 MOV      LDB      0,X
000541 005144 A  274          IXR
000542 074013 A  275          STX      JUMP+1
000543 034014 A  276          LDX      START
000544 005322 A  277 DBR1     DBR
000545 006017 A  278          LDAE*    STRT1
000546 100561 R
000547 044011 A  279          INR      STRT1
000550 055000 A  280          STA      0,X
000551 005144 A  281          IXR
000552 001026 A  282          JBNZ     DBR1
000553 000544 R
000554 074003 A  283          STX      START
000555 001000 A  284 JUMP     JMP      *
000556 000555 R
                 285 *
                 286 *
                 287 *
000557 000573 R  288 HDAD     DATA     HEAD
000560 000000 A  289 START    DATA     0
000561 000000 A  290 STRT1    DATA     0
000562 000000 A  291 IPKWRT   DATA     0
000563 000000 A  292 TMP      DATA     0
000564 000000 A  293 TMP1     DATA     0
000565 000000 A  294 N1       DATA     0
000566 000000 A  295 NBITS    DATA     0
000567 000000 A  296 NTOUP    DATA     0
000570 000014 A  297 TWELVE   DATA     12
```

```
000571 004540 A   298 LLSR   LLSR    0
000572 000000 A   299 CUR    DATA    0
                  300 *
                  301 *
                  302 *
000573 000001 A   303 HEAD   DATA    1,0
000574 000000 A
000575 000000 A   304 NWDS   DATA    0
000576            305 STAT   BSS     1
000577 000000 A   306 IPACK  DATA    0
       000577 R   307 IPK1   EQU     IPACK
000600            308        BSS     7
000607            309 SEC    BSS     1
000610            310 MILLI  BSS     1
000611            311 BUFF   BSS     1
000612            312 SCX    BSS     240
001172            313 TEMP   BSS     49
001253            314 BUFR   BSS     120
       000000 R   315        END     ARC
ENTRY NAMES
EXTERNAL NAMES
000247 E $UNPK  000000 E SNAP   000213 E V$EXEC 000116 E V$IOC
SYMBOLS
000247 E $UNPK  000171 R ABUF   000067 R ADD    000000 R ARC
000020 R ARCH1  000211 R ARCHIV 000002 A B      000611 R BUFF
001253 R BUFR   000360 R CMPRS  000572 R CUR    000170 R D120
000544 R DBR1   000173 R DCB    000212 R DELAY  000217 R DELY1
000435 R DN1    000441 R DONE   000041 A DUNIT  000254 R DXR1
000424 A EIGHT  000175 R FCB    000472 A FIFTN  000144 R FILL
000156 R FILLIN 000465 A FIVE   000423 A FOUR   000557 R HEAD
000573 R HEAD   000425 R IAR    000532 R INSRT  000577 R IPACK
000577 R IPK1   000562 R IPKWRT 000555 R JUMP   000375 R LDA
000167 R LEFT   000105 R LFT1   000466 R LLS    000571 R LLSR
000514 R LODX   000443 A M5     000610 R MILLI  000540 R MOV
000527 R N      000565 R N1     000566 R NBITS  000207 R NSECT
000567 R NTOWD  000172 R NWD1   000575 R NWDS   000463 R NXT1
000075 R OUT    000307 R PACK   000353 R RET1   000612 R SCX
000607 R SEC    000467 A SEVEN  000466 A SIX    000000 E SNAP
000560 R START  000576 R STAT   000561 R STRT1  000410 R SUB
001172 R TEMP   000464 A THREE  000563 R TMP    000564 R TMP1
000025 A TUNIT  000521 R TWEL   000570 R TWELVE 000422 A TWO
000210 R TWOTWO 000213 E V$EXEC 000116 E V$IOC  000115 R WRITE
000001 A X      000510 R ZBITS
  0 ERRORS ASSEMBLY COMPLETE
```

Program ARCT4P

```
                                        VORTEX  DASMR                   0000 HOURS

          000002 A      1 B       EQU     2
          000462 A      2 LHW     EQU     0462
          000005 A      3 TUNIT   EQU     5
          000006 A      4 DUNIT   EQU     6
          000004 A      5 AFCB    EQU     4
000000    005111 A      6 START   IAR
000001    054116 A      7         STA     FCB
000002    010006 A      8         LDA     DUNIT
000003    124007 A      9         ADD     REW+3
000004    054006 A     10         STA     REW+3
000005    010006 A     11         LDA     DUNIT
000006    124020 A     12         ADD     READ+3
000007    054017 A     13         STA     READ+3
                       14 REW     REW     FCB
000010    006505 A
000011    000000 E
000012    100000 A
000013    001400 A
000014    000120 R
000015    000000 A
000016    000000 A
000017    020004 A     15         LDB     AFCB
000020    016003 A     16         LDA     3,B
000021    006140 A     17         SUBI    22
000022    000026 A
000023    054077 A     18         STA     FCB+3
                       19 READ    READ    FCB,,,3
000024    006505 A
000025    000011 E
000026    100000 A
000027    030000 A
000030    000120 R
000031    000000 A
000032    000000 A
000033    002000 A     20         CALL    WRITRY
000034    000055 R
000035    001010 A     21         JAZ     EXIT
000036    000052 R
000037    006010 A     22 FOR3    LDAI    43
000040    000053 A
000041    140005 A     23         SUB     TUNIT
000042    050005 A     24         STA     TUNIT
000043    002000 A     25         CALL    WRITRY
000044    000055 R
000045    001010 A     26         JAZ     EXIT
000046    000052 R
000047    000000 A     27         HLT
000050    001000 A     28         JMP     FOR3
000051    000037 R
                       29 EXIT    EXIT
000052    006505 A
000053    000000 E
000054    000200 A
```

```
000055 000000 A    30 WRITRY ENTR
000056 014006 A    31        LDA     WRITE+3
000057 150462 A    32        ANA     LHW
000060 120005 A    33        ADD     TUNIT
000061 054003 A    34        STA     WRITE+3
                   35 WRITE  WRITE   FCB,21,,4
000062 006505 A
000063 000025 E
000064 100000 A
000065 040425 A
000066 000120 R
000067 000030 A
000070 000000 A
                   36        STAT    WRITE,NEXT,NEXT,EOT,NEXT
000071 006505 A
000072 000000 E
000073 000062 R
000074 000100 R
000075 000100 R
000076 000103 R
000077 000100 R
000100 005001 A    37 NEXT   TZA
000101 001000 A    38        JMP*    WRITRY
000102 100055 R
000103 014006 A    39 EOT    LDA     SREC+3
000104 150462 A    40        ANA     LHW
000105 120005 A    41        ADD     TUNIT
000106 054003 A    42        STA     SREC+3
                   43 SREC   SREC    FCB,21,,1
000107 006505 A
000110 000063 E
000111 100000 A
000112 012025 A
000113 000120 R
000114 000000 A
000115 000000 A
000116 001000 A    44        JMP*    WRITRY
000117 100055 R
                   45 FCB    FCB     2560,BUFR,3
000120 005000 A
000121 000132 R
000122 001400 A
000123 000000 A
000124 000000 A
000125 000000 A
000126 000000 A
000127 000000 A
000130 000000 A
000131 000000 A
000132             46 BUFR   BSS     2560
       000000 R    47        END     START
ENTRY NAMES
EXTERNAL NAMES
```

VORTEX DASMR 0000 HOURS

```
000053 E V$EXEC 000110 E V$IOC  000072 E V$IOST
SYMBOLS
000004 A AFCB   000002 A B      000132 R BUFR   000006 A DUNIT
000103 R EOT    000052 R EXIT   000120 R FCB    000037 R FOR3
000462 A LHW    000100 R NEXT   000024 R READ   000010 R REW
000107 R SREC   000000 R START  000005 A TUNIT  000053 E V$EXEC
000110 E V$IOC  000072 E V$IOST 000062 R WRITE  000055 R WRITRY
  0 ERRORS ASSEMBLY COMPLETE
```

Function I$TPY

```
                                        VORTEX  DASMR                   0001 HOURS

                                1               EXT     $SE,$BUFF
                                2               NAME    I$TPY
          000001 A              3 X             EQU     1
          000002 A              4 B             EQU     2
000000 014106 A                 5 START         LDA     INUM
000001 024076 A                 6               LDB     IA
000002 064110 A                 7               STB     FCB+1
000003 001016 A                 8               JANZ    NOSCED
000004 000030 R
000005 006020 A                 9               LDBI    CALSEQ
000006 000103 R
                               10               SCHED   5,1,106,'F','DC','MP','RS'
000007 006505 A
000010 000000 E
000011 010105 A
000012 143152 A
000013 142303 A
000014 146720 A
000015 151323 A
000016 014071 A                11               LDA     IST
000017 001004 A                12               JAN     DECRM
000020 000066 R
                               13               REW     FCB,10
000021 006505 A
000022 000000 E
000023 100000 A
000024 001412 A
000025 000112 R
000026 000000 A
000027 000000 A
                               14 NOSCED        READ    FCB,10,,3
000030 006505 A
000031 000022 E
000032 100000 A
000033 030012 A
000034 000112 R
000035 000000 A
000036 000000 A
000037 024040 A                15               LDB     IA
000040 016364 A                16               LDA     244,B
000041 001010 A                17               JAZ     WIP
000042 000050 R
000043 001002 A                18               JAP     DECRM-1
000044 000065 R
000045 044047 A                19               INR     FCB+3
000046 001000 A                20               JMP     NOSCED
000047 000030 R
000050 005001 A                21 WIP           TBA
000051 006140 A                22               SUBI    5
000052 000005 A
000053 054041 A                23               STA     FCB+3
                               24               WRITE   FCB,10,,3
000054 006505 A
```

```
000055 000031 E
000056 100000 A
000057 030412 A
000060 000112 R
000061 000000 A
000062 000000 A
000063 001000 A    25          JMP     DECRM
000064 000066 R
000065 044027 A    26          INR     FCB+3
000066 014020 A    27 DECRM    LDA     INUM
000067 005311 A    28          DAR
000070 054016 A    29          STA     INUM
000071 014016 A    30          LDA     IST
000072 001000 A    31          JMP
000073 000000 A
000073             32 I$TPY    BES     0
000074 002000 A    33          CALL    $SE
000075 000000 E
000076 000002 A    34          DATA    2
000077 000000 A    35 IUNIN    DATA    0
000100 000000 A    36 IA       DATA    0
000101 001000 A    37          JMP     START
000102 000000 R
000103 100077 R    38 CALSEQ   DATA    (IUNIN)*,IST,IUNOT,INUM
000104 000110 R
000105 000111 R
000106 000107 R
000107 000000 A    39 INUM     DATA    0
000110 000000 A    40 IST      DATA    0
000111 000012 A    41 IUNOT    DATA    10
                   42 FCB      FCB     245,0,0
000112 000365 A
000113 000000 A
000114 000000 A
000115 000000 A
000116 000000 A
000117 000000 A
000120 000000 A
000121 000000 A
000122 000000 A
000123 000000 A
                   43          END
ENTRY NAMES
000073 R I$TPY
EXTERNAL NAMES
000000 E $BUFF   000075 E $SE     000010 E V$EXEC 000055 E V$IOC
SYMBOLS
000000 E $BUFF   000075 E $SE     000002 A B      000103 R CALSEQ
000066 R DECRM   000112 R FCB     000073 R I$TPY  000100 R IA
000107 R INUM    000110 R IST     000077 R IUNIN  000111 R IUNOT
000030 R NOSCED  000000 R START   000010 E V$EXEC 000055 E V$IOC
000050 R WIP     000001 A X
  0 ERRORS ASSEMBLY COMPLETE
```

Program DCMPRS

```
                                        VORTEX  DASMR                   0000 HOURS

                                1               EXT     $SE
          000001 A              2 X             EQU     1
          000420 A              3 ZERO          EQU     0420
          000423 A              4 FOUR          EQU     0423
          000002 A              5 B             EQU     2
                                6               EXT     $$BUFF
000000    002000 A              7 START         CALL    $$BUFF,(CALSEQ)*,BUFR,D2560,ZERO,FOUR
000001    000000 E
000002    100037 R
000003    000056 R
000004    000055 R
000005    000420 A
000006    000423 A
000007    034030 A              8               LDX     CALSEQ+1
000010    055000 A              9               STA     0,X
000011    001004 A             10               JAN     ERROR
000012    000025 R
000013    014042 A             11               LDA     BUFR
000014    001010 A             12               JAZ     START
000015    000000 R
000016    064036 A             13               STB     NWDS
                               14               EXT     DCMP
000017    002000 A             15               CALL    DCMP,(CALSEQ+2)*,(CALSEQ+3)*,BUFR,NWDS
000020    000000 E
000021    100041 R
000022    100042 R
000023    000056 R
000024    000055 R
                               16 ERROR         EXIT
000025    006505 A
000026    000000 E
000027    000200 A
000030    064002 A             17 ENTRY         STB     *+3
000031    001000 A             18               JMP     *+3
000032    000034 R
000033    000045 R             19               DATA    TST
000034    002000 A             20               CALL    $SE
000035    000000 E
000036    000004 A             21               DATA    4
000037    000000 A             22 CALSEQ        DATA    0,0,0,0
000040    000000 A
000041    000000 A
000042    000000 A
000043    001000 A             23               JMP     START
000044    000000 R
          000045 R             24 TST           EQU     *
                               25               DUP     4
000045    000051 R             26               DATA    *+4
000046    000052 R             26               DATA    *+4
000047    000053 R             26               DATA    *+4
000050    000054 R             26               DATA    *+4
000051    000027 A             27 TST1          DATA    23,0,33,0
000052    000000 A
```

```
                                        VORTEX  DASMP                   0000 HOURS

000053 000041 A
000054 000000 A
000055 005000 A    28 D2560   DATA     2560
       000055 R    29 NWDS    EQU      D2560
000056             30 BUFR    BSS      2560
       000030 R    31         END      ENTRY
ENTRY NAMES
EXTERNAL NAMES
000001 E $$BUFF 000035 E $SE     000020 E DCMP    000026 E V$EXEC
SYMBOLS
000001 E $$BUFF 000035 E $SE     000002 A B       000056 R BUFR
000037 R CALSEQ 000055 R D2560   000020 E DCMP    000030 R ENTRY
000025 R ERROR  000423 A FOUR    000055 R NWDS    000000 R START
000045 R TST    000051 R TST1    000026 E V$EXEC  000001 A X
000420 A ZERO
  0 ERRORS ASSEMBLY COMPLETE
```

```
                                        VORTEX  DASMR   $BUFF     0000 HOURS

                                1       TITLE   $BUFF
                                2       NAME    $$BUFF
                                3       NAME    LRECL
                                4       EXT     $SE
        000421 A                5 ONE   EQU     0421
000000  000000 A                6 $$BUFF ENTR
000001  002000 A                7       CALL    $SE
000002  000000 E
000003  000005 A                8       DATA    5
000004                          9 UNIT  BSS     1
000005                         10 WHERE BSS     1
000006                         11 LENG  BSS     1
000007                         12 OP    BSS     1
000010                         13 MODE  BSS     1
000011  006017 A                14      LDAE    WHERE
000012  000005 R
000013  054047 A                15      STA     LRECL+1
000014  006017 A                16      LDAE*   LENG
000015  100006 R
000016  054043 A                17      STA     LRECL
000017  006017 A                18      LDAE*   MODE
000020  100010 R
000021  004244 A                19      LRLA    4
000022  006127 A                20      ADDE*   OP
000023  100007 R
000024  004250 A                21      LRLA    8
000025  006127 A                22      ADDE*   UNIT
000026  100004 R
000027  054003 A                23      STA     *+4
        000030 R                24 WRITE EQU    *
                                25      WRITE   LRECL,5,,1   WILL BE MODIFIED
000030  006505 A
000031  000000 E
000032  100000 A
000033  010405 A
000034  000062 R
000035  000000 A
000036  000000 A
000037  006027 A                26      LDBE    WRITE+5
000040  000035 R
000041  010421 A                27      LDA     ONE
                                28      STAT    WRITE,ERR,ERR+1,ERR+2,ERR+3
000042  006505 A
000043  000000 E
000044  000030 R
000045  000053 R
000046  000054 R
000047  000055 R
000050  000056 R
000051  001000 A                29      RETU*   $$BUFF
000052  100000 R
000053  005311 A                30 ERR   DAR
000054  005311 A                31      DAR
```

```
                                VORTEX  DASMR   $BUFF     0000 HOURS

000055 005311 A    32          DAR
000056 005311 A    33          DAR
000057 001000 A    34          RETU*   $$BUFF
000060 100000 R
000061             35 STA      BSS     1
                   36 LRECL    FCB     0,0,1
000062 000000 A
000063 000000 A
000064 000400 A
000065 000000 A
000066 000000 A
000067 000000 A
000070 000000 A
000071 000000 A
000072 000000 A
000073 000000 A
                   37          END
ENTRY NAMES
000000 R $$BUFF 000062 R LRECL
EXTERNAL NAMES
000002 E $SE    000031 E V$IOC  000043 E V$IOST
SYMBOLS
000000 R $$BUFF 000002 E $SE    000053 R ERR    000006 R LENG
000062 R LRECL  000010 R MODE   000421 A ONE    000007 R OP
000061 R STA    000004 R UNIT   000031 E V$IOC  000043 E V$IOST
000005 R WHERE  000030 R WRITE
  0 ERRORS ASSEMBLY COMPLETE
```

```
                                        VORTEX  DASMR   $BUFF     0000 HOURS

                          1             NAME    IUNPK
000000 006037 A           2 START       LDXE*   NTOWD
000001 100035 R
000002 006017 A           3             LDAE*   IPK
000003 100034 R
000004 124033 A           4             ADD     LLS
000005 054010 A           5             STA     LLS1
000006 014032 A           6             LDA     LASL
000007 006147 A           7             SUBE*   IPK
000010 100034 R
000011 054005 A           8             STA     LSL1
000012 006017 A           9             LDAE*   IA
000013 100032 R
000014 005344 A          10 DXR         DXR
000015 005002 A          11             TZB
000016 004540 A          12 LLS1        LLSR    0
000017 004117 A          13 LSL1        ASRB    15
000020 006067 A          14             STBE*   IB
000021 100033 R
000022 044010 A          15             INR     IB
000023 001046 A          16             JXNZ    DXR
000024 000014 R
000025 001000 A          17             JMP     *
000026 000025 R
000026                   18 IUNPK       BES     0
                         19             EXT     $SE
000027 002000 A          20             CALL    $SE
000030 000000 E
000031 000004 A          21             DATA    4
000032 000000 A          22 IA          DATA    0
000033 000000 A          23 IB          DATA    0
000034 000000 A          24 IPK         DATA    0
000035 000000 A          25 NTOWD       DATA    0
000036 001000 A          26             JMP     START
000037 000000 R
000040 004540 A          27 LLS         LLSR    0
000041 004117 A          28 LASL        ASRB    15
                         29             END

ENTRY NAMES
000026 R IUNPK
EXTERNAL NAMES
000030 E $SE
SYMBOLS
000030 E $SE     000014 R DXR     000032 R IA      000033 R IB
000034 R IPK     000026 R IUNPK   000041 R LASL    000040 R LLS
000016 R LLS1    000017 R LSL1    000035 R NTOWD   000000 R START
  0 ERRORS ASSEMBLY COMPLETE
```

```
                                        VORTEX  DASMR   $BUFF     0000 HOURS

                          1
                          2             NAME    ISHF
        000001 A          3 X           EQU     1
000000  006037 A          4 START       LDXE*   N
000001  100022 R
000002  005344 A          5 DXR         DXR
000003  006215 A          6             LDAE*   IA,X,0200
000004  100020 R
000005  004244 A          7             LRLA    4
000006  004304 A          8             ASRA    4
000007  006255 A          9             STAE*   IB,X,0200
000010  100021 R
000011  001046 A         10             JXNZ    DXR
000012  000002 R
000013  001000 A         11             JMP     *
000014  000013 R
000014                    12 ISHF       BES     0
                          13            EXT     $SE
000015  002000 A         14             CALL    $SE
000016  000000 E
000017  000003 A         15             DATA    3
000020                    16 IA         BSS     1
000021                    17 IB         BSS     1
000022                    18 N          BSS     1
000023  001000 A         19             JMP     START
000024  000000 R
                          20            END

ENTRY NAMES
000014 R ISHF
EXTERNAL NAMES
000016 E $SE
SYMBOLS
000016 E $SE     000002 R DXR     000020 R IA      000021 R IB
000014 R ISHF    000022 R N       000000 R START   000001 A X
  0 ERRORS ASSEMBLY COMPLETE
```

```
                                        VORTEX  DASMR   $BUFF     0000 HOURS

                          1
                          2             NAME    $$BF1
                          3             EXT     $SE
000000 014044 A     4 START   LDA     FRST
000001 001016 A     5         JANZ    WRITE
000002 000025 R
000003 044041 A     6         INR     FRST
000004 006027 A     7         LDBE*   UNIT
000005 100041 R
000006 005021 A     8         TBA
000007 124011 A     9         ADD     REW+3
000010 054010 A    10         STA     REW+3
000011 005021 A    11         TBA
000012 124015 A    12         ADD     WRITE+3
000013 054014 A    13         STA     WRITE+3
000014 014025 A    14         LDA     IB
000015 054031 A    15         STA     FCB+1
                   16 REW     REW     FCB
000016 006505 A
000017 000000 E
000020 100000 A
000021 001400 A
000022 000046 R
000023 000000 A
000024 000000 A
                   17 WRITE   WRITE   FCB,,,3
000025 006505 A
000026 000017 E
000027 100000 A
000030 030400 A
000031 000046 R
000032 000000 A
000033 000000 A
000034 001000 A    18         JMP     *
000035 000034 R
000035             19 $$BF1   BES     0
000036 002000 A    20         CALL    $SE
000037 000000 E
000040 000002 A    21         DATA    2
000041 000000 A    22 UNIT    DATA    0
000042 000000 A    23 IB      DATA    0
000043 001000 A    24         JMP     START
000044 000000 R
000045 000000 A    25 FRST    DATA    0
                   26 FCB     FCB     245,0,1
000046 000365 A
000047 000000 A
000050 000400 A
000051 000000 A
000052 000000 A
000053 000000 A
000054 000000 A
000055 000000 A
```

```
                                VORTEX  DASMR   $BUFF      0000 HOURS

000056 000000 A
000057 000000 A
                      27        END
ENTRY NAMES
000035 R $$BF1
EXTERNAL NAMES
000037 E $SE      000026 E V$IOC
SYMBOLS
000035 R $$BF1   000037 E $SE     000046 R FCB     000045 R FRST
000042 R IB      000016 R REW     000000 R START   000041 R UNIT
000026 E V$IOC   000025 R WRITE
  0 ERRORS ASSEMBLY COMPLETE
```

```
                                VORTEX  FTN IV              0000 HOURS

    1           SUBROUTINE DCMP(IUNIT,INUM,IA,NBUFF)
    2           DIMENSION IA(2)
    3           DATA NTMP,NN/0,1/
    4           INUM=0
    5  1       CALL IDCMP(IUNIT,INUM,NTMP,IA(NN),NN)
    6           NN=NN+IA(NN+2)
    7            IF(IA(NN).EQ.4095)RETURN
    8            GO TO 1
    9           END
ENTRY/COMMON BLOCK NAMES
  000110 R    DCMP
EXTERNAL NAMES
  000002 E      $SE
  000032 E   IDCMP
SYMBOL TABLE
  100004 R   IUNIT
  100005 R    INUM
  100006 R      IA
  100007 R   NBUFF
  000002 E      $SE
  000012 R    NTMP
  000013 R      NN
  000103 R 000000
  000021 R       1
  000032 E   IDCMP
  000104 R 177777
  000105 R    $1
  000057 R      0$
  000107 R 007777
  000106 R    $1 0
  0 ERRORS COMPILATION COMPLETE
```

```
PAGE    1                        VORTEX  FTN IV              0000 HOURS

     1
     2
     3          SUBROUTINE IDCMP(IUNIT,INUM,NTMP,IA,NN)
     4          DIMENSION IA(1),IB(245)
     5          IF(IA(1).NE.1)GO TO 99
     6          INUM=INUM+1
     7           CALL ISHF(IA(4),IA(4),1)
     8           IST=IA(4)+1
     9           NWDS=240
    10          IF(IST.LT.0)GO TO 1
    11   3       IF(NTMP.NE.0)CALL $$BF1(IUNIT,IB)
    12          NB=240
    13           IOUT=1
    14          NTMP=0
    15          IF(IST.GT.0)GO TO 4
    16          CALL ISHF(IA(8),IB,NWDS)
    17          GO TO 6
    18   4      MM=16
    19          DO 11 I=1,3
    20          LL=50*I-49
    21   11     CALL JDCMP(IA(I+4),IA(MM),IB(LL),50,MM)
    22          DO 12 I=1,3
    23          LL=10*I+141
    24   12     CALL JDCMP(IA(I+7),IA(MM),IB(LL),10,MM)
    25          CALL ISHF(IA(MM),IB(121),9)
    26          MM=MM+9
    27          CALL JDCMP(IA(11),IA(MM),IB(190),23,MM)
    28          CALL JDCMP(IA(12),IA(MM),IB(213),15,MM)
    29          CALL ISHF(IA(MM),IB(228),13)
    30          NBEG=13
    31   6      IB(NB+1)=IA(4)
    32          IB(NB+2)=IA(NBEG)
    33          IB(NB+3)=IA(NBEG+1)
    34          CALL ISHF(IA(NBEG+2),IB(NB+4),1)
    35          IB(NB+5)=NWDS
    36           IF(NTMP.NE.0)IB(NB)=-1
    37           IF(NTMP.EQ.48)IOUT=1
    38           IF(IOUT.EQ.0)RETURN
    39   98      CALL$$BF1(IUNIT,IB)
    40           NTMP=0
    41           RETURN
    42   99      IF(NTMP.EQ.0)RETURN
    43           IF(IA(NN+1).EQ.4095)GO TO 98
    44           RETURN
    45   1       NWDS=IA(3)-7
    46           IF(NWDS.NE.0)GO TO 3
    47           IOUT=0
    48           NBEG=5
    49           NTMP=NTMP+1
    50           NB=245-5*NTMP
    51           GO TO 6
    52           END
ENTRY/COMMON BLOCK NAMES
```

```
  001447 R  IDCMP
EXTERNAL NAMES
  000002 E    $SE
  001131 E   ISHF
  001235 E  $$BF1
  001002 E  JDCMP
  000703 E    $DO
SYMBOL TABLE
  001367 R 000001
  001444 R 000002
  100004 R  IUNIT
  100005 R   INUM
  100006 R   NTMP
  100007 R     IA
  100010 R     NN
  000002 E    $SE
  000013 R     IB
  001446 R 000365
  001237 R     99
  001131 E   ISHF
  001365 R 000003
  001366 R   $1
  001356 R     0$
  001370 R    IST
  001372 R   NWDS
  001371 R 000360
  001373 R 000000
  001306 R      1
  000467 R      3
  001225 E  $$BF1
  001374 R     NB
  001375 R   IOUT
  000552 R      4
  001376 R 000007
  001033 R      6
  001400 R     MN
  001377 R 000020
  000570 R     11
  001401 R      I
  001404 R     LL
  001402 R 000062
  001403 R 000061
  001002 E  JDCMP
  001405 R 177777
  001406 R   $1 0
  001407 R 000012
  001410 R   $1 1
  000703 E    $DO
  000646 R     12
  001411 R 000012
  001412 R 000215
  001413 R 000006
  001415 R 000011
```

```
001414 R 000277
001420 R 000013
001417 R 000027
001416 R 000310
001422 R 000017
001421 R 000337
001424 R 000015
001423 R 000356
001425 R   NBEG
001426 R 000013
001427 R 000014
001430 R    $1 2
001431 R    $1 3
001432 R    $1 4
001433 R 000015
001434 R    $1 5
001435 R    $1 6
001436 R    $1 7
001437 R 000016
001440 R    $1 8
001441 R 000017
001442 R 000060
001224 R      98
001443 R 007777
001445 R 000005
0 ERRORS COMPILATION COMPLETE
```

```
     1          SUBROUTINE JDCMP(IPK,IA,IB,N,MM)
     2          DIMENSION IB(N),IA(2)
     3          IF(IPK.NE.12)GO TO 1
     4          CALL ISHF(IA,IB,N)
     5          MM=MM+N
     6          RETURN
     7  1       IF(IPK.NE.0)GO TO 2
     8          CALL ISHF(IA,IB,1)
     9          DO 22 I=2,N
    10  22      IB(I)=IB(1)
    11          MM=MM+1
    12          RETURN
    13  2       CALL ISHF(IA,IB,1)
    14          NTOWD=12/IPK
    15          L=2
    16          NN=N-1
    17          DO 3 I=1,NN,NTOWD
    18          CALL IUNPK(IA(L),IB(I+1),IPK,NTOWD)
    19          L=L+1
    20  3       CONTINUE
    21          MM=MM+L
    22          DO 4 I=2,N
    23  4       IB(I)=IB(I)+IB(I-1)
    24          RETURN
    25          END
ENTRY/COMMON BLOCK NAMES
  000341 R  JDCMP
EXTERNAL NAMES
  000002 E    $SE
  000145 E   ISHF
  000315 E    $DO
  000225 E  IUNPK
SYMBOL TABLE
  000326 R 000001
  000327 R 000002
  100004 R     IPK
  100005 R      IA
  100006 R      IB
  100007 R       N
  100010 R      MM
  000002 E    $SE
  000324 R 000014
  000311 R      0$
  000054 R       1
  000145 E   ISHF
  000325 R 000000
  000144 R       2
  000101 R      22
  000330 R       I
  000331 R 177777
  000332 R    $1
  000315 E    $DO
  000333 R  NTOWD
```



```
  000334 R       L
  000335 R      NN
  000237 R       3
  000225 E  IUNPK
  000336 R    $1 0
  000264 R       4
  000337 R 177776
  000340 R    $1 1
  0 ERRORS COMPILATION COMPLETE
```

# Appendix C

Each physical record consists of 2560 or fewer 12-bit words, (512 CDC 6600 words), and contains as many logical records as can be completely contained in 2559 words. All logical records contain 3 control words, except for type -1.

Word 1 = type of record
Word 2 = subtype
Word 3 = number of 12-bit words this logical record (including control words).

At present, three types of records have been reserved.

Type 0 = header record, not yet defined
Type 1 = frame data
Type -1 = end of physical record (12-bit, 2's complement -1).

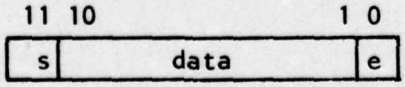
For type 1 records, subtype 0 has been defined.

Word 1 = 1
Word 2 = 0
Word 3 = N = number of words
1 - good frame
Word 4 = Status = -1 - error frame
-2 - timeout

The rest of the frame is as follows for the various frame statuses.

| | Status | | |
|---|---|---|---|
| | -2 | -1 | -1 |
| Word 5 | "received" | time | seconds |
| Word 6 | " | " | milliseconds |
| Word 7 | Buffer contents | | |
| Word 8-N | partial frame received, if any. | | 240 data words received in arranged form by instrument as given in AFCRL-TR-75-0588. |

Data and buffer contents are two's complement




Status 1

| | | |
|---|---|---|
| Word 5 - | packing factor | SCX |
| Word 6 - | " | SCY |
| Word 7 - | " | SCZ |
| Word 8 - | " | FFX |
| Word 9 - | " | FFY |
| Word 10 - | " | FFZ |

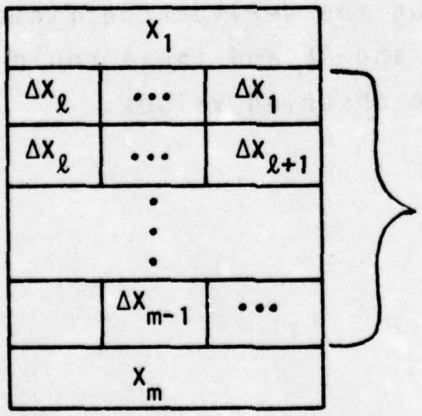
Word 11 - packing factor  Analog spares
Word 12 -  "  "  Digital spares

The rest of the frame is then the same as words 5-N of status -1 frames except that relevant data is compressed according to the scheme detailed below.

Packing Scheme

If a particular instrument gives in readings, $X_1, \ldots\ldots, X_m$, m-1 $\Delta X_i$ are calculated where $\Delta X_i = X_{i+1} - X_i$. If all the $\Delta X_i$ are 0, the packing factor K is chosen to be 0. Otherwise K is the minimum of 2, 3, 4, 6, 12 such that $-2^{K-1} \leq \Delta X_i \leq 2^{K-1} - 1$ for all $i=1, \ldots, m-1$.

When K=12, actual data is stored. When K≠12, the packed instrument format is as follows, where $\ell = 12/K$




It should be noted that $X_1$ and $X_m$ are stored with error bits (which are zero because we only pack error-free frames) but that the $\Delta X_i$ are data deltas only, in two's complement form.

Therefore, after the $\Delta X_i$'s are expanded to full words, $X_i = X_{i-1} + 2\Delta X_{i-1}$.

## 2. MATHEMATICAL ANALYSIS

The main activity of the past year has been writing programs to handle large amounts of data.

### (a) MULTIPLE CORRELATION

A multiple correlation program, MLTCOR, (Program 1) was written that yields the partial regression coefficients between ground measured geophysical parameters such as Kp, Ap and AE and various solar wind parameters such as bulk velocity, magnetic field, and the direction of the magnetic field. Using the derived coefficients predictions were made of Kp, Ap, and AE and these could be plotted and compared with the observed values.

Program 1

```
BELMC,CM110000,T150.                                    2261     BELLEW     10
ATTACH,TAPE1,CORDTX3218,ID=BELLEW,MR=1.                                    11
ATTACH,TAPE2,IMFX3218,ID=BELLEW,MR=1.                                      12
ATTACH,PEN,ONLINEPEN,MR=1.                                                 13
FTN,SL,PL=77777.                                                           14
LOSET(LIB=PEN)                                                             15
LGO.                                                                       16
```

```
      PROGRAM MLTCOR(INPUT,OUTPUT,TAPE1,TAPE2,TAPE3,TAPE6=OUTPU          10
     +T)                                                                 11
                                                                         12
         CATALOGED AS MLTCORX3218,CY=5                                   13
                                                                         14
      DIMENSION B(6),SB(6)                                               15
      DIMENSION PROGID(3)                                                16
      COMMON/SUB/B,SB,SY,BNOT                                            17
      COMMON/DAT/LAG,JMPDT                                               18
      DATA PROGID/6RBELLEW,4R2261,2RB9/                                  19
      CALL PLTID3(PROGID,200.0,11.0,0.75)                                20
      F=2.5                                                              21
      TOL=.0001                                                          22
      N=6                                                                23
      M=100                                                              24
      SIGN=-1                                                            25
      JMPDT=0                                                            26
      DO 2 J=1,2                                                         27
      LAG=1                                                              28
      DO 1 I=1,1                                                         29
      CALL SETUP(N,M,SIGN)                                               30
      CALL SUB1(F,TOL,M)                                                 31
      LAG=LAG+1                                                          32
      JMPDT=JMPDT+1                                                      33
      CALL ENDPLT                                                        34
      STOP                                                               35
      END                                                                36
```

```
      SUBROUTINE SETUP(N,M,SIGN)                                        37
                                                                        38
      SETUP READS DATA FROM TAPE1 AND CONSTRUCTS                        39
      A MATRIX OF THE CROSS CORRELATION COEFFICIENTS                    40
                                                                        41
      DIMENSION A(6,6),Y(600,6),YB(6),SIG(6),R(6,6)                     42
      COMMON/SET/YB,SIG,R,NM,DF,Y                                       43
      COMMON/DAT/LAG,JMPDT                                              44
      COMMON/MUL/KNT                                                    45
      REWIND 3                                                          46
      PRINT 100                                                         47
      FORMAT(1H1)                                                       48
      FORMAT()                                                          49
      PRINT 300, SIGN,LAG                                               50
      FORMAT(/*      SIGN=*,F5.1,*       LAG=*,I2///)                   51
        N NUMBER OF DEPENDENT AND INDEPENDENT VARIABLES                 52
        M NUMBER OF OBSERVATIONS                                        53
      MSAV=M                                                            54
      CALL DATIN(M)                                                     55
      PI=2.0*ACOS(0.0)                                                  56
      FP=1.0/(4*PI)                                                     57
      REWIND 3                                                          58
      NM=N-1                                                            59
      DF=M-1                                                            60
      KNT=0                                                             61
      DO 1 J=1,M                                                        62
      READ(3) IDATE,B,BX,BY,BZ,SGBX,SGBY,SGBZ,                          63
     +VTH,RHO,V,VX,VY,VZ,AE                                             64
      PRINT 75,B,BX,BY,BZ,VTH,RHO,V,VX,VY,VZ,AE                         65
      FORMAT(4F10.4/7F10.4)                                             66
      IF(SIGN*BZ.LT.0.0) GO TO 1                                        67
      KNT=KNT+1                                                         68
      Y(KNT,1)=RHO*V*V                                                  69
      SX=FP*((BY*BY+BZ*BZ)*VX-(VY*BY+VZ*BZ)*BX)                         70
      SY=FP*((BX*BX+BZ*BZ)*VY-(VX*BX+VZ*BZ)*BY)                         71
      SZ=FP*((BX*BX+BY*BY)*VZ-(VX*BX+VY*BY)*BZ)                         72
      S=SQRT(SX*SX+SY*SY+SZ*SZ)                                         73
      PRINT 11,S,SX,SY,SZ                                               74
      Y(KNT,2)=S                                                        75
      Y(KNT,2)=BZ                                                       76
      Y(KNT,3)=V*BZ                                                     77
      Y(KNT,5)=SQRT(SGBX*SGBX+SGBY*SGBY+SGBZ*SGBZ)                      78
      Y(KNT,4)=BZ*BZ*V                                                  79
      Y(KNT,2)=Y(KNT,3)*Y(KNT,5)                                        80
      Y(KNT,6)=AE                                                       81
    1 CONTINUE                                                          82
      M=KNT                                                             83
      PRINT 15,N,M                                                      84
      FORMAT(/2I10/)                                                    85
      PRINT 11,((Y(J,I),I=1,N),J=1,M)                                   86
      PRINT 200                                                         87
```

```
11    FORMAT(1X,1P6E12.3)                                              88
C        DO 6 FINDS THE MEAN OF EACH VARIABLE                          89
      DO 6 I=1,N                                                       91
      YB(I)=0                                                          91
      DO 5 J=1,M                                                       92
5     YB(I)=YB(I)+Y(J,I)                                               93
6     YB(I)=YB(I)/M                                                    94
      PRINT 11,(YB(I),I=1,N)                                           95
      PRINT 200                                                        96
C        DO 7 FINDS THE DEVIATIONS                                     97
      DO 7 I=1,N                                                       98
      DO 7 J=1,M                                                       99
7     Y(J,I)=Y(J,I)-YB(I)                                              100
      PRINT 12,((Y(J,I),I=1,N),J=1,M)                                  101
      PRINT 200                                                        102
C        DO 4 CONSTRUCTS A MATRIX OF SQUARES AND                       103
C        CROSS PRODUCTS OF DEVIATIONS                                  104
      DO 4 I=1,N                                                       105
      DO 4 K=I,N                                                       106
      A(I,K)=0                                                         107
      DO 8 J=1,M                                                       108
8     A(I,K)=A(I,K)+Y(J,I)*Y(J,K)                                      109
      A(K,I)=A(I,K)                                                    110
4     CONTINUE                                                         111
C     PRINT 12,((A(J,I),I=1,N),J=1,N)                                  112
      PRINT 200                                                        113
12    FORMAT(1P6E12.3).                                                114
C                                                                      115
      DO 13 I=1,N                                                      116
13    SIG(I)=SQRT(A(I,I))                                              117
      PRINT 11,(SIG(I),I=1,N)                                          118
      PRINT 200                                                        119
      DO 14 I=1,N                                                      120
      DO 14 J=I,N                                                      121
      R(I,J)=A(I,J)/(SIG(I)*SIG(J))                                    122
14    R(J,I)=R(I,J)                                                    123
      PRINT 12,((R(I,J),J=1,N),I=1,N)                                  124
      M=MSAV                                                           125
C                                                                      126
      RETURN                                                           127
      END                                                              128
```

```
      SUBROUTINE SUB1(F,TOL,M)                                          129
                                                                        130
      DIMENSION B(6),SB(6)                                              131
      DIMENSION YB(6),SIG(6),R(6,6)                                     132
      DIMENSION Y(600,6)                                                133
      COMMON/SUB/B,SB,SY,BNOT                                           134
      COMMON/SET/YB,SIG,R,NM,DF,Y                                       135
                                                                        136
      PRINT 200                                                         137
      FORMAT()                                                          138
      CLRL=0                                                            139
      F1=F                                                              140
      N=NM+1                                                            141
      F2=F                                                              142
      PRINT 940,F1,F2                                                   143
      FORMAT(*     F1,F2 =*1P2E12.3)                                    144
      VMIN=10**10                                                       145
      VMAX=0.0                                                          146
      NMAX=0                                                            147
      NMIN=0                                                            148
      SY=SIG(N)*SQRT(R(N,N)/DF)                                         149
      PRINT 120,SY                                                      150
      FORMAT(*       SY=*F10.5)                                         151
      DO 10 J=1,NM                                                      152
      B(J)=0                                                            153
                                                                        154
      I=1                                                               155
      CONTINUE                                                          156
         PRINT*,R(I,I),I,I                                              157
                                                                        158
         THIS TEST REDUCES THE POSSIBILITY OF DEGENERACY                159
         WHEN AN INDEPENDENT VARIABLE IS APPROXIMATELY A                160
         LINEAR COMBINATION OF OTHER INDEPENDENT VARIABLES.             161
      IF(R(I,I).LE.TOL) GO TO 3                                         162
                                                                        163
         LINEAR COMBIMATION OF OTHER INDEPENDENT VARIABLES              164
      VI=R(I,N)*R(N,I)/R(I,I)                                           165
         PRINT*,VI,VMAX,I                                               166
      IF(VI.EQ.0) GO TO 3                                               167
      IF(VI.LT.0.0) GO TO 1                                             168
      IF(VI.LE.VMAX) GO TO 3                                            169
      VMAX=VI                                                           170
      NMAX=I                                                            171
         PRINT*,VMAX,NMAX                                               172
      GO TO 3                                                           173
      B(I)=SIG(N)/SIG(I)*R(I,N)                                         174
         PRINT*,B(I),I                                                  175
      SB(I)=SY/SIG(I)*SQRT(1.0/R(I,I))                                  176
      SB(I)=SY/SIG(I)*SQRT(R(I,I))                                      177
      PRINT 430,B(I),SB(I),I                                            178
      FORMAT(1P2E12.3,0PI10)                                            179
```

```
      IF(ABS(VI).GE.ABS(VMIN)) GO TO 3                                  180
      VMIN=VI                                                           181
      NMIN=I                                                            182
      IF(I.EQ.NM) GO TO 100                                             183
      I=I+1                                                             184
      GO TO 4                                                           185
      SUM=0                                                             186
      DO 101 J=1,I                                                      187
      SUM=SUM+B(J)*YB(J)                                                188
      BNOT=YB(N)-SUM                                                    189
      IF(CLRL.NE.0) CALL COMLC(B)                                       190
      FIN=ABS(VMIN)*DF/R(N,N)                                           191
      PRINT 710,FIN,F2,BNOT                                             192
      FORMAT(*      FIN,F2,BNOT =*1P3E12.3)                             193
      IF(FIN.GE.F2) GO TO 6                                             194
      K=NMIN                                                            195
      PRINT 620,K                                                       196
      FORMAT(*      K=NMIN=*I2)                                         197
      DF=DF+1                                                           198
      GO TO 5                                                           199
      CONTINUE                                                          200
      FOUT=VMAX*(DF-1.0)/(R(N,N)-VMAX)                                  201
      PRINT 711,FOUT,F1                                                 202
      FORMAT(*      FOUT,F1 =*1P2E12.3)                                 203
      CONTINUE                                                          204
      IF(VMAX*(DF-1.0)/(R(N,N)-VMAX).LE.F1) RETURN                      205
      K=NMAX                                                            206
      PRINT 123,K                                                       207
      FORMAT(*      K=NMAX=*I2)                                         208
      DF=DF-1                                                           209
      CALL NUMAT(K,N)                                                   210
      CLRL=1                                                            211
      GO TO 7                                                           212
                                                                        213
      END                                                               214
```

```
      SUBROUTINE NUMAT(K,N)                                              215
      DIMENSION YB(6),SIG(6),R(6,6),D(6,6)                               216
      DIMENSION Y(600,6)                                                 217
      COMMON/SET/YB,SIG,R,NM,DF,Y                                        218
      PRINT 200                                                          219
      PRINT 50,((R(I,J),J=1,N),I=1,N)                                    220
      FORMAT(1P6E12.3)                                                   221
      FORMAT()                                                           222
      DO 1 I=1,N                                                         223
      DO 1 J=1,N                                                         224
      IF(I.EQ.K.AND.J.EQ.K) GO TO 10                                     225
      IF(I.EQ.K.AND.J.NE.K) GO TO 9                                      226
      IF(I.NE.K.AND.J.EQ.K) GO TO 8                                      227
      D(I,J)=(R(I,J)*R(K,K)-R(I,K)*R(K,J))/R(K,K)                        228
      GO TO 1                                                            229
    8 D(I,J)=-R(I,K)/R(K,K)                                              230
      GO TO 1                                                            231
    9 D(I,J)=R(K,J)/R(K,K)                                               232
      GO TO 1                                                            233
   10 D(I,J)=1.0/R(I,J)                                                  234
    1 CONTINUE                                                           235
      DO 2 I=1,N                                                         236
      DO 2 J=1,N                                                         237
    2 R(I,J)=D(I,J)                                                      238
      PRINT 200                                                          239
      PRINT 50,((R(I,J),J=1,N),I=1,N)                                    240
      RETURN                                                             241
      END                                                                242
```

```
      SUBROUTINE DATIN(M)                                               243
      DIMENSION X(14),IX(14),ID(14)                                     244
      COMMON/DAT/LAG,JMPDT                                              245
      IF(JMPDT.NE.0) GO TO 10                                           246
      REWIND 1                                                          247
      REWIND 2                                                          248
      REWIND 3                                                          249
      LAGL=LAG                                                          250
      LAGM=1                                                            251
      LLAG=LAG-1                                                        252
      KOUNT=0                                                           253
      READ(1) IDATE1,(X(I),I=1,13)                                      254
      IF(EOF(1)) 5,2                                                    255
      READ(2) IDATE2,IX11                                               256
      IF(EOF(2)) 7,9                                                    257
      X(14)=IX11                                                        258
      IF(IDATE1.GT.IDATE2) GO TO 2                                      259
      IF(IDATE2.EQ.IDATE1) GO TO 4                                      260
      READ(1) IDATE1,(X(I),I=1,13)                                      261
      IF(IDATE1.GT.IDATE2) GO TO 2                                      262
      IF(EOF(1)) 5,3                                                    263
      CONTINUE                                                          264
      IF(LAG.EQ.0) GO TO 12                                             265
      IF(LAGL.EQ.LAG.OR.LAG.EQ.1) GO TO 18                              266
      DO 17 I=1,LLAG                                                    267
      ID(I)=ID(I+1)                                                     268
      IX(I)=IX(I+1)                                                     269
      CONTINUE                                                          270
      I=LAG                                                             271
      GO TO 22                                                          272
      I=1                                                               273
      READ(2) ID(I),IX(I)                                               274
      IX11=IX(I)                                                        275
      IDD1=IDATE1-(IDATE1/100)*100                                      276
      IDD2=ID(I)-(ID(I)/100)*100                                        277
      IDIF=IDD2-IDD1                                                    278
      IF(IDIF.EQ.LAG.OR.IDIF.EQ.LAG-24) GO TO 19                        279
      IF(I.EQ.LAG.AND.LAGL.EQ.LAG) GO TO 19                             280
      IF(ID(I)-IDATE1.EQ.LAG) GO TO 19                                  281
      I=I+1                                                             282
      GO TO 22                                                          283
      LAGM=LAG                                                          284
      X(14)=IX11                                                        285
      WRITE(3) IDATE1,(X(I),I=1,14)                                     286
      KOUNT=KOUNT+1                                                     287
      PRINT 54, IDATE1,(X(I),I=1,7)                                     288
      FORMAT(I10,7F10.3)                                                289
      PRINT 56,(X(I),I=8,14)                                            290
      FORMAT(7F10.3)                                                    291
      IF(KOUNT.EQ.M) RETURN                                             292
      LAGL=1                                                            293
```

```
      IF(LAG.EQ.0) GO TO 1                                              294
      I=1                                                               295
      READ(1) IDATE1,(X(J),J=1,13)                                      296
      IF(EOF(1)) 5,55                                                   297
      CONTINUE                                                          298
      IF(IDATE1.EQ.ID(I)) GO TO 4                                       299
      I=I+1                                                             300
      IF(I.LE.LAG) GO TO 55                                             301
      LAGM=1                                                            302
      LAGL=LAG                                                          303
      GO TO 2                                                           304
      PRINT 6                                                           305
      FORMAT(*      EOF(1)*)                                            306
      RETURN                                                            307
      PRINT 8                                                           308
      FORMAT(*      EOF(2)*)                                            309
      RETURN                                                            310
      END                                                               311
      SUBROUTINE COMLC(B)                                               312
C                                                                       313
C     THIS SUBROUTINE CALCULATES THE TOTAL CORRELATION                  314
C     COEFFICIENT OF THE OBSERVED DEPENDENT VARIABLE Y,                 315
C     AND THE PREDICTED DEPENDENT VARIABLE YJ.                          316
C                                                                       317
      DIMENSION YB(6),SIG(6),R(6,6),Y(600,6)                            318
      DIMENSION B(6)                                                    319
      DIMENSION PY(100),PX(100)                                         320
      COMMON/SET/YB,SIG,R,NM,DF,Y                                       321
      COMMON/MUL/M                                                      322
      N=NM+1                                                            323
      CYJY=0                                                            324
      SIGYJ=0                                                           325
      DO 2 I=1,M                                                        326
      YJ=0                                                              327
      DO 1 J=1,NM                                                       328
      YJ=YJ+B(J)*Y(I,J)                                                 329
      PY(I)=Y(I,N)                                                      330
      PX(I)=YJ                                                          331
      CYJY=CYJY+YJ*Y(I,N)                                               332
      SIGYJ=SIGYJ+YJ*YJ                                                 333
      SIGYJ=SQRT(SIGYJ)                                                 334
      RML=CYJY/(SIGYJ*SIG(N))                                           335
      P=0                                                               336
      DO 3 I=1,NM                                                       337
      IF(B(I).NE.0) P=P+1                                               338
      CONTINUE                                                          339
      RR=RML*RML                                                        340
      FTEST=RR/(1-RR)*(M-P-1)/P                                         341
      PRINT*,"   RMUL,FTEST,M,P=",RML,FTEST,M,P                         342
      CALL SPLOT(PX,PY,M,99,XMAX,YMAX,X0,Y0)                            343
      CALL PLOTTER(PX,PY,-1000.0,1000.0,5.0,0.0,6,11,1,M)               344
      RETURN                                                            345
      END                                                               346
```

```
      SUBROUTINE SPLOT(X,Y,N,ISCAL,XMAX,YMAX,X0,Y0)                    347
      DIMENSION X(1),Y(1),G(101,61)                                    348
      DATA BLANK/1H /,AXIS/1H./,POINT/1H*/                             349
***                                                                    350
* X(I)= ARRAY FOR X-AXIS                                              351
* Y(I)= ARRAY FOR Y-AXIS                                              352
* N= NUMBER OF POINTS                                                 353
* X0,Y0= ORIGIN OF GRAPH                                              354
* DX,DY= SCALE FACTOR MULTIPLYING POINT VALUES                        355
* **NOTE**TOO HIGH OR TOO LOW X,Y WILL CAUSE ERROR IF ISCAL.NE.       356
     +99                                                              356
* IF ISCAL=99 THEN DO AUTO-SCALING                                    357
      IF(ISCAL.NE.99) GO TO 99                                         358
***                                                                    359
      X0=XMAX=X(1)                                                     360
      Y0=YMAX=Y(1)                                                     361
      DO 98 K=2,N                                                      362
      IF(X(K).LT.X0) X0=X(K)                                           363
      IF(Y(K).LT.Y0) Y0=Y(K)                                           364
      IF(X(K).GT.XMAX) XMAX=X(K)                                       365
      IF(Y(K).GT.YMAX) YMAX=Y(K)                                       366
98    CONTINUE                                                         367
99    CONTINUE                                                         368
      DX=100./(XMAX-X0)                                                369
      DY=60./(YMAX-Y0)                                                 370
      DO 1 I=1,101                                                     371
1     G(I,1)=AXIS                                                      372
      DO 7 I=1,61                                                      373
7     G(1,I)=AXIS                                                      374
      DO 2 I=2,101                                                     375
      DO 2 J=2,61                                                      376
2     G(I,J)=BLANK                                                     377
* PLOT POINTS                                                         378
      DO 3 I=1,N                                                       379
* 1. IS ADDED SO X=0 DOES NOT GIVE A 0 INDEX                          380
      IX=(X(I)-X0)*DX+1.                                               381
      IY=(Y(I)-Y0)*DY+1.                                               382
3     G(IX,IY)=POINT                                                   383
      DELX=1./DX                                                       384
      DELY=1./DY                                                       385
      PRINT 6, X0,XMAX,DELX,Y0,YMAX,DELY,N                             386
6     FORMAT(1H1,10X,*XMIN,XMAX,DELX=*,3F10.3,2X,                      387
     +*YMIN,YMAX,DELY=*,3F10.3,2X,*NO. PTS.=*,I6)                      388
      DO 4 I=1,61                                                      389
* STEP NECESSARY SO Y-AXIS COMES OUT RIGHT SIDE UP                    390
      II=62-I                                                          391
4     PRINT 100,(G(J,II),J=1,101)                                      392
100   FORMAT (1H ,10X,101A1)                                           393
      RETURN                                                           394
      END                                                              395
```

```
      SUBROUTINE PLOTTER(X,BF,XB,XE,XO,YO,NOX,NNX,NORM,N)                396
      DIMENSION X(1),BF(1)                                               397
      COMMON/YMB/YM,B                                                    398
         IF(NORM.EQ.0) GO TO 3                                           399
      CALL SLIN(BF,NORM,FB,FS,N)                                         400
         GO TO 2                                                         401
         YM=1.0                                                          402
         B=5.0                                                           403
      CALL BOXER(XO,YO,10.0,10.0,XB,XE,NOX,NNX)                          404
         PRINT 10                                                        405
         FORMAT(*      BOXER CALLED*)                                    406
      EX=(X(1)-XB)*10.0/(XE-XB)                                          407
         IF(EX.LT.0.0.OR.EX.GT.10.0) GO TO 25                            408
      Y=YM*BF(1)+B                                                       409
      CALL PLOT(EX,Y,3)                                                  410
      CALL SYMBOL(EX,Y,.2,1,0.0,-1)                                      411
         PRINT 9,X(1),BF(1),EX,Y                                         412
      GO TO 21                                                           413
      CALL PLOT(0,0,3)                                                   414
      DO 1 I=2,N                                                         415
      EX=(X(I)-XB)*10.0/(XE-XB)                                          416
         IF(EX.LT.0.0.OR.EX.GT.10.0) GO TO 1                             417
      Y=YM*BF(I)+B                                                       418
         IF(MOD(I,50).EQ.0) PRINT 9,X(I),BF(I),EX,Y,I                    419
         FORMAT(2(0PF12.4,1PE13.4),I10)                                  420
      CALL PLOT(EX,Y,2)                                                  421
      CALL SYMBOL(EX,Y,.2,1,0.0,-1)                                      422
    1 CONTINUE                                                           423
         RETURN                                                          424
      END                                                                425
```

```
      SUBROUTINE BOXER(XO,YO,XL,YL,XB,XE,NDX,NNX)                       426
      COMMON/YMB/YM,R                                                   427
      COMMON /CONST/PI,XC,ALPHA,XMINUS,X1,X2,X3,X4,XPLUS,XJN,XN         428
     +U,XM                                                              428
     1,XN,DEL,DEL2,A,EXPA,XNU2,ALFA4,ALFA2,ALFA42                       429
      CALL PLOT(XO,YO,-3)                                               430
      CALL PLOT(XL,0,2)                                                 431
      CALL PLOT(XL,YL,2)                                                432
      CALL PLOT(0,YL,2)                                                 433
      CALL PLOT(0,0,2)                                                  434
      SIGN=1.0                                                          435
      YB=0.0                                                            436
      DY=YB+SIGN*.12                                                    437
      X=XL/NDX                                                          438
      LU=NDX-1                                                          439
         WRITE(6,130) X,LU                                              440
         FORMAT(F12.5,I10)                                              441
      DO 1 I=1,LU                                                       442
      CALL PLOT(X,YB,3)                                                 443
      CALL PLOT(X,DY,2)                                                 444
      X=X+XL/NDX                                                        445
      IF(SIGN.LT.0.0) GO TO 5                                           446
      SIGN=-SIGN                                                        447
      YB=10.0                                                           448
      GO TO 4                                                           449
      XFI=0.0                                                           450
         WRITE(6,110)                                                   451
         FORMAT(*          X-AXIS COMPLETED    *)                      452
      SIGN=1.0                                                          453
      DX=XFI+SIGN*.12                                                   454
      Y=.5                                                              455
      DO 2 I=1,19                                                       456
      CALL PLOT(XFI,Y,3)                                                457
      CALL PLOT(DX,Y,2)                                                 458
      Y=Y+.5                                                            459
      IF(SIGN.LT.0.0) GO TO 7                                           460
      SIGN=-SIGN                                                        461
      XFI=XL                                                            462
      GO TO 6                                                           463
      CONTINUE                                                          464
         WRITE(6,120)                                                   465
         FORMAT(*          Y-AXIS COMPLETED    *)                      466
      CALL PLOT(XO,YO,3)                                                467
      Y=-1.0                                                            468
      X=1.0                                                             469
      DO 3 I=1,9                                                        470
      XI=X*(XE-XB)/XL+XB                                                471
      IF(XI.LT.XO) GO TO 3                                              472
      SQ=SQRT(XO/XI)                                                    473
      DEG=57.2958*ACOS(SQ)                                              474
         PRINT 10,X,XI,DEG,I                                            475
```

```
   FORMAT(3F12.5,I10)                                                    476
CALL NUMBER(X,Y,.15,DEG,0.),1)                                           477
X=X+1.0                                                                  478
XNUM=XB                                                                  479
Y=-.5                                                                    480
X=0                                                                      481
DO 8 I=1,NNX                                                             482
XNUM=X*(XE-XB)/XL+XB                                                     483
PRINT 100,X,XNUM                                                         484
FORMAT(2F12.5)                                                           485
CALL NUMBER(X,Y,.15,XNUM,0.0,1)                                          486
X=X+XL/(NNX-1)                                                           487
X=-1.0                                                                   488
Y=0                                                                      489
YNUM=(Y-B)/YM                                                            490
DO 9 I=1,11                                                              491
CALL NUMBER(X,Y,.15,YNUM,0).0,1)                                         492
   PRINT 10,X,Y,YNUM,I                                                   493
Y=Y+1.0                                                                  494
YNUM=(Y-B)/YM                                                            495
RETURN                                                                   496
END                                                                      497
```

```
      SUBROUTINE SLIN(F,NORM,FB,FS,N)                                   493
      DIMENSION F(1)                                                    499
      COMMON/YMB/YM,B                                                   500
      FB=F(1)                                                           501
      FS=F(1)                                                           502
      DO 1 I=2,N                                                        503
      IF(F(I).GT.FB) FB=F(I)                                            504
      IF(F(I).LT.FS) FS=F(I)                                            505
    1 CONTINUE                                                          506
      IF(NORM.NE.1) GO TO 25                                            507
      IF(ABS(FS).GT.FB) FB=FS                                           508
      DO 20 I=1,N                                                       509
      F(I)=F(I)/FB                                                      510
      B=0.0                                                             511
      YM=10.0                                                           512
      PRINT 12,FB,FS,YM,B                                               513
      IF(FS.LT.0.0) GO TO 10                                            514
      RETURN                                                            515
      YM=5.0                                                            516
      B=5.0                                                             517
      PRINT 12,FB,FS,YM,B                                               518
      RETURN                                                            519
      IF(FS.LT.0) GO TO 2                                               520
      B=0                                                               521
      U=.00001                                                          522
      IF(U.GT.FB) GO TO 3                                               523
      U=U*10.0                                                          524
      GO TO 4                                                           525
      YM=10.0/U                                                         526
      IF(YM*FB.LE.5.0) YM=2.0*YM                                        527
      PRINT 12,FB,FS,YM,B                                               528
      RETURN                                                            529
      IF(FB.LT.0) GO TO 7                                               530
      B=5.0                                                             531
      IF(ABS(FS).GT.FB) FB=ABS(FS)                                      532
      U=.00001                                                          533
      IF(U.GT.FB) GO TO 5                                               534
      U=U*10.0                                                          535
      GO TO 6                                                           536
      YM=5.0/U                                                          537
      IF(YM*FB.LE.2.5) YM=2.0*YM                                        538
      PRINT 12,FB,FS,YM,B                                               539
      RETURN                                                            540
      B=10.0                                                            541
      FB=ABS(FS)                                                        542
      U=.00001                                                          543
      IF(U.GT.FB) GO TO 8                                               544
      U=U*10.0                                                          545
      GO TO 9                                                           546
      YM=10.0/U                                                         547
      IF(YM*FB.LE.5) YM=2.0*YM                                          548
      PRINT 12,FB,FS,YM,B                                               549
   12 FORMAT(4F12.5)                                                    550
      RETURN                                                            551
      END                                                               552
```

# COMPARISON OF SOLAR-WIND PARAMETERS AND GEOMAGNETIC ACTIVITY INDICES

Philip M. Pazich

William F. Bellew

March 1976

Magnetospheric Dynamics Branch

Space Physics Division

## I. DATA

A comparison of solar-wind parameters and geomagnetic indices was made to determine the parameters that could best be used for prediction purposes. The solar-wind data consisted of 1000 hourly averages of plasma ($v$ and $\rho$) and magnetic field data ($Bx$, $By$, $Bz$, $\sigma x$, $\sigma y$, $\sigma z$) from the Explorer 33 satellite. The data were measured between 19 January and 6 April 1968. An earlier study by Garrett et al. (1974) also used data from the Explorer 33 satellite measured during this period. Approximately half of the data used in the current study coincided with the data set of Garrett et al.

Solar-wind data for each hour were recorded on magnetic tape along with the AE index for the same hour and the following two hours so that correlations could be done easily. The Kp and ap indices were read from an additional data file as needed. A given hour of solar-wind data was correlated with the Kp and ap indices for both the three-hour interval in which the solar-wind data were measured and the following three-hour interval.

## II. CORRELATIONS

Functions of solar-wind variables were determined on the basis of physical arguments and previous studies to be those that gave the highest correlations with magnetic activity indices.

The indices are:

| | |
|---|---|
| AE 0, AE 1, AE 2 | - Auroral electrojet index lagged 0, 1, and 2 hours |
| Kp 0, Kp 3<br>ap 0, ap 3 | - Kp and ap indices from the same 3-hour interval as the solar-wind data (Kp 0, ap 0) and from the following 3-hour interval (kp 3, ap 3) |

The solar-wind variables are:

| | |
|---|---|
| BZSE | - magnetic field component Bz, SE coordinates (gammas) |
| BZSM | - " " " " SM " " |
| RHO | - solar wind ion density ($\#/cm^3$) |
| v | - solar wind velocity (km/sec) |
| SIGT or ST | - $(\sigma_x^2+\sigma_y^2+\sigma_z^2)^{\frac{1}{2}}$ for the hourly average |

Correlations between the magnetic indices and functions determined from the solar-wind variables are listed in Table I. Brackets indicate that only negative values of the quantity were used; values > 0 were set equal to 0.

## III. MULTIPLE REGRESSION

A multiple linear regression was performed using the functions and indices listed in Table I. The highest, consistent correlations were for the functions $v\cdot\langle BZSM\rangle$ and $v\cdot SIGT$ for indices AE 1, Kp 0, ap 0. The regression equations for these three indices as determined from the entire data set are given in Table II. The function $\langle BZSM\rangle$ is the solar magnetospheric Bz component when that component is negative and is set equal to zero for positive values of the component. In the 1000 hours of data

there were 509 hours with an average negative Bz and 491 hours with an average positive Bz. Additional terms to the regression equation did not significantly improve the multiple correlation coefficient.

The data was divided into two 500 hour segments to determine the effect of a specific data sample on the regression. The results of these regression analyses are listed in Table III. For the Kp and ap equations, the solar-wind functions were selected in opposite order by the regression procedure for the two data segments. This indicates that the inclusion of the additional term serves only to increase the accuracy of the fit for the specific data.

The regression analyses for the AE index lagged by one hour always gave the highest multiple correlation coefficient and the computed coefficients in the regression equation were the most consistent between data samples. An equation for Kp and ap determined by multiple linear regression on specific data of v, $\langle BZSM \rangle$, and SIGT could be expected to result in correlation coefficients of between .5 and .6 when applied to another data set. An equation to predict AE 1 would give a somewhat better correlation of between .6 and .7 when applied to a different data set

REFERENCE

Garrett, H.B., A.J. Dessler, and T.W. Hill, Influence of solar wind variability on geomagnetic activity, JGR, 79, No. 31, November 1974, p. 4603.

Table I. CORRELATIONS FOR 1000 HOURLY AVERAGES OF SOLAR-WIND DATA AND GEOMAGNETIC INDICES

| | | 1 | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | AE 1 | - | 2 | | | | | | | | | | | | | | | | | |
| 2 | ap 0 | .521 | - | 3 | | | | | | | | | | | | | | | | |
| 3 | Kp 0 | .638 | .878 | - | | | | | | | | | | | | | | | | |
| | | | | | 4 | | | | | | | | | | | | | | | |
| 4 | $-\langle BZSM \rangle$ | .660 | .403 | .466 | - | 5 | | | | | | | | | | | | | | |
| 5 | $-v\cdot\langle BZSM \rangle$ | .700 | .418 | .492 | .982 | - | 6 | | | | | | | | | | | | | |
| 6 | $v\cdot\langle BZSM \rangle^2$ | .542 | .404 | .419 | .919 | .888 | - | 7 | | | | | | | | | | | | |
| 7 | $-\langle BZSM \rangle ST$ | .499 | .451 | .447 | .813 | .795 | .841 | - | 8 | | | | | | | | | | | |
| 8 | $-v\cdot\langle BZSM \rangle ST$ | .555 | .470 | .479 | .816 | .828 | .823 | .984 | - | | | | | | | | | | | |
| | | | | | | | | | | 9 | | | | | | | | | | |
| 9 | SIGT | .217 | .468 | .440 | .070 | .085 | .124 | .372 | .387 | - | 10 | | | | | | | | | |
| 10 | v·SIGT | .228 | .463 | .457 | .028 | .016 | .029 | .235 | .278 | .939 | - | | | | | | | | | |
| | | | | | | | | | | | | 11 | | | | | | | | |
| 11 | -v·BZSM | .627 | .262 | .362 | .816 | .836 | .673 | .637 | .671 | .044 | .117 | - | 12 | | | | | | | |
| 12 | -v·BZSE | .563 | .254 | .311 | .654 | .688 | .534 | .488 | .532 | .012 | .021 | .846 | - | | | | | | | |
| | | | | | | | | | | | | | | 13 | | | | | | |
| 13 | -BZSM | .620 | .279 | .376 | .850 | .848 | .718 | .669 | .684 | .035 | .106 | .984 | .830 | - | 14 | | | | | |
| 14 | -BZSE | .550 | .248 | .307 | .678 | .695 | .566 | .504 | .534 | .016 | .020 | .830 | .985 | .842 | - | | | | | |
| | | | | | | | | | | | | | | | | 15 | | | | |
| 15 | $RHO\cdot v^2$ | .227 | .402 | .390 | .258 | .210 | .303 | .350 | .316 | .373 | .254 | .030 | .054 | .054 | .062 | - | | | | |
| | | | | | | | | | | | | | | | | | 16 | | | |
| 16 | $-\langle BZSE \rangle$ | .592 | .313 | .383 | .782 | .792 | .730 | .594 | .621 | .037 | .001 | .682 | .793 | .694 | .809 | .156 | - | | | |
| | | | | | | | | | | | | | | | | | | 17 | | |
| 17 | AE 0 | .765 | .599 | .699 | .557 | .594 | .466 | .428 | .478 | .237 | .249 | .519 | .461 | .515 | .447 | .270 | .483 | - | 18 | |
| 18 | AE 2 | .770 | .407 | .534 | .531 | .557 | .437 | .402 | .441 | .173 | .181 | .513 | .439 | .511 | .434 | .172 | .457 | .533 | - | 19 |
| 19 | Kp 3 | .620 | .560 | .678 | .477 | .498 | .438 | .435 | .466 | .398 | .411 | .361 | .299 | .373 | .297 | .369 | .386 | .544 | .679 | - |
| 20 | ap 3 | .582 | .564 | .602 | .475 | .490 | .479 | .483 | .506 | .359 | .370 | .351 | .297 | .365 | .293 | .387 | .388 | .504 | .632 | .917 |

Table II. MULTIPLE LINEAR REGRESSION FOR 1000 HOURS OF SOLAR-WIND DATA

R= Multiple Correlation Coefficient

| | | |
|---|---|---|
| Kp 0 = | $-.00219\ v\cdot\langle BZSM\rangle + 5.43$ | R= 0.492 |
| = | $-.00216\ v\cdot\langle BZSM\rangle + .00382\ v\cdot SIGT + 2.68$ | R= 0.666 |
| ap 0 = | $.0117\ v\cdot SIGT + 2.99$ | R= 0.463 |
| = | $.0115\ v\cdot SIGT - .00546\ v\ BZSM + .00319$ | R= 0.618 |
| AE 1 = | $-.146\ v\cdot\langle BZSM\rangle + 114.5$ | R= 0.699 |
| = | $-.145\ v\cdot\langle BZSM\rangle + .0863\ v\cdot SIGT + 52.3$ | R= 0.732 |

Table III. MULTIPLE LINEAR REGRESSION FOR TWO SEGMENTS OF DATA

a) First 500 Hours

| | | |
|---|---|---|
| Kp 0 = | $-.00242\ v\cdot\langle BZSM\rangle + 5.23$ | R= 0.470 |
| = | $-.00249\ v\cdot\langle BZSM\rangle + .00344\ v\cdot SIGT + 2.54$ | |
| ap 0 = | $-.00572\ v\cdot\langle BZSM\rangle + 78.5$ | R= 0.450 |
| = | $-.00590\ v\cdot\langle BZSM\rangle + .00817\ v\cdot SIGT + 1.47$ | R= 0.621 |
| AE 1 = | $-.140\ v\cdot\langle BZSM\rangle + 102.9$ | R= 0.631 |
| = | $-.142\ v\cdot\langle BZSM\rangle + .101\ v\cdot SIGT + 23.6$ | R= 0.701 |

b) Second 500 Hours

| | | |
|---|---|---|
| Kp 0 = | $.00510\ v\cdot SIGT + 3.59$ | R= 0.532 |
| = | $.00467\ v\cdot SIGT - .00183\ v\cdot\langle BZSM\rangle + 2.61$ | R= 0.695 |
| ap 0 = | $.0185\ v\cdot SIGT + .0140$ | R= 0.569 |
| = | $.0175\ v\cdot SIGT - .00459\ v\cdot\langle BZSM\rangle - 2.46$ | R= 0.658 |
| AE 1 = | $-.145\ v\cdot\langle BZSM\rangle + 129.5$ | R= 0.731 |
| = | $-.142\ v\cdot\langle BZSM\rangle + .0767\ v\cdot SIGT + 79.7$ | R= 0.749 |

## (b) SPECTRAL ANALYSIS

The second area of data handling was in preparing micropulsation data for power spectral analysis. The completed programs from this effort

a. take the data from the archive tape and convert it into engineering units and give a quick plot of the data (Program 2) and

b. filter the data so that it will be in a form suitable for spectral analysis (Program 3).

Work is going on to produce a contour plotting routine to display dynamic spectra, that is, spectral density contours will be plotted on a time - frequency graph.

## PROGRAM 2

This program unpacks data from the archive tape and plots it on the printer for a quick look.

The subroutine LEVELER suppresses points that differ from the average by more than one RMS. This is to ensure that the whole plot will not be dominated by a few large spurious values.

## PROGRAM 3

In order to properly frequency analyze a time series it must first be filtered to avoid aliasing. We assume what we are analyzing is a continuous process in time, however it is sampled at discrete intervals, say $\Delta\tau$.

For instance let the power spectrum, P(f), of a stationary process, X(t), be

$$P(f) = \int_{-\infty}^{\infty} C(\tau) e^{-i\omega\tau} d\tau \qquad (\omega=2\pi f)$$

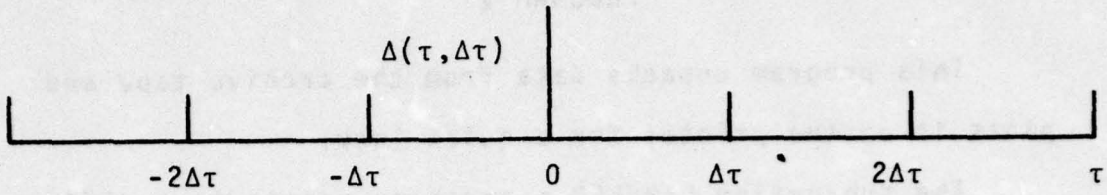
where

$$C(\tau) = \lim_{\tau\to\infty} \int_{-\tau}^{\tau} X(t)\ X(t+\tau) dt$$

is the auto covariance. Let the values of $C(\tau)$ be given only at uniformly spaced values of $\tau$, $\tau=0, \pm\Delta\tau, \pm 2\Delta\tau \pm \ldots$. we can calculate the aliased power spectrum Pa(f) by

$$Pa(f) = \int_{-\infty}^{\infty} \nabla(\tau,\Delta\tau)\ C(\tau) e^{-i\omega\tau} d\tau \qquad (1)$$

where $\nabla(\tau,\Delta\tau)$ is an infinite Dirac comb




The teeth of the comb are δ-functions. Applying the convolution theorem to equation (1) we get

$$Pa(f) = A(f;1/\Delta\tau)\cdot P(f)$$

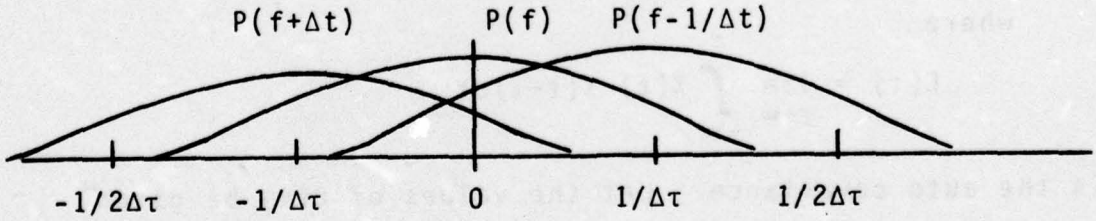
where A is also an infinite Dirac comb

$$A(f;1/\Delta t) = \sum_{q=-\infty}^{q=\infty} \delta(f-q/\Delta t)$$

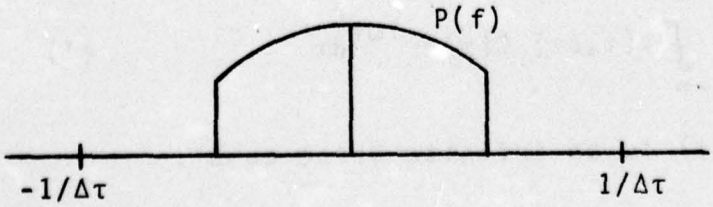
Explicitly we have

$$Pa(f) = \sum_{q=-\infty}^{q=\infty} P(f-q/\Delta t) \qquad (2)$$

From eq.(2) it is easy to see how high frequency components make contributions to low frequency components in the spectrum, see Fig. 1




The only way to eliminate aliasing is for Pa(f) to cut off for $f \geq 1/2\Delta t$



With this cut off Pa(f) is equal to P(f) for $f < f_N = 1/2\Delta\tau$, this frequency is defined as the Nyquist frequency.

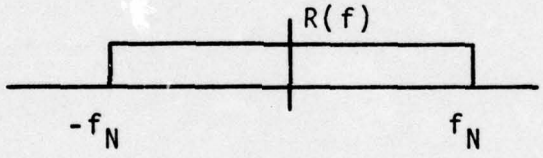
To avoid aliasing the stationary time series I(t) must be passed through a filter $F(\tau)$

$$\phi(t) = \int_{-\infty}^{\infty} F(\tau) I(t-\tau) d\tau$$

Transforming to frequency space we have

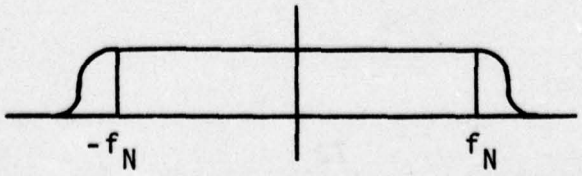
$$\phi(f) = R(f) I(f)$$

If $\phi(f)$ is to reproduce I(f) exactly for $f < f_N$ then R(f), the filter response must be of the form




To achieve such a filter response requires an infinitely long filter in the time domain (This is analogous to a finite wave packet in space requiring an infinite number of frequency components). Taking only a finite amount of data - letting the filter cut off after some time leads to Gibbs phenomenon, that is the filter response oscillates in the neighborhood of the cut off frequency.

This can be mitigated by not requiring a sharp cut off. This is affected in Program 3 by replacing the sharp cut off by a sine-terminator, adding half a sine wave to the filter response



The program then calculates the weights $W_n = F(k\Delta\tau)$ that make up the filter. This filter is described in a NASA report by Behannon and Ness (Design of Numerical Filters for Geomagnetic Data Analysis, NASA TN-D-3341).

Program 2

```
      PROGRAM UPACK(OUTPUT,TAPE1,TAPE2,TAPE3,TAPE4,TAPE6=OUTPUT
     +)
      DIMENSION NP(72),NU(240),NBY(10),NEWT(4),IOLDT(4),IX(1600
     +)
      DIMENSION X(1600),Y(1600),Z(1600)
      DIMENSION IMAX(4),IMIN(4),IMINOLD(4)
NSTATOP=NUMBER OF THE STATION TO BE PLOTTED.
      DO 37 NSTATOP=1,1
      K1=K2=K3=K4=K5=NREC=IEOF=0
      REWIND 1
      REWIND 2
      ERROR=1H
      CALL DECODER(NU,NREC,IEOF)
      IF(IEOF) 21,7
      CONTINUE
      IF(MOD(NREC,240).EQ.1) PRINT*," >>>>>>>>>>>NREC=",NREC,">
     +>>>>>>>>"
      NSTU=2*NSTATOP
      IF(NU(186).NE.0.A.NU(231).EQ.NSTU)PRINT*,"  SCALE CHANGE
     +AT NRE
     1C#=",NREC,"   ",NU(184),NU(185),NU(186),"   ",NU(161),NU(
     +170)
      IF(NU(226).NE.0) ERROR=1H*
      NSTA=6
      IF(NU(231).EQ.2) NSTA=1
      IF(NU(231).EQ.6) NSTA=2
      IF(NU(231).EQ.10) NSTA=3
      IF(NU(231).EQ.14) NSTA=4
      IF(NU(231).EQ.18) NSTA=5
      GO TO (2,6,11,14,18,5) NSTA
      K1=K1+1
      GO TO 55
      K2=K2+1
      GO TO 55
      K3=K3+1
      GO TO 55
      K4=K4+1
      GO TO 55
      K5=K5+1
      IF(NSTA.NE.NSTATOP) GO TO 5
      GO TO (41,42,43,44,45,5) NSTA
      KS=K1
      GO TO 58
      KS=K2
      GO TO 58
      KS=K3
      GO TO 58
      KS=K4
      GO TO 58
      KS=K5
      IMIN(NSTA)=IMAX(NSTA)=NU(151)
```

```
      ISUMX=0                                                           56
      ISUMY=0                                                           57
      ISUMZ=0                                                           58
      DO 17 I=1,10                                                      59
      ISUMX=ISUMX+NU(150+I)                                             60
      ISUMY=ISUMY+NU(160+I)                                             61
      ISUMZ=ISUMX+NU(170+I)                                             62
      IX(KS)=2000*NU(223)+32*NU(224)+NU(225)/2                          63
      X(KS)=ISUMX/320.+NU(181)*32.                                      64
      Y(KS)=ISUMY/320.+NU(184)*32.                                      65
      Z(KS)=ISUMZ/320.+NU(187)*32.                                      66
      IF(NU(185).NE.0) X(KS)=X(KS-1)                                    67
      IF(NU(186).NE.0) Y(KS)=Y(KS-1)                                    68
      IF(NU(189).NE.0) Z(KS)=Z(KS-1)                                    69
      PRINT 191,NU(181),NU(182),NU(183),IX(KS),(NU(I),I=151,160)        70
     +)                                                                 71
      PRINT 191,NU(184),NU(185),NU(186),IX(KS),(NU(I),I=161,170)        72
     +)                                                                 73
      PRINT 191,NU(187),NU(188),NU(189),IX(KS),(NU(I),I=171,180)        74
     +)                                                                 75
  191 FORMAT(3I7,I12/10I7)                                              76
      WRITE(2) Y(KS)                                                    77
      GO TO 5                                                           78
      PRINT*," EOF ENCOUNTERED ON TAPE1"                                79
      PRINT*," LAST RECORD=",NREC                                       80
      PRINT*,"  KS, FIRST, & LAST TIME VALUES ARE:",KS,IX(1),IX        81
     +(KS)                                                              82
      PRINT*," K1,K2,K3,K4,K5,& STATION NUMBER OF PLOTTED DATA=         83
     +",                                                                84
     1K1,K2,K3,K4,K5,NSTAT3P                                            85
  192 FORMAT(*    WRONG PATH*)                                          86
      CALL LEVELER(IX,Y,KS,10)                                          87
      CALL GRAPH(IX,X,KS,1,1,1,6H TIME ,6HMICROP,0.,0.,5HTITLE,         88
     +1,0)                                                              89
      CALL GRAPH(IX,Y,KS,1,1,1,6H TIME ,6HMICROP,0.,0.,5HTITLE,         90
     +1,0)                                                              91
      CALL GRAPH(IX,Z,KS,1,1,1,6H TIME ,6HMICROP,0.,0.,5HTITLE,         92
     +1,0)                                                              93
      CONTINUE                                                          94
      STOP 66666                                                        95
      END                                                               96
                                                                        97
                                                                        98
```

```
      SUBROUTINE GRAPH  (X,Y,N,IDY,I1,I2,XNAME,YNAME,Y1,Y2,BCD,      89
     *                   NBCD,ISZ)                                    90
                                                                      91
C     PRINTER PLOT ROUTINE      OUTPUT ON TAPE6                       92
                                                                      93
C     PARAMETERS                                                      94
C        X(I)           =X COORD MATRIX                               95
C        Y(I,J)         =Y COORD MATRIX                               96
C        N              =NUMBER OF POINTS ON X-AXIS                   97
C        IDY            =VALUE 1ST DIMENSION OF Y(I,J)                98
C                    MUST MATCH DIMENSION OF Y(I,J) IN MAIN PROG.     99
C        I1             =FIRST Y FIELD TO BE PLOTED                   100
C        I2             =LAST  Y FIELD TO BE PLOTED                   101
C        XNAME          =6 CHARACTER TITLE X-AXIS                     102
C        YNAME          =6 CHARACTER TITLE Y-AXIS                     103
C        Y1             =MIN VALUE Y   IF Y<Y1 PLOT AS Y1             104
C        Y2             =MAX VALUE Y   IF Y>Y2 PLOT AS Y2             105
C                  IF Y1=Y2 PROGRAM WILL COMPUTE YMIN AND YMAX         106
C        BCD            =GRAPH TITLE   (A6 FORMAT)                    107
C        NBCD           =NO A6 WRDS IN BCD   MAX=20 FOR 129 COL       108
C                                            MAX=10 FOR  65 COL       109
C                        0=HEADER WRITTEN IN MAIN PROGRAM              110
C        ISZ            =PLOT SIZE CONTROL                            111
C                        0=129 PRINT COL                               112
C                        1= 65 PRINT COL      (TELETYPE)               113
                                                                      114
      DIMENSION X(1),Y(IDY,1),A(101),AXIS(6),BCD(1),MARK(11)          115
      INTEGER STAR,BLANK,A,ZI,X                                       116
       DATA STAR,BLANK/1H*,1H /                                       117
       DATA DASH,ZI/1H-,1HI/                                          118
      DATA MARK /1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H9,1H0,1HX/         119
      NMARK=11                                                        120
      IF(NBCD .GT. 20) NBCD=20                                        121
      IHALF=(22-NBCD)/2                                               122
      LAST=101                                                        123
      HUND=100.0                                                      124
      IF(ISZ .NE. 1) GO TO 10                                         125
      IF(NBCD .GT. 10) NBCD=10                                        126
      IHALF=(11-NBCD)/2                                               127
      LAST=51                                                         128
      HUND=50.0                                                       129
      YMIN=Y1                                                         130
      YMAX=Y2                                                         131
      IF(YMIN.NE.YMAX) GO TO 101                                      132
      YMIN = Y(I1,1)                                                  133
      YMAX = Y(I1,1)                                                  134
      DO 99 J = 1,N                                                   135
      DO 98 I = I1,I2                                                 136
      IF(YMAX.LT.Y(I,J)) YMAX=Y(I,J)                                  137
      IF(YMIN.GT.Y(I,J)) YMIN=Y(I,J)                                  138
   98 CONTINUE                                                        139
```

```
      CONTINUE                                                          140
      ICOUNT=0                                                          141
      IF(YMAX .EQ. 0.0) YMAX=0.0                                        142
      IF(YMIN .EQ. 0.0) YMIN=0.0                                        143
      DY = (YMAX-YMIN)/5.0                                              144
      AXIS(1) = YMIN                                                    145
      DO 102 I = 2,6                                                    146
      AXIS(I)=AXIS(I-1) + DY                                            147
      IF(AXIS(I) .EQ. 0.0) AXIS(I)=0.0                                  148
  102 CONTINUE                                                          149
      IF(NBCD .GT. 0) WRITE(6,100) (BLANK,I=1,IHALF),(BCD(I),I=         150
     +1,NBCD)                                                           151
      IF(ISZ .NE. 1) GO TO 103                                          151
      WRITE(6,500)AXIS                                                  152
      WRITE(6,600)XNAME,ZI,(DASH,I=1, 9),ZI,(DASH,I=1, 9),              153
     *                ZI,(DASH,I=1, 9),ZI,(DASH,I=1, 9),              154
     *                ZI,(DASH,I=1, 9),ZI                             155
      GO TO 104                                                         156
  103 WRITE(6,200) AXIS                                                 157
      WRITE(6,300)XNAME,ZI,(DASH,I=1,19),ZI,(DASH,I=1,19),              158
     *                ZI,(DASH,I=1,19),ZI,(DASH,I=1,19),              159
     *                ZI,(DASH,I=1,19),ZI,YNAME                       160
  104 LINE=3                                                            161
      DELTA = YMAX - YMIN                                               162
      DO 999 M = 1,N                                                    163
      DO 201 K = 1,101                                                  164
      A(K) = BLANK                                                      165
      KK=0                                                              166
      DO 203 K=I1,I2                                                    167
      KK=KK+1                                                           168
      IF(KK .GT. NMARK) KK=1                                            169
      IF(Y(K,M) .EQ. 0.0) Y(K,M)=0.0                                    170
      IY=(((Y(K,M)-YMIN)/DELTA)*HUND)+1.5                               171
      IF (IY .LT. 1) IY=1                                               172
      IF(IY .GT. LAST) IY=LAST                                          173
      IF(I1 .NE. I2) GO TO 202                                          174
       A(IY) = STAR                                                     175
      IF(A(IY) .NE. BLANK) A(IY)=STAR                                   176
      IF(A(IY) .EQ. BLANK) A(IY)=MARK(KK)                               177
  202 CONTINUE                                                          178
      IF(A(LAST) .EQ. BLANK) A(LAST)=ZI                                 179
      IF(A(1   ) .EQ. BLANK) A(1   )=ZI                                 180
      IF(ISZ .NE. 1) GO TO 302                                          181
      WRITE(6,700)X(M),(A(L),L=1,LAST)                                  182
      GO TO 999                                                         183
  302 WRITE(6,800)X(M),A,Y(I1,M)                                        184
      LINE=LINE+1                                                       185
      LIN1=LINE+1                                                       186
      IF(ISZ .NE. 1) GO TO 1001                                         187
      WRITE(6,600)XNAME,ZI,(DASH,I=1, 9),ZI,(DASH,I=1, 9),              188
     *                ZI,(DASH,I=1, 9),ZI,(DASH,I=1, 9),              189
```

```
     *                    ZI,(DASH,I=1, 9),ZI                          190
      WRITE(6,500)AXIS                                                191
      GO TO 9999                                                      192
 1001 IF(LINE.LE.60)WRITE(6,300)XNAME,ZI,(DASH,I=1,19),ZI,(DASH       193
     +,I=1,19),                                                       193
     *                    ZI,(DASH,I=1,19),ZI,(DASH,I=1,19),         194
     *                    ZI,(DASH,I=1,19),ZI,YNAME                  195
      LINE=LINE+1                                                     196
      IF(LINE .LE. 60) WRITE(6,200)AXIS                               197
  100 FORMAT(1H1,21X6,A6)                                             198
  200 FORMAT(G22.3,5G20.3)                                            199
  300 FORMAT(4X36,4X101A1,5XA6)                                       200
  400 FORMAT(1X19,4X,101A1,2XG12.5)                                   201
  500 FORMAT(1PE21.2,5E10.2)                                          202
  600 FORMAT(4X46,4X51A1)                                             203
  700 FORMAT(1XG12.5,1X51A1)                                          204
 9999 RETURN                                                          205
      END                                                             206
C THIS IS PF DECODE3714 1+0C 29JAN76 ZAWALICK.                        207
```

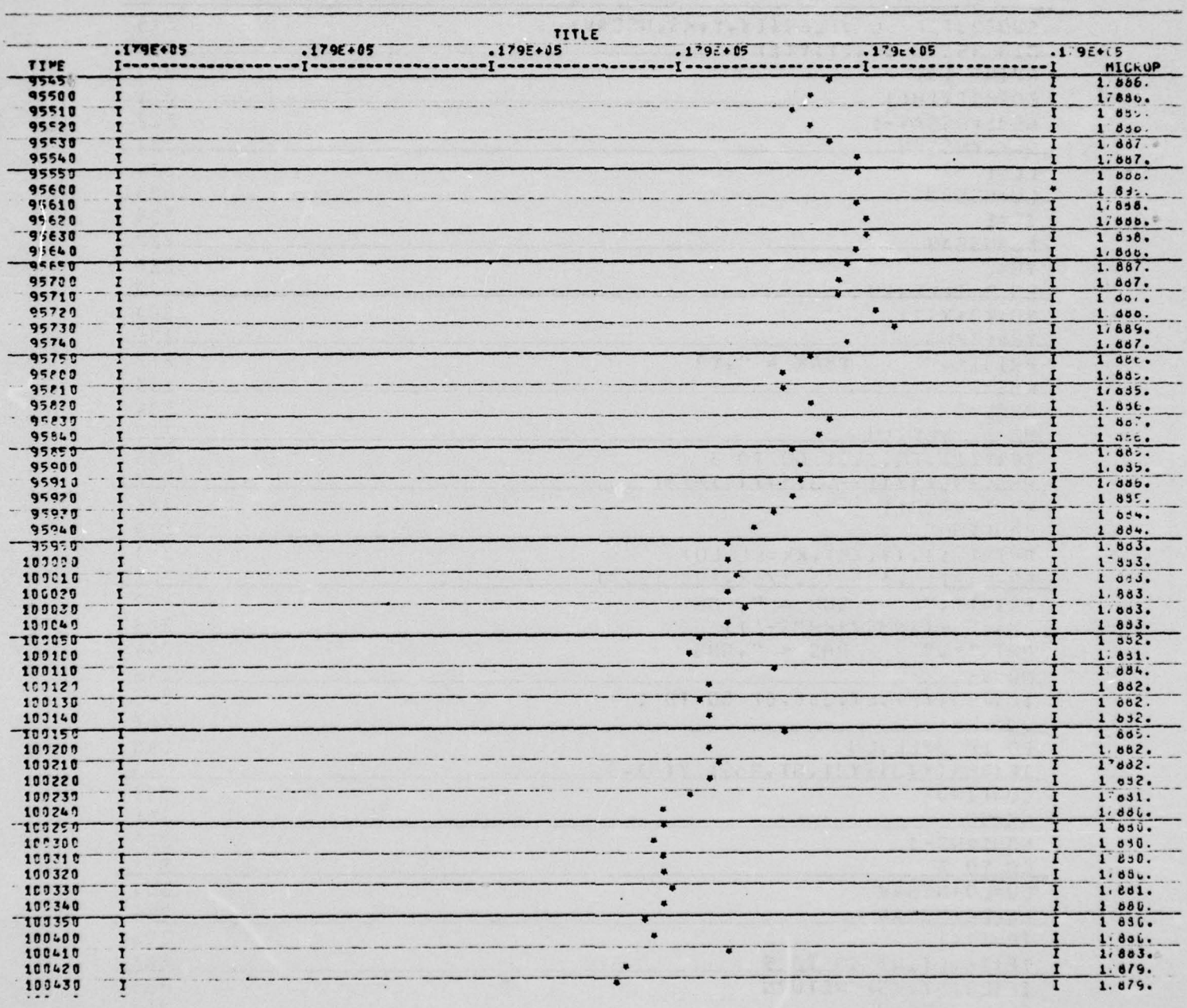
```
      SUBROUTINE DECODER(IB,K,IEOF)                                     203
      DIMENSION IBUFF(74),IA(240),IB(240),INX(240)                      204
      DATA (INX(I),I=1,120)/150,166,178,189,7,50,100,1,51,101,2         210
     +,52,102,3                                                         210
     1,53,103,4,54,104,151,161,171,190,5,55,105,6,56,106,7,57,1         211
     +07,8,58,                                                          211
     2108,9,59,109,152,162,172,191,10,60,110,11,61,111,12,62,11         212
     +2,13,63,                                                          212
     3113,14,64,114,153,163,173,192,15,65,115,16,66,116,17,67,1         213
     +17,18,68,                                                         213
     4118,19,69,119,154,164,174,193,20,70,120,21,71,121,22,72,1         214
     +22,23,                                                            214
     573,123,24,74,124,227,228,229,199,200,201,202,203,204,205,         215
     +206,207,                                                          215
     6208,209,210,211,155,165,175,194,25,75,125,26,76/                  216
      DATA (INX(I),I=121,240)/126,27,77,127,28,78,128,29,79,129         217
     +,156,166,                                                         217
     1175,195,30,80,130,31,81,131,32,82,132,33,83,133,34,84,134         218
     +,157,167,                                                         218
     2177,196,35,85,135,36,86,136,37,87,137,38,88,138,39,89,139         219
     +,158,168,                                                         219
     3178,197,40,90,140,41,91,141,42,92,142,43,93,143,44,94,144         220
     +,159,169,                                                         220
     4179,198,45,95,145,46,96,146,47,97,147,48,98,148,49,99,149         221
     +,160,                                                             221
     5193,185,181,184,187,182,185,188,212,213,214,215,216,217,2         222
     +18,219,                                                           222
     6220,221,222,223,224,225,226,230,231,232,233,234,235,236,2         223
     +37,238,23                                                         223
     79/                                                                224
      DATA K1/0/                                                        225
      IF(K1.EQ.1)K=0                                                    226
      K1=1                                                              227
      DO 3 K=1,600                                                      228
      K=K+1                                                             229
      READ(1) IBUFF                                                     230
      IF(EOF(1)) 5,10                                                   231
      BUFFER IN(1,1)(IBUFF(1),IBUFF(74))                                232
      IF(UNIT(1)) 10,30,20                                              233
      GO TO 4                                                           234
      IS=1                                                              235
      DO 1 I=1,72,3                                                     236
      IE=IS+9                                                           237
      DECODE(30,10,IBUFF(I)) (IA(J),J=IS,IE)                            238
      FORMAT(10A3)                                                      239
      IS=IS+10                                                          241
      DO 2 I=1,240                                                      241
      IERROR=IA(I).AND.1B                                               242
      IIB=SHIFT((IA(I).AND.1777408),56)                                 243
      IA(I)=IIB+IERROR                                                  244
      J=INX(I)                                                          245
```

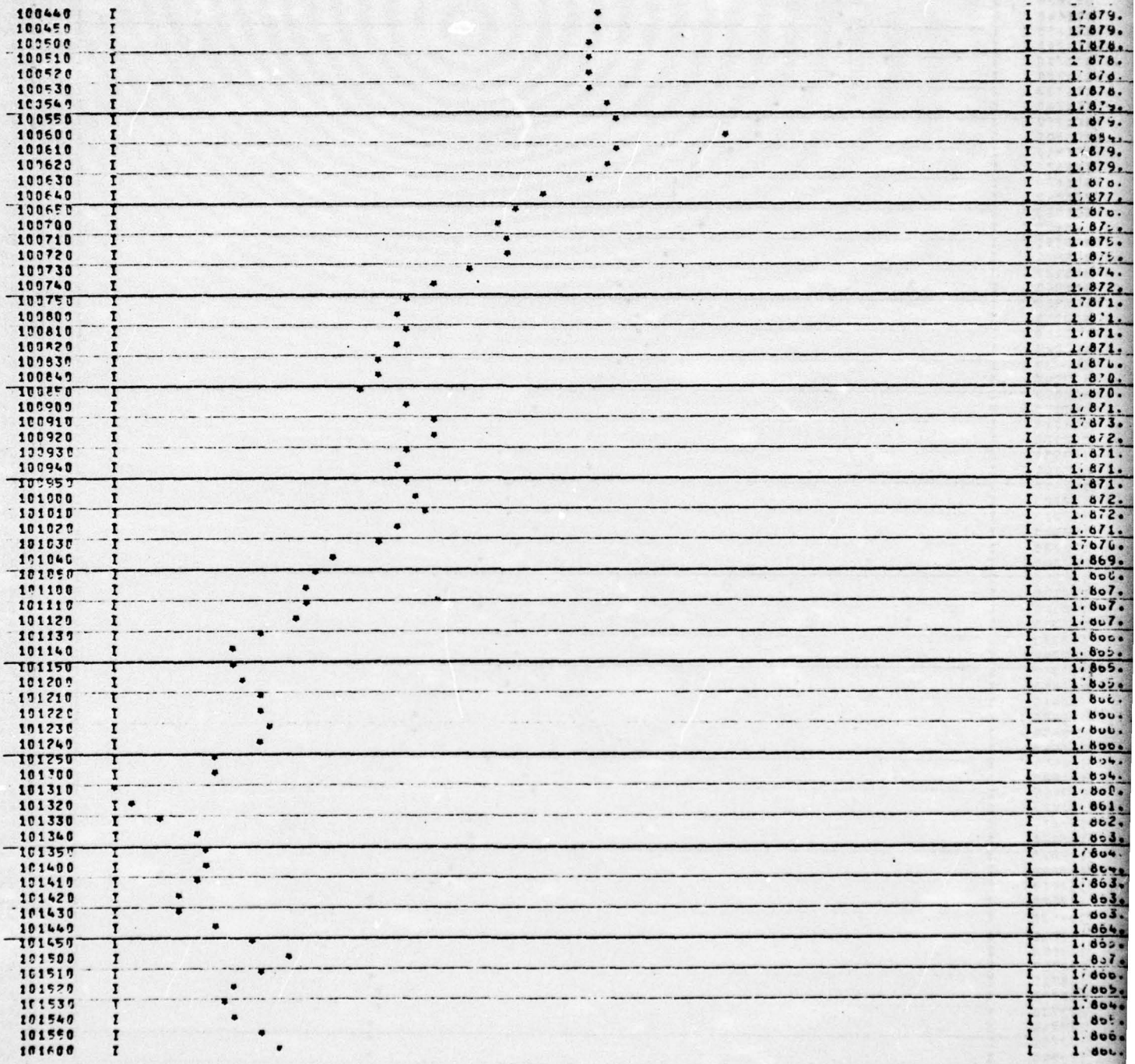
```
      IB(J+1)=IA(I)                                                     245
      PRINT*," STATION#,ERRORS?=",IA(231),IA(241)                       247
      IB(226)=IA(241)                                                   248
      IS10=10*SHIFT(IB(238),A.20,59)                                    249
      IS20=20*SHIFT(IB(238),A.40,58)                                    250
      IS40=40*SHIFT(IB(238),A.100,57)                                   251
      ISEC=IS40+IS20+IS10                                               252
      PRINT*,ISEC,IS40,IS20,IS10                                        253
      IB(225)=2*ISEC                                                    254
      M1=SHIFT(IB(238),A.200,56)                                        255
      M2=2*SHIFT(IB(238),A.400,55)                                      256
      M4=4*SHIFT(IB(238),A.1000,54)                                     257
      M8=8*SHIFT(IB(238),A.2000,53)                                     258
      M10=10*SHIFT(IB(238),A.4000,52)                                   259
      M20=20*SHIFT(IB(238),A.10000,51)                                  260
      M40=40*SHIFT(IB(238),A.20000,50)                                  261
      MIN=M40+M20+M10+M8+M4+M2+M1                                       262
       PRINT*," MIN=",MIN,M40,M20,M10,M8,M4,M2,M1                       263
      IB(224)=2*MIN                                                     264
      IH1=SHIFT(IB(239),A.20,59)                                        265
      IH2=2*SHIFT(IB(239),A.40,58)                                      266
      IH4=4*SHIFT(IB(239),A.100,57)                                     267
      IH8=8*SHIFT(IB(239),A.200,56)                                     268
      IH10=10*SHIFT(IB(239),A.400,55)                                   269
      IH20=20*SHIFT(IB(239),A.1000,54)                                  270
      IHR=IH20+IH10+IH8+IH4+IH2+IH1                                     271
      PRINT*," IHR=",IHR,IH20,IH10,IH8,IH4,IH2,IH1                      272
      IB(223)=2*IHR                                                     273
      ID1=SHIFT(IB(239),A.200,53)                                       274
      ID2=2*SHIFT(IB(239),A.400,52)                                     275
      ID4=4*SHIFT(IB(239),A.1000,51)                                    276
      ID8=8*SHIFT(IB(239),A.2000,50)                                    277
      ID10=10*SHIFT(IB(240),A.20,59)                                    278
      ID20=20*SHIFT(IB(240),A.40,58)                                    279
      ID40=40*SHIFT(IB(240),A.100,57)                                   280
      ID80=80*SHIFT(IB(240),A.200,56)                                   281
      ID100=100*SHIFT(IB(240),A.400,55)                                 282
      ID200=200*SHIFT(IB(240),A.1000,54)                                283
      IDY=ID200+ID100+ID80+ID40+ID20+ID10+ID8+ID4+ID2+ID1               284
      PRINT*," IDY=",IDY,ID200,ID100,ID80,ID40,ID20,ID10,ID8,ID
     +4,ID2,ID1                                                         285
      IB(222)=2*IDY                                                     286
      IY1=SHIFT(IB(240),A.200,53)                                       287
      IY2=2*SHIFT(IB(240),A.400,52)                                     288
      IY4=4*SHIFT(IB(240),A.1000,51)                                    289
      IY8=8*SHIFT(IB(240),A.2000,50)                                    291
      IYR=IY8+IY4+IY2+IY1                                               291
      PRINT*," IYR=",IYR,IY8,IY4,IY2,IY1                                292
      IB(221)=2*IYR                                                     293
      DO 31 I=1,240                                                     294
      IF(IB(I).GE.2048) IB(I)=IB(I)-4096                                295
```

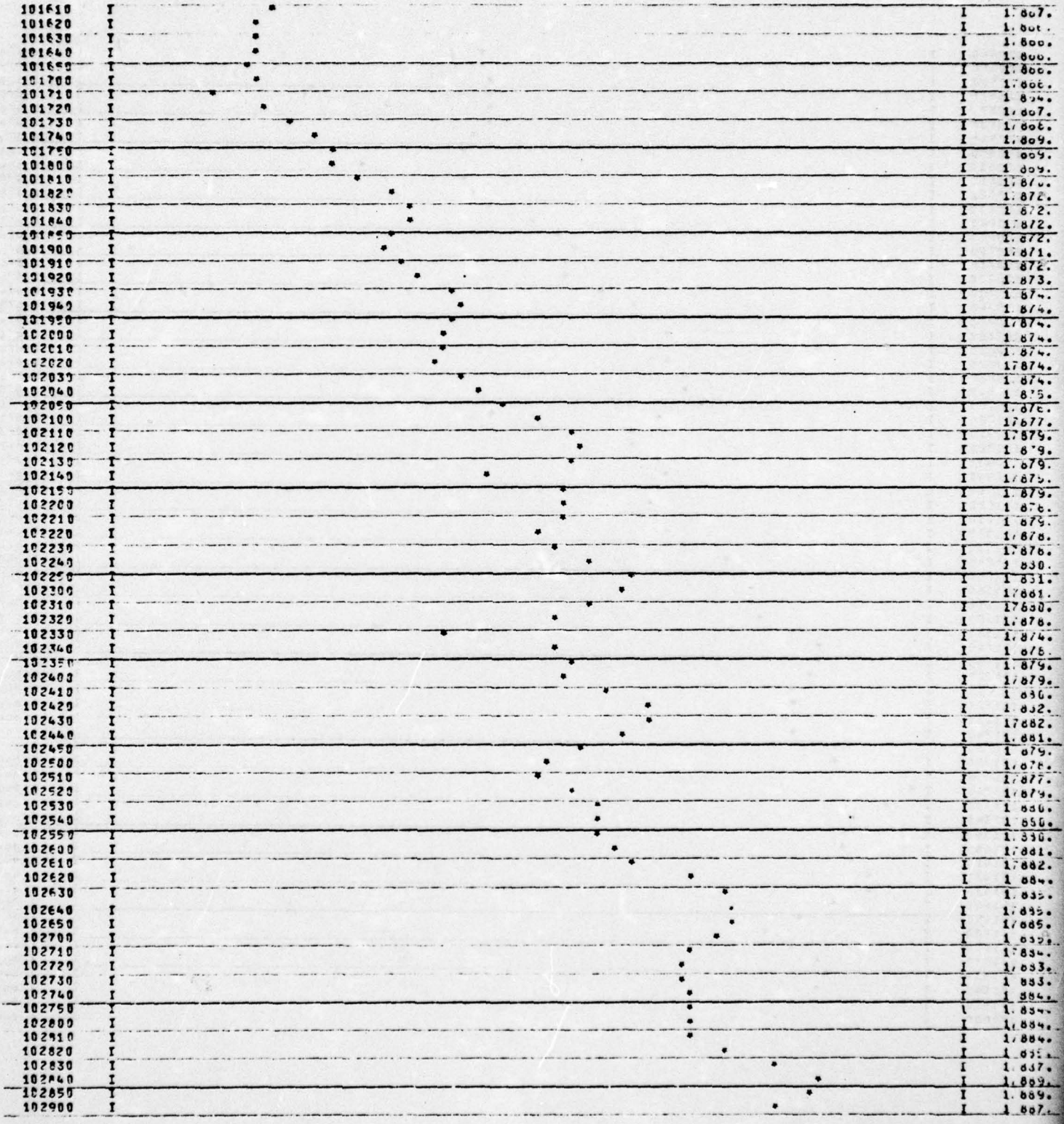
```
      CONTINUE                                                          296
      GO TO 5                                                           297
      PRINT*,"PARITY ERROR ON TAPE1. NREC=",K                           298
      WRITE(2) K,(ID(I),I=1,240)                                        299
      RETURN                                                            300
      PRINT*," NREC=",K                                                 301
      ISP=1                                                             302
      DO 5 L=1,5                                                        303
      DO 6 LI=1,10                                                      304
      IEP=ISP+4                                                         305
      IF(L.EQ.5) IEP=ISP+3                                              306
      PRINT200,(ID(IP),IP=ISP,IEP,10)                                   307
      ISP=ISP+1                                                         308
      ISP=ISP+40                                                        309
      PRINT 300                                                         310
      CONTINUE                                                          311
      CONTINUE                                                          312
      IEOF=1                                                            313
      RETURN                                                            314
      FORMAT(3025)                                                      315
      FORMAT(1X,1H ,///)                                                316
      END                                                               317
```

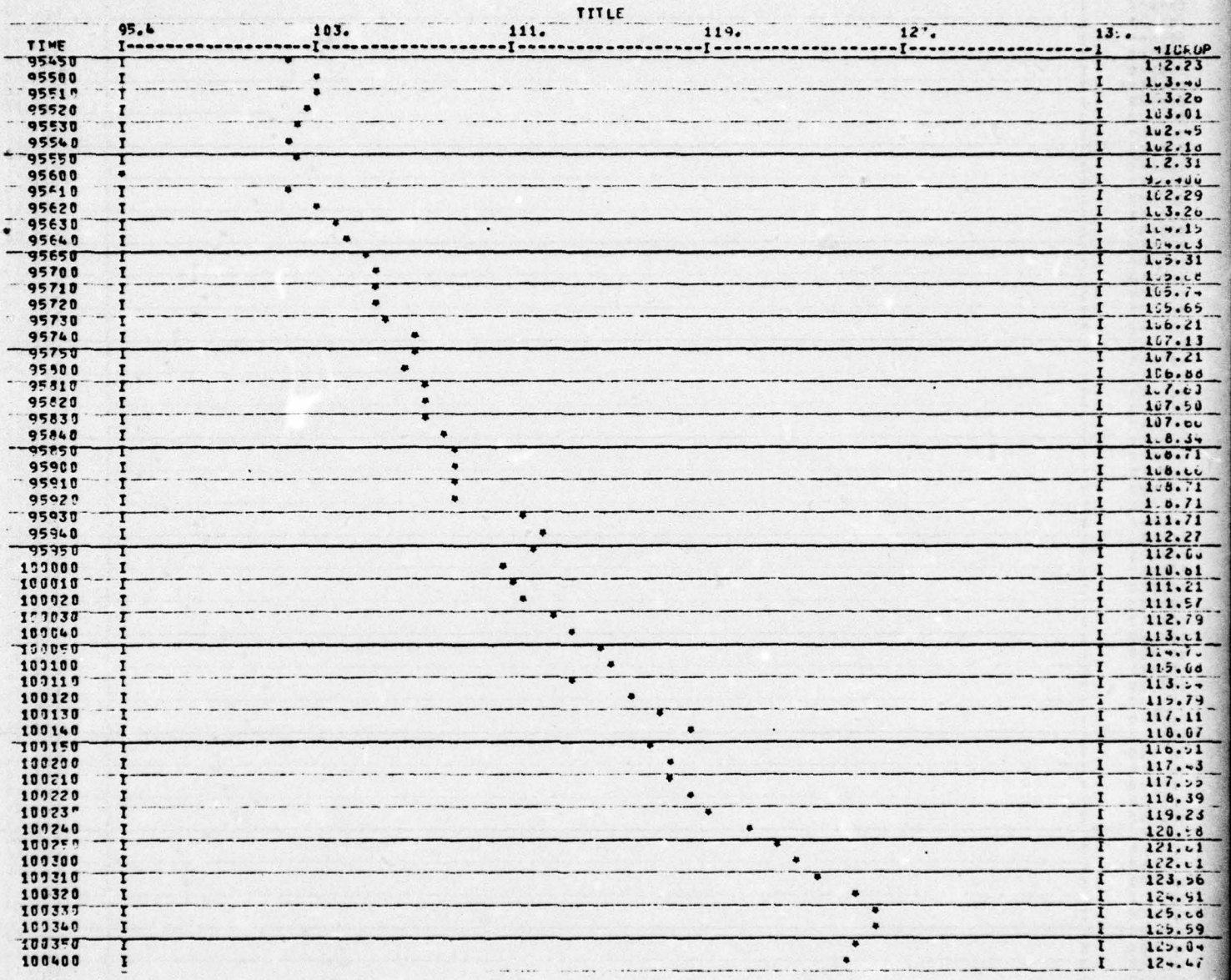
```
      SUBROUTINE LEVELER(IX,Y,KS,NSSAV)                                 318
      DIMENSION IX(1),Y(1)                                              319
      PRINT 100                                                         320
  100 FORMAT(1H1)                                                       321
      NS1=NSSAV-1                                                       322
      N=KS/NSSAV                                                        323
      LL=1                                                              324
      LU=NSSAV                                                          325
      IR=1                                                              326
      NS=NSSAV                                                          327
      YB=0                                                              328
      DO 2 I=LL,LU                                                      329
      YB=YB+Y(I)                                                        330
      YB=YB/NS                                                          331
      PRINT*,"      YBAR = ",YB                                         332
      RMS=0                                                             333
      KRMS=0                                                            334
      DO 3 I=LL,LU                                                      335
      IF(Y(I).EQ.0.0) GO TO 3                                           336
      RMS=RMS+(Y(I)-YB)*(Y(I)-YB)                                       337
      KRMS=KRMS+1                                                       338
    3 CONTINUE                                                          339
      PRINT 10,(Y(KK),KK=LL,LU)                                         340
   10 FORMAT(1X,10F12.3/1X,10F12.3)                                     341
      PRINT*,"      SUM = ",RMS                                         342
      RMS=SQRT(RMS/(KRMS-1))                                            343
      PRINT*,"      RMS = ",RMS                                         344
      NC=NS                                                             345
      IF(ABS(YB).LT.100.0) GO TO 1                                      346
      JJ=LL-1                                                           347
      DO 12 J=LL,LU                                                     348
      IF(ABS(Y(J)-YB).GT.R1S) Y(J)=0                                    349
   12 CONTINUE                                                          350
      NS=NC                                                             351
      NSM1=NS-1                                                         352
      GO TO 7                                                           353
    1 LU=LU+NSSAV                                                       354
      LL=LL+NSSAV                                                       355
      IR=IR+1                                                           356
      IF(IR.LT.N) GO TO 5                                               357
      IF(LU.GT.KS) RETURN                                               358
      LL=KS-NSSAV                                                       359
      LU=KS                                                             360
      PRINT*,"      GOING TO 5"                                         361
      GO TO 5                                                           362
      END                                                               363
```

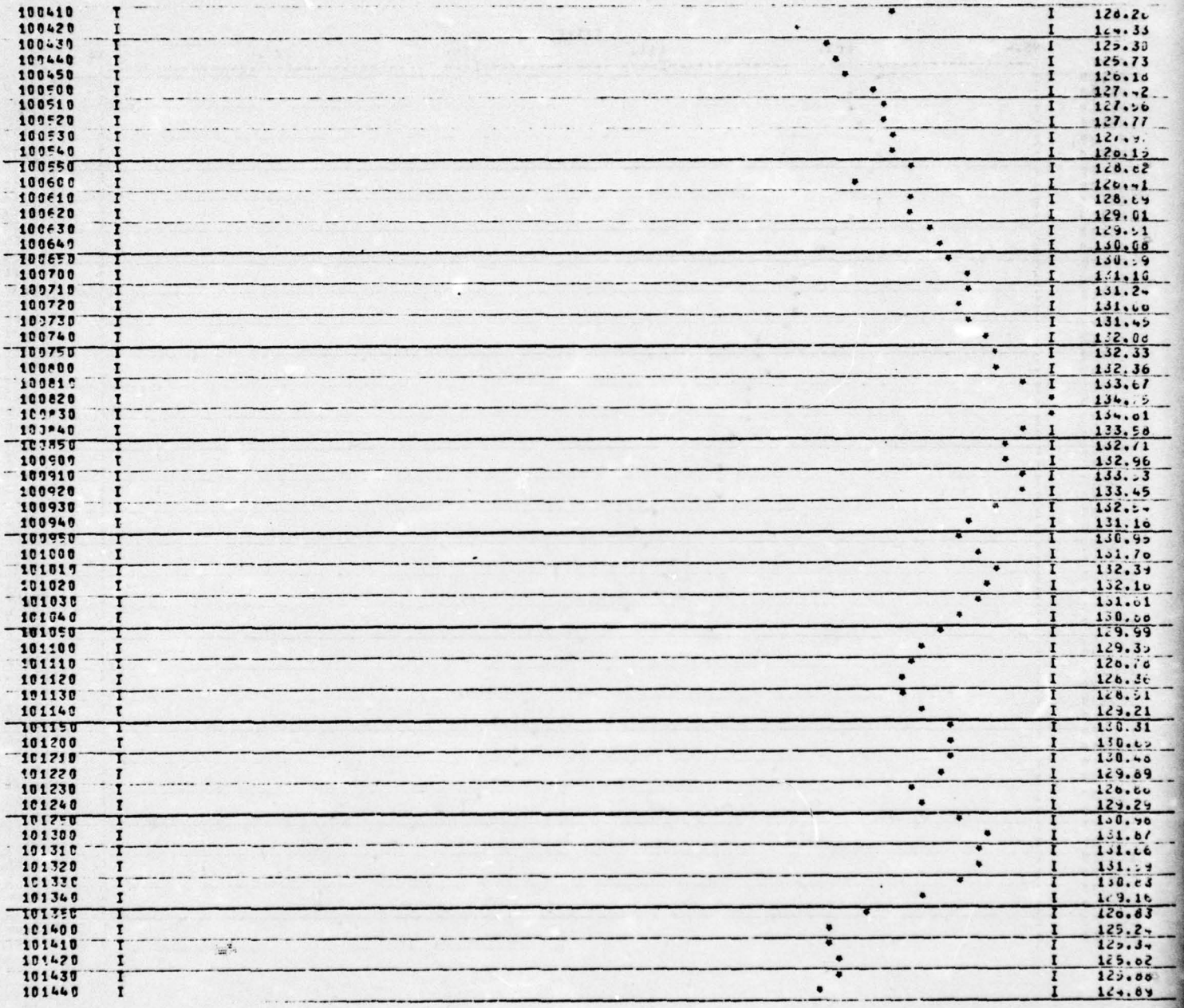
TITLE

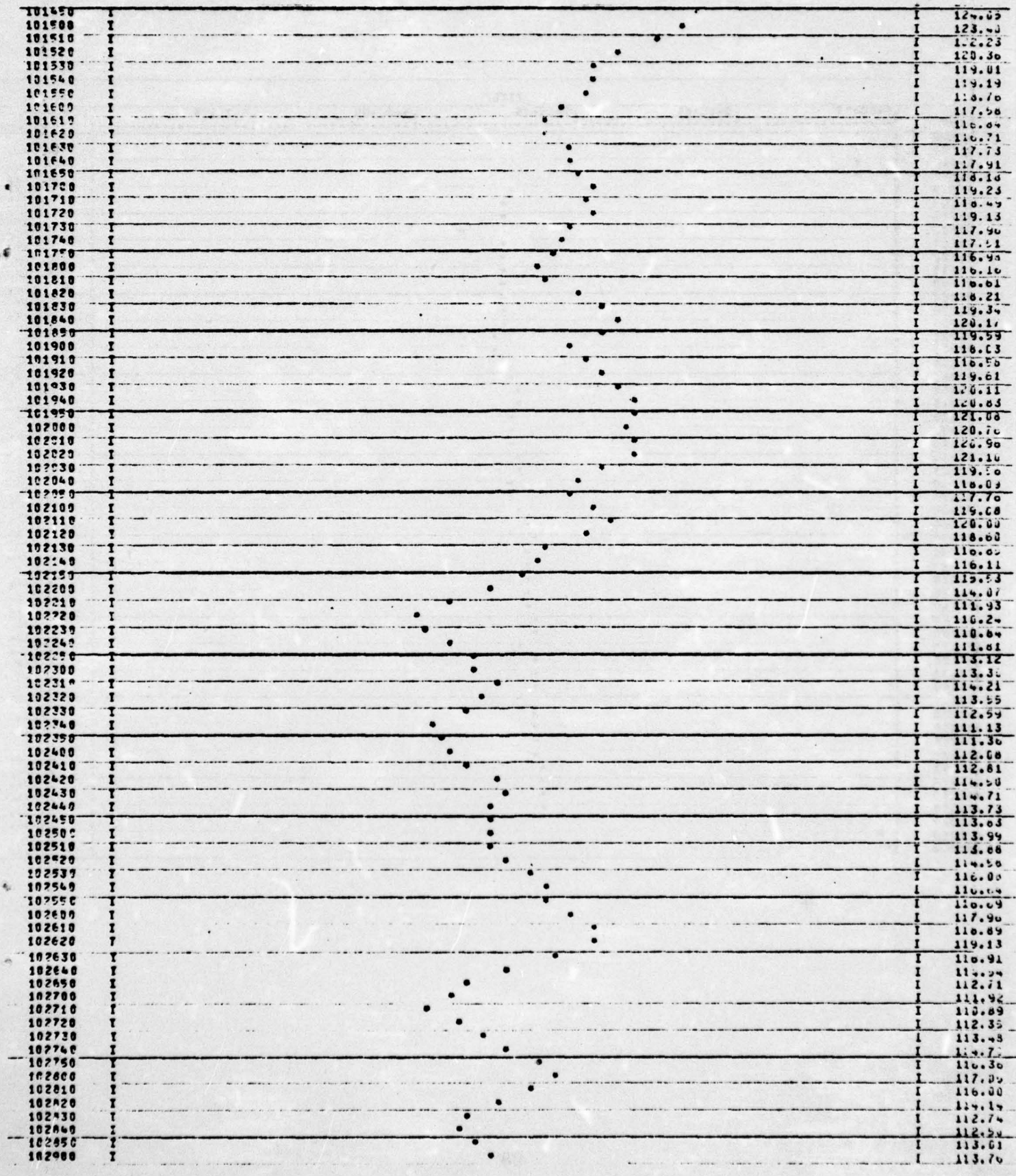

.179E+05
.179E+05
.179E+05
.179E+05
.179E+05
.179E+05
TIME
MICKUP
95450 1.886.
95500 1.886.
95510 1.885.
95520 1.886.
95530 1.887.
95540 1.887.
95550 1.886.
95600 1.885.
95610 1.886.
95620 1.886.
95630 1.886.
95640 1.886.
95650 1.887.
95700 1.887.
95710 1.887.
95720 1.886.
95730 1.889.
95740 1.887.
95750 1.886.
95800 1.885.
95810 1.885.
95820 1.886.
95830 1.887.
95840 1.886.
95850 1.885.
95900 1.885.
95910 1.886.
95920 1.885.
95930 1.884.
95940 1.884.
95950 1.883.
100000 1.883.
100010 1.883.
100020 1.883.
100030 1.883.
100040 1.883.
100050 1.882.
100100 1.881.
100110 1.884.
100120 1.882.
100130 1.882.
100140 1.882.
100150 1.885.
100200 1.882.
100210 1.882.
100220 1.882.
100230 1.881.
100240 1.886.
100250 1.880.
100300 1.880.
100310 1.880.
100320 1.886.
100330 1.881.
100340 1.880.
100350 1.880.
100400 1.886.
100410 1.883.
100420 1.879.
100430 1.879.

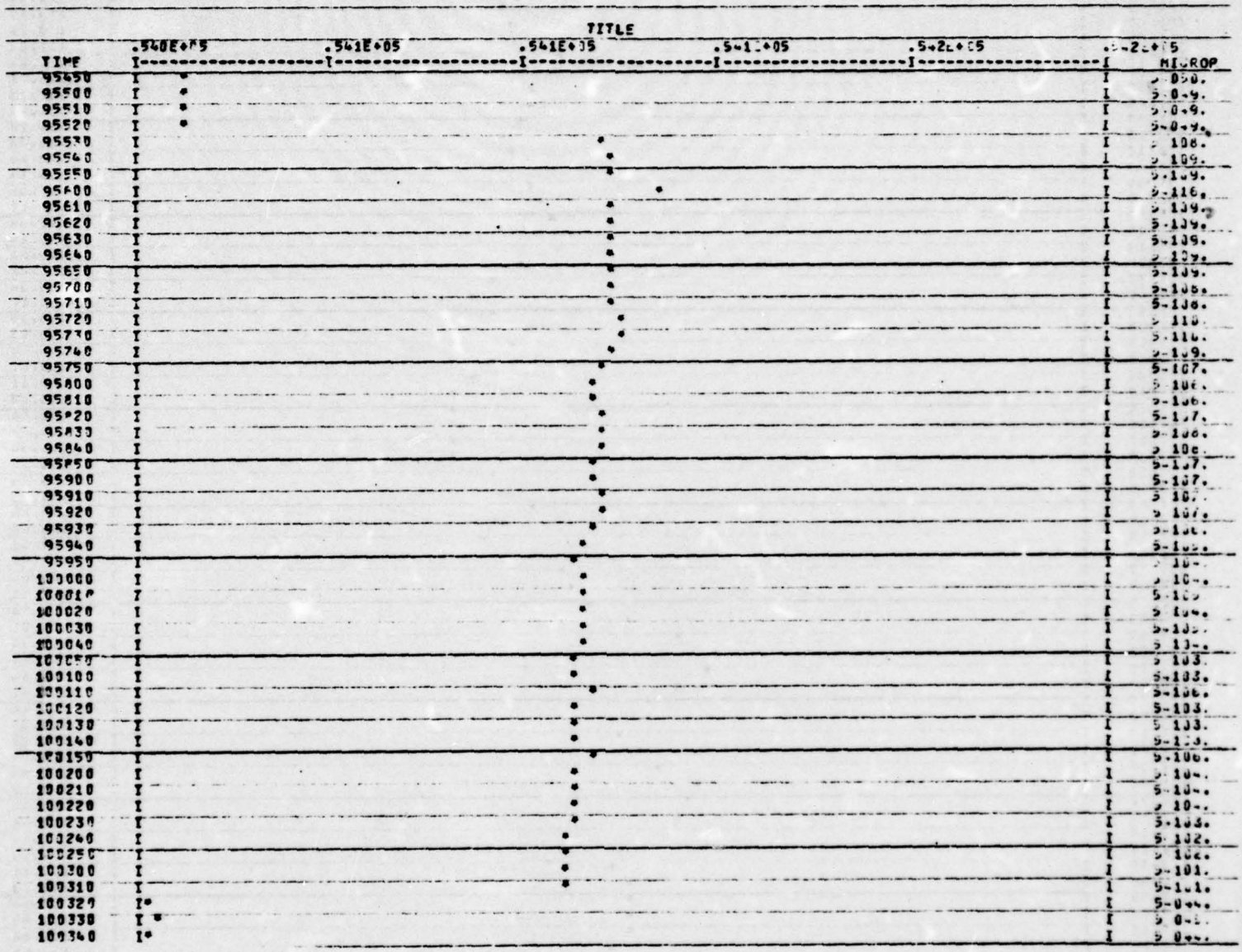

100440 I 1.879.
100450 I 1.879.
100500 I 1.878.
100510 I 1.878.
100520 I 1.878.
100530 I 1.878.
100540 I 1.879.
100550 I 1.879.
100600 I 1.879.
100610 I 1.879.
100620 I 1.879.
100630 I 1.880.
100640 I 1.877.
100650 I 1.880.
100700 I 1.875.
100710 I 1.875.
100720 I 1.875.
100730 I 1.874.
100740 I 1.872.
100750 I 1.871.
100800 I 1.871.
100810 I 1.871.
100820 I 1.871.
100830 I 1.870.
100840 I 1.870.
100850 I 1.870.
100900 I 1.871.
100910 I 1.873.
100920 I 1.872.
100930 I 1.871.
100940 I 1.871.
100950 I 1.871.
101000 I 1.872.
101010 I 1.872.
101020 I 1.871.
101030 I 1.870.
101040 I 1.869.
101050 I 1.868.
101100 I 1.867.
101110 I 1.867.
101120 I 1.867.
101130 I 1.866.
101140 I 1.865.
101150 I 1.865.
101200 I 1.865.
101210 I 1.866.
101220 I 1.866.
101230 I 1.866.
101240 I 1.866.
101250 I 1.864.
101300 I 1.864.
101310 I 1.860.
101320 I 1.861.
101330 I 1.862.
101340 I 1.863.
101350 I 1.864.
101400 I 1.864.
101410 I 1.863.
101420 I 1.863.
101430 I 1.863.
101440 I 1.864.
101450 I 1.865.
101500 I 1.867.
101510 I 1.866.
101520 I 1.865.
101530 I 1.864.
101540 I 1.865.
101550 I 1.866.
101600 I 1.866.

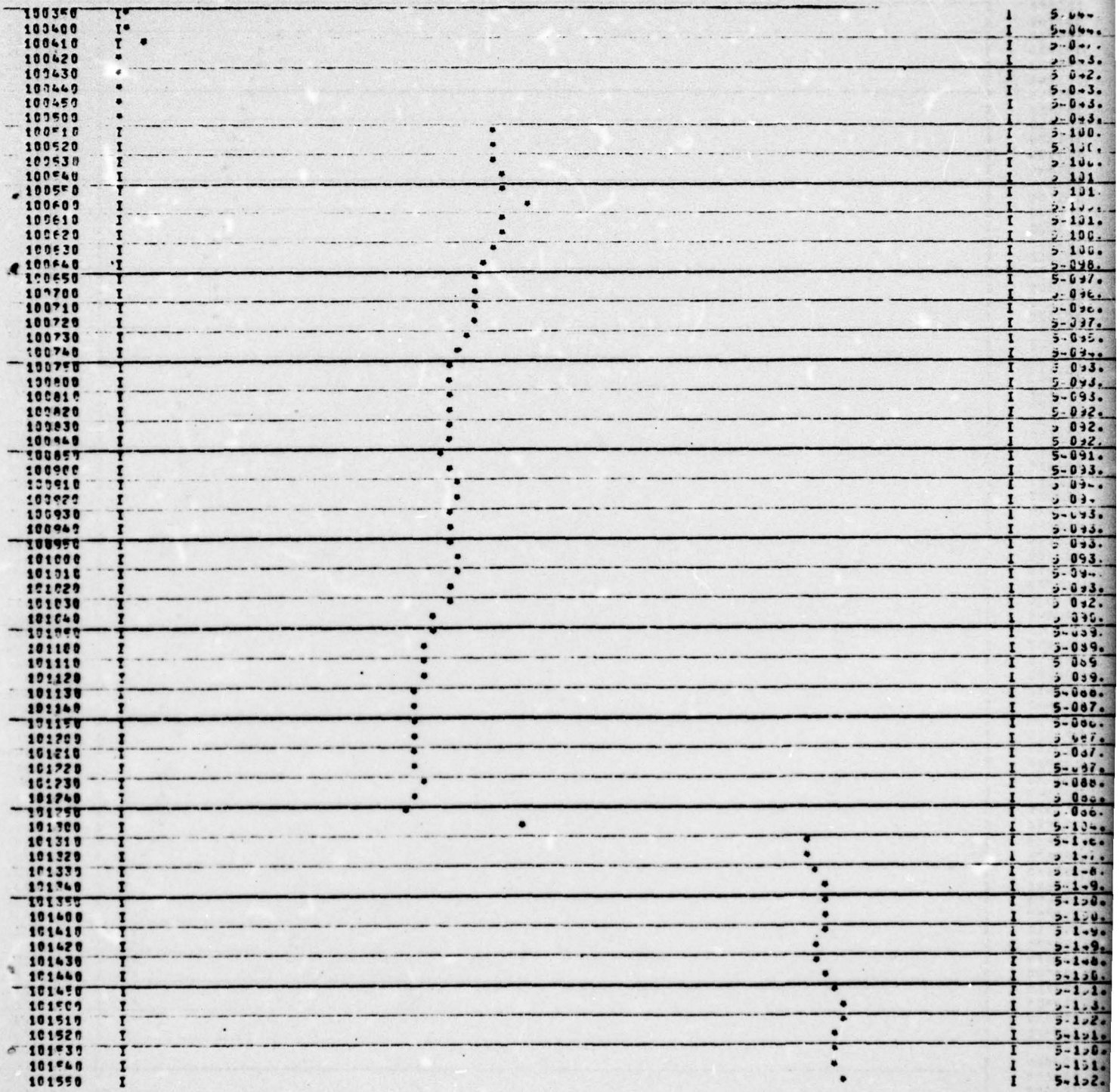

101610 I 1.867.
101620 I 1.867.
101630 I 1.866.
101640 I 1.866.
101650 I 1.866.
101700 I 1.866.
101710 I 1.866.
101720 I 1.867.
101730 I 1.868.
101740 I 1.869.
101750 I 1.869.
101800 I 1.869.
101810 I 1.870.
101820 I 1.872.
101830 I 1.872.
101840 I 1.872.
101850 I 1.872.
101900 I 1.871.
101910 I 1.872.
101920 I 1.873.
101930 I 1.874.
101940 I 1.874.
101950 I 1.874.
102000 I 1.874.
102010 I 1.874.
102020 I 1.874.
102030 I 1.874.
102040 I 1.875.
102050 I 1.876.
102100 I 1.877.
102110 I 1.879.
102120 I 1.879.
102130 I 1.879.
102140 I 1.879.
102150 I 1.879.
102200 I 1.876.
102210 I 1.879.
102220 I 1.878.
102230 I 1.876.
102240 I 1.880.
102250 I 1.881.
102300 I 1.881.
102310 I 1.880.
102320 I 1.878.
102330 I 1.874.
102340 I 1.878.
102350 I 1.879.
102400 I 1.879.
102410 I 1.880.
102420 I 1.882.
102430 I 1.882.
102440 I 1.881.
102450 I 1.879.
102500 I 1.878.
102510 I 1.877.
102520 I 1.879.
102530 I 1.880.
102540 I 1.880.
102550 I 1.880.
102600 I 1.881.
102610 I 1.882.
102620 I 1.884.
102630 I 1.885.
102640 I 1.885.
102650 I 1.885.
102700 I 1.885.
102710 I 1.884.
102720 I 1.883.
102730 I 1.883.
102740 I 1.884.
102750 I 1.884.
102800 I 1.884.
102810 I 1.884.
102820 I 1.887.
102830 I 1.887.
102840 I 1.889.
102850 I 1.889.
102900 I 1.887.

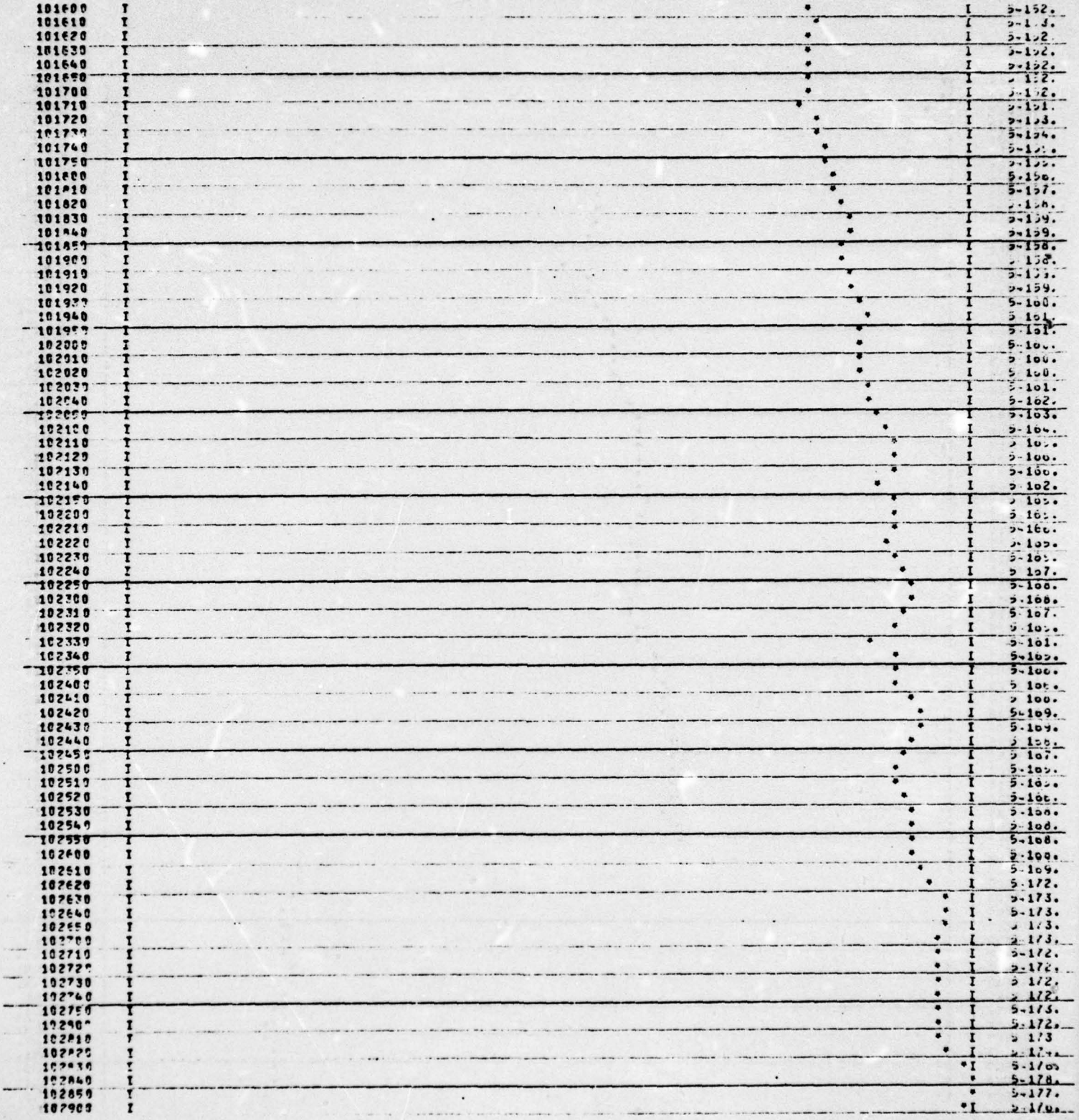

TITLE
95.4
103.
111.
119.
127.
13..
TIME
MICROP
95450 112.23
95500
95510
95520 103.01
95530 102.45
95540 102.10
95550 102.31
95600
95610 102.29
95620
95630
95640
95650 105.31
95700
95710
95720 105.65
95730 106.21
95740 107.13
95750 107.21
95800 106.86
95810 107.03
95820 107.50
95830 107.00
95840
95850 108.71
95900
95910 108.71
95920 108.71
95930 111.71
95940 112.27
95950 112.00
100000 110.81
100010 111.21
100020 111.57
100030 112.79
100040 113.21
100050
100100 115.06
100110
100120 115.79
100130 117.11
100140 118.07
100150
100200 117.43
100210
100220 118.39
100230
100240 120.18
100250 121.21
100300 122.21
100310 123.96
100320 124.91
100330
100340 125.59
100350 125.04
100400 124.47


100410
100420
100430
100440
100450
100500
100510
100520
100530
100540
100550
100600
100610
100620
100630
100640
100650
100700
100710
100720
100730
100740
100750
100800
100810
100820
100830
100840
100850
100900
100910
100920
100930
100940
100950
101000
101010
101020
101030
101040
101050
101100
101110
101120
101130
101140
101150
101200
101210
101220
101230
101240
101250
101300
101310
101320
101330
101340
101350
101400
101410
101420
101430
101440




TITLE
.540E+05
.541E+05
.541E+05
.542E+05
TIME
MICROP





```
101600    I                                                                  I    5-152.
101610    I                                                                  I    5-153.
101620    I                                                                  I    5-152.
101630    I                                                                  I    5-152.
101640    I                                                                  I    5-152.
101650    I                                                                  I    5-152.
101700    I                                                                  I    5-152.
101710    I                                                                  I    5-151.
101720    I                                                                  I    5-153.
101730    I                                                                  I    5-154.
101740    I                                                                  I    5-155.
101750    I                                                                  I    5-155.
101800    I                                                                  I    5-156.
101810    I                                                                  I    5-157.
101820    I                                                                  I    5-158.
101830    I                                                                  I    5-159.
101840    I                                                                  I    5-159.
101850    I                                                                  I    5-158.
101900    I                                                                  I    5-158.
101910    I                                                                  I    5-159.
101920    I                                                                  I    5-159.
101930    I                                                                  I    5-160.
101940    I                                                                  I    5-161.
101950    I                                                                  I    5-161.
102000    I                                                                  I    5-160.
102010    I                                                                  I    5-160.
102020    I                                                                  I    5-160.
102030    I                                                                  I    5-161.
102040    I                                                                  I    5-162.
102050    I                                                                  I    5-163.
102100    I                                                                  I    5-164.
102110    I                                                                  I    5-165.
102120    I                                                                  I    5-166.
102130    I                                                                  I    5-166.
102140    I                                                                  I    5-162.
102150    I                                                                  I    5-165.
102200    I                                                                  I    5-165.
102210    I                                                                  I    5-166.
102220    I                                                                  I    5-165.
102230    I                                                                  I    5-165.
102240    I                                                                  I    5-167.
102250    I                                                                  I    5-168.
102300    I                                                                  I    5-168.
102310    I                                                                  I    5-167.
102320    I                                                                  I    5-165.
102330    I                                                                  I    5-161.
102340    I                                                                  I    5-165.
102350    I                                                                  I    5-166.
102400    I                                                                  I    5-164.
102410    I                                                                  I    5-166.
102420    I                                                                  I    5-169.
102430    I                                                                  I    5-169.
102440    I                                                                  I    5-166.
102450    I                                                                  I    5-167.
102500    I                                                                  I    5-165.
102510    I                                                                  I    5-165.
102520    I                                                                  I    5-164.
102530    I                                                                  I    5-166.
102540    I                                                                  I    5-168.
102550    I                                                                  I    5-168.
102600    I                                                                  I    5-169.
102610    I                                                                  I    5-169.
102620    I                                                                  I    5-172.
102630    I                                                                  I    5-173.
102640    I                                                                  I    5-173.
102650    I                                                                  I    5-173.
102700    I                                                                  I    5-173.
102710    I                                                                  I    5-172.
102720    I                                                                  I    5-172.
102730    I                                                                  I    5-172.
102740    I                                                                  I    5-172.
102750    I                                                                  I    5-173.
102800    I                                                                  I    5-172.
102810    I                                                                  I    5-173
102820    I                                                                  I    5-174.
102830    I                                                                  I    5-175
102840    I                                                                  I    5-178.
102850    I                                                                  I    5-177.
102900    I                                                                  I    5-176.

**********        B9B9IA9  //// END OF LIST ////
**********        B9B9IB9  //// END OF LIST ////
```

## Program 3

```
      PROGRAM READTP(INPUT,OUTPUT,TAPE1,TAPE2)                          10
CC PROGRAM READS TAPE2 WRITES TAPE1 CCCCCC                              11
      DIMENSION WT(79),INX(2700),OUTX(500)                              12
      COMMON WT(157),IREAD,NX(2648)                                     13
      REAL INX                                                          14
      DATA KEOF,LINEC,NPPR,LASTREC/2*0,2048,1/                          15
      DATA LINX,LINY,LINZ,LOUTX,LOUTY,LOUTZ,KPX,KPY,KPZ,NTO/10*         16
     +0/                                                                16
      DATA LB,LFIL,IDEC,XMAX,XMIN,YMAX,YMIN,ZMAX,ZMIN/450,49,1,         17
     + 0.,100.,-*3./                                                    18
      FORMAT(1H1)                                                       19
      IREAD=0                                                           20
      IP=1                                                              21
      NVAR=1                                                            22
      NPTS(IP)=0                                                        23
      LWORKA=NPPR+LFIL-1                                                24
      REWIND 1                                                          25
      REWIND 2                                                          26
      CALL READA(A,WT,LFIL,IP,1.)                                       27
      I=0                                                               28
      I=I+1                                                             29
      READ(2) INX(I)                                                    30
      IF(EOF(2)) 2,1                                                    31
      PRINT*,"      EOF(2)"                                             32
      NPPR=I-1                                                          33
      NPSAV=NPPR                                                        34
      PRINT*,"      NPPR,NPSAV=",NPPR,NPSAV                             35
      PRINT 5,(INX(I),I=1,NPPR)                                         36
      IREAD=IREAD+1                                                     37
      CALL FILTER(INX,NPPR,OUTX,LOUTX,XMAX,XMIN,KPX,LFIL,IDEC,L         38
     +ASTREC)                                                           38
      CALL FILTER(INX,NPPR,OUTX,LOUTX,LFIL,IDEC)                        39
      PRINT 100                                                         40
      PRINT*,"      LOUTX=",LOUTX                                       41
      PRINT 5,(INX(I),I=1,NPSAV)                                        42
      PRINT 111                                                         43
      PRINT 5,(OUTX(I),I=1,NPSAV)                                       44
      FORMAT(10F12.3)                                                   45
      CALL SKRATCH(OUTX,LOUTX)                                          46
      FORMAT(//)                                                        47
      IF(IREAD.EQ.LASTREC) STOP 107                                     48
      NPPR=2048                                                         49
      GO TO 44                                                          50
      PRINT 1300                                                        51
      FORMAT(*  EOF ENCOUNTERED ON INPUT TAPE*)                         52
      STOP 1010                                                         53
      END                                                               54
```

```
      SUBROUTINE MAVE(IX,NPTS,IXMAIX,IMAIX,IXMIN,IMIN,IXAVE,IR          55
     +)                                                                 55
      DIMENSION IX(1)                                                   56
      IXMAIX=-99999999                                                  57
      IXMIN=99999999                                                    58
      IXSUM=0.                                                          59
      DO 50 I=1,NPTS                                                    60
      IXSUM=IXSUM+IX(I)                                                 61
      IF(IXMAIX-IX(I)) 46,47,47                                         62
   46 IXMAIX=IX(I)                                                      63
      IMAIX=I                                                           64
   47 IF(IXMIN-IX(I)) 50,50,48                                          65
   48 IXMIN=IX(I)                                                       66
      IMIN=I                                                            67
   50 CONTINUE                                                          68
      IXAVE=IXSUM/NPTS                                                  69
      IF(IR.EQ.0) GO TO 70                                              70
      DO 60 I=1,NPTS                                                    71
      IX(I)=IX(I)-IXAVE                                                 72
      RETURN                                                            73
      END                                                               74
```

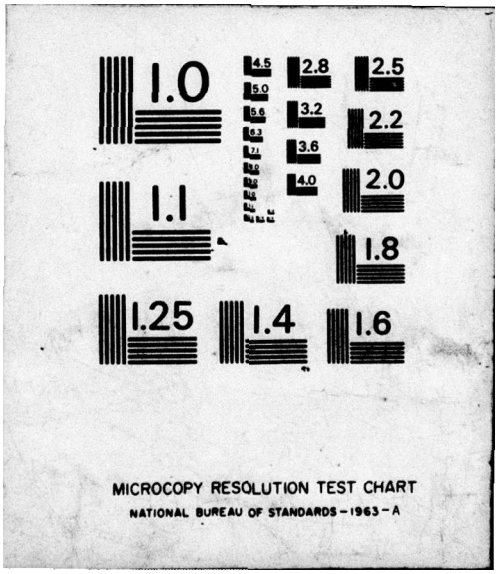
```
      SUBROUTINE PATCH(Y,N)                                              75
C     THIS PROGRAM PLOTS N SEQUENTIAL Y'S IN A SERIES OF CHARTS W        76
     +ITH THE                                                            76
C     SAME VERTICAL AXIS.   THIS IS CYCLE 2 OF WIGGLE828                 77
      DIMENSION HL(7),Y(N),IY(1000),HH(132),KIL(132)                     78
      DATA HL/1H ,1H6,1H6,1H4,1H2,1H0,1H./                               79
      BIGY=-1.E99;SMALLY=1.E99                                           80
      DO 3 I=1,N                                                         81
      IF(Y(I).GT.BIGY)GO TO 12                                           82
      GO TO 13                                                           83
   12 BIGY=Y(I)                                                          84
      MAXYI=I                                                            85
   13 IF(Y(I).LT.SMALLY)GO TO 14                                         86
      GO TO 3                                                            87
   14 SMALLY=Y(I)                                                        88
      MINYI=I                                                            89
    3 CONTINUE                                                           90
      DIFF=BIGY-SMALLY;QUANT=BIGY-56.*SMALLY                             91
      DO 4 I=1,N                                                         92
    4 IY(I)=(QUANT+55.*Y(I))/DIFF                                        93
      IY(MINYI)=1;IY(MAXYI)=56                                           94
      IMK=QUANT/DIFF                                                     95
      IK=N/130+MINO(MOD(N,130),1)                                        96
      DO 20 JJ=1,IK                                                      97
      DO 5 J=1,132                                                       98
    5 KIL(J)=0                                                           99
      HH(J)=1H                                                          100
      IST=(JJ-1)*130+1                                                  101
      PRINT 1000,IST                                                    102
 1000 FORMAT(*1ENUMERATION BEGINS WITH*,I4,*ST ITEM*)                   103
      DO 9 I=1,56                                                       104
      IF(MOD(I,11).NE.1)HH(1)=HL(7)                                     105
      IF(MOD(I,11).EQ.1)GO TO 10                                        106
      GO TO 11                                                          107
   10 KM=I/11+1;HH(1)=HL(KM)                                            108
   11 NN=MIN0(130,N+1-IST)                                              109
      DO 7 K=1,NN                                                       110
      KL=K+1;IF(KIL(KL).EQ.1)GO TO 8                                    111
      IF(IY(K+IST-1).NE.57-I)GO TO 7                                    112
      KIL(KL)=1                                                         113
      IF(57-I.GT.IMK)HH(KL)=1H+                                         114
      IF(57-I.LE.IMK)HH(KL)=1H*                                         115
    8 CONTINUE                                                          116
      PRINT 1001,(HH(L),L=1,132)                                        117
 1001 FORMAT(2X,132A1)                                                  118
    9 CONTINUE                                                          119
 1002 FORMAT(*  ....;....1....;...2...;...3...;....4....;.              120
     +...5....;                                                         120
     1....6....;....7....;....8....;....9....;....0....;....1..        121
     +..;....2.                                                         121
     2...;....3+)                                                       122
      PRINT 1002                                                        123
   20 CONTINUE                                                          124
      PRINT 1003,MAXYI,BIGY,MINYI,SMALLY                                125
 1003 FORMAT(* MAXIMUM VALUE IS *,I3,*TH=*,E21.12,* MINIMUM VAL         126
     +UE IS *,I                                                         126
     3,*TH=*,E21.12)                                                    127
      RETURN                                                            128
      END                                                               129
```

```
      SUBROUTINE IFLTER(IN,LIN,OUT,LOUT,XMAX,XMIN,KR,LF,IDEC,LA      130
     +STREC)                                                         131
      DIMENSION IN(1),OUT(500)                                       132
      COMMON WT(1,2),IREAD                                           133
      REAL IN                                                        134
      DATA NLINE,NENTS/2*0/                                          135
      NENTS=NENTS+1                                                  136
      MIDFIL=LF/2+1                                                  137
      IB=1                                                           138
      IE=LF                                                          139
      JE=(LIN-LF)/IDEC+1                                             140
      FTO=0.                                                         141
      DO 55 J=1,JE                                                   142
      JS=J                                                           143
      DO 44 I=IB,IE                                                  144
      IW=I-IDEC*(J-1)                                                145
      IF(IW.EQ.MIDFIL) NOFILPT=I                                     146
      IF(I.EQ.IE) GO TO 43                                           147
      GO TO 42                                                       148
   43 IF(IW.NE.LF) STOP 10                                           149
   42 CONTINUE                                                       150
      FTO=WT(IW)*(IN(I))+FTO                                         151
      LOUT=LOUT+1                                                    152
      IF(LOUT.GT.500) STOP 11                                        153
      OUT(LOUT)=FTO                                                  154
      FTO=0.                                                         155
      IF(XMAX.LT.OUT(LOUT)) XMAX=OUT(LOUT)                           156
      IF(XMIN.GT.OUT(LOUT)) XMIN=OUT(LOUT)                           157
      IF(LOUT.EQ.500) GO TO 66                                       158
      IF(J.EQ.JE) GO TO 66                                           159
   44 GO TO 54                                                       160
   66 KR=KR+1                                                        161
      WRITE(1) LOUT,(OUT(K),K=1,LOUT)                                162
      PRINT 77,IREAD,KR,LOUT,XMAX,XMIN,(OUT(L),L=1,10)               163
      IF(MOD(KR,6).EQ.2) CALL SKRATCH(OUT,500)                       164
      PRINT *,IN(IE-2),IN(IE-1),IN(IE)                               165
   77 FORMAT(1X,3I10,2F12.4/(10F12.4/))                              166
      XMIN=999999999999.                                             167
      XMAX=-999999999999.                                            168
      LLINEFD=NLINE+J*IDEC                                           169
      PRINT*," LINE COUNT AT END OF FILTERED DATA RECORD=",LLIN      170
     +EFD                                                            171
   54 IB=IB+IDEC                                                     172
      IE=IE+IDEC                                                     173
      NUSED=JE*IDEC                                                  174
      NLINE=NLINE+NUSED                                              175
      LIN=LIN-NUSED                                                  176
      IF(LIN.GT.(LF-1)) STOP 12                                      177
      IF(MOD(IREAD,3).EQ.0.AND.JS.EQ.JE) GO TO 88                    178
      DO 87 I=1,LIN                                                  179
   87 IN(I)=IN(I+NUSED)                                              180
      RETURN                                                         181
   88 LIN=0                                                          182
      RETURN                                                         183
      END                                                            184
```

AD-A035 941 EMMANUEL COLL BOSTON MASS F/G 17/6
INVESTIGATION OF MICROPULSATION ACTIVITY. 1. MAGAF SYSTEM ADDIT--ETC(U)
AUG 76 W F BELLEW, C J CANTOR, M P HAGAN F19628-76-C-0013
UNCLASSIFIED SCIENTIFIC-1 AFGL-TR-76-0244 NL

2 OF 2
AD A035941

END
DATE FILMED
3-77


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS - 1963 - A

```
      SUBROUTINE READW(N,WT,LFIL,IP,DT)                                 183
      DIMENSION W(1),WT(157),FREQRES(130)                               184
      N=LFIL/2                                                          185
      PLP=1./FLOAT(N)                                                   186
      H=1./N                                                            187
      NP1=N+1                                                           188
      TWODT=2.*DT                                                       189
      PI=3.14159265358979                                               190
      W(1)=PLP                                                          191
      PSUM=0.                                                           192
      DO 7 K=1,N                                                        193
      IF(K.EQ.N/2) GO TO 4                                              194
      F=COS(H*K*PI)/(1.-4.*H*H*K*K)                                     195
      GO TO 5                                                           196
    4 F=.7853981634                                                     197
      PRINT*,"   IN SUBR READW   K=",K                                  198
    5 S=SIN(PLP*K*PI)/(PI*K)                                            199
      IF(K.EQ.100) PRINT*,F,S,PLP*K*PI,PI*K                             200
      W(K+1)=F*S                                                        201
      PSUM=PSUM+W(K+1)                                                  202
    7 CONTINUE                                                          203
      SUMOFLS=W(1)+2.*PSUM                                              204
      DELTA=1.-SUMOFLS                                                  205
      DELTAD=DELTA/FLOAT(2*N+1)                                         206
      PRINT*," SUM OF UNCORRECTED FILTER WEIGHTS=",SUMOFLS              207
      SUMWTS=0.                                                         208
      DO 11 I=1,NP1                                                     209
      W(I)=W(I)+DELTAD                                                  211
      SUMWTS=SUMWTS+W(I)                                                211
      WT(I+N)=W(I)                                                      212
      IC=NP1+1-I                                                        213
      WT(IC)=W(I)                                                       214
   11 PRINT*," SUM OF CORRECTED FILTER WEIGHTS=",2.*SUMWTS-W(1)         215
      IF(IP.EQ.0) GO TO 30                                              216
      CALL SKRATCH(WT,2*N+1)                                            217
      PRINT 100, NP1,WT                                                 218
  100 FORMAT(1H1,I5,*  LOW-PASS FIL. WEIGHTS,W0,W1,...,WN*,/(1X         219
     +10E13.6))                                                         219
   30 DO 50 I=1,130                                                     220
      SP=FLOAT(I-1)/129.*.2                                             221
      FREQ=SP/TWODT                                                     222
      SUMC=0.                                                           223
      DO 40 K=1,N                                                       224
      FK=K                                                              225
      SUMC=SUMC+WT(NP1+K)*COS(PI*FK*SP)                                 226
   40 CONTINUE                                                          227
      FREQRES(I)=WT(NP1)+2.*SUMC                                        228
      PRINT 102,I,FREQ,1./FREQ,100.*FREQRES(I)                          229
   50 CONTINUE                                                          230
      IF(IP.EQ.0) GO TO 60                                              231
      CALL SKRATCH(FREQRES,130)                                         232
```

```
      PRINT 111,FREQRES                                                 233
111   FORMAT(1H ,* FREQUENCY RESPONSE DATA FOLLOWS:*,/(1X,10G13         234
     +.5))                                                              234
      FORMAT(1X,I3,3E15.3)                                              235
      RETURN                                                            236
      END                                                               237
```

```
      SUBROUTINE FILTER(X,LIN,OUT,LOUT,LF,IDEC)                         238
      DIMENSION X(1),OUT(1)                                             239
      COMMON WT(157),IREAD                                              241
      LOUT=0                                                            241
      IB=1                                                              242
      IE=LF                                                             243
      JE=(LIN-LF)/IDEC+1                                                244
      DO 3 J=1,JE                                                       245
      FTD=0                                                             246
      DO 2 I=IB,IE                                                      247
      IW=1-IDEC*(J-1)                                                   248
2     FTD=FTD+WT(IW)*X(I)                                               249
      LOUT=LOUT+1                                                       251
3     OUT(LOUT)=FTD                                                     251
      IF(XMAX.LT.FTD) XMAX=FTD                                          252
      IF(XMIN.GT.FTD) XMIN=FTD                                          253
      IB=IB+IDEC                                                        254
      IE=IE+IDEC                                                        255
      PRINT 4,(OUT(I),I=1,LOUT)                                         256
4     FORMAT(10F12.3)                                                   257
      RETURN                                                            258
      END                                                               259
```

## 3. PROGRAMS

### (a) In Progress

Work is going on to recognize spurious points in the data and eliminate them before the data are analyzed.

### (b) Programs Planned

The next major area of concern will be to develop plotting routines for the Varian computer. After that we will begin to get magnetic indices from the network data.