National Aeronautics and
Space Administration

NASA CR-
160534

Lyndon B. Johnson Space Center
Houston, Texas 77058

JSC-14807

PROGRAM DOCUMENTATION

FOR THE

DATA NOISING PROGRAM

Job Order 53-409

CPD 912

Prepared By

Lockheed Electronics Company, Inc.

Systems and Services Division

Houston, Texas

Contract NAS 9-15800

For

ENGINEERING ANALYSIS DIVISION

June 1979

LEC-13266

(NASA-CR-160534) PROGRAM DOCUMENTATION FOR THE DATA NOISING PROGRAM (Lockheed Electronics Co.) 44 p HC A03/MF A01
CSCL 05B

N80-18087

Unclas
G3/16 13350

JSC-14807

PROGRAM DOCUMENTATION

FOR THE

DATA NOISING PROGRAM

Job Order 53-409

CPD 912

PREPARED BY

M. T. Nguyen

APPROVED BY

P. H. Horsley, Supervisor
Data Management Section

W. J. Reicks, Manager
Applied Mechanics Department

Prepared By

Lockheed Electronics Company, Inc.
Systems and Services Division
Houston, Texas

For

Engineering Analysis Division

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
LYNDON B. JOHNSON SPACE CENTER
HOUSTON, TEXAS

June 1979

LEC-13266

CONTENTS

1. INTRODUCTION

The Data Noising Program (DNP) will provide a new program which simulates the effect of various noise sources on flight data. The program will read a Space Shuttle Functional Simulator (SSFS) format file, perform various noising operations on it, and rewrite a file in the SSFS format. The first noise function implemented was the time skew function, since this function determines the first and last output data record. Following implementation of the time skew function, up to six additional data noising functions may be implemented; these are noise, location, bias, alignment, quantization, and sample rate. Three of these possible six functions have been implemented at the time of this documentation, and are described herein (time skew, noise, and quantization).

The task to design, check out, and document the Data Noising Program was performed for the EX32 organization at the Johnson Space Center.

2. PROGRAM DESCRIPTION

2.1 TIME SKEW FUNCTION

2.1.1 PURPOSE

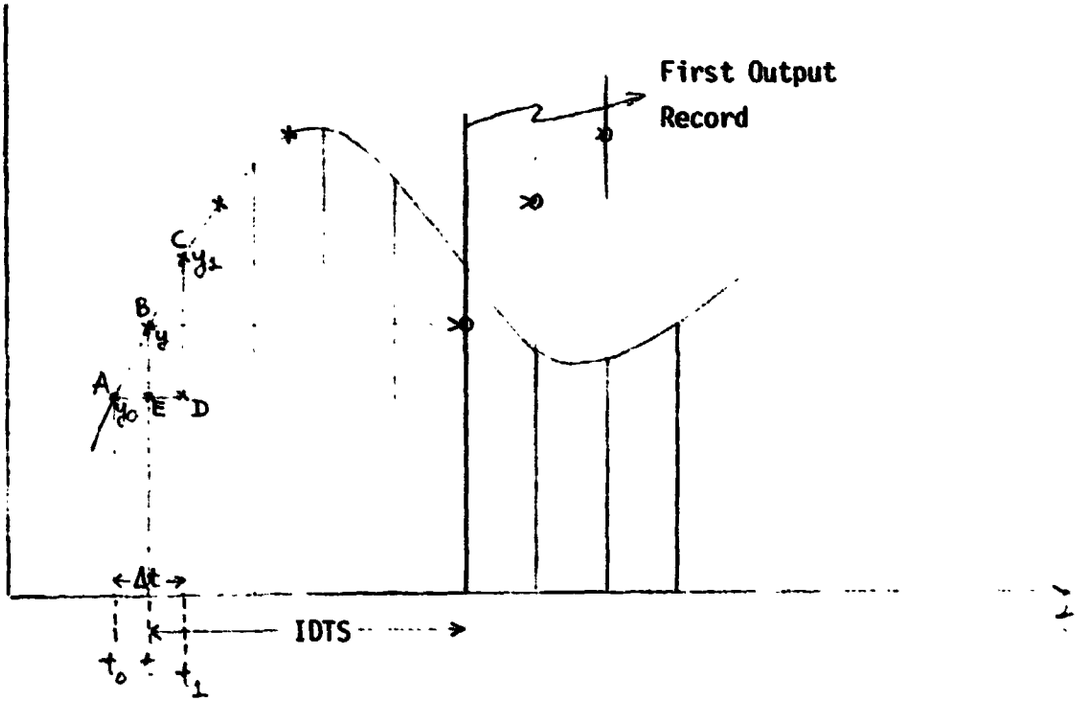
The DNP introduces a time shift, either in a positive (right) direction or negative (left) direction, for a given measurement, with respect to a normal (unskewed) measurement.

2.1.2 INPUT DATA

For a given measurement, the skew time can be input positively or negatively. However, the magnitude of skew time can not be greater than the magnitude of absolute value of 5 times the different time between two records ($\Delta\tau$) positively or negatively.

2.1.3 ANALYTICAL DESCRIPTION

a. Positive skew time



From the graph above,

$$\frac{BE}{CD} = \frac{AE}{AD}$$

Then

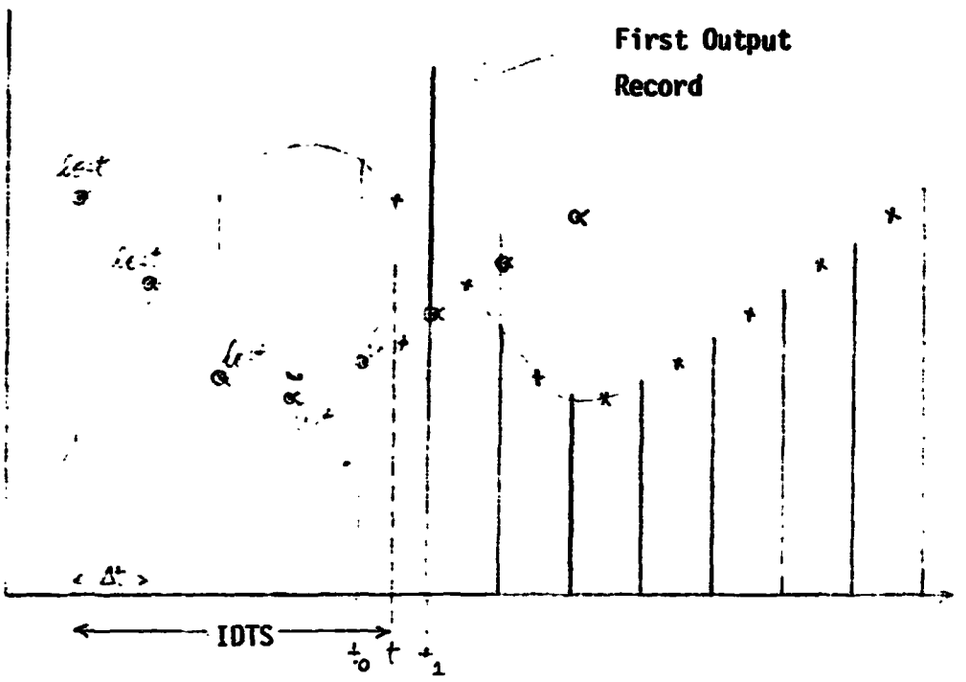
$$BE = \frac{AE}{AD} CD$$

$$y - y_0 = \frac{t - t_0}{\Delta t} (y_1 - y_0)$$

Therefore,

$$y = \frac{1}{\Delta t} \left[y_0(t_1 - t) + y_1(t - t_0) \right] \quad (1)$$

b. Negative skew time



Equation (1) is also applied for negative skew. The "lost" skew data will not be computed. Therefore, that data will not be written on the output file.

2.2 NOISE FUNCTION

2.2.1 PURPOSE

This function is used to change a measurement by adding the noise. It includes the frequency, amplitude, and phase.

2.2.2 INPUT DATA

For a given measurement, the following information must be input when the noise function is applied

a. Number of frequencies

b. For each frequency:
   Frequency in Hertz: FREQ
   Amplitude of frequency: AMPL
   Phase angle in degree: PHASE

c. Random number, which is a 5-7 digit integer where the last digit is odd: NRANDM

d. Type of random number:
   0 designates uniform
   1 designates normal: NTYPE

e. Amplitude of standard deviation: STANDN

2.2.3 ANALYTICAL DESCRIPTION

Given a measurement (whether or not it has been applied by another function) for each number of frequencies, it will be applied by the following equation:

$$Y = Y + AMPL(J) * SIN[FREQ(J) * TLOCAL + PHASE(J)]$$

where

Y = the measurement

TLOCAL = local time, which is zero from start of the output record. Time is incremented by Δt.

In the case where the input random number of this given measurement is 0, Y will be the final value after noise function is applied; otherwise, it will have the random noise added.

$$Y_1 = Y + STANDN * X$$

where

$$X = ZOR\ (NTYPE)$$

2.3 QUANTIZATION FUNCTION

2.3.1 PURPOSE

This function is used to change a measurement from a continuous form to one with discrete steps to simulate digital data.

2.3.2 INPUT DATA

For a given measurement, the following information must be input when the quantization function is applied.

a. Number of bits of quantization: IBIT

b. The upper and lower limits of range for which the measurement is to be quantized: QMAX, QMIN

2.3.3 ANALYTICAL DESCRIPTION

For a given measurement Y: if Y is less than QMIN, let Y = QMIN or if greater than QMAX, let Y = QMAX. Then Y is the final value returned from the quantization function. Otherwise, two additional parameters must be defined as follows:

a. DELTA = Number of steps into which measurement range is divided.

$$DELTA = (QMAX - QMIN)/BITS$$

where

$$BITS = FLOAT\ (2**IBIT)$$

b. INCR - Number of increments of size DELTA into which a measurement value will be divided.

$$INCR = INT\ ((Y-(QMIN + DELTA/2.))\ /DELTA\ )$$

and

$$Y_j = QMIN + FLOAT(INCR + 1) * DELTA$$

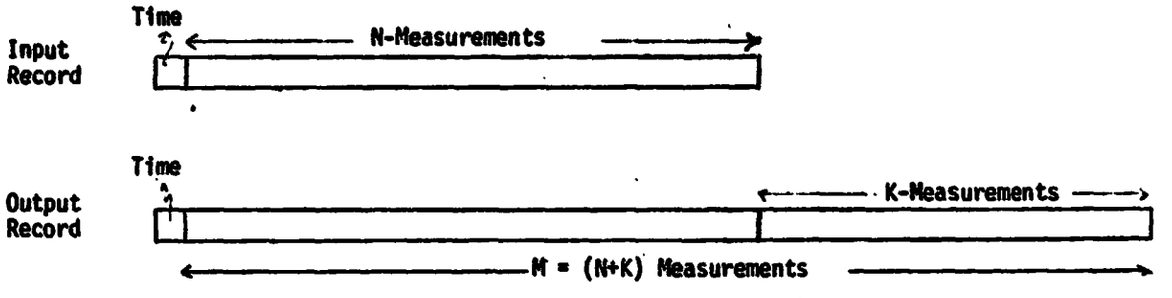
where

Y = the measurement that has been applied by another function previously, or

Y = the measurement from the input data file.

Y_j = final value after it has been quantized.

2.4 INPUT AND OUTPUT RECORDS FORMAT



Each input record consists of time and N-measurements. The output record will consist of time and M-measurements, where

$$M = N + K$$

where K is the number of measurements that are modified.

$$(1 \leq K \leq 100)$$

3. PROGRAM USAGE

The main program and its subroutine subprograms reside in the secured file EX32-L72330*DAPSPT. To execute the program, assuming that the user has assigned the input data file as well as the output data file, the user enters:

@XQT EX32-L72330*DAPSPT.RUN

The following message is displayed:

PLEASE ENTER THE INPUT AND OUTPUT LOGICAL UNITS

Key in two desired units (which must not be 5 and 6) that the user has assigned to the input and output data files. The program then will read the header record. The last word of the header record is the number of words in each subsequent record. Therefore, the following is then displayed:

THE NUMBER OF WORDS IN EACH DATA RECORD IS XXX

The program will ask for the following information on the display:

PLEASE ENTER START AND END TIME IN MILLISECOND INTEGER

The user will key in two legitimate times, and the following message will be displayed:

PLEASE ENTER CHANNEL NUMBER

The user enters any desirable channel number, and the program will then display the following function menu:

PLEASE ENTER THE NUMERICAL VALUE REPRESENTING THE FOLLOWING FUNCTIONS

1 - SKEW	2 - NOISE
3 - LOCATION	4 - BIAS
5 - ALIGN.	6 - QUANT.
7 - SAMPLE RATE	

NOTE: Only skew, noise, and quantization are presently available.

a. If "1" is entered, the message

PLEASE ENTER SKEW TIME IN MILLISECOND INTEGER

appears on the screen, the program will always let the user know the current maximum or minimum value of skew time. The program then lists the function menu

b. If "2" is entered, the following message is displayed:

PLEASE ENTER NUMBER OF FREQUENCIES

Key in the legitimate value for number of frequencies, then the following message is displayed

PLEASE ENTER VALUES OF FREQUENCY IN HERTZ
AMPLITUDE
PHASE IN DEGREE

to request the user to enter each set of frequency, amplitude and phase. Enter one at a time. The program will convert internally input values into radians and print them out. After that, the message following is displayed:

PLEASE ENTER THE VALUES REPRESENTING THE VARIABLES AS

* RANDOM NUMBER WHERE IT IS 5-7 DIGIT INTEGER, LAST DIGIT IS ODD
* TYPE OF RANDOM WHERE 0 IS UNIFORM AND 1 IS NORMAL
* AMPLITUDE OF STANDARD DEVIATION.

Enter one set of those values; the program will print them out and return the user to the function menu:

c. If "6" is entered, the message is then displayed:

PLEASE ENTER NUMBER OF BITS

Key in the legitimate value, then the program will ask:

PLEASE ENTER THE UPPER AND LOWER LIMITS OF RANGE FOR WHICH THIS CHANNEL IS TO BE QUANTIZED.

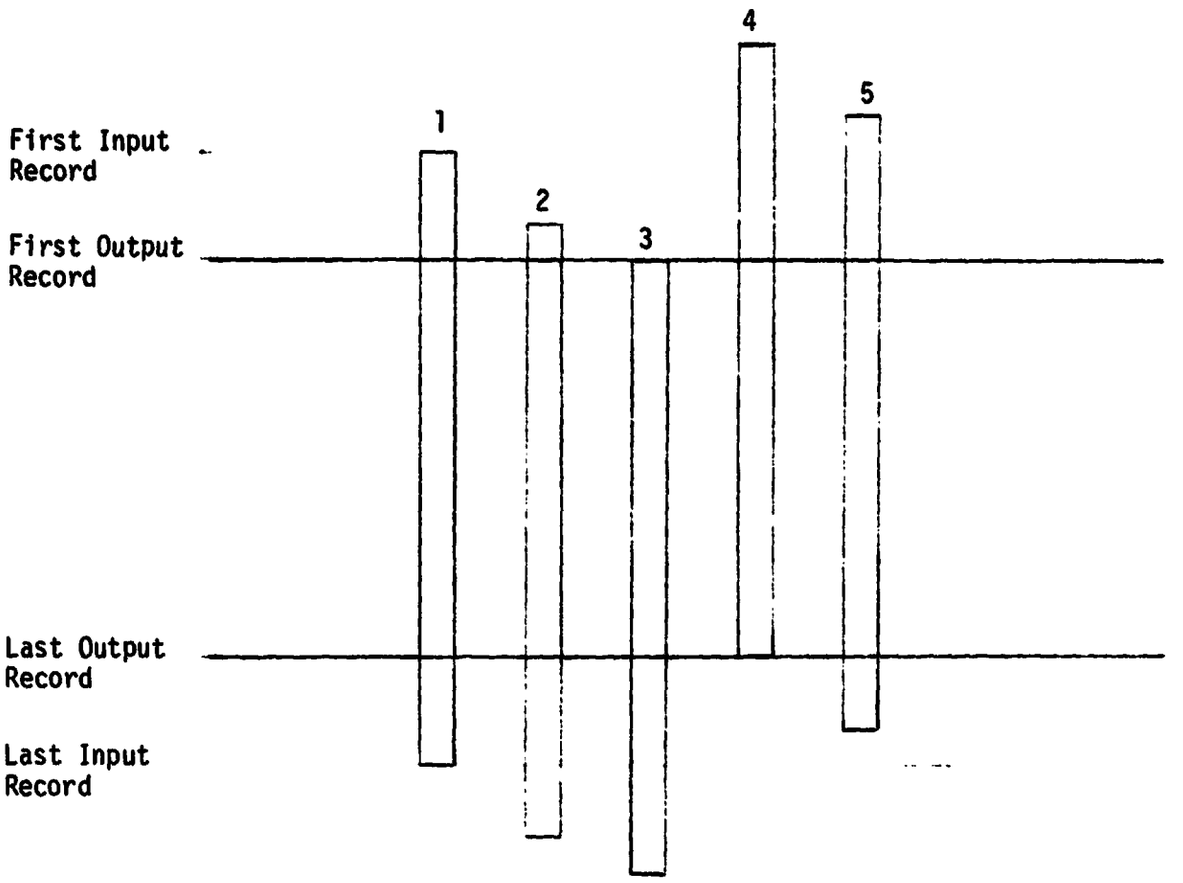
Enter two values; the program then will print them out and immediately return the user to the function menu.

In order to get out the function menu, the user simply enters "0", the program will ask for another channel number. However, if "0" is entered when the program requests the channel number, the program will then display the message:

TOTAL NUMBERS OF CHANNEL ARE XXX

From this point on, the output records are printed. Each output record consists of the number of channels in each input record and the total number of modified channels.

4. DIAGRAM



Because the time skew function dictates the outcome of the output records, the above chart shows how the output file looks, depending upon the input skew time. Five possible cases may occur:

1 - No skew time

2 - Input positive skew time is less than maximum positive skew time

3 - Maximum positive skew time

4 - Maximum negative skew time

5 - Magnitude of the absolute value of input negative skew time is less than magnitude of the absolute value of maximum negative.

APPENDIX A

FORTRAN SOURCE LISTING

```
@FOR,S DAPSPT.NOISNG/LEC
FOR 90E3-06/13/79-12:43:19 ([illegible]

   MAIN PROGRAM

   STORAGE USED: CODE(1) 001736; DATA(0) 00[illegible]; BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003    ARRAY  004230
    0004    HERTZ  000764
    0005    DISTC  000764
    0006    DEGRE  000764
    0007    TIMEX  000144

   EXTERNAL REFERENCES (BLOCK, NAME)

    0010   CONVER
    0011   RANDOM
    0012   SKEW
    0013   NOISE
    0014   QUANTN
    0015   NINTR$
    0016   NWDU$
    0017   NIO2$
    0020   NRDU$
    0021   NREW$
    0022   NRBU$
    0023   NIO3$
    0024   NERR2$
    0025   NSTOP$
    0026   NWBU$
    0027   NIO1$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)
```

0001	001627 1L	0001	000164 10L	0001	000134 100L	0001	001605 1000L	0001	001613 1010L
0001	000330 1020L	0001	000260 1030L	0001	000432 1050L	0001	000336 1051L	0001	001644 1055L
0001	000516 1210L	0001	001666 196L	0001	001674 197L	0001	000002 199L	0001	000620 2L
0001	000165 20L	0001	000321 2000L	0001	001702 201L	0001	001710 202L	0001	000237 210G
0001	000256 224G	0001	000377 23L	0001	001007 27L	0001	001376 310L	0001	001364 311L
0001	001247 320L	0001	001346 321L	0001	001331 330L	0001	001307 331L	0001	001313 332L
0001	001336 334L	0001	000545 400L	0001	000552 405L	0001	000674 410G	0001	000565 410L
0001	001201 415L	0001	001205 420L	0001	000716 424G	0001	001636 43L	0001	001652 49L
0001	000563 5L	0000	002371 500F	0001	001621 501L	0001	001165 552G	0001	001214 566G
0001	001067 570L	0001	001226 574G	0001	001113 575L	0001	001121 580L	0001	001126 585L
0000	002372 600F	0000	003235 608F	0000	003261 609F	0000	003305 6115F	0000	002710 6123F
0000	00240 615F	0000	003122 619F	0000	002416 620F	0000	003137 621F	0000	003160 6222F
0000	003170 6233F	0000	002433 625F	0000	002440 630F	0000	002455 632F	0000	002465 635F
0000	002545 640F	0001	001422 643G	0000	002637 652F	0001	001434 652G	0000	002577 655F
0000	003050 6555F	0000	003061 660F	0000	003071 665F	0000	003057 6677F	0000	003101 670F
0001	001475 671G	0000	003004 6710F	0000	003110 672F	0000	003011 6720F	0000	003032 6730F
0000	002566 676F	0000	002555 677F	0000	002530 691F	0000	002361 6990F	0000	003211 6991F
0000	003223 6993F	0001	001527 703G	0001	001536 707G	0001	001575 725G	0001	001716 745L
0001	001134 749L	0001	001045 750L	0001	001505 777L	0001	001660 79L	0000	002661 850F
0000	002667 851F	0000	002723 852F	0000	002771 857F	0001	000755 9L	0005 R	000000 AMPL

ORIGINAL PAGE IS OF POOR QUALITY

```
0000 R 002331 DTR       0004 R 000000 FREQ      0000 I 002275 I         0000 I 001446 IBIT      0000 I 002333 IBITS
0000 I 002342 IBUF      0000 I 002344 ICALL     0000 I 002316 ICHAN     0000 I 002343 ICLT      0000 I 002303 IDT
0000 I 002321 IDTS      0000 I 002350 IDTSAB    0000 I 002324 IDTSAS    0000 I 002347 IDTS+1    0000 I 002313 IDTSMN
0000 I 002314 IDTSMX    0000 I 000000 IDUM      0000 I 003323 IDUMMY    0000 I 002300 IET       0000 I 003323 IFCNTB
0000 I 002320 IFCNUM    0000 I 002352 IFIRST    0000 I 002317 IFUNC     0000 I 002274 IK        0000 I 002354 IMN
0000 I 000626 INSKEW    0000 I 002337 IRFOUT    0000 I 002340 IRLOUT    0000 I 002322 ISKEW     0000 I 002277 IST
0000 I 002305 ISTREC    0000 I 002336 ISUM      0000 I 002302 ITM       0000 I 002304 ITOTAL    0000 I 002346 ITYPE
0000 I 002272 IUNT      0000 I 002345 J         0000 I 002273 JUNT      0000 I 002360 K         0000 I 002355 KK
0000 I 002306 KOUNT     0000 I 002311 MAXDEL    0000 I 002312 MINDEL    0000 I 002315 NC        0000 I 002323 NCAP
0000 I 002310 NEGIDT    0000 L 002267 NEGSKW    0000 I 002341 NGOTO     0000 I 002351 NLOCAL    0000 I 002327 NMFREQ
0000 I 002307 NMOTRC    0000 I 002356 NMOVE     0000 I 002325 NMXNEG    0000 I 002353 NN        0000 I 002772 NRANDM
0000 I 002357 NSTOP     0000 I 001136 NTYPE     0000 I 002122 NUMFRE    0000 I 002301 NUMREC    0000 I 002276 NWORD
0000 I 002326 NWORK     0006 R 000000 PHASE     0000 L 002270 POSSKW    0000 R 001612 QMAX      0000 R 002334 QMAXX
0000 R 001756 QMIN      0000 R 002335 QMINN     0011 R 000000 RANDOM    0000 R 000400 RECORD    0000 L 002266 SKWFLG
0000 R 001302 STANDN    0007   000000 TLOCAL    0000 R 002330 TWOPIE    0003 R 000000 WORK      0000 R 002271 XEND
0000 R 002332 Z1
```

```
00100      1*      C
00100      2*      C
00100      3*      C                PROGRAM NAME    -    DATA NOISING PROGRAM
00100      4*      C
00100      5*      C              DEPT.      -      APPLIED MECHANICS
00100      6*      C              SECT.      -      DATA MANAGEMENT
00100      7*      C              DATE       -      NOVEMBER  1978
00100      8*      C              AUTHOR     -      M.T. NGUYEN
00100      9*      C
00100     10*      C
00100     11*      C
00100     12*      C        FOR EX43-ENGINEERING ANALYSIS DIVISION AT NASA/JSC
00100     13*      C
00100     14*      C
00100     15*      C        PURPOSE :
00100     16*      C
00100     17*      C
00100     18*      C           THE DATA NOISING PROGRAM WILL PROVIDE A NEW PROGRAM WHICH
00100     19*      C           SIMULATES THE EFFECT OF VARIOUS NOISE SOURCES ON DATA.
00100     20*      C           THE PROGRAM WILL READ A SPACE SHUTTLE FUNCTIONAL SIMULATOR
00100     21*      C           (SSFS) FORMAT FILE, PERFORM VARIOUS NOISING OPERATIONS
00100     22*      C           ON IT, AND REWRITE A FILE IN THE SSFS FORMAT.
00100     23*      C
00100     24*      C           THE FIRST AND NECESSARY NOISE FUNCTION TO BE IMPLEMENTED
00100     25*      C           IS THE TIME-SKEW FUNCTION. FOLLOWING IMPLEMENTATION OF
00100     26*      C           THE TIME-SKEW FUNCTION, UP TO SIX ADDITIONAL DATA
00100     27*      C           NOISING FUNCTIONS ARE TO BE IMPLEMENTED. THEY ARE LISTED
00100     28*      C           AS FOLLOWS : NOISE, LOCATION, BIAS, ALIGNMENT, QUANTATIZATION
00100     29*      C           AND SAMPLE RATE.
00100     30*      C
00100     31*      C
00101     32*            PARAMETER IDTMIN=20,IDTSMA=5*IDTMIN
00103     33*            PARAMETER NWORK1=2*IDTSMA/IDTMIN+1
00104     34*            COMMON /ARRAY/ WORK(NWORK1,200)
00105     35*            COMMON /HERTZ/ FREQ(100,5)
00106     36*            COMMON /DISTC/ AMPL(100,5)
00107     37*            COMMON /DEGRE/ PHASE(100,5)
00110     38*            COMMON /TIMEX/ TLOCAL(100)
```

```
00111     39*          DIMENSION IDUM(256),RECORD(150),IDUMMY(1),INSKEW(100)                  000002
00112     40*          DIMENSION IFCNTB(100,8),NRANDM(100),NTYPE(100),STANDN(100)             000002
00113     41*          DIMENSION IBIT(100),QMAX(100),QMIN(100),NUMREC(100)                    000002
00114     42*          LOGICAL SKWFLG,NEGSKW,POSSKW                                           000002
00115     43*          EQUIVALENCE (IDUMMY(1),IFCNTB(1,1))                                    000002
00116     44*          DATA XEND /3HEOF/                                                      000002
00116     45*     C                                                                           000002
00116     46*     C  ***  INPUT LOGICAL UNIT - IUNT                                         000002
00116     47*     C  *** OUTPUT LOGICAL UNIT - JUNT                                         000002
00116     48*     C                                                                           000002
00120     49*       199 WRITE(6,6990)                                                       000002
00122     50*      6990 FORMAT(1X,'ENTER THE INPUT & OUTPUT LOGICAL UNITS')                 000006
00123     51*           READ(5,500) IUNT,JUNT                                              000006
00127     52*       500 FORMAT( )                                                          000015
00127     53*     C                                                                           000015
00130     54*           IF(IUNT .LT. 1  .OR.  IUNT .EQ. 5  .OR.  IUNT .EQ. 6  .OR.  IUNT .  000015
00130     55*          *GT. 30) GO TO 196                                                  000015
00132     56*           IF(JUNT .LT. 1  .OR.  JUNT .EQ. 5  .OR.  JUNT .EQ. 6  .OR.  IUNT .  000047
00132     57*          *GT. 30) GO TO 197                                                  000047
00132     58*     C                                                                           000047
00132     59*     C  ***  INITIALIZE THE LOGICAL FLAGS  ***                                 000047
00132     60*     C                                                                           000047
00134     61*           SKWFLG = .FALSE.                                                    000101
00136     62*           POSSKW = .FALSE.                                                    000102
00136     63*           NEGSKW = .FALSE.                                                    000103
00137     64*           IK = 0                                                              000104
00137     65*     C                                                                           000104
00137     66*     C   ***  REWIND BOTH INPUT & OUTPUT DATA FILES  ***                       000104
00137     67*     C                                                                           000104
00140     68*           REWIND IUNT                                                         000105
00141     69*           REWIND JUNT                                                         000110
00141     70*     C                                                                           000110
00141     71*     C  ***  READ HEADER RECORD (256 WORDS)  ***                              000110
00141     72*     C  ***  DETERMINE NUMBER OF POINTS IN EACH DATA RECORD ***               000110
00141     73*     C                                                                           000110
00142     74*           READ(IUNT,ERR=501)(IDUM(I),I=1,256)                                 000113
00145     75*           NWORD = IDUM(256)                                                   000123
00146     76*           WRITE(6,600) NWORD                                                  000125
00151     77*       600 FORMAT(1X,'THE NUMBER OF POINTS IN EACH DATA RECORD ARE',I5)        000134
00151     78*     C                                                                           000134
00151     79*     C  ***  USER(S) INPUT START AND END TIMES  ***                           000134
00151     80*     C                                                                           000134
00152     81*       100 WRITE(6,615)                                                        000134
00154     82*       615 FORMAT(1X,'ENTER START AND END TIME IN MILLISECOND INTEGER')       000140
00155     83*           READ(5,500) IST,IET                                                 000140
00155     84*     C                                                                           000140
00161     85*           IF(IST .EQ. 0) GO TO 1000                                           000147
00163     86*           IF(IET .GT. IST) GO TO 10                                           000151
00165     87*           WRITE(6,620)                                                        000155
00167     88*       620 FORMAT(1X,'ERROR - TRY AGAIN',/                                     000162
00167     89*          *,1X,'END TIME HAS TO BE GREATER THAN START TIME')                   000162
00170     90*           GO TO 100                                                           000162
00170     91*     C                                                                           000162
00170     92*     C  ***  LOOKING FOR START TIME  ***                                      000162
00170     93*     C                                                                           000162
00171     94*       10 NUMREC = 0                                                           000164
00171     95*     C                                                                           000164
00171     96*     C  ***  ROUTINE CONVER IS DEPICTED FROM EX43-N04482*BIGFILE  ***        000164
```

```
00171     97*     C  ***  ITS ORIGINAL NAME IS SSFSR  ***                                  000164
00171     98*     C                                                                       000164
00172     99*        20 CALL CONVER(IUNT,NWORD,ITM,IDT,RECORD,IK)                         000165
00173    100*           IF(IST .LT. ITM) GO TO 43                                         000174
00175    101*           IF(IK .EQ. 1) GO TO 1010                                          000200
00177    102*           NUMREC = NUMREC + 1                                               000205
00177    103*     C                                                                       000205
00177    104*     C  *** DETERMINE THE TOTAL RECORDS NEEDED IN THE PROGRAM  ***           000205
00177    105*     C                                                                       000205
00200    106*           IF(NUMREC .EQ. 2) ITOTAL = (IET-IST)/IDT + 1                      000211
00202    107*           IF(ITM .LT. IST) GO TO 20                                         000222
00202    108*     C                                                                       000222
00202    109*     C  ***  START TIME HAS BEEN FOUND  ***                                 000222
00202    110*     C                                                                       000222
00204    111*           IST = ITM                                                         000226
00205    112*           ISTREC = 1                                                        000230
00206    113*           KOUNT = 1                                                         000232
00206    114*     C                                                                       000232
00206    115*     C  ***  STORE THE FIRST RECORD IN THE WORK - ARRAY  ***                000232
00206    116*     C                                                                       000232
00207    117*           DO 11 I=1,NWORD                                                   000237
00212    118*        11 WORK(1,I) = RECORD(I)                                             000237
00212    119*     C                                                                       000237
00212    120*     C  ***  INITIALIZATION  ***                                           000237
00212    121*     C                                                                       000237
00214    122*           NNOTRC = 1                                                        000241
00215    123*           NEGIDT = - IDT                                                    000243
00216    124*           MAXDEL = 5*IDT                                                    000245
00217    125*           MINDEL = - MAXDEL                                                 000247
00220    126*           IDTSMN = 0                                                        000250
00221    127*           IDTSMX = 0                                                        000251
00222    128*           NC = 0                                                            000252
00223    129*           DO 13 I=1,800                                                     000256
00226    130*        13 IDUMMY(I) = 0                                                     000256
00226    131*     C                                                                       000256
00226    132*     C  ***  USER INPUT CHANNEL NUMBER TO BE MODIFIED  ***                  000256
00226    133*     C                                                                       000256
00230    134*      1030 WRITE(6,625)                                                      000260
00232    135*       625 FORMAT(1X,'ENTER CHANNEL NUMBER')                                 000264
00233    136*           READ(5,500) ICHAN                                                 000264
00233    137*     C                                                                       000264
00233    138*     C  ***  TESTING USER'S INPUT OF CHANNEL NUMBER                         000264
00233    139*     C                                                                       000264
00236    140*           IF(ICHAN .EQ. 0) GO TO 2000                                       000272
00240    141*           IF(ICHAN .GT. 1 .AND. ICHAN .LE. NWORD) GO TO 1020                000274
00242    142*           WRITE(6,630)                                                      000312
00244    143*        630 FORMAT(1X,'ERROR - TRY AGAIN',/                                  000317
00244    144*          *,1X,'YOU HAVE INPUT AN INVALID CHANNEL NUMBER')                   000317
00245    145*           GO TO 1030                                                        000317
00245    146*     C                                                                       000317
00246    147*      2000 WRITE(6,632) NC                                                   000321
00251    148*       632 FORMAT(//25X,'NUMBER OF MODIFIED CHANNELS ARE',I5)                000326
00252    149*           GO TO 750                                                         000326
00252    150*     C                                                                       000326
00252    151*     C  *** INCREMENT NUMBER OF CHANNEL BY 1 ***                           000326
00252    152*     C  ***  THEN STORE IT IN THE FUNCTION TABLE  ***                       000326
00252    153*     C                                                                       000326
00253    154*      1020 NC = NC + 1                                                       000330
```

```
00254    155*          IFCNTB(NC,1) = ICHAN                                                    000333
00254    156*     C                                                                            000333
00254    157*     C  ***  DISPLAY THE FUNCTION MENU  ***                                      000333
00254    158*     C                                                                            000333
00255    159*      1051 WRITE(6,635)                                                           000336
00257    160*       635 FORMAT(/,1X,'ENTER THE NUMERICAL VALUE REPRESENTING THE FOL           000342
00257    161*          *LOWING FUNCTIONS',/,10X,'1-SKEW ',10X,'2-NOISE ',/                    000342
00257    162*          *,10X,'3-LOCATION ',6X,'4-BIAS',/,10X,'5-ALIGN.',8X,'6-QUANT. ',/      000342
00257    163*          *,18X,'7-SAMPLE RATE'/)                                                000342
00257    164*     C                                                                            000342
00257    165*     C  ***  READ THE FUNCTION NUMBER ***                                       000342
00257    166*     C                                                                            000342
00260    167*          READ(5,500) IFUNC                                                       000342
00263    168*          IF(IFUNC .EQ. 0) GO TO 1030                                             000350
00265    169*          IF(IFUNC .GE. 1 .AND. IFUNC .LE. 7) GO TO 23                            000352
00265    170*     C                                                                            000352
00267    171*          WRITE(6,691)                                                            000370
00271    172*        691 FORMAT(1X,'ERROR - TRY AGAIN',/                                       000375
00271    173*          *,' YOU HAVE INPUT AN INVALID FUNCTION NUMBER')                         000375
00272    174*          GO TO 1051                                                              000375
00273    175*         23 CONTINUE                                                              000377
00273    176*     C                                                                            000377
00273    177*     C   ***  TURN ON THE SKEW FLAG  ***                                        000377
00273    178*     C                                                                            000377
00274    179*          IF (IFUNC .EQ. 1) SKWFLG = .TRUE.                                       000377
00274    180*     C                                                                            000377
00274    181*     C  ***  BUILDING THE FUNCTION TABLE  ***                                   000377
00274    182*     C                                                                            000377
00276    183*          IFCNUM = IFUNC + 1                                                      000403
00276    184*     C                                                                            000403
00276    185*     C  ***  TEST IF THAT FUNCTION HAS BEEN INPUT  ***                          000403
00276    186*     C                                                                            000403
00277    187*          IF(IFCNTB(NC,IFCNUM) .GT. 0) GO TO 1055                                 000410
00301    188*          IFCNTB(NC,IFCNUM) = IFUNC                                               000413
00301    189*     C                                                                            000413
00301    190*     C  ***  INPUT DATA RELATED TO FUNCTION(S)  ***                             000413
00301    191*     C                                                                            000413
00302    192*          GO TO (1050,2,1,1,1,9,1),IFUNC                                          000415
00302    193*     C                                                                            000415
00302    194*     C  ***  INPUT DATA RELATED TO FUNCTION SKEW  ***                           000415
00302    195*     C                                                                            000415
00303    196*      1050 WRITE(6,640)                                                           000432
00305    197*       640 FORMAT(1X,'ENTER SKEW TIME IN MILLISECOND INTEGER')                   000436
00306    198*          READ(5,500) IDTS                                                        000436
00306    199*     C                                                                            000436
00306    200*     C  ***  CHECKING THE USER'S INPUT OF SKEW TIME  ***                        000436
00306    201*     C                                                                            000436
00311    202*          IF(IDTS .GT. MAXDEL) GO TO 201                                          000445
00313    203*          IF(IDTS .LT. MINDEL) GO TO 202                                          000451
00313    204*     C                                                                            000451
00313    205*     C   ***  TURN ON THE NEGATIVE SKEW FLAG  ***                               000451
00313    206*     C                                                                            000451
00315    207*          IF(IDTS .LT. 0) NEGSKW = .TRUE.                                         000456
00315    208*     C                                                                            000456
00315    209*     C   ***  TURN ON THE POSITIVE SKEW FLAG  ***                               000456
00315    210*     C                                                                            000456
00317    211*          IF(IDTS .GT. 0) POSSKW = .TRUE.                                         000463
00317    212*     C                                                                            000463
```

```
00317    213*     C  ***  STORE THE INPUT SKEW TIME IN THE ARRAY  ***                                  000463
00317    214*     C                                                                                   000463
00321    215*      1200 CONTINUE                                                                      000470
00322    216*           INSKEW(NC) = IDTS                                                             000470
00323    217*           ISKEW = IABS(INSKEW(NC))                                                      000473
00324    218*           IF(INSKEW(NC) .GT. 0) GO TO 1210                                              000475
00324    219*     C                                                                                   000475
00324    220*     C    ***  DETERMINE THE CURRENT MINIMUM VALUE OF SKEW TIME  ***                     000475
00324    221*     C                                                                                   000475
00326    222*           IF(INSKEW(NC) .LT. IDTSMN) IDTSMN = IDTS                                      000475
00330    223*           WRITE(6,677) IDTSMN                                                           000500
00333    224*       677 FORMAT(1X,'CURRENT MINIMUM VALUE OF SKEW TIME IS ',I5/)                       000507
00333    225*     C                                                                                   000516
00333    226*     C  ***  DETERMINE THE CURRENT MAXIMUM VALUE OF SKEW TIME  ***                       000516
00333    227*     C                                                                                   000516
00334    228*      1210 IF(ISKEW .GT. IDTSMX) IDTSMX = ISKEW                                          000516
00336    229*           WRITE(6,676) IDTSMX                                                           000516
00341    230*       676 FORMAT(1X,'CURRENT MAXIMUM VALUE OF SKEW TIME IS ',I5/)                       000523
00341    231*     C                                                                                   000531
00341    232*     C  ***  DETERMINE NUMBER OF RECORDS NEEDED TO BE READ  ***                         000531
00341    233*     C                                                                                   000531
00342    234*           IF(MOD(IDTSMX,IDT) .GT. 0) GO TO 400                                          000531
00344    235*           NCAP = IDTSMX/IDT + 1                                                         000531
00345    236*           GO TO 405                                                                     000536
00345    237*     C                                                                                   000543
00346    238*       400 NCAP = IDTSMX/IDT + 2                                                         000543
00347    239*       405 IF(NCAP .LE. ITOTAL) GO TO 410                                              000545
00351    240*           WRITE(6,655)                                                                  000552
00353    241*       655 FORMAT(1X,'ERROR - YOU HAVE MADE A CRITICAL MISTAKE'/                        000555
00353    242*          *1X,'THE TOTAL NUMBER OF RECORDS DICTATED BY START AND END'/                  000563
00353    243*          *1X,'TIMES SHOULD BE GREATER THAN THE NUMBER OF STARTING'/                    000563
00353    244*          *1X,'OUTPUT RECORD')                                                          000563
00354    245*           STOP                                                                          000563
00354    246*     C                                                                                   000563
00354    247*     C  ***  DETERMINE THE TOTAL NUMBER OF RECORDS ( NWORK ) NEEDED  ***                000563
00354    248*     C  ***          IN THE WORKING ARRAY             ***                                000563
00354    249*     C                                                                                   000563
00355    250*       410 CONTINUE                                                                      000563
00356    251*           IDTSAS = IABS(IDTSMN)                                                         000565
00357    252*           NMXNEG = IDTSAS/IDT + 1                                                       000565
00360    253*           IF(MOD(IDTSAS,IDT) .GT. 0) NMXNEG = NMXNEG + 1                                000566
00362    254*           NWORK = NCAP + NMXNEG - 1                                                     000572
00363    255*           IF(NWORK .LT. ITOTAL) GO TO 1051                                              000602
00365    256*           WRITE(6,652)                                                                  000606
00367    257*       652 FORMAT(1X,'ERROR - YOU HAVE MADE A CRITICAL MISTAKE'/                        000611
00367    258*          *1X,'WE CANNOT FIND THE RECORD NEEDED FOR OUTPUTTING')                        000616
00370    259*           GO TO 5                                                                       000616
00370    260*     C                                                                                   000616
00370    261*     C  ***  INPUT DATA RELATED TO FUNCTION NOISE  ***                                   000616
00370    262*     C                                                                                   000616
00371    263*         2 WRITE(6,850)                                                                  000616
00373    264*       850 FORMAT(1X,'ENTER NUMBER OF FREQUENCIES')                                      000620
00374    265*           READ(5,500) NMFREQ                                                            000624
00374    266*     C                                                                                   000624
00377    267*           IF(NMFREQ .LE. 0  .OR.  NMFREQ .GT. 5) GO TO 745                              000624
00401    268*           NUMFRE(NC) = NMFREQ                                                           000636
00402    269*           TWOPIE = 6.2831853                                                            000653
00403    270*           DTR = .01745329                                                               000656
                                                                                                     000660
```

```
00483    271*     C
00483    272*     C  ***  INPUT FREQUENCIES, AMPLITUDES AND PHASES  ***
00483    273*     C
00484    274*           WRITE(6,851)
00486    275*       851 FORMAT(1X,'ENTER VALUES OF FREQUENCY IN HERTZ',/
00486    276*          *,24X,'AMPLITUDE                ',/
00486    277*          *,24X,'PHASE IN DEGREE')
00486    278*     C
00487    279*           DO 4 I = 1,NMFREQ
00512    280*           READ(5,500) FREQ(NC,I),AMPL(NC,I),PHASE(NC,I)
00517    281*           FREQ(NC,I) = FREQ(NC,I)*TWOPIE
00520    282*           PHASE(NC,I) = PHASE(NC,I)*DTR
00521    283*         4 CONTINUE
00521    284*     C
00523    285*           DO 3 I=1,NMFREQ
00526    286*           WRITE(6,6123) FREQ(NC,I),AMPL(NC,I),PHASE(NC,I)
00533    287*      6123 FORMAT(5X,'FREQUENCY',F15.5,5X,'AMPLITUDE',F15.5,5X,'PHASE',F15.5)
00534    288*         3 CONTINUE
00534    289*     C
00534    290*     C  ***  INPUT RANDOM NUMBER, TYPE OF RANDOM AND STANDARD DEVIATION  ***
00534    291*     C
00536    292*           WRITE(6,852)
00540    293*       852 FORMAT(1X,'ENTER THE VALUES REPRESENTING THE VARIABLES AS'/
00540    294*          *,10X,'RANDOM NUMBER WHERE IT IS 5-7 DIGIT INTEGER, LAST DIGIT IS 0
00540    295*          *DD',/,10X,'TYPE OF RANDOM WHERE 0 IS UNIFORM AND 1 IS NORMAL'/
00540    296*          *,10X,'AMPLITUDE OF STANDARD DEVIATION')
00540    297*     C
00541    298*           READ(5,500) NRANDM(NC),NTYPE(NC),STANDN(NC)
00546    299*           IF(NRANDM(NC) .LE. 0) GO TO 1051
00550    300*           Z1 = RANDOM(NRANDM(NC))
00550    301*     C
00551    302*           WRITE(6,857) NRANDM(NC),NTYPE(NC),STANDN(NC)
00556    303*       857 FORMAT(/1X,'RANDOM NUMBER',I8,2X,'TYPE',I2,'  STNDRD DEV.',F10.5/)
00557    304*           GO TO 1051
00557    305*     C
00557    306*     C   ***  INPUT DATA RELATED TO FUNCTION QUANTATIZATION  ***
00557    307*     C
00560    308*         9 WRITE(6,6710)
00562    309*      6710 FORMAT(1X,'ENTER NUMBER OF BITS')
00563    310*           READ(5,500) IBITS
00566    311*           IF(IBITS .LE. 0) GO TO 49
00570    312*           IBIT(NC) = IBITS
00570    313*     C
00571    314*        27 WRITE(6,6720)
00573    315*      6720 FORMAT(1X,'ENTER THE UPPER & LOWER LIMITS OF RANGE',/
00573    316*          *,1X,'FOR WHICH THIS CHANNEL IS TO BE QUANTATIZED')
00573    317*     C
00574    318*           READ(5,500) QMAXX,QMINN
00500    319*           IF(QMAXX .LE. QMINN) GO TO 79
00502    320*           QMAX(NC) = QMAXX
00503    321*           QMIN(NC) = QMINN
00503    322*     C
00504    323*           WRITE(6,6730) IBIT(NC),QMAX(NC),QMIN(NC)
00511    324*      6730 FORMAT(/10X,'NUMBER OF BITS',I3,5X,'MAX. NUMBER',F15.5,5X,
00511    325*          *'MIN. NUMBER',F15.5/)
00512    326*           GO TO 1051
00512    327*     C
00512    328*     C  ***  START PROCESSING THE DATA NOISNG PROGRAM  ***
```

```
000660
000660
000660
000662
000674
000674
000674
000674
000674
000674
000703
000706
000716
000716
000716
000716
000726
000726
000726
000726
000726
000726
000733
000733
000733
000733
000733
000733
000743
000746
000746
000754
000764
000764
000764
000764
000766
000772
000772
001001
001004
001004
001007
001013
001013
001013
001013
001023
001027
001031
001031
001033
001043
001043
001043
001043
001043
```

```
00512      C                                                                      001043
00513        750 ISUM = [illegible]                                               001045
00513      C                                                                      001045
00513      C  ***  STORE IN THE 256TH WORD OF HEADER RECORD  ***                  001045
00513      C                                                                      001045
00514            IDUM(256) = ISUM                                                 001047
00514      C                                                                      001047
00514      C  ***  WRITE ON THE OUTPUT TAPE THE HEADER RECORD  ***                001047
00514      C                                                                      001047
00515            WRITE(JUNT)(IDUM(I),I=1,256)                                     001050
00520            IF(SKWFLG) GO TO 570                                             001057
00520      C                                                                      001057
00520      C  ***  IRFOUT : INPUT RECORD FIRST OUTPUT  ***                        001057
00520      C  ***  IRLOUT : INPUT RECORD LAST  OUTPUT  ***                        001057
00520      C                                                                      001057
00520      C  ***  DETERMINE THE BEGINNING AND ENDING OF OUTPUT RECORDS  ***      001057
00520      C                                                                      001057
00520      C  ***  SKEW FLAG HAS NOT BEEN TURNED ON AT ALL  ***                   001057
00520      C                                                                      001057
00522            IRFOUT = 1                                                       001061
00523            IRLOUT = ITOTAL                                                  001063
00524            GO TO 420                                                        001065
00525        570 NGOTO = 1                                                        001067
00526            IF(POSSKW  .AND.  .NOT.  NEGSKW) NGOTO = 2                       001070
00530            IF(NEGSKW  .AND.  .NOT.  POSSKW) NGOTO = 3                       001075
00532            GO TO (575,580,585),NGOTO                                        001102
00532      C                                                                      001102
00532      C  ***  BOTH POSITIVE AND NEGATIVE FLAGS ARE TURNED ON  ***            001102
00532      C                                                                      001102
00533        575 IRFOUT = NCAP                                                    001113
00534            IRLOUT = ITOTAL - NCAP + 1                                       001114
00535            GO TO 749                                                        001117
00535      C                                                                      001117
00535      C  ***  ONLY POSITIVE FLAG IS TURNED ON  ***                           001117
00535      C                                                                      001117
00536        580 IRFOUT = NCAP                                                    001121
00537            IRLOUT = ITOTAL                                                  001122
00540            GO TO 749                                                        001124
00540      C                                                                      001124
00540      C  ***  ONLY NEGATIVE FLAG IS TURNED ON  ***                           001124
00540      C                                                                      001124
00541        585 IRFOUT = 1                                                       001126
00542            IRLOUT = ITOTAL - NCAP + 1                                       001127
00542      C                                                                      001127
00542      C  ***  READ MORE DATA RECORD FROM INPUT DATA FILE  ***                001127
00542      C                                                                      001127
00543        749 CALL CONVER(IUNT,NWORD,ITM,IDT,RECORD,IK)                        001134
00543      C                                                                      001134
00543      C  ***  CHECKING THE END OF FILE MARK  ***                             001134
00543      C                                                                      001134
00544            IF(IK .EQ. 1) GO TO 1010                                         001143
00546            KOUNT = KOUNT + 1                                                001146
00547            ISTREC = ISTREC + 1                                              001151
00550            IBUF = 0                                                         001154
00550      C                                                                      001154
00550      C  ***  STORE DATA IN THE WORK - ARRAY  ***                            001154
00550      C                                                                      001154
00551            DO 31 I = 1,NWORD                                                001165
```

```
00554    387*        31 WORK(KOUNT,I) = RECORD(I)                                                    001165
00554    388*     C                                                                                  001165
00554    389*     C  ***  CHECKING AGAINST THE TOTAL NUMBER OF RECORDS AND THE NUMBER ***           001165
00554    390*     C  ***  OF RECORDS STORED IN THE WORKING ARRAY  ***                                001165
00554    391*     C                                                                                  001165
00556    392*           IF(ISTREC .LT. ITOTAL) GO TO 415                                            001167
00560    393*           IF(KOUNT .GT. 1) GO TO 420                                                  001173
00562    394*           GO TO 5                                                                     001177
00563    395*        415 IF(KOUNT .LT. NWORK) GO TO 749                                             001201
00563    396*     C                                                                                  001201
00563    397*     C  ***  IN  PROGATE THE FUNCTION TABLE FOR NUMBER OF CHANNEL  ***                  001201
00563    398*     C                                                                                  001201
00565    399*        420 DO 380 ICNT = 1,NC                                                         001205
00565    400*     C                                                                                  001205
00565    401*     C  ***  THE FLAG ( ICALL ) IS SET FOR TESTING IF THIS CHANNEL IS  ***              001205
00565    402*     C  ***  BEING APPLIED BY ANOTHER FUNCTION  ***                                     001205
00565    403*     C                                                                                  001205
00570    404*           ICALL = 0                                                                   001214
00571    405*           I = ICNT                                                                    001215
00572    406*           ICHAN = IFCNTB(I,1)                                                         001221
00572    407*     C                                                                                  001221
00573    408*               DO 310 J=2,8                                                            001226
00576    409*               IF(IFCNTB(I,J) .LE. 0) GO TO 310                                        001226
00600    410*               ITYPE = IFCNTB(I,J)                                                     001230
00601    411*               GO TO (320,321,310,310,310,311,310),ITYPE                               001232
00601    412*     C                                                                                  001232
00601    413*     C  ***  SEARCH FOR THE INPUT OF SKEW FUNCTION  ***                                001232
00601    414*     C                                                                                  001232
00602    415*        320 IDTSKW = INSKEW(I)                                                         001247
00603    416*           IF(IDTSKW .LT. IDT  .AND.  IDTSKW .GT. NEGIDT) GO TO 334                    001250
00605    417*           IDTSAB = IABS(IDTSKW)                                                       001265
00606    418*           NLOCAL = IDTSAB/IDT + 1                                                     001267
00607    419*           IF(MOD(IDTSAB,IDT) .GT. 0) NLOCAL = NLOCAL + 1                              001273
00611    420*           IF(IDTSKW .LT. 0) GO TO 330                                                 001303
00613    421*        331 IFIRST = NCAP - NLOCAL + 1                                                 001307
00613    422*     C                                                                                  001307
00613    423*     C  ***  APPLY FUNCTION AS DICTATED IN IFCNTB  ***                                  001307
00613    424*     C                                                                                  001307
00614    425*        332 CALL SKEW(IDT,ICHAN,NWORD,ICNT,IFIRST,IRFOUT,IDTSAB,ICALL)                 001313
00615    426*           NN = NWORD + ICNT                                                           001324
00616    427*           GO TO 310                                                                   001327
00616    428*     C                                                                                  001327
00617    429*        330 IFIRST = NCAP + NLOCAL - 2                                                 001331
00620    430*           GO TO 332                                                                   001334
00620    431*     C                                                                                  001334
00621    432*        334 IDTSAB = IABS(IDTSKW)                                                      001336
00622    433*           NLOCAL = 1                                                                  001337
00623    434*           IF(IDTSKW .GT. 0) GO TO 330                                                 001341
00625    435*           GO TO 331                                                                   001344
00625    436*     C                                                                                  001344
00625    437*     C  ***  CALL ROUTINE NOISE  ***                                                   001344
00625    438*     C                                                                                  001344
00626    439*        321 CALL NOISE(ICHAN,NUMFRE,STANDN,NTYPE,IDT,ICNT,NWORD,IBUF,                  001346
00626    440*       *                NRANDM,ICALL,IRFOUT)                                          001346
00627    441*           GO TO 310                                                                   001362
00627    442*     C                                                                                  001362
00627    443*     C  ***  CALL ROUTINE QUANTATIZATION  ***                                          001362
00627    444*     C                                                                                  001362
```

```
00630    445*      311 CALL QUANTN(IBIT,QMAX,QMIN,ICNT,NWORD,ICALL,ICHAN,IRFOUT)                001364
00630    446*    C                                                                             001364
00631    447*      310     CONTINUE                                                            001401
00631    448*    C                                                                             001401
00633    449*      300 CONTINUE                                                                001401
00633    450*    C                                                                             001401
00633    451*    C  ***  WRITE ON THE OUTPUT FILE OR TAPE  ***                                 001401
00633    452*    C                                                                             001401
00635    453*          WRITE(6,6555) NMOTRC                                                    001401
00640    454*     6555 FORMAT(/20X,'OUTPUT RECORD NUMBER = ',I4/)                              001412
00641    455*          WRITE(6,6677) (WORK(IRFOUT,INN),INN=1,ISUM)                             001412
00647    456*     6677 FORMAT(2X,8F16.5)                                                       001425
00647    457*    C                                                                             001425
00650    458*          WRITE(JUNT) (WORK(IRFOUT,KK),KK=1,ISUM)                                 001425
00656    459*          IBUF = 1                                                                001437
00657    460*          NMOTRC = NMOTRC + 1                                                     001441
00660    461*          IF(SKWFLG) GO TO 777                                                    001443
00660    462*    C                                                                             001443
00660    463*    C  ***  SKEW FLAG HAS NOT BEEN TURNED ON AT ALL  ***                         001443
00660    464*    C                                                                             001443
00662    465*          CALL CONVER(IUNT,NWORD,ITM,IDT,RECORD,IK)                               001445
00663    466*          IF(ITM .GT. IET) GO TO 5                                                001455
00665    467*          IF(IK .EQ. 1) GO TO 1010                                                001461
00667    468*          ISTREC = ISTREC + 1                                                     001466
00667    469*    C                                                                             001466
00670    470*          DO 15 I=1,NWORD                                                         001475
00673    471*       15 WORK(1,I) = RECORD(I)                                                   001475
00673    472*    C                                                                             001475
00675    473*          IF(ISTREC .GT. ITOTAL) GO TO 5                                          001477
00677    474*          GO TO 420                                                               001503
00677    475*    C                                                                             001503
00677    476*    C  ***  INITIALIZE TO REPOSITION THE WORK - ARRAY  ***                       001503
00677    477*    C                                                                             001503
00700    478*      777 NMOVE = 1                                                               001505
00701    479*          NSTOP = NWORK - 1                                                       001506
00701    480*    C                                                                             001506
00701    481*    C  ***  REPOSITION THE WORK - ARRAY  ***                                     001506
00701    482*    C                                                                             001506
00702    483*          DO 99 K=1,NSTOP                                                         001520
00705    484*          NMOVE = NMOVE + 1                                                       001527
00706    485*             DO 98 J=1,NWORD                                                      001531
00711    486*             WORK(K,J) = WORK(NMOVE,J)                                            001536
00712    487*       98    CONTINUE                                                            001544
00714    488*       99 CONTINUE                                                               001544
00714    489*    C                                                                             001544
00714    490*    C  ***  BRING IN MORE DATA FROM INPUT FILE  ***                              001544
00714    491*    C                                                                             001544
00716    492*          CALL CONVER(IUNT,NWORD,ITM,IDT,RECORD,IK)                               001544
00717    493*          IF(ITM .GE. IET) GO TO 5                                                001554
00721    494*          IF(IK .EQ. 1) GO TO 1010                                                001560
00723    495*          ISTREC = ISTREC + 1                                                     001565
00723    496*    C                                                                             001565
00723    497*    C  ***  AND THEN STORE DATA IN THE WORK - ARRAY  ***                         001565
00723    498*    C                                                                             001565
00724    499*          DO 82 I=1,NWORD                                                         001575
00727    500*       82  WORK(NWORK,I) = RECORD(I)                                            001575
00727    501*    C                                                                             001575
00731    502*          IF(ISTREC .GT. ITOTAL) GO TO 5                                          001577
```

```
00733          GO TO 420                                                                  001603
00733   C                                                                                 001603
00733   C    ******* E R R O R   M E S S A G E S  ********                                001603
00733   C                                                                                 001603
00734    1000 WRITE(6,660)                                                                001605
00736     660 FORMAT(1X,'ERROR - WE CANNOT FIND START TIME !!!')                          001611
00737         GO TO 5                                                                     001611
00740    1010 WRITE(6,665)                                                                001613
00742     665 FORMAT(1X,'ERROR - END OF FILE ENCOUNTERED !!!')                            001617
00743         GO TO 5                                                                     001617
00744     501 WRITE(6,670)                                                                001621
00746     670 FORMAT(1X,'ERROR ON READING INPUT FILE !!!')                                001625
00747         GO TO 5                                                                     001625
00750      1  WRITE(6,672) IFUNC                                                          001627
00753     672 FORMAT(1X,'WE DO NOT HAVE THE INFORMATION FOR THIS FUNCTION',I5)            001634
00754         GO TO 1051                                                                  001634
00755      43 WRITE(6,619)                                                                001636
00757     619 FORMAT(1X,'ERROR - TRY AGAIN',/                                             001642
00757        *,1X,'YOU HAVE INPUT A NON-EXISTING START TIME')                             001642
00760         GO TO 100                                                                   001642
00761    1055 WRITE(6,621)                                                                001644
00763     621 FORMAT(1X,'YOU ARE NOT ALLOWED TO INPUT THE SAME FUNCTION NUMBER A          001650
00763        *T THE SAME CHANNEL IN ONE PROCESS')                                         001650
00764         GO TO 1051                                                                  001650
00765      49 WRITE(6,6222)                                                               001652
00767    6222 FORMAT(1X,'NUMBER OF BITS HAS TO BE GREATER THAN 0')                        001656
00770         GO TO 9                                                                     001656
00771      79 WRITE(6,6233)                                                               001660
00773    6233 FORMAT(1X,'YOU HAVE INPUT THE INVALID VALUES FOR MAX.'/                     001664
00773        *,1X,'NUMBER AND MIN. NUMBER OF QUANTATIZATION')                             001664
00774         GO TO 27                                                                    001664
00775     196 WRITE(6,6991)                                                               001666
00777    6991 FORMAT(1X,'YOU HAVE INPUT AN INVALID NUMBER FOR INPUT UNIT')                001672
01000         GO TO 199                                                                   001672
01001     197 WRITE(6,6993)                                                               001674
01003    6993 FORMAT(1X,'YOU HAVE INPUT AN INVALID NUMBER FOR OUTPUT UNIT')               001700
01004         GO TO 199                                                                   001700
01005     201 WRITE(6,608)                                                                001702
01007     608 FORMAT(1X,'YOU HAVE INPUT AN UNACCEPTABLE POSITIVE SKEW TIME'/              001706
01007        *1X,'IT HAS TO BE LESS THAN FIVE POSITIVE SAMPLE RATES')                     001706
01010         GO TO 1050                                                                  001706
01011     202 WRITE(6,609)                                                                001710
01013     609 FORMAT(1X,'YOU HAVE INPUT AN UNACCEPTABLE NEGATIVE SKEW TIME'/              001714
01013        *1X,'IT HAS TO BE GREATER THAN FIVE NEGATIVE SAMPLE RATES')                  001714
01014         GO TO 1050                                                                  001714
01015     745 WRITE(6,6115)                                                               001716
01017    6115 FORMAT(1X,'ERROR - TRY AGAIN',/                                             001722
01017        *,1X,'YOU HAVE INPUT AN INVALID NUMBER OF FREQUENCY')                        001722
01020         GO TO 2                                                                     001722
01021         END                                                                         001725
END FOR
>
```

```
@FOR,S DAPSPT.NOISE/LEC
FOR S0E3-06/13/79-12:55:29 (0,)

   SUBROUTINE NOISE     ENTRY POINT 000140

   STORAGE USED: CODE(1) 000162; DATA(0) 000025; BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003   ARRAY  004230
    0004   HERTZ  000764
    0005   DISTC  000764
    0006   DEGRE  000764
    0007   TIMEX  000144

   EXTERNAL REFERENCES (BLOCK, NAME)

    0010   ZOR
    0011   SIN
    0012   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001     000056 126G      0001     000041 135L      0001     000107 155L      0001     000111 160L      0001     000047 200L
    0001     000051 205L      0005 R 000000 AMPL      0000 R 000003 COMPUT    0004 R 000000 FREQ      0000     000005 INJP$
    0000 I 000002 J         0000 I 000000 NMFREQ    0006 R 000000 PHASE     0007 R 000000 TLOCAL    0003 R 000000 WORK
    0000 R 000001 Y         0010 R 000000 ZOR


00100         C                                                                              000032
00100         C                                                                              000032
00100         C       SUBROUTINE NAME  -  NOISE                                              000032
00100         C                                                                              000032
00100         C            AUTHOR       :      M.T. NGUYEN                                   000032
00100         C            DATE         :      DECEMBER  1978                                000032
00100         C                                                                              000032
00100         C       FOR EX43 - ENGINEERING ANALYSIS DIVISION AT NASA/JSC                   000032
00100         C                                                                              000032
00100         C       PURPOSE :                                                              000032
00100         C                                                                              000032
00100         C            THIS FUNCTION IS USED TO CHANGE A MEASUREMENT BY MEANS OF         000032
00100         C            FREQUENCY, AMPLITUDE, PHASE ANGLE OR RANDOM NUMBER                000032
00100         C                                                                              000032
00100         C                                                                              000032
00101               SUBROUTINE NOISE(ICHAN,NUMFRE,STANDN,NTYPE,IDT,ICNT,NWORD,               000032
00101              *                 IBUF,NRANDM,ICALL,IRFOUT)                               000032
00103               PARAMETER IDTMIN=20,IDTSMA=5*IDTMIN                                      000032
00104               PARAMETER NWORK1=2*IDTSMA/IDTMIN+1                                       000032
00105               COMMON /ARRAY/ WORK(NWORK1,200)                                          000032
00106               COMMON /HERTZ/ FREQ(100,5)                                               000032
00107               COMMON /DISTC/ AMPL(100,5)                                               000032
00110               COMMON /DEGRE/ PHASE(100,5)                                              000032
```

```
00111          COMMON /TIMEX/ TLOCAL(100)                                                       000032
00112          DIMENSION NTYPE(100),STANDN(100),NRANDM(100),NUMFRE(100)                         000032
00113          NMFREQ = NUMFRE(ICNT)                                                            000032
00113    C                                                                                      000032
00113    C  ***  TEST TO SEE IF THIS IS THE FIRST COMPUTED INPUT DATA RECORD  ***               000032
00113    C                                                                                      000032
00114          IF(IBUF) 130,130,135                                                             000034
00114    C                                                                                      000034
00114    C  ***  INITIALIZE TIME OF NOISE  ***                                                  000034
00114    C                                                                                      000034
00117      130 TLOCAL(ICNT) = 0.                                                                000037
00117    C                                                                                      000037
00117    C  ***  ADD NOISE ON THE WORKED MEASUREMENT  ***                                       000037
00117    C                                                                                      000037
00117    C     ICALL - FLAG TO DETERMINE WHETHER THE PRESENT INPUT MEASUREMENT                  000037
00117    C             HAS BEEN APPLIED BY ANOTHER FUNCTION                                     000037
00117    C                                                                                      000037
00120      135 IF(ICALL .LE. 0) GO TO 200                                                       000041
00122          Y = WORK(IRFOUT,NWORD+ICNT)                                                      000043
00123          GO TO 205                                                                        000045
00123    C                                                                                      000045
00124      200 Y = WORK(IRFOUT,ICHAN)                                                           000047
00125      205 DO 10 J = 1,NMFREQ                                                               000051
00130          Y = Y + AMPL(ICNT,J)*SIN(FREQ(ICNT,J)                                            000056
00130         1  *TLOCAL(ICNT)+PHASE(ICNT,J))                                                   000056
00131      10  CONTINUE                                                                         000070
00131    C                                                                                      000070
00131    C  ***  ARE WE ADDING RANDOM NOISE ON THAT MEASUREMENT  ***                            000070
00131    C                                                                                      000070
00133          IF(NRANDM(ICNT) .LE. 0) GO TO 155                                                000070
00135          COMPUT = ZOR(NTYPE(ICNT))                                                        000073
00136          Y = Y + STANDN(ICNT) * COMPUT                                                    000101
00137          WORK(IRFOUT,NWORD+ICNT) = Y                                                      000104
00140          GO TO 160                                                                        000105
00140    C                                                                                      000105
00140    C     RANDOM NOISE WILL NOT BE ADDED TO THE MEASUREMENT IN WHICH                       000105
00140    C     THE RANDOM NUMBER IS EQUAL ZERO                                                  000105
00140    C                                                                                      000105
00141      155 WORK(IRFOUT,NWORD+ICNT) = Y                                                      000107
00141    C                                                                                      000107
00141    C     INCREMENT TIME WHEN NEXT DATA RECORD BROUGHT IN                                  000107
00141    C                                                                                      000107
00142      160 TLOCAL(ICNT) = TLOCAL(ICNT)+ FLOAT(IDT)                                          000111
00143       5  CONTINUE                                                                         000115
00144          ICALL = 1                                                                        000115
00145          RETURN                                                                           000117
00146          END                                                                              000161
END FOR
>
```

```
@FOR,S DAPSPT.SKEW/LEC
FOR S0E3-06/13/79-12:57:21 (0,)

   SUBROUTINE SKEW      ENTRY POINT 000117

   STORAGE USED: CODE(1) 000132; DATA(0) 000025; BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003   ARRAY  004230

   EXTERNAL REFERENCES (BLOCK, NAME)

    0004   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0000 R 000005 DELT      0000 R 000004 DTMTO     0000 R 000003 DT1MT     0000 I 000002 IDTMTO    0000 I 000001 IDT1MT
    0000   000012 INJP$     0000 I 000000 ITIME     0003 R 000000 WORK      0000 R 000006 Y


00100       1*     C                                                                   000014
00100       2*     C                                                                   000014
00100       3*     C        SUBROUTINE NAME   =   SKEW                                 000014
00100       4*     C                                                                   000014
00100       5*     C           AUTHOR    :     M.T. NGUYEN                             000014
00100       6*     C           DATE      :     NOVEMBER  1978                          000014
00100       7*     C                                                                   000014
00100       8*     C                                                                   000014
00100       9*     C        FOR EX43 - ENGINEERING ANALYSIS DIVISION AT NASA/JSC       000014
00100      10*     C                                                                   000014
00100      11*     C                                                                   000014
00100      12*     C        PURPOSE :                                                  000014
00100      13*     C                                                                   000014
00100      14*     C           THIS FUNCTION IS USED TO INTRODUCE A TIME-SHIFT, EITHER IN A   000014
00100      15*     C           POSITIVE (RIGHT) DIRECTION OR NEGATIVE (LEFT) DIRECTION, FOR   000014
00100      16*     C           A GIVEN MEASUREMENT, WITH RESPECT TO A NORMAL (UNSKEWED)       000014
00100      17*     C           MEASUREMENT                                             000014
00100      18*     C                                                                   000014
00100      19*     C                                                                   000014
00101      20*              SUBROUTINE SKEW(IDT,ICHAN,NWORD,ICNT,IFIRST,IRFOUT,IDTSAB,ICALL)   000014
00103      21*              PARAMETER IDTMIN=20,IDTSMA=5*IDTMIN                        000014
00104      22*              PARAMETER NWORK1=2*IDTSMA/IDTMIN+1                         000014
00105      23*              COMMON /ARRAY/ WORK(NWORK1,200)                            000014
00105      24*     C                                                                   000014
00106      25*              ITIME = IFIX(WORK(IRFOUT,1)*1000.+.05) - IDTSAB            000014
00107      26*              IDT1MT = IFIX(WORK(IFIRST+1,1)*1000.+.05)                  000027
00110      27*              IDTMTO = IFIX(WORK(IFIRST,1)*1000.+.05)                    000041
00111      28*              DT1MT = FLOAT(IDT1MT - ITIME)                              000053
00112      29*              DTMTO = FLOAT(ITIME - IDTMTO)                              000057
00113      30*              DELT = FLOAT(IDT)                                          000063
00113      31*     C                                                                   000063
```

```
C     ***  PERFORM A LINEAR INTERPOLATION  ***                          000063
C                                                                       000063
      Y = 1./DELT*(WORK(IFIRST,ICHAN)*DT1MT+WORK(IFIRST+1,ICHAN)*DTMT0) 000066
      WORK(IRFOUT,NWORD+ICNT) = Y                                       000077
      ICALL = 1                                                         000100
       RETURN                                                           000102
      END                                                               000131
END FOR
>
```

```
@FOR,$ DAPSPT.QUAMTM/LEC
FOR S0E3-06/13/79-12:58:38 (0,

   SUBROUTINE QUANTM    ENTRY P INT 00013

   STORAGE USED: CODE(1) 000143; DATA(0) 000017; BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003   ARRAY  004230

   EXTERNAL REFERENCES (BLOCK, NAME)

    0004   XPII
    0005   MERR38

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000111 100L      0001   000104 105L      0001   000107 110L      0001   000032 200L      0001   000034 205L
    0000 R 000001 BITS      0000 R 000002 DELTA     0000 I 000003 INCR      0000   000006 INJP$     0003 R 000000 WORK
    0000 R 000000 Y
```

```
00100      1*      C                                                                    000023
00100      2*      C                                                                    000023
00100      3*      C        SUBROUTINE NAME      *      QUAMTM                          000023
00100      4*      C                                                                    000023
00100      5*      C          AUTHOR      :     M.T. NGUYEN                           000023
00100      6*      C          DATE        :     DECEMBER  1978                        000023
00100      7*      C                                                                    000023
00100      8*      C                                                                    000023
00100      9*      C        FOR EX43 - ENGINEERING ANALYSIS DIVISION AT NASA/JSC        000023
00100     10*      C                                                                    000023
00100     11*      C                                                                    000023
00100     12*      C        PURPOSE :                                                   000023
00100     13*      C                                                                    000023
00100     14*      C           THIS FUNCTION IS USED TO CHANGE A MEASUREMENT FROM A     000023
00100     15*      C           CONTINUOS FORM TO ONE WITH DISCRETE STEPS TO SIMULATE    000023
00100     16*      C           DIGITAL DATA .                                           000023
00100     17*      C                                                                    000023
00100     18*      C                                                                    000023
00101     19*            SUBROUTINE QUANTM(IBIT,QMAX,QMIN,ICNT,NWORD,ICALL,ICHAN,IRFOUT)  000023
00103     20*            PARAMETER IDTMIN=20,IDTSMA=5*IDTMIN                              000023
00104     21*            PARAMETER NWORK1=2*IDTSMA/IDTMIN+1                               000023
00105     22*            COMMON /ARRAY/ WORK(NWORK1,200)                                  000023
00106     23*            DIMENSION IBIT(100),QMAX(100),QMIN(100)                          000023
00106     24*      C                                                                      000023
00106     25*      C     ICALL - FLAG TO DETERMINE WHETHER THE PRESENT INPUT CHANNEL HAS  000023
00106     26*      C             BEEN APPLIED BY ANOTHER FUNCTION                         000023
00106     27*      C                                                                      000023
00107     28*            IF(ICALL .LE. 0) GO TO 200                                       000023
00111     29*            Y = WORK(IRFOUT,NWORD+ICNT)                                      000026
```

```
00112          GO TO 205                                                          000030
00113     200 Y = WORK(IRFOUT,ICHAN)                                              000032
00114     205 IF(Y .LT. QMIN(ICNT)) GO TO 105                                     000034
00116         IF(Y .GE. QMAX(ICNT)) GO TO 110                                     000037
00120         BITS = FLOAT(2**IBIT(ICNT))                                         000043
00120   C                                                                         000043
00120   C   DELTA - NUMBER OF STEPS INTO WHICH MEASUREMENT RANGE IS DIVIDED       000043
00120   C                                                                         000043
00120   C   INCR  -  NUMBER OF INCREMENTS OF SIZE DELTA INTO WHICH A              000043
00120   C            MEASUREMENT VALUE WILL BE DIVIDED                            000043
00120   C                                                                         000043
00121         DELTA = (QMAX(ICNT) - QMIN(ICNT))/BITS                              000054
00122         INCR = INT((Y - (QMIN(ICNT) + DELTA/2.))/DELTA)                     000060
00123         Y = QMIN(ICNT) + FLOAT(INCR + 1) * DELTA                            000073
00124         GO TO 100                                                           000102
00124   C                                                                         000102
00125     105 Y = QMIN(ICNT)                                                      000104
00126         GO TO 100                                                           000105
00127     110 Y = QMAX(ICNT)                                                      000107
00130     100 WORK(IRFOUT,NWORD+ICNT) = Y                                         000111
00131       5 CONTINUE                                                            000112
00132         ICALL = 1                                                           000112
00133         RETURN                                                              000114
00134         END                                                                 000142
END FOR
>
```

```
@FOR,S DAPSPT.CONVER
FOR S0E3-06/13/79-13:00:04 (1,)

   SUBROUTINE CONVER    ENTRY POINT 000074

   STORAGE USED: CODE(1) 000124; DATA(0) 000050; BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   SSRD
    0004   NWDU$
    0005   NIO2$
    0006   MERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000064 10L       0000   000004 110F      0000   000022 115F      0001   000042 50L       0001   000052 55L
    0000 I 000003 IFST      0000 I 000002 ILST      0000   000043 INJP$     0000 I 000000 NK        0000 R 000001 P


00101      1*          SUBROUTINE CONVER(IUNT,NWORD,ITM,IDT,RECORD,IK)                                  000000
00101      2*     C....THIS SUBROUTIINE READS A SSFS TYPE CALCOMP                                       000000
00101      3*     C      FILE AND RETURNS TIME, DELTA TME, AND                                         000000
00101      4*     C       A DATA RECORD                                                                000000
00103      5*          DIMENSION RECORD(150)                                                           000000
00104      6*        110 FORMAT(//,1X,'EOF ENCOUNTERED READING UNIT ',I4,                              000000
00104      7*           1' IN DAP.FACCES AFTER TIME=',I10,)                                           000000
00105      8*        115 FORMAT(//,1X,'PARITY ERROR ON READING UNIT ',I4,                              000000
00105      9*           1' IN DAP.FACCES AFTER TIME=',I10,)                                           000000
00106     10*           NK = 1                                                                        000000
00107     11*           CALL SSRD(IUNT,NK,NWORD,P,IK,RECORD)                                          000000
00110     12*           IF( IK .EQ. 1 ) GO TO 55                                                      000002
00112     13*           IF( IK .GT. 1 ) GO TO 50                                                      000012
00114     14*           ITM=IFIX( RECORD(1)*1000. + .05 )                                             000015
00115     15*           ILST = IFST                                                                   000021
00116     16*           IFST = ITM                                                                    000033
00117     17*           IDT = IFST - ILST                                                             000035
00120     18*           GO TO 10                                                                      000036
00121     19*        50 WRITE (6,115) IUNT,ITM                                                        000040
00125     20*           GO TO 10                                                                      000042
00126     21*        55 WRITE(6,110)IUNT, ITM                                                         000050
00132     22*           ITM=ITM+IDT                                                                   000052
00133     23*        10 RETURN                                                                        000060
00134     24*           END                                                                           000064
END FOR                                                                                             000123
>
```

```
@FOR,S DAPSPT.SSRD
FOR S0E3-06/13/79-17:00:54 (0,)

   SUBROUTINE SSRD      ENTRY POINT 000054

   STORAGE USED: CODE(1) 000067; DATA(0) 000017; BLANK COMMON(2) 00000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   NRBU$
    0004   NIO1$
    0005   NIO2$
    0006   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000024 114G      0001   000036 180L      0001   000042 190L      0000 I 000004 I          0000 I 000001 IKP
    0000 I 000002 IKPT      0000   000006 INJP$     0000 I 000003 LTH       0000 R 000000 XBUFF


00101      1%            SUBROUTINE SSRD(IUNT,NW,MW,IP,IK,BUFF)                                    000007
00101      2%      C                                                                               000007
00101      3%      C   IUNT=UNIT NUMBER                                                           000007
00101      4%      C   NW=BEGINING BUFFER INDEX                                                   000007
00101      5%      C   MW=ENDING BUFFER INDEX                                                      000007
00101      6%      C   IP=PARITY INDICATOR                                                         000007
00101      7%      C   IK= 0, READ OK                                                              000007
00101      8%      C       1, EOF ENCOUNTERED                                                      000007
00101      9%      C       2, PARITY ERROR                                                         000007
00101     10%      C                                                                               000007
00103     11%           DIMENSION BUFF(1)                                                          000007
00104     12%           DATA XBUFF /3HEOF /                                                        000007
00106     13%           IKP=0                                                                      000007
00107     14%           IKPT=0                                                                     000010
00110     15%       100 CONTINUE                                                                   000011
00111     16%           LTH=MW-NW+1                                                                000011
00112     17%           READ(IUNT) (BUFF(I),I=NW,MW)                                               000015
00120     18%           IF( BUFF(1) -XBUFF ) 180,170,180                                           000027
00123     19%       170 IK=1                                                                       000032
00124     20%           GO TO 190                                                                  000034
00125     21%       180 IF( IK .NE. 2 ) IK=0                                                       000036
00127     22%       190 RETURN                                                                     000042
00130     23%           END                                                                        000066
END FOR
>
```

```
@PRT,S DAPSPT.MAPNOISE
FURPUR 87R3A       E33 SL73R1 06/13/79 13:02:23



EX32-L72330*DAPSPT(1).MAPNOISE
     1        LIB EX32-L72330*DAPSPT.
     2        LIB MSC*LOCALIB.
     3          IN NOISNG,SKEW,NOISE,CONVER,SSRD,QUANTN
     4          NOT NOISNG/TEST,NOISE/TEST,SKEW/TEST
     5          NOT QUANTN/TEST
     6         END
>
```

APPENDIX B

A TEST CASE

```
@<GT ILLIN RUN
 PLEASE ENTER THE INPUT & OUTPUT LOGICAL UNITS
>8,9
 THE NUMBER OF MEASUREMENTS IN EACH DATA RECORD ARE   35
 PLEASE ENTER START AND END TIME IN MILISECOND INTEGER
>3181020,3181560
 PLEASE ENTER CHANNEL NUMBER
>2

 PLEASE ENTER THE NUMERICAL VALUE REPRESENTING THE FOLLOWING FUNCTIONS
          1-SKEW          2-NOISE
          3-LOCATION      4-BIAS
          5-ALIGN         6-QUANT
                  7-SAMPLE RATE

>1
 PLEASE ENTER SKEW TIME IN MILLISECOND INTEGER
>70
 CURRENT MAXIMUM VALUE OF IDTS IS   70


 PLEASE ENTER THE NUMERICAL VALUE REPRESENTING THE FOLLOWING FUNCTIONS
          1-SKEW          2-NOISE
          3-LOCATION      4-BIAS
          5-ALIGN         6-QUANT
                  7-SAMPLE RATE

>6
 PLEASE ENTER NUMBER OF BITS
>4
 PLEASE ENTER THE UPPER & LOWER LIMITS OF RANGE
 FOR WHICH THIS CHANNEL IS TO BE QUANTATIZED
>4 051.1.870

          NUMBER OF BITS  4     MAX  NUMBER        4 05100     MIN  NUMBER        1 87000

 PLEASE ENTER THE NUMERICAL VALUE REPRESENTING THE FOLLOWING FUNCTIONS
          1-SKEW          2-NOISE
          3-LOCATION      4-BIAS
          5-ALIGN         6-QUANT
                  7-SAMPLE RATE

>0
 PLEASE ENTER CHANNEL NUMBER
>0

 PLEASE ENTER THE NUMERICAL VALUE REPRESENTING THE FOLLOWING FUNCTIONS
          1-SKEW          2-NOISE
          3-LOCATION      4-BIAS
          5-ALIGN         6-QUANT
                  7-SAMPLE RATE

>1
 PLEASE ENTER SKEW TIME IN MILLISECOND INTEGER
>90
 CURRENT MAXIMUM VALUE OF IDTS IS   90


 PLEASE ENTER THE NUMERICAL VALUE REPRESENTING THE FOLLOWING FUNCTIONS
          1-SKEW          2-NOISE
          3-LOCATION      4-BIAS
          5-ALIGN         6-QUANT
                  7-SAMPLE RATE
```

```
>0
 PLEASE ENTER CHANNEL NUMBER
>15

 PLEASE ENTER THE NUMERICAL VALUE REPRESENTING THE FOLLOWING FUNCTIONS
          1-SKEW          2-NOISE
          3-LOCATION      4-BIAS
          5-ALIGN         6-QUANT
                  7-SAMPLE RATE

>2
 PLEASE ENTER NUMBER OF FREQUENCIES
>3
 PLEASE ENTER VALUES OF FREQUENCY IN HERTZ
                        AMPLITUDE
                        PHASE IN DEGREE

>3.45. 030.30
>4.56. 040.40.
>5.67. 050.50
     FREQUENCY        21.67699     AMPLITUDE          03000     PHASE          52360
     FREQUENCY        28.65132     AMPLITUDE          04000     PHASE          69813
     FREQUENCY        35.62566     AMPLITUDE          05000     PHASE          87266
 PLEASE ENTER THE VALUES REPRESENTING THE VARIABLES AS
          RANDOM NUMBER WHERE IT IS 5-7 DIGIT INTEGER. LAST DIGIT IS ODD
          TYPE OF RANDOM WHERE 0 IS UNIFORM AND 1 IS NORMAL
          AMPLITUDE OF STANDARD DEVIATION
>42335.1..020

 RANDOM NUMBER   42335  TYPE 1  STNDRD DEV      02000


 PLEASE ENTER THE NUMERICAL VALUE REPRESENTING THE FOLLOWING FUNCTIONS
          1-SKEW          2-NOISE
          3-LOCATION      4-BIAS
          5-ALIGN         6-QUANT
                  7-SAMPLE RATE

>0
 PLEASE ENTER CHANNEL NUMBER
>24

 PLEASE ENTER THE NUMERICAL VALUE REPRESENTING THE FOLLOWING FUNCTIONS
          1-SKEW          2-NOISE
          3-LOCATION      4-BIAS
          5-ALIGN         6-QUANT
                  7-SAMPLE RATE

>6
 PLEASE ENTER NUMBER OF BITS
>8
 PLEASE ENTER THE UPPER & LOWER LIMITS OF RANGE
 FOR WHICH THIS CHANNEL IS TO BE QUANTATIZED
>1.005. 00230
          NUMBER OF BITS  8     MAX. NUMBER        1.00500     MIN. NUMBER          00230


 PLEASE ENTER THE NUMERICAL VALUE REPRESENTING THE FOLLOWING FUNCTIONS
          1-SKEW          2-NOISE
          3-LOCATION      4-BIAS
          5-ALIGN         6-QUANT
                  7-SAMPLE RATE
```

```
PLEASE ENTER CHANNEL NUMBER
>3

PLEASE ENTER THE NUMERICAL VALUE REPRESENTING THE FOLLOWING FUNCTIONS
        1-SKEW          2-NOISE
        3-LOCATION      4-BIAS
        5-ALIGN         6-QUANT
                7-SAMPLE RATE

>1
PLEASE ENTER SKEW TIME IN MILLISECOND INTEGER
>-75
CURRENT MINIMUM VALUE OF IDTS IS  -75

CURRENT MAXIMUM VALUE OF IDTS IS   90

PLEASE ENTER THE NUMERICAL VALUE REPRESENTING THE FOLLOWING FUNCTIONS
        1-SKEW          2-NOISE
        3-LOCATION      4-BIAS
        5-ALIGN         6-QUANT
                7-SAMPLE RATE

>2
PLEASE ENTER NUMBER OF FREQUENCIES
>5
PLEASE ENTER VALUES OF FREQUENCY IN HERTZ
                        AMPLITUDE
                        PHASE IN DEGREE
>1.00,.010,10
>2.00,.020,20
>3.00,.030,30
>4.00,.040,40
>5.00,.050,50
    FREQUENCY       6 28319     AMPLITUDE       01000     PHASE       17453
    FREQUENCY      12 56637     AMPLITUDE       02000     PHASE       34907
    FREQUENCY      18 84956     AMPLITUDE       03000     PHASE       52360
    FREQUENCY      25 13274     AMPLITUDE       04000     PHASE       69813
    FREQUENCY      31 41593     AMPLITUDE       05000     PHASE       87266
PLEASE ENTER THE VALUES REPRESENTING THE VARIABLES AS
        RANDOM NUMBER WHERE IT IS 5-7 DIGIT INTEGER, LAST DIGIT IS ODD
        TYPE OF RANDOM WHERE 0 IS UNIFORM AND 1 IS NORMAL
        AMPLITUDE OF STANDARD DEVIATION
>51237,0,.035

RANDOM NUMBER   51237  TYPE 0  STNDRD DEV      03500

PLEASE ENTER THE NUMERICAL VALUE REPRESENTING THE FOLLOWING FUNCTIONS
        1-SKEW          2-NOISE
        3-LOCATION      4-BIAS
        5-ALIGN         6-QUANT
                7-SAMPLE RATE

>6
PLEASE ENTER NUMBER OF BITS
>12
PLEASE ENTER THE UPPER & LOWER LIMITS OF RANGE
FOR WHICH THIS CHANNEL IS TO BE QUANTATIZED
>4 03591,1 06433

        NUMBER OF BITS 12     MAX. NUMBER       4 03591     MIN. NUMBER       1 06433
```

```
 PLEASE ENTER THE NUMERICAL VALUE REPRESENTING THE FOLLOWING FUNCTIONS
          1-SKEW          2-NOISE
          3-LOCATION      4-BIAS
          5-ALIGN         6-QUANT
                7-SAMPLE RATE

>3
 PLEASE ENTER CHANNEL NUMBER
>5

 PLEASE ENTER THE NUMERICAL VALUE REPRESENTING THE FOLLOWING FUNCTIONS
          1-SKEW          2-NOISE
          3-LOCATION      4-BIAS
          5-ALIGN         6-QUANT
                7-SAMPLE RATE

>1
 PLEASE ENTER SKEW TIME IN MILLISECOND INTEGER
>15
 CURRENT MAXIMUM VALUE OF IDTS IS    90


 PLEASE ENTER THE NUMERICAL VALUE REPRESENTING THE FOLLOWING FUNCTIONS
          1-SKEW          2-NOISE
          3-LOCATION      4-BIAS
          5-ALIGN         6-QUANT
                7-SAMPLE RATE

>2
 PLEASE ENTER NUMBER OF FREQUENCIES
>2
 PLEASE ENTER VALUES OF FREQUENCY IN HERTZ
                        AMPLITUDE
                        PHASE IN DEGREE
>5.074, .070,70.
>3.254, .050,50.
     FREQUENCY        31.88000      AMPLITUDE          .07000     PHASE        1.22173
     FREQUENCY        20.44640      AMPLITUDE          .05000     PHASE         .87266
 PLEASE ENTER THE VALUES REPRESENTING THE VARIABLES AS
          RANDOM NUMBER WHERE IT IS 5-7 DIGIT INTEGER, LAST DIGIT IS ODD
          TYPE OF RANDOM WHERE 0 IS UNIFORM AND 1 IS NORMAL
          AMPLITUDE OF STANDARD DEVIATION
>0,0,.030

 PLEASE ENTER THE NUMERICAL VALUE REPRESENTING THE FOLLOWING FUNCTIONS
          1-SKEW          2-NOISE
          3-LOCATION      4-BIAS
          5-ALIGN         6-QUANT
                7-SAMPLE RATE

>0
 PLEASE ENTER CHANNEL NUMBER
>20

 PLEASE ENTER THE NUMERICAL VALUE REPRESENTING THE FOLLOWING FUNCTIONS
          1-SKEW          2-NOISE
          3-LOCATION      4-BIAS
          5-ALIGN         6-QUANT
                7-SAMPLE RATE

>2
 PLEASE ENTER NUMBER OF FREQUENCIES
>4
```

```
PLEASE ENTER VALUES OF FREQUENCY IN HERTZ
                       AMPLITUDE
                       PHASE IN DEGREE
>3.05  050.50
>2.05  040.40
>1.05  030.30
>5.05  020.20
    FREQUENCY        19.16372     AMPLITUDE          .05000     PHASE          .87266
    FREQUENCY        12.88053     AMPLITUDE          .04000     PHASE          .69813
    FREQUENCY         6.50734     AMPLITUDE          .03000     PHASE          .52360
    FREQUENCY        31.73009     AMPLITUDE          .02000     PHASE          .34907
PLEASE ENTER THE VALUES REPRESENTING THE VARIABLES AS
         RANDOM NUMBER WHERE IT IS 5-7 DIGIT INTEGER, LAST DIGIT IS ODD
         TYPE OF RANDOM WHERE 0 IS UNIFORM AND 1 IS NORMAL
         AMPLITUDE OF STANDARD DEVIATION
>23651.0,.055

RANDOM NUMBER   23651  TYPE 0  STNDRD DEV      .05500


PLEASE ENTER THE NUMERICAL VALUE REPRESENTING THE FOLLOWING FUNCTIONS
         1-SKEW           2-NOISE
         3-LOCATION       4-BIAS
         5-ALIGN.         6-QUANT.
                 7-SAMPLE RATE

>6
PLEASE ENTER NUMBER OF BITS
>16
PLEASE ENTER THE UPPER & LOWER LIMITS OF RANGE
FOR WHICH THIS CHANNEL IS TO BE QUANTATIZED
> .14562,-.03492

         NUMBER OF BITS 16     MAX. NUMBER          .14562     MIN. NUMBER        -.03492


PLEASE ENTER THE NUMERICAL VALUE REPRESENTING THE FOLLOWING FUNCTIONS
         1-SKEW           2-NOISE
         3-LOCATION       4-BIAS
         5-ALIGN.         6-QUANT.
                 7-SAMPLE RATE

>0
PLEASE ENTER CHANNEL NUMBER
>0
```

NUMBER OF MODIFIED CHANNELS ARE 7

OUTPUT RECORD NUMBER = 1

3181.18000	2.73180	.08474	.13600	3.62632	.65931	-.21468	.52002
212.78867	2.50503	1.28400	-15.46766	-23.73484	2.12004	3.00397	-12.14671
2.34013	.08300	-.00665	-.71206	1.51947	.93007	-.19636	-.04915
-.53072	179.80400	-.00030	.01786	-.00002	1.18050	.62036	-.72000
.00000	-.00000	-1.56762	2.82419	218.28074	3.18080	.00236	1.06433
3.73040	-.03400						

OUTPUT RECORD NUMBER = 2

3181.13000	2.71070	.00003	.13600	3.62638	.65931	-.21571	.52010
212.01043	2.61000	1.27496	-15.46967	-23.74629	2.07247	10.82890	-6.30254
1.20321	.00053	-.56467	-.00754	1.51626	.92787	-.19955	-.03740
-.54107	179.68012	-.00007	.01790	-.01506	2.08594	.48966	-.72000

.30000	-.50500	-1.56618	2.68419	212.30360	10.00500	.00236	1.06433
3.61311	-.03492						

OUTPUT RECORD NUMBER = 3

3181.16000	2.60682	.00218	.14459	3.63969	.65931	-.21643	.52023
213.05000	2.63864	1.27430	-15.45083	-23.75512	2.06925	19.76704	4.23989
-.05944	.69252	-.51726	-.69739	1.51311	.92787	-.19978	-.01785
-.54695	179.06326	-.00024	.01811	-.00492	2.23099	.15181	-.70000
.31200	-.47000	-1.56618	2.68788	212.49085	19.75743	.00236	1.06433
3.73500	-.03492						

OUTPUT RECORD NUMBER = 4

3181.17000	2.65423	.00132	.15593	3.66197	.65931	-.21112	.52035
213.10135	2.64603	1.27507	-15.44057	-23.76210	2.07380	9.11003	4.46007
-1.71887	.69251	-.44734	-.69716	1.51133	.93007	-.20412	.00558
-.54701	179.69740	-.00021	.01815	-.00020	1.96409	.00628	-.68000
.20000	-.45000	-1.57066	2.68788	212.61538	9.12553	.00628	1.06433
3.62740	-.03492						

OUTPUT RECORD NUMBER = 5

3181.20001	2.62163	.07047	.16727	3.68425	.65931	-.20580	.52048
213.33181	2.64852	1.27747	-15.43031	-23.76907	2.08476	-8.07870	1.26051
-8.17724	.69251	-.44117	-.69714	1.50055	.93447	-.21063	.01337
-.54106	179.73154	-.00019	.01815	-.03014	1.37794	.00441	-.66000
.27000	-.43000	-1.59046	2.68788	212.73090	-8.13788	.01411	1.06433
3.70042	-.03492						

OUTPUT RECORD NUMBER = 6

3181.20000	2.58903	.05061	.17061	3.70665	.65931	-.20048	.52067
213.47403	2.65253	1.27772	-15.42005	-23.77706	2.08597	-5.67228	-4.18259
-6.00544	.69251	-.40620	-.69703	1.5[illegible]7	.93667	-.20409	.00160
-.52366	179.70568	-.00019	.01815	-.00011	1.09002	.00349	-.64000
.25000	-.41000	-1.60778	2.68788	212.84066	-5.59637	.00236	1.06433
3.63060	-.03492						

OUTPUT RECORD NUMBER = 7

3181.23000	2.60571	.07330	.18145	3.74218	.65931	-.19453	.52096
213.60053	2.67273	1.27577	-15.40079	-23.78300	2.07643	3.20506	-10.37054
-17.03360	.69251	-.40003	-.69701	1.50617	.93667	-.20462	-.01798
-.50741	179.70000	-.00018	.01815	-.00015	1.40079	.00451	-.59600
.24000	-.39000	-1.60778	2.68788	212.90066	3.30276	.00236	1.06433
3.71303	-.03492						

OUTPUT RECORD NUMBER = 8

3181.20001	2.63100	.00140	.18145	3.78873	.65831	-.17052	.52126
213.77023	2.[illegible]	1.[illegible]	-15.30063	-23.79000	2.08846	1.37510	3.72423
-14.[illegible]	.[illegible]	-.[illegible]	-.[illegible]	1.58780	.94327	-.20359	-.02067
-.51400	179.[illegible]	-.00018	.01831	.06800	1.28483	.00872	-.51000
.20400	-.37000	-1.61250	2.68788	213.12118	1.36548	.00236	1.06433
3.[illegible]	-.03492						

OUTPUT RECORD NUMBER = 9

3181.20000	2.65047	.10000	.18146	3.83120	.65931	-.15061	.52155
213.[illegible]	2.[illegible]	1.00070	-15.30007	-23.7[illegible]	2.14060	-4.12530	24.57000
-6.[illegible]	.[illegible]	-.30304	-.69677	1.60081	.96447	-.19567	-.03340
-.54000	179.00010	-.00010	.01860	1.68540	1.50134	.05974	-.43600
.30400	-.35000	-1.[illegible]	2.66166	213.00160	-4.10000	.00236	1.06433

3.76372	-.03492

OUTPUT RECORD NUMBER = 10

3181.29000	2.68184	.12772	.18145	3.07500	.65931	-.14050	.52109
214.07597	2.72300	1.29030	-15.37001	-23.80395	2.19300	-5.50030	5.72[illegible]
.45837	.70072	-.31777	-.00640	1.51133	.97047	-.19527	-.0[illegible]
-.56416	179.90224	-.00019	.01881	2.40646	1.97437	.00871	-.3[illegible]
.44400	-.33000	-1.61730	2.55156	213.40332	-5.55967	.00236	1.06[illegible]
3.79735	-.03492						

OUTPUT RECORD NUMBER = 11

3181.30001	2.67332	.13043	.22300	3.86604	.65931	-.12245	.52160
214.00000	2.75604	1.31103	-15.36876	-23.81003	2.25330	-5.50030	-20.35606
7.70023	.05604	-.27047	-.69124	1.51290	.98526	-.19920	-.01784
-.55004	179.93630	-.00019	.01906	2.33825	2.76230	.35250	-.35300
.40000	-.30000	-1.60650	2.56156	213.55017	-5.42448	.00236	1.06433
3.85244	-.03492						

OUTPUT RECORD NUMBER = 12

3181.34000	2.66801	.13042	.28068	3.82238	.65931	-.10366	.52150
214.30415	2.70004	1.31000	-15.35050	-23.81790	2.29517	-4.46907	-9.79758
7.56304	.05603	-.18820	-.69098	1.51331	.99266	-.19995	-.01010
-.52000	179.97052	-.00019	.01915	-.13946	3.19485	.55091	-.41300
.40000	-.20000	-1.60218	2.60780	213.70087	-4.43217	.00236	1.06433
3.87200	-.03492						

OUTPUT RECORD NUMBER = 13

3181.35000	2.64431	.13048	.33738	3.77782	.65931	-.08480	.52141
214.50004	2.70613	1.31937	-15.34749	-23.82319	2.20005	-3.09397	22.97561
4.81506	.04002	-.11240	-.66427	1.51365	.99266	-.20046	-.01789
-.53001	180.00407	-.00019	.01915	-2.49506	2.40576	.79078	-.47300
.40000	-.20000	-1.60218	2.60780	213.85157	-3.10360	.00236	1.06433
3.80067	-.03400						

OUTPUT RECORD NUMBER = 14

3181.38000	2.62000	.13041	.39400	3.73320	.65931	-.06610	.52144
214.64000	2.83005	1.30811	-15.33475	-23.82453	2.33005	2.40642	20.41871
1.71007	.00113	-.07787	-.66487	1.51300	1.00406	-.20720	-.02527
-.55307	180.03001	-.00019	.01915	-.00032	1.23440	.74932	-.53300
.40000	-.10000	-1.58004	2.60780	214.00145	2.42192	.00236	1.06433
3.83001	-.03400						

OUTPUT RECORD NUMBER = 15

3181.30000	2.62003	.10730	.45527	3.72001	.65931	-.04674	.52106
214.77000	2.80318	1.34018	-15.30001	-23.82507	2.43001	0.20100	24.75178
-1.40000	.00033	-.07738	-.63001	1.51304	1.00000	-.21104	-.02306
-.[illegible]	180.07000	-.00019	.01915	.09311	.43411	.05002	-.50050
.40000	-.14000	-1.51618	2.60780	214.14001	0.82274	.00236	1.06433
3.74001	-.03400						

OUTPUT RECORD NUMBER = 16

3181.41000	2.65440	.21000	.44706	3.75100	.65031	-.01051	.52100
214.81036	2.00004	1.30004	-15.30000	-23.02700	2.65361	0.46300	0.53707
-3.00003	.00004	-.07733	-.63307	1.40777	1.05300	-.21481	-.00000
-.00424	180.10700	-.00019	.01003	4.11003	.94646	.01269	-.57550
.53000	-.11000	-1.40164	2.60700	214.20010	0.52971	.00236	1.06433
3.81000	-.03400						

OUTPUT RECORD NUMBER = 17

3181 44000	2 66888	25489	47064	3 77337	05931	01371	52210
215 04104	2 90047	1 39636	-15 89858	-23 82854	8 72804	7 39116	-11 86023
-5 50030	05558	- 06095	- 64940	1 46251	1 00505	- 21761	- 02185
- 58100	100 14123	- 00017	01935	4 12161	2 00082	- 18815	- 58550
57600	- 00600	-1 46304	2 68788	214 43619	7 42806	00236	1 06433
3 73130	- 03498						

OUTPUT RECORD NUMBER = 18

3181 46000	2 08336	29776	49332	3 79565	05931	04304	52206
215 10007	3 00093	1 41036	-15 28378	-23 82988	8 00548	6 87540	-12 40848
-8 53707	1 00001	01322	- 64307	1 43784	1 10065	- 20054	- 02163
- 50041	100 17537	- 00014	01948	2 46373	2 46770	17049	- 58050
61000	- 00500	-1 45420	2 68788	214 57861	6 06587	00236	1 06433
3.83476	- 03498						