NATIONAL RADIO ASTRONOMY OBSERVATORY
CHARLOTTESVILLE, VIRGINIA

ELECTRONICS DIVISION INTERNAL REPORT No. 232

FREQUENCY STANDARD TEST PROGRAM

S. WEINREB

JUNE 1983

NUMBER OF COPIES: 150

FREQUENCY STANDARD TEST PROGRAM

S. Weinreb

Figures

Appendix

FREQUENCY STANDARD TEST PROGRAM

S. Weinreb

I. Introduction

This report describes a program which computes the root Allan variance [1], SIGMA, as a function of time interval, TAU, which varies from 1 to 10,000 seconds in 1-2-5 steps. While taking data the program presents a video display of current samples of phase, frequency, and temperature and also plots vs. time SIGMA for TAU = 1, frequency, and temperature. BASIC language for the Apple II Plus computer is utilized. The test configuration will first be described and then be followed by descriptions of program start-up, function, and interrupt options. A listing of the program is in Appendix I.

II. Test Configuration

The test configuration is shown in Figure 1. The two frequency standard outputs, usually at 5 MHz, are connected to an NRAO Precision Phase Comparator. This unit consists of a phase detector followed by a DC amplifier and also has a finely adjustable phase shifter in the input line. At the start of a test the phase shifter is adjusted to put the phase detector inputs in quadrature and output at null. The DC gain following the phase detector is selectable in decade steps so that full scale output of ±10 volts is produced by time shifts of ±10 ps, ±100 ps, ±1,000 ps, or ±10,000 ps. For comparison of hydrogen masers the 1,000 ps scale is appropriate; thus, to stay on scale for 10^4 seconds the frequencies must be identical within 1 pp 10^{13}.

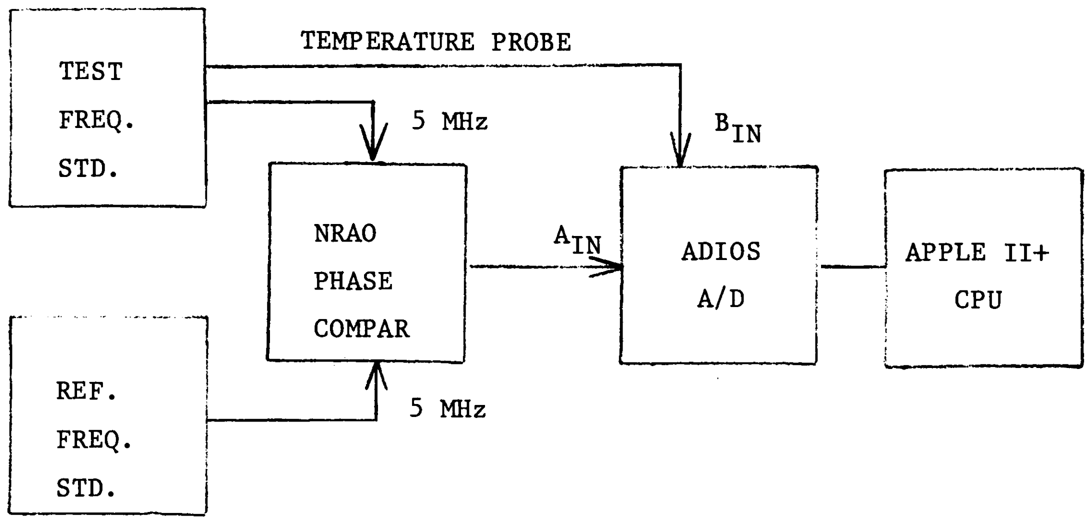


Fig. 1. Test configuration

The phase comparator output drives the A input of an ADIOS integrating A/D converter which is described in NRAO EDIR #212. The A/D conversion is accomplished with a precision 1 MHz voltage-to-frequency converter followed by a 32-bit counter.* For this application the integration time or COUNT in the program is set at 160 ms followed by a dead time or BLANK of 840 ms. This process can be modeled as a 1 Hz low-pass filter followed by 1 Hz sampling. A second analog input channel, B, is driven by a temperature sensor with

*The V/F output is 1 MHz with +10 volts input and 0 Hz at -10 volts input; inputs < -9 volts should be avoided to prevent counter resolution problems.

schematic shown in Figure 2. The toggle switches on the ADIOS front panel must be set at +/- and 10 volts for A and, for B, at + and 10 volts.

The required computer hardware is an Apple II Plus computer with 48k byte memory, one 5¼" floppy disk (slot 6), CRT display, a Trendcom 200 thermal printer (slot 1) and California Computer Systems 7724 clock (slot 4). The program is stored on disk with the name FST along with required binary utility programs LIB 3.1 (for ADIOS service, see EDIR's #224 and #225), SHAPES (for plotting symbols), and HGR CHR GEN SHORT (for labeling plot). Also included on the disk are a text file, FST LOADER, which loads all of the above programs, and a program, SET TIME, to set the clock, if necessary.

III. Start-Up

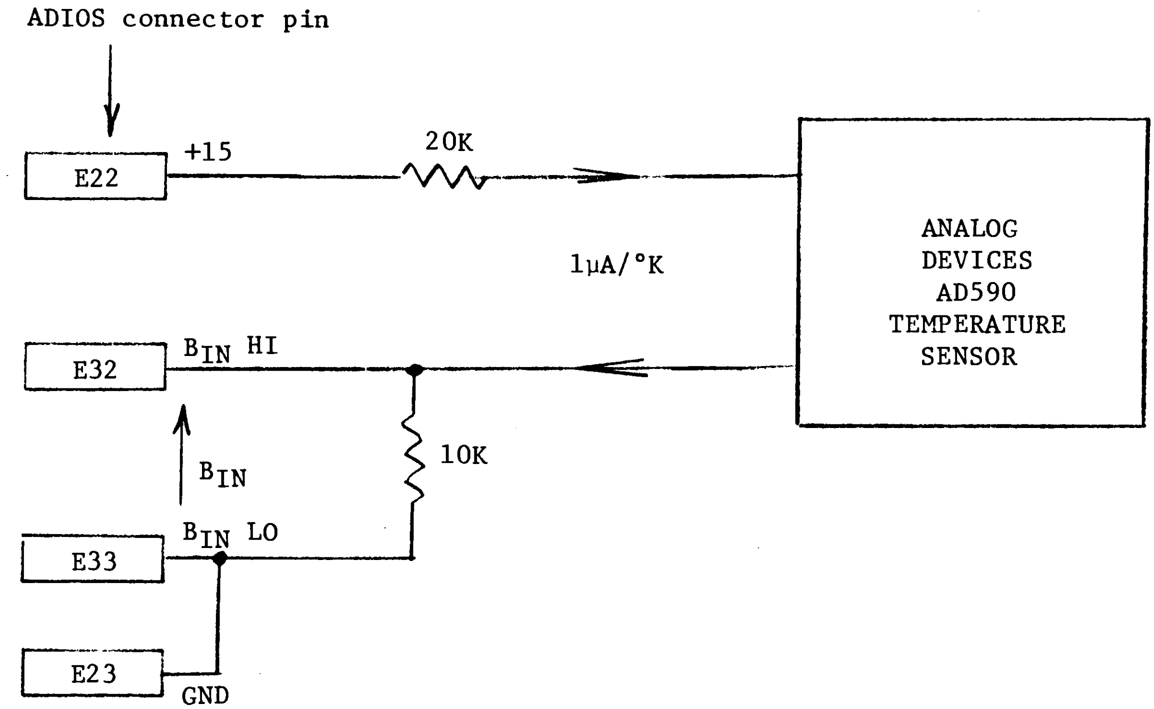
The program disk is inserted in the disk drive and AC power is turned on or IN#6 (CR)* is typed. The disk catalog is displayed and then EXEC FST LOADER (CR) is typed to load FST and binary utilities and then run FST.

The following "edit parameters" screen will appear on the CRT:

```
PRESENT PARAMETERS ARE:
(1)  PHASE DETECTOR OUTPUT, FULL SCALE, IN PS=1000
(2)  PERFECT (P) OR EQUALLY (E) UNSTABLE REFERENCE STANDARD;NOW IS P
(3)  SECONDS PER DATA BATCH=100
(4)  VIEW BEFORE HARD COPY (V) OR CONTINUOUS DATA TAKING (C); NOW V
(5)  LOG SIGMA PLOT, MAX IS EXP -11
(6)  LIN DRIFT PLOT, MAX *EXP-12 IS 100
(7)  TEMPERATURE PLOT SPAN IS 5C
     CHANGE (1-7) OR (RETURN)?
```

---

*Carriage return key.



$B_{IN}(mV) = 10*TE(°K)$

$B_{IN}(mV) = 10*(TE(°C) + 273.2)$

$IN(1)$ (COUNTS) $= .100*MODE\%(2)*B_{IN}(mV)$

Thus, $TE(°C) = CK*IN(1) - CN$

where $CK = 1/MODE\%(2)$ $\quad$ $CN = 273.2$

Fig. 2. Schematic of temperature sensor and scaling equations to relate program variable TE(°C) to counts, IN(1), produced by temperature sensor.

To change a parameter, type the number (1 thru 7) of the parameter and a new value of the parameter will then be requested on the screen. The new value and the return key are then typed; the process can be repeated as often as desired. The editing is terminated by typing the return key instead of a number; data taking will then begin.

The meaning of the seven program parameters is as follows:

(1) Phase detector full scale - This is the number of ps time shift of one standard relative to the other which will cause a 10 volt change in the phase comparator output.

(2) Perfect or equal reference stability - If P is typed, it is assumed that the reference standard has perfect stability and all instability is assigned to the standard under test. If E is typed, all output rms deviations are divided by $\sqrt{2}$ so that they apply to one of two equally unstable standards.

(3) Duration of data batches, KM - The data taken in batches of length KM seconds. After each batch statistics are viewed on the CRT or printed on the printer. Data taking stops during this viewing and printing time.

(4) View before hard copy or continuous data taking - If V is chosen, after each batch is complete statistics will be presented on the CRT and a choice may be made as to whether to hard copy or not - an operator must be present in order for data taking to continue. If C is chosen, statistics are hard copied and a new batch starts without operator intervention.

(5) Log SIGMA plot maximum - During each batch the 1 second SIGMA is plotted (□ symbol) vs. time; after the batch SIGMA vs. TAU is plotted. The abscissa of both plots cover 5 decades with maximum value entered here (i.e., 10^{-11} to 10^{-16} if the parameter is -11).

(6) Linear drift plot scale, DM - During each batch, frequency averaged over ∿KM/100 seconds is also plotted (+ symbol) with linear scale of + DM * 10^{-12} to - DM * 10^{-12}.

(7) Temperature plot span, TS - During each batch, temperature vs. time is plotted (solid line) with a total span of TS centered upon the temperature at batch-start time.

IV. Program Function

The program samples, once per second, the difference in phase of the two frequency standards; the current sample is denoted as PØ and a sample taken TAU(L) seconds back in time is labeled P1(L). Since the computed frequencies must be normalized to the phase comparison frequency, fo, we divide each sample phase (in radians) by 2πfo in the process of scaling the ADIOS A/D output. This normalization puts the phase samples in units of time; picoseconds, ps, are used as units. Thus, the fractional frequency, FØ, (i.e., Δf/f) over the time interval TAU(L) is given by

$$F\emptyset(L) = \frac{P\emptyset - P1(L)}{TAU(L)} \tag{1}$$

in units of 10^{-12}. A statistical estimate of the Allan variance, $\sigma^2(L)$ as given in NBS Monograph 140, Eq. 8.13A, is then one-half of the mean square of differences of successive values of FØ

$$\sigma^2(L) = \frac{1}{2N} \sum_{k=1}^{N} [F\emptyset(L,k) - F\emptyset(L,k-1)]^2 \tag{2}$$

where the index k is the sample number.

The program computes $\sigma^2(L)$ for TAU(L) running from 1 to 10,000 seconds in a 1-2-5 sequence. The samples used for a particular TAU(L) are selected so that the computation load is spread out among all the samples as shown in the table below:

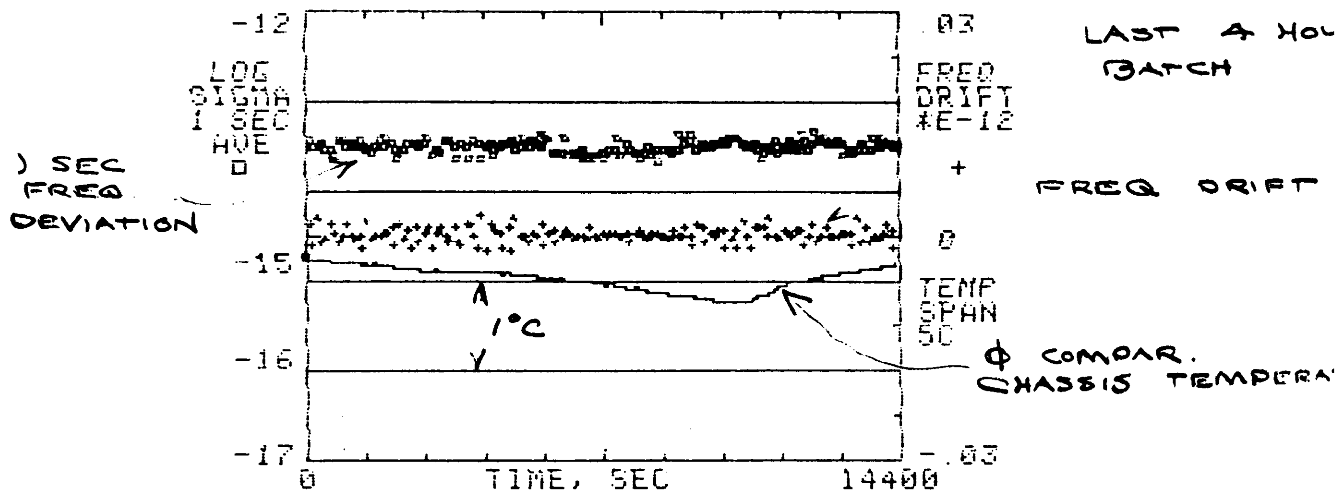
L	TAU(L)	SAMPLES UTILIZED
1	1	1, 2, 3, 4, ...
2	2	2, 4, 6, 8, ...
3	5	5, 10, 15,
4	10	1, 11, 21,
5	20	6, 26, 46,
6	50	5, 55, 105,
7	100	7, 107, 207,
8	200	1, 201, 401,
9	500	2, 502, 1002,
10	1,000	8, 1008, 2008,
11	2,000	4, 2004, 4004,
12	5,000	7, 5007, 10007,
13	10,000	9, 10009, 20009,

While taking data, the CRT displays the values of every other phase sample, two-second frequency, FØ(2), and probe temperature. A plot of three variables versus time is also presented on the CRT; the time axis extends to the batch duration, KM. Points are plotted every DP seconds where DP is the largest value of TAU(L) less than KM/100. The three plotted variables are the 1 second root Allan variance estimate based upon DP samples, the fractional frequency, FØ, over the time interval DP, and the probe temperature, TE. A example of a completed time plot is shown as the upper graph of Figure 3.

TESTS OF NRAO 5 MHZ PHASE COMPARATOR

FST PROGRAM OF 01/14/83 | PERFECT REFERENCE ASSUMMED
BATCH START 04:57.9 02/03/83 | TEMP=29.07C | PHASE=13.4PS
BATCH END 08:57.9 02/03/83 | TEMP=29.01C | PHASE=13.4PS
RUN START 08:40.9 02/02/83 | TLOW=26.69C, THIGH=29.05C

	LAST BATCH		CUMULATIVE		TIME
TAU SEC	SAMPLES	SIGMA *EXP-15	SAMPLES	SIGMA *EXP-15	ERROR PS
1	14398	31.2	86388	33.6	0
2	7198	19.3	43188	17.0	0
5	2878	10.3	17268	8.7	0
10	1438	6.3	8628	5.1	.1
20	718	4.0	4308	3.1	.1
50	286	2.0	1716	1.5	.1
100	142	1.2	852	0.8	.1
200	70	0.5	420	0.4	.1
500	29	0.2	172	0.2	.1
1000	15	0.1	88	0.2	.2
2000	8	0.1	46	0.2	.4
5000	3	0.0	16	0.1	.6
10000	2	0.0	10	0.0	.4



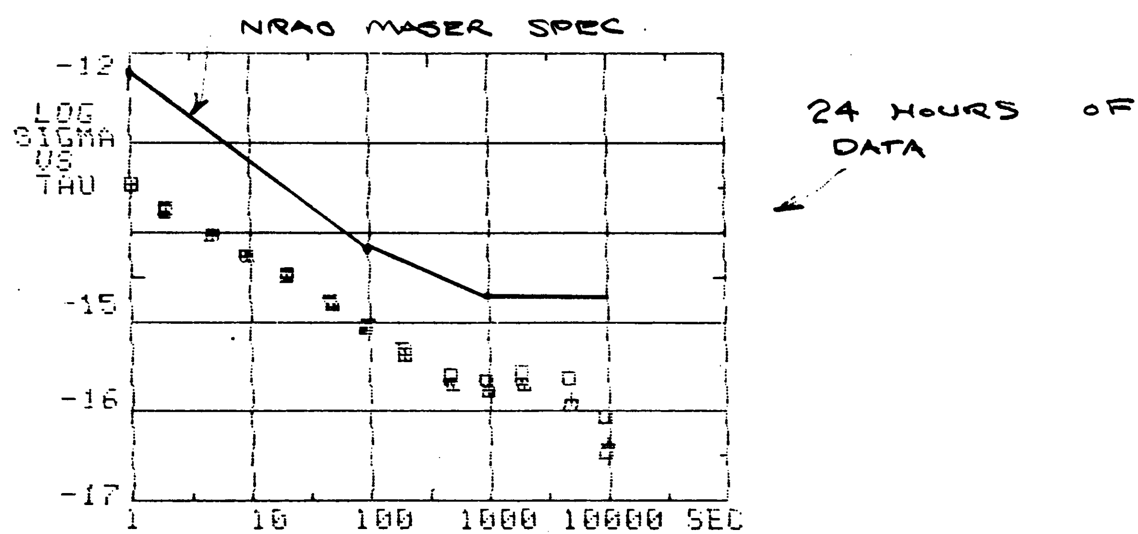


Fig. 3. Example of hard copy at end of data batch.

At the end of a data batch, all of the root Allan variance values are tabulated and plotted as shown in Figure 3. The batch values are also averaged with previously taken data (until a new run is started by typing RUN (CR)). On the plot of root-Allan variance, SIGMA vs. TAU, the cumulative average is plotted with a + symbol and the rms limits of SIGMA x $(1 \pm 1/\sqrt{N})$ are plotted with □ symbols to give a confidence range of the SIGMA value.

V. Data Interruption

The display and optional hard copy at the end of a batch interrupt the data taking sequence. Other interruptions can be produced by typing keys B, E, P, F, or H during data taking. The consequences and uses of the key interrup will be discussed here.

An interruption of the program could cause the time duration between phase samples to be > TAU(L) and also cause the mean square frequency differences in the Allan variance to have some "dead time" between time intervals. If the interruption is < TAU(L) both effects will be negligble. The batch-end interruption lasts for ∿ 200 seconds and thus the last phase and frequency samples for TAU(L) < 500 are discarded. The phase and frequency samples for TAU = 500, 1000, 2000, 5000, and 10,000 are kept as it would be wasteful of time and unnecessary to discard them.

The key interrupts have the following effects:

B - Batch end. Causes an immediate end to the batch with tables, plots, and resets as in a normal batch-end.

E - Edit parameters. Jumps to the edit screen as described in III. After parameters are edited, frequency and phase are reset as for batch-end and the batch continues.

P - Phase adjustment. Data taking stops and phase-comparator phase may be adjusted. All phase samples are discarded and the batch resumes when (CR) is typed.

F - Frequency adjustment. Data taking stops and phase and frequency may be adjusted. All phase and frequency samples are discarded and the batch resumes when (CR) is typed.

H - Halt. Program execution stops. Type RUN (CR) to make a fresh start - all variables erased and default parameters installed. GOTO 2600 will have same effect as B. GOTO 2100 will start a new batch but does not reset phase and frequency.

VI. Program Test

The program can be checked by generating fictitious phase samples using the pseudo-random number generator command, RND, included in Apple BASIC. This is accomplished by removing the REM (remark) word in program lines 2176 and 2247. The program will then input from lines 4200 and 4210 phase and temperature samples with uniform probability from 100 to 110 ps and 25° to 26°C respectively.

The theoretical value for the root-Allan variance can be computed as follows. An estimate of SIGMA in terms of phase samples, P(k), is:

$$(\mathrm{SIGMA})^2 = \frac{1}{2N} \sum_{k=1}^{N} \left[\frac{P(k + L) - 2P(k) + P(k - L)}{L}\right]^2 \tag{3}$$

where L is the time difference between phase samples. The statistical average of $(\mathrm{SIGMA})^2$ can then be expressed in terms of the autocorrelation function, R(k), of the samples,

$$\overline{(\mathrm{SIGMA})^2} = \frac{3R(0) - 4R(L) + R(2L)}{L^2} \tag{4}$$

In the case of independent samples (i.e., white phase noise), R(L) = 0 = R(2L) and R(0) is equal to the variance of a phase sample. For the uniform distribution of 100 to 110 ps R(0) = 8.333 $(ps)^2$ and finally,

$$SIGMA = 5/L \tag{5}$$

in units of 10^{-12} or 5000/L in units of 10^{-15} tabulated by the program. A long test of the program gave the following result which is within statistical limits of 5000/L.

	LAST BATCH		CULMATIVE		TIME
TAU	SAMPLES	SIGMA	SAMPLES	SIGMA	ERRO
SEC		*EXP-15		*EXP-15	PS
1	49998	4985.8	349986	4993.6	5
2	24998	2506.9	174986	2512.9	5
5	9998	998.6	69986	1005.4	5
10	4998	495.9	34986	500.0	5
20	2498	247.8	17486	250.0	5
50	998	100.4	6986	102.9	5.1
100	498	46.8	3486	49.2	4.9
200	248	23.8	1736	24.8	5
500	100	9.5	698	9.5	4.7
1000	50	4.5	348	5.0	5
2000	25	2.4	173	2.5	5
5000	10	1.0	68	1.1	5.5
10000	5	0.7	33	0.5	4.9

REFERENCE

NBS Monograph 140, "Time and Frequency - Theory and Fundamentals National Bureau of Standards, Boulder, CO, pp. 156-157, 181-182.

```
01/17/83  15:52.4

PROGRAM LENGTH= 13472 BYTES         VARIABLES= 174 BYTES
FREE MEMORY= 6833 BYTES
START=16385 LOMEM=29857 FREE=29948 STRING=36781 HIMEM=36864


10  REM  FST - FREQUENCY STANDARD TEST - PROGRAM. COMPUTES ROOT ALLAN
 VARIANCE AS FUNCTION OF AVERAGING TIME. REFERENCE  NBS MONOGRAPH #140;
 ERROR IN EQUATION 8.9 NOTED.
20  REM  WHILE PROGRAM IS TAKING DATA KEYBOARD RESPONDS TO (H) HALT,(B)
 END BATCH,(E) EDIT PARAMETERS,(P) ADJUST PHASE, AND (F) ADJUST FREQUENCY.

30  REM  FOR PROGRAM TESTS EDIT LINES 2176,2247, AND 4810
90  HGR : TEXT : HOME : SPEED= 255: HCOLOR= 3
100 NM$ = "FST PROGRAM OF 01/14/83"
105  PRINT NM$
110  PRINT : PRINT "REQUIRES LIB 3.1,SHAPES,HGR CHR GEN SHORT IN MEMORY"
120  REM   KEY VARIABLES FOLLOW
130  REM   L IS AN INDEX WHICH DETERMINES  WHICH AVERAGING TIME,TAU(L)
 IS RELEVENT. L RUNS 1 TO 13.
140  REM  P0 IS THE MOST RECENT PHASE SAMPLE
145  REM  P1(L) IS A PHASE SAMPLE TAKEN TAU(L) UNITS BACK IN TIME
150  REM  F0 IS MOST RECENT FREQUENCY COMPUTED FROM P0 AND P1(L)
155  REM  F1(L) IS A FREQUENCY COMPUTATION TAU(L) UNITS BACK IN TIME
160  REM  K IS A RUNNING SAMPLE INDEX
165  REM  K(L) IS THE NEXT DESIRED SAMPLE
170  REM   KP(L) IS A FLAG WHICH IS 0 UNTIL THE FIRST P1(L) IS VALID.
 IT MUST BE RESET IF PHASE IS ADJUSTED.
175  REM  KD(L)IS THE NUMBER OF SQUARED FREQUENCY DIFFERENCES,D0, IN
 THE SUM,SD(L)
180  REM   GD(L) IS THE  GRAND SUM OF ND(L) FREQUENCY DIFFERENCES
190  REM   TE IS THE MEASURED TEMPERATURE
250  DIM TAU(13),P1(13),F1(13),KP(13),KF(13),K(13),SD(13),KD(13),GD(13),ND
(13)
290  FOR L = 1 TO 13: READ TAU(L),K(L): NEXT L
295  REM  INITIAL K(L) DETERMINES WHICH SAMPLES ARE USED FOR A PARTICULAR
 L. A   SAMPLE OCCURS WHEN K=K(L)+N*TAU(L).
300  DATA  1,0,2,0,5,0,10,1,20,6,50,5,100,7,200,1,500,2,1000,8,2000,4,5000
,7,10000,9
305  RESTORE
310 GP = 3: REM   POINT PLOTTED AT K=3,3+DP,3+2*DP,ETC.
340  REM   C1 IS THE NUMBER OF MV INPUT PER PS OF TIME ERROR
355  REM   C2=0.707 IF RMS VALUES ARE TO APPLY TO ONE OF TWO IDENTICAL
 STANDARDS.C2=1 IF REFERENCE IS PERFECT.SEE 5150.
360 C3 = .70707: REM  C3*C2 MULTIPLIES RMS FREQUENCY DIFFERENCE
400  REM         KM IS NUMBER OF SAMPLES BEFORE PRINT OUT
435 CM = 1000:CC = 100:CY = .21714
440 CA = 10:CB = .5:EP = 1E - 18
450  DEF  FN R1(X) =  INT (CA * X + CB) / CA
460  DEF  FN R2(X) =  INT (CC * X + CB) / CC
510  POKE 232,191: POKE 233,31: REM  SHAPES IN A8127,L24
600 AI = 5126:AS = 5129: REM  ENTRY POINTS FOR ADIOS INIT AND SERVICE
610 MODE%(1) = 840:MODE%(2) = 160: REM  BLANK AND COUNT TIMES IN MS
620 CS = 20 / MODE%(2):CQ = 10000: REM  SCALINGCONSTANTS FOR ADIOS
 IN +/-10 VOLT MODE
630 CK = 1 / MODE%(2):CN = 273.2: REM   SCALINT CONSTANTS FOR AD590
 TEMP PROBE IN BIN WITH +10 VOLT ADIOS SCALE
650  CALL AI,MODE%(0),OUT%(0): REM   INITIALIZE ADIOS; SEE EDIR #224
899  REM  INITAL PARAMETERS NEXT
900 FS = 1000:RS$ = "P":KM = 100:O$ = "V":SM =  - 11:DM = 100:TS =
 5
1000  GOSUB 5000: REM  EDIT PARAMETERS
1010  GOSUB 4800:TR$ = "RUN   START " + TM$ + DT$
2100  REM  PREPARE FOR TAKING DATA BATCH
2105  POKE 34,21: REM  TEXT BELOW GRAPHICS
2110  FOR L = 1 TO 13: READ TAU(L),K(L):KD(L) = 0:SD(L) = 0: NEXT L
2125  RESTORE
2130 XL = 0:XH = KM:YH = SM:YL = SM - 5
2135 XC$ = "TIME, SEC"
2140  GOSUB 10000: GOSUB 10200:PD = 0:GP = 3: REM  ERASE AND INIT TIME
 PLOT
2150  GOSUB 4800:TS$ = "BATCH START " + TM$ + DT$
2160 TL = 99:TH = 00: REM   INITAL LOW AND HIGH TEMPS
2170  CALL AS,IN%(0),OUT%(0)
2172 P0 = CV * IN%(0) - CM
2175 TE = CK * IN%(1) - CN
2176  REM    GOSUB 4200
2177 TG = TE:PG = P0: REM   BATCH START TEMP AND PHASE
2178 YC = TE + .55 * TS:YD = TE - .45 * TS: REM     TEMP PLOT LIMITS
2180 DJ = (Y9 - Y8) / (YC - YD):DK = Y8 - DJ * YD:YJ = DJ * TE + DK:
 REM    TEMP PLOT PARAMETERS
2199  REM  MAIN MEASUREMENT LOOP FOLLOWS
2200  FOR K = 0 TO KM
2210  IF K = KM THEN  GOSUB 3400: GOTO 2100
2215 IR =  PEEK ( - 16384): POKE  - 16368,0
2217  IF IR > 127 THEN  GOSUB 4300
2220  CALL AS,IN%(0),OUT%(0)
2240 P0 = CV * IN%(0) - CM
2245 TE = CK * IN%(1) - CN
2247  REM  GOSUB 4200: REM     RANDOM NUMBERS FOR PROGRAM TEST
2248  IF TE < TL THEN TL = TE
2250  IF TE > TH THEN TH = TE
2260  IF K = GP THEN  GOSUB 2900: REM   PLOT
2300  FOR L = 1 TO 13
2309  REM  NEXT IF DATA POINT IS NOT USED AT THIS L
2310  IF K <  > K(L) THEN  NEXT L: NEXT K
2330 K(L) = K(L) + TAU(L)
2349  REM   NEXT IF FIRST POINT AT THIS L
2350  IF KP(L) = 0 THEN P1(L) = P0:KP(L) = 1: NEXT L: NEXT K
2360 F0 = (P0 - P1(L)) / TAU(L)
2370 P1(L) = P0
2380  IF L = 2 THEN  VTAB 24: PRINT K;" PHASE="; FN R1(P0); TAB( 17);"
 FREQ="; FN R1(F0); TAB( 30);"TEMP="; FN R2(TE)
2399  REM   NEXT IF F0 IS FIRST FREQUENCY MEASUREMENT
2400  IF KF(L) = 0 THEN F1(L) = F0:KF(L) = 1: NEXT L: NEXT K
2420 D0 = F0 - F1(L):SD(L) = SD(L) + D0 * D0:KD(L) = KD(L) + 1
2430 F1(L) = F0: NEXT L: NEXT K
2599  REM    KM   SAMPLES HAVE BEEN TAKEN
2600  GOSUB 3400
2610  GOTO 2100
2900  REM  PLOT SIGMA VS TIME
2905 X = SD(1) - PD: IF X < EP THEN YP = YL: GOTO 2920
2910 YP = CZ + CY *  LOG (X / DP)
2920 XP = K: GOSUB 10400
2925 YP = F1(LP): GOSUB 10500
2930 GP = GP + DP:PD = SD(1)
2940  IF K < DP THEN  RETURN
2945 YK = DJ * TE + DK
2950  IF YK < Y9 THEN YK = Y9
2955  IF YK > Y8 THEN YK = Y8
```

```
2957 XJ = D8 * (K - DP) + D9:XK = D8 * K + D9
2960  HPLOT XJ,YJ TO XK,YK
2970 YJ = YK: RETURN
3400  REM  PROCESSING AFTER BATCH COMPLETE
3402  GOSUB 4800:TE$ = "BATCH  END  " + TM$ + DT$
3405  CALL  - 1059: CALL  - 1059: REM  BEEP
3407 T7 = TE:P7 = P0: REM  BATCH END TEMP AND PHASE
3410  FOR L = 1 TO 13
3420 NO(L) = NO(L) + KO(L)
3430 GO(L) = GO(L) + SO(L)
3435  NEXT L
3440  REM   NEXT STARTS NEW P1 AND F1 SAMPLES FOR TAU(L)<500
3450  FOR L = 1 TO 8
3455 KP(L) = 0:KF(L) = 0
3460  NEXT L
3470  IF O$ = "C" THEN  GOSUB 5600: GOSUB 5800: GOSUB 3900: GOSUB 5800:
 GOSUB 5400: RETURN : REM      HARD COPY OF TABLE,TIME PLOT, AND SIGMA
 PLOT
3480  GOSUB 3600: REM  VIEW TABLE
3490  PRINT "HARD COPY (Y OR N)? ";: GET Z$: PRINT Z$: HOME : POKE
 34,21
3500  IF Z$ = "Y" THEN  GOSUB 5600
3505 IR$ = Z$: GOSUB 4320
3510  GOSUB 4100: REM   SHOW TIME PLOT
3520  PRINT "HARD COPY (Y OR N)? ";: GET Z$: PRINT Z$
3530  IF Z$ = "Y" THEN  GOSUB 5800
3535 IR$ = Z$: GOSUB 4320
3540  GOSUB 3900: REM  SIGMA PLOT
3550  PRINT "HARD COPY (Y OR N)? ";: GET Z$: PRINT Z$
3560  IF Z$ = "Y" THEN  GOSUB 5800
3565 IR$ = Z$: GOSUB 4320
3570  GOTO 2100: REM  START NEW BATCH
3600  REM  VIEW TABLE
3610  TEXT : HOME
3620  PRINT TS$: PRINT TE$: PRINT TR$
3625  PRINT "TEMPERATURE: LOW="; FN R2(TL);"C, HIGH="; FN R2(TH);"C"
3627  PRINT "START "; FN R2(T6);"C,"; FN R1(P6);"PS END "; FN R2(T7);"C,";
 FN R1(P7);"PS"
3630  PRINT : PRINT  TAB( 10);"LAST BATCH"; TAB( 26);"CULMATIVE"
3640  PRINT "TAU"; TAB( 8);"SAMPLES"; TAB( 17);"SIGMA"; TAB( 24);"SAMPLES"
; TAB( 33);"SIGMA"
3650  PRINT "SEC"; TAB( 16);"*EXP-15"; TAB( 32);"*EXP-15"
3660  FOR L = 1 TO 13
3670  PRINT TAU(L); TAB( 8);KO(L); TAB( 15);
3680 X = C4 *  SQR (SO(L) / (KO(L) + EP))
3685 WI = 7:DI = 1: GOSUB 11000
3690  PRINT  TAB( 24);NO(L); TAB( 32);
3695 X = C4 *  SQR (GO(L) / (NO(L) + EP))
3698 DI = 1:WI = 7: GOSUB 11000
3700  PRINT : NEXT L: PRINT
3710  RETURN
3900  REM  SIGMA VS TAU(L) PLOT
3910 XL = 0:XH = 5:YL = SH - 5:YH = SH
3920  GOSUB 10000: GOSUB 10700: REM   ERASE AND INIT PLOT
3930  FOR L = 1 TO 13
3935  IF NO(L) < 1 THEN 4020
3940 XP = 2 * CY *  LOG (TAU(L))
3950 YP = CZ + CY *  LOG (GO(L) / NO(L))
3960  SCALE= 2:SH = 1: GOSUB 10400
3970 YP = YP +  LOG (1 + 1 /  SQR (NO(L)))
3980 SH = 2: GOSUB 10400
3985  IF NO(L) = 1 THEN YP = SH - 5 + EP: GOTO 4000
3990 YP = YP +  LOG (1 - 1 /  SQR (NO(L)))
4000 SH = 2: GOSUB 10400
4020  NEXT L
4030  SCALE= 1: RETURN
4100  REM  SHOW GRAPHICS
4110  POKE  - 16304,0: POKE  - 16300,0: POKE  - 16297,0: POKE  - 16301,0:
 HCOLOR= 3: VTAB 21: HTAB 1
4120  RETURN
4200 P0 = 100 + 10 *  RND (K)
4210 TE = 25 +  RND (1)
4220  RETURN
4300  REM  KEYBOARD INTERRUPT SERVICE
4305 IR = IR - 128
4310 IR$ =  CHR$ (IR)
4320  IF IR$ = "P" THEN  GOSUB 4400
4325  IF IR$ = "F" THEN  GOSUB 4500
4330  IF IR$ = "E" THEN  GOSUB 4600
4340  IF IR$ = "B" THEN 2600: REM  BATCH END
4350  IF IR$ = "H" THEN  GOSUB 4700
4390  RETURN
4400  REM  ADJUST PHASE
4410  PRINT "ADJUST PHASE AND PRESS (RETURN)": GET ZZ$: PRINT ZZ$
4420  FOR L = 1 TO 13:KP(L) = 0: NEXT L
4430  RETURN
4500  REM  ADJUST FREQUENCY
4510  PRINT "ADJUST FRQUENCY AND PRESS (RETURN)": GET ZZ$: PRINT ZZ$
4520  FOR L = 1 TO 13:KF(L) = 0:KP(L) = 0: NEXT L
4530  RETURN
4600  REM  EDIT PARAMETERS
4610  GOSUB 5000
4620  FOR L = 1 TO 8
4630 KP(L) = 0:KF(L) = 0: NEXT L
4640  REM  ABOVE RESET PHASE AND FREQ SAMPLES FOR TAU <500. APPROPRIATE
 FOR <60 SEC PAUSE.
4650 XH = KM:YH = SH:YL = SH - 5: GOSUB 10200
4690  RETURN
4700  REM  HALT DATA TAKING
4710  TEXT : HOME
4720  PRINT "HALT DATA TAKING": PRINT
4730  PRINT "(RUN) NEW RUN- FRESH START."
4740  PRINT "(GOTO 2100) NEW BATCH."
4750  PRINT "(GOTO 2600) OUTPUT DATA."
4760  END
4800  REM   SETS T$ TO TIME
4810  REM      REMOVE RETURN IF YOU HAVE A CLOCK
4820 YR$ = "83"
4825  PRINT
4830 D$ =  CHR$ (4)
4840  PRINT D$;"IN#4"
4850  REM  PRINT D$;"PR#4": REM THIS IS ONLY FOR MOUNTAIN HARDWARE
 CLOCK
4860  INPUT " ";T$
4870  CALL 1013
4880 DT$ =  LEFT$ (T$,5) + "/" + YR$ + " "
4890 SC$ =  MID$ (T$,13,2)
4900 EM =  INT ( VAL (SC$) / 6)
4910 EM$ =  STR$ (EM)
4920 TM$ =  MID$ (T$,7,2) + ":" +  MID$ (T$,10,2) + "." + EM$ + " "
4930  RETURN
5000  REM   EDIT PARAMETERS
```

```
5010  TEXT : HOME : PRINT
5020  PRINT "     PRESENT PARAMETERS ARE:": PRINT
5030  PRINT "(1) PHASE DETECTOR OUTPUT, FULL SCALE, IN PS=";
5035  FLASH : PRINT FS: NORMAL : PRINT
5040  PRINT "(2) PERFECT (P) OR EQUALLY (E) UNSTABLE REFERENCE STANDARD;NO
W IS ";
5045  FLASH : PRINT RS$: NORMAL : PRINT
5049  PRINT "(3) SECONDS PER DATA BATCH=";
5050  FLASH : PRINT KM: NORMAL : PRINT
5060  PRINT "(4) VIEW BEFORE HARD COPY (V) OR CONTINUOUS DATA TAKING
 (C); NOW ";
5065  FLASH : PRINT O$: NORMAL : PRINT
5070  PRINT "(5) LOG SIGMA PLOT, MAX IS EXP ";
5075  FLASH : PRINT SM: NORMAL : PRINT
5080  PRINT "(6) LIN DRIFT PLOT, MAX *EXP-12 IS ";
5085  FLASH : PRINT DM: NORMAL : PRINT
5090  PRINT "(7) TEMPERATURE PLOT SPAN IS ";: FLASH : PRINT TS;: NORMAL
 : PRINT "C": PRINT
5100  PRINT "    CHANGE (1-7) OR (RETURN)? ": PRINT
5120  GET J$
5125  IF  ASC (J$) > 55 THEN 5100
5130  ON  VAL (J$) GOTO 5210,5220,5230,5240,5250,5260,5270
5140 C1 = FS / 10000: REM  INPUT PS/MV
5142 CV = C1 * CS:CH = C1 * CQ
5145 YA = DM:YB =  - DM: REM    DRIFT PLOT SCALE
5150  IF RS$ = "E" THEN C2 = .707:RF$ = "EQUAL REFERENCE ASSUMMED"
5152  IF RS$ = "P" THEN C2 = 1:RF$ = "PERFECT REFERENCE ASSUMMED"
5155 C4 = C3 * C2 * CM: REM  . COLOR= *(.7 OR 1)*1000
5157 CZ = .4343 *  LOG (C3 * C2) - 12:CY = .217: REM  CONSTANTS FOR
 SCALING LOG SIGMA
5160  REM  NEXT FINDS INTERVAL,DP, BETWEEN POINTS IN TIME PLOT
5170 LP = 13
5180 LP = LP - 1: IF LP = 1 THEN 5190
5185  IF TAU(LP) > KM / 100 THEN 5180
5190 DP = TAU(LP): REM  DP AND LP ARE NOW KNOWN
5205  RETURN
5210  INPUT "(1) INPUT PS= ";FS
5215  GOTO 5010
5220  INPUT "(2) REFERENCE, E OR P IS ";RS$
5225  GOTO 5010
5230  INPUT "(3) SECONDS PER BATCH=";KM
5235  GOTO 5010
5240  INPUT "(4) VIEW (V) OR CONTINUOUS (C) ";O$
5245  GOTO 5010
5250  INPUT "SIGMA PLOT MAX, EXP ";SM
5255  GOTO 5010
5260  INPUT "DRIFT PLOT SCALE, *EXP-12=";DM
5265  GOTO 5010
5270  INPUT "TEMP PLOT SPAN=";TS
5275  GOTO 5010
5300  REM  TURN ON TRENDCOM PRINTER
5308  PRINT
5310  PRINT  CHR$ (4);"PR#1"
5320  PRINT  CHR$ (0);
5330  POKE 1913,6: POKE 1785,72: POKE 1657,80: REM  MARGINS
5340  RETURN
5400  REM  MOVE TO END OF PAGE
5410  GOSUB 5300
5420  FOR LL = 1 TO 5: PRINT : NEXT
5430  CALL 1013: RETURN
5600  REM  HARD COPY TABLE
5610  GOSUB 5300: REM  TURN ON PRINTER
5615  HOME : PRINT  CHR$ (9);"N": REM  DISABLE CRT
5618  POKE 33,80: REM  MARGIN
5619  PRINT NM$; TAB( 40);RF$
5620  PRINT TS$; TAB( 40);"TEMP="; FN R2(T6);"C"; TAB( 55);" PHASE=";
 FN R1(P6);"PS"
5623  PRINT TE$; TAB( 40);"TEMP="; FN R2(T7);"C"; TAB( 55);" PHASE=";
 FN R1(P7);"PS"
5626  PRINT TR$; TAB( 40);"TLOW="; FN R2(TL);"C, THIGH="; FN R2(TH);"C"
5630  PRINT : PRINT "          ";"LAST BATCH"; TAB( 37);"CUMULATIVE";
 TAB( 53);"TIME"
5640  PRINT "TAU"; TAB( 14);"SAMPLES"; TAB( 23);"SIGMA"; TAB( 35);"SAMPLES
"; TAB( 44);"SIGMA"; TAB( 53);"ERROR"
5650  PRINT "SEC"; TAB( 22);"*EXP-15"; TAB( 43);"*EXP-15"; TAB( 54);"
 PS "
5660  FOR L = 1 TO 13
5670  PRINT TAU(L); TAB( 14);KD(L); TAB( 21);
5675 DI = 1:WI = 8
5680 X = C4 *  SQR (SD(L) / (KD(L) + EP)): GOSUB 11000
5690  PRINT  TAB( 35);ND(L); TAB( 43);
5695 X1 = C4 *  SQR (GD(L) / (ND(L) + EP))
5698 X = X1:DI = 1:WI = 7: GOSUB 11000
5710 TE = X1 * TAU(L) / CH
5720  PRINT  TAB( 53); FN R1(TE)
5730  NEXT L: PRINT
5740  POKE 33,40: PRINT  CHR$ (9);"I": CALL 1013: REM  ENABLE CRT
5750  RETURN
5800  REM  PRINT GRAPHICS
5810  GOSUB 5300
5820  POKE 1145,105
5830  CALL  - 16038
5840  CALL 1013: RETURN
6000  REM  FORMATTED LIST
6010  GOSUB 4000: REM   GET TIME
6020  POKE 33,33
6030  GOSUB 5300: REM  TURN ON PRINTER
6040  DEF  FN CT(AD) =  PEEK (AD) + 256 *  PEEK (AD + 1)
6050 SR =  FN CT(103)
6060 LM =  FN CT(105):FR =  FN CT(109)
6070 HM =  FN CT(115):ST =  FN CT(111)
6080  PRINT : PRINT DT$,TM$: PRINT
6090  PRINT "PROGRAM LENGTH= ";LM - SR;" BYTES        VARIABLES= ";HM
 - ST + FR - LM;" BYTES"
6100  PRINT "FREE MEMORY= ";ST - FR;" BYTES"
6110  PRINT "START=";SR;" LOMEM=";LM;" FREE=";FR;" STRING=";ST;" HIMEM=";H
M
6120  PRINT
6130  LIST
6140  CALL  - 26868: REM  TURN OFF PRINTER AND RECONNECT PLE
6150  END
10000  REM  RECTANGULAR PLOT ROUTINES, ERASE @10000, INIT @10200,PLOT
 POINT @10400, COMMENT @10600
10005  HGR : HCOLOR= 3
10010 I = 0: POKE 8125,I: REM    PLOT SHAPE PARAMETER
10015 SH = 2: ROT= 0: SCALE= 1
10020 X8 = 39:X9 = 239:Y8 = 150:Y9 = 0
10025 X7 = (X9 - X8) / 10:Y7 = (Y8 - Y9) / 10
10030  REM   PLOT BORDER NEXT
10035  HPLOT X8,Y8 TO X8,Y9 TO X9,Y9 TO X9,Y8 TO X8,Y8
10040  FOR YT = Y9 TO Y8 STEP 2 * Y7
10045  HPLOT X8,YT TO X9,YT
```

```
10047  NEXT
10048  FOR YT = Y9 TO Y8 STEP Y7
10050  HPLOT X9,YT TO X9 - 2,YT
10055  NEXT YT
10060 X6 = (X9 + X8) / 2:Y6 = (Y9 + Y8) / 2
10065  HPLOT X8,Y6 TO X8 + 4,Y6: HPLOT X9 - 4,Y6 TO X9,Y6
10070  HPLOT X6,Y8 - 4 TO X6,Y8: HPLOT X6,Y9 TO X6,Y9 + 4
10075  FOR XT = X8 TO X9 STEP X7
10080  HPLOT XT,Y8 - 2 TO XT,Y8
10085  HPLOT XT,Y9 TO XT,Y9 + 2
10090  NEXT XT
10095  RETURN
10199  REM
10200  REM      INITIALIZE PLOT, SET LIMITS (XL,XH,YL,YH) AND (YA,YB)
10205 X8 = 39:X9 = 239:Y8 = 150:Y9 = 0
10210 X7 = (X9 - X8) / 10:Y7 = (Y8 - Y9) / 10
10215 D8 = (X9 - X8) / (XH - XL):D9 = X8 - D8 * XL
10220 D6 = (Y9 - Y8) / (YH - YL):D7 = Y8 - D6 * YL
10222 D4 = (Y9 - Y8) / (YA - Y8):D5 = Y8 - D4 * Y8
10225  POKE  - 16304,0: POKE  - 16300,0
10230  POKE  - 16297,0: POKE  - 16301,0: HCOLOR= 3
10235  REM   LABEL WITH HGR CHR GEN
10240  CALL 3072
10245  PRINT  CHR$ (1); CHR$ (17)
10250  VTAB (1): HTAB (3): PRINT YH
10251  VTAB (3): HTAB (1): PRINT " LOG"
10252  VTAB (4): HTAB (1): PRINT "SIGMA"
10253  VTAB (5): HTAB (1): PRINT "1 SEC"
10254  VTAB (6): HTAB (1): PRINT " AVE"
10255  VTAB (19): HTAB (3): PRINT YL
10256  VTAB (15): HTAB (3): PRINT YL + 1
10257  VTAB (11): HTAB (3): PRINT YL + 2
10260  VTAB (20): HTAB (6): PRINT XL
10261  VTAB (1): HTAB (36): PRINT YA
10262  VTAB (19): HTAB (36): PRINT YB
10263  VTAB (3): HTAB (36): PRINT "FREQ "
10264  VTAB (4): HTAB (36): PRINT "DRIFT"
10265  VTAB (20): HTAB (32): PRINT XH
10266  VTAB (5): HTAB (36): PRINT "*E-12"
10267  VTAB (10): HTAB (36): PRINT " 0"
10268  VTAB (20): HTAB (15): PRINT XC$
10269  VTAB (12): HTAB (36): PRINT "TEMP"
10270  VTAB (13): HTAB (36): PRINT "SPAN"
10271  VTAB (14): HTAB (36): PRINT TS;"C"
10274  CALL 1013: REM    TURN OFF HGR CHR GEN
10275  POKE  - 16301,0
10276  SCALE= 2
10277  DRAW 1 AT 260,52: DRAW 2 AT 17,52
10278  SCALE= 1
10280  REM    CHANGES PLOT POINT SHAPE
10285 X8 = 39:X9 = 239:Y8 = 150:Y9 = 0
10290 X7 = (X9 - X8) / 10:Y7 = (Y8 - Y9) / 10
10295 SH = 2: ROT= 0: SCALE= 1
10300  ON I GOSUB 10310,10315,10320,10325,10330,10335
10305 I = I + 1: GOTO 10340
10310 SH = 1: ROT= 0: SCALE= 1: RETURN
10315 SH = 2: ROT= 8: SCALE= 2: RETURN
10320 SH = 3: ROT= 0: SCALE= 1: RETURN
10325 SH = 3: ROT= 32: SCALE= 1: RETURN
10330 SH = 2: ROT= 0: SCALE= 2: RETURN
10335 SH = 1: ROT= 16: SCALE= 2: RETURN
10340  RETURN
10399  REM
10400  REM    PLOT XP,YP IN LIMITS XL,XH,YL,YH
10410 XT = D8 * XP + D9:YT = D6 * YP + D7
10420  IF XT < X8 THEN XT = X8
10430  IF XT > X9 THEN XT = X9
10440  IF YT < Y9 THEN YT = Y9
10450  IF YT > Y8 THEN YT = Y8
10460  DRAW SH AT XT,YT
10470  RETURN
10500  REM  ALTERNATE PLOT YP IN Y8,YT
10510 XT = D8 * XP + D9:YT = D4 * YP + D5
10520  IF XT < X8 THEN XT = X8
10530  IF XT > X9 THEN XT = X9
10540  IF YT < Y9 THEN YT = Y9
10550  IF YT > Y8 THEN YT = Y8
10560  DRAW 1 AT XT,YT
10570  RETURN
10599  REM
10600  REM   COMMENT ON PLOT
10610  DRAW SH AT 4,8 * (1.5 + I)
10620  INPUT "TYPE QUOTE THEN COMMENT  ";CM$
10630  HCOLOR= 0: REM   ERASE BORDER
10640  HPLOT X8,8 * (1 + I) TO X8,8 * (2 + I)
10650  HCOLOR= 3
10660  CALL 3072: PRINT  CHR$ (1): PRINT  CHR$ (17)
10670  VTAB (2 + I): HTAB (2): PRINT CM$
10680  CALL 1013: POKE  - 16301,0
10690  RETURN
10700  REM     INITIALIZE PLOT, SET LIMITS (XL,XH,YL,YH)
10705 X8 = 39:X9 = 239:Y8 = 150:Y9 = 0
10710 X7 = (X9 - X8) / 10:Y7 = (Y8 - Y9) / 10
10715 D8 = (X9 - X8) / (XH - XL):D9 = X8 - D8 * XL
10720 D6 = (Y9 - Y8) / (YH - YL):D7 = Y8 - D6 * YL
10730  POKE  - 16304,0: POKE  - 16300,0
10735  POKE  - 16297,0: POKE  - 16301,0: HCOLOR= 3
10740  REM   LABEL WITH HGR CHR GEN
10745  CALL 3072
10750  PRINT  CHR$ (1); CHR$ (17)
10755  VTAB (1): HTAB (3): PRINT YH
10760  VTAB (3): HTAB (1): PRINT " LOG"
10765  VTAB (4): HTAB (1): PRINT "SIGMA"
10770  VTAB (5): HTAB (1): PRINT " VS  "
10775  VTAB (6): HTAB (1): PRINT " TAU"
10780  VTAB (19): HTAB (3): PRINT YL
10785  VTAB (15): HTAB (3): PRINT YL + 1
10790  VTAB (11): HTAB (3): PRINT YL + 2
10795  VTAB (20): HTAB (6): PRINT "1"
10796  VTAB (20): HTAB (12): PRINT "10"
10797  VTAB (20): HTAB (17): PRINT "100"
10800  VTAB (20): HTAB (22): PRINT "1000"
10820  VTAB (20): HTAB (27): PRINT "10000"
10830  VTAB (20): HTAB (33): PRINT "SEC"
10840  CALL 1013: REM    TURN OFF HGR CHR GEN
10845  POKE  - 16301,0
10850  FOR XT = X8 TO X9 STEP 2 * X7
10852  HPLOT XT,Y8 TO XT,Y9
10854  NEXT
10860  SCALE= 1
10865  REM    CHANGES PLOT POINT SHAPE
10870 X8 = 39:X9 = 239:Y8 = 150:Y9 = 0
```

```
10875 X7 = (X9 - X8) / 10:Y7 = (Y8 - Y9) / 10
10880 SH = 2: ROT= 0: SCALE= 1
10885  ON I GOSUB 10310,10315,10320,10325,10330,10335
10890 I = I + 1: GOTO 10340
10895 SH = 1: ROT= 0: SCALE= 1: RETURN
10900 SH = 2: ROT= 8: SCALE= 2: RETURN
10905 SH = 3: ROT= 0: SCALE= 1: RETURN
10910 SH = 3: ROT= 32: SCALE= 1: RETURN
10915 SH = 2: ROT= 0: SCALE= 2: RETURN
10920 SH = 1: ROT= 16: SCALE= 2: RETURN
10925  RETURN
10999  REM  FORMATTER NEXT. FORMATS X INTO FIELD WIDTH WI WITH DI DIGITS.
 PRINTS X;
11000  REM *FORMATTER*
11001 XX = 10 ^ DI:X =  INT (X * XX + .5):X$ =  STR$ (X):XX$ = ""
11002  IF  ABS (X) > 1E9 THEN X$ =  STR$ (X / XX):XX$ =  MID$ (X$,
 LEN (X$) - 3,1): GOTO 11006
11003 XX =  LEN (X$): IF X < 0 THEN  IF XX < DI + 2 THEN X$ = "-" +
  RIGHT$ ("000000000" +  MID$ (X$,2),DI):XX =  LEN (X$)
11004  IF XX <  = DI THEN X$ =  RIGHT$ ("000000000" + X$,DI + 1):XX
 =  LEN (X$)
11005  IF DI > 0 THEN X$ =  LEFT$ (X$,XX - DI) + "." +  RIGHT$ (X$,DI)
11006  IF XX$ = "E" THEN  IF WI > 4 THEN X$ =  LEFT$ (X$,WI - 3) +
  RIGHT$ (X$,3)
11007  IF  LEN (X$) > WI  THEN X$ =  LEFT$ (X$,WI - 1) + "*"
11008  PRINT  RIGHT$ ("         " + X$,WI);: RETURN
```